Using RFID In Augmented Campus Environments

Masters Thesis, Computer Science

Dervis Mansuroglu

October 24, 2008
Halden, Norway

Østfold University College
Mobile Applications Group

Abstract

This thesis uses RFID in a mobile information system to augment and to be an addition to existing campus services. The thesis has focus on and explores two different but yet relative problem domains. These two focuses of the thesis are a user-oriented part discussing RFID and relevant technologies and a technical part that gives an overview of the RFID programming libraries and the implementation of a prototype that uses these libraries. The main objective of the thesis is to find out whether this prototype does augment the existing services and what students and employees of the campus think of it. Interviews and user surveys are carried out which gives an overview of what Norwegian students at Østfold University College think about a augmented campus environment. The technical part shows how to achieve automatic identification, including detailed description of the memory layout. The resulting prototype have shown several advantages, but also challenges, in integrating RFID in a campus environment.

Keywords: RFID, Java JSR 257, Mobile Applications, Campus Services

Acknowledgements

Among the people I would like to thank is my supervisor Gunnar Misund and the Mobile Applications Group, who helped me structure and build this thesis. He made some important decisions at the start, which helped me reach my goals. Then I would like to thank my mother, sister and brother for their endless support. Without their help I would not even be here and made through two challenging years for a Master degree. I also thank Ehsan and Cecilie for correcting the grammar and the people who tested my prototype.

Table of Contents

Chapter 1

Introduction

Research on the field of ubiquitous computing has reached new arenas rapidly and scientists are putting great effort into finding new ways of improving our effectiveness and automating systems to make life easier for most people. To be able to do this, Schilit *et al.* [1] emphasizes the need for computer systems that provide ubiquitous access to information, communication and computing, but also for mobile distributed computer systems that react to the constant change in computer users environments. Most of the research done has involved in making computers understand the surrounding context and allowing them to automatically collect context information, thus eliminating the need for users to explicitly provide the needed information [2].

A new way to collect information about the context is by labelling objects. The environment and the surrounding context of users are identified by tags, allowing them to *speak* when *spoken* to. The physical environment is thus recognized and mapped by a computer system. Chapter 2 explains how this is possible, by presenting background information about the technologies used.

Mark Weisers vision of the disappearing computer [3], seems to become true as we reach the end of another decade. I believe that desktop services will decrease and mobile services increase in the future. In my thesis I will try to find out if wireless electronic tags can augment existing desktop services. The contributions of this thesis may pinpoint whether simple and available technology can be used to indicate the usability of mobile RFID based services in a campus environment.

RFID devices are under continues development and the technology improves every year. The devices are getting smaller and the wireless reading ranges are getting longer, which reinforces the vision of the disappearing computer and a totally digitalized environment. To accomplish the digitalization, however, standards are required to have a common language and communication platform among the increasing number of context-sensitive devices. The thesis will present an important standard called NFC (*Near Field Communication*), which is mainly developed by Nokia. Sun has approved NFC as the primary interface against RFID hardware in the Java programming language. Chapter 3 will give an overview of the most important API's and also purpose techniques for handling advanced features like queueing.

1.1 Research Objectives

Numerous research has been done in the field of augmented environments in the last eight years. Tagging objects and places around us using one dimensional vertical barcodes and recently two dimensional barcodes, has been two of the most used information carriers among research groups. A new technology based on radio frequencies called *radio frequency identification* (RFID), is said to replace barcodes in the future. RFID devices have not only been a subject to heavy research in augmented environments for the past six decades, but also in smart-homes and smart devices. However, in most of the works, it has either been used in personal digital assistants (PDA) or hybrid phones with RFID readers attached to their shells.

With the release of the Nokia 6131 NFC phone, with integrated RFID reader and writer, hole new possibilities are emerging. This thesis has a divided focus with one user-oriented part and a technical part. The user-oriented part focusing on augmented campus environments with RFID as main subject and one technical part exploring the Nokia programming API's. The main objective of this thesis is:

Investigate the usability of RFID to augment an campus environment and analyse the technical application programming interfaces

I will specifically focus on these questions:

- Will tags and mobile RFID readers augment existing campus services?
- How can the Nokia programming API's be used to handle RFID tags?
- What do the users think about RFID powered campus services?

This thesis can be seen as a continuation of previous projects I have done on context-awareness [4] [5]. In this thesis, I am approaching context-aware services from a different angle. Instead of using Bluetooth, which was the main force behind the previous projects, I am using RFID as explained above.

1.2 Motivation

We are slowly moving towards the end of another decade and a lot has happened since the eighties, when the personal computer had just started to operate in our homes. In about thirty years, we have moved from using huge terminals to writing on small mobile devices. The history and evolution of computers is therefor one of the motivations for writing this thesis.

The overall motivation for my work is to make a contribution to the ubiquitous computing domain, by addressing its hardware related problem areas. Better sensor and actuator technology is continuously being developed and as mobile phones are becoming more powerful, new types of services can be made available.

Our daily environment is evolving with a growing number of sensors and technologies that seek to increase the efficiency of everyday trivial tasks as well as more complicated work related tasks. RFID is just one of many building stones towards an automated environment and this development process will help me understand more about the technologies that will be as pervasive as the Internet is today.

1.3 Expected Deliveries, Research Contributions

The expected deliveries from this thesis are:

- A technical overview with analysis of standard API's and coding examples
- An implemented prototype
- User experiences from real users.

1.4 Methodology

This thesis discusses a technology that have unequal amount of distribution in different parts of the world and in industries. A problem statement and a concept has therefore been developed by experimenting and exploring, hence explorative design techniques will be used. Agile design methods will be used to assist the design process of the prototype. The overall methodology is based on iterative processes. Some of the steps taken in this thesis are:

- Literature studies
- Finding research objectives
- Analysing the campus environment and designing a prototype
- Implementing the prototype
- Testing the prototype with real users
- Discussing findings and proposing future work

Literature Studies gave better understanding of the problem domain and was used to gain knowledge about theories and related technologies. Literature surveys were conducted several times during this project and has helped me construct my research questions and form my scenarios.

Analytical Study was done to obtain a deeper knowledge of the problem in hand. The analytical study contributed to the decisions that were taken in the design and implementation phase.

The last phase of the project has been a validation process, were I compared test results to the results of the analytical study. This process also clarified whether the hypotheses and scenarios from the analytical study and literature studies became true when testing with real users.

1.5 Limitations

This thesis will utilize and discuss RFID tags, both active and passive. However, active tags will not be used. This is due to cost and poor availability of mobile readers. Among the very few mobile RFID readers, only the Nokia 6131 NFC mobile reader will be discussed and used.

The prototype that I will design and implement is meant to be a proof-of-concept. This limits the time I will spend on issues such as user interface and robustness. I expect, however, the prototype to perform the identified tasks in the analysis step.

1.6 Challenges

There is several challenges in this thesis. For instance the poor documentation of the technical aspects of technologies used and the poor documentation of their software development kits. The architecture of the RFID tags are also badly documented, which makes it difficult to get a complete overview of the inner workings of the tags. Since the only documentation I have was the Nokia 6131 NFC SDK documentation, it has been a great challenge to learn how to use the JSR-257 API. Another challenge was how to carry out the user studies and to make a survey of opinions. However, thanks to enthusiastic students and employees, it turned out to be a great experience.

1.7 Selection Of Hardware

Since this thesis was written for the Mobile Applications Group (MAG), it was obvious that mobile equipment was needed. I had no in-depth knowledge about RFID or barcodes when I started my search. Therefore it was important to learn as much as possible about radio frequency identification before purchases where made. The selection of hardware took about a month. RFID as a technology is well explained in numerous resources and it is easy to come by information about the technology. However there is different information, especially about the reading ranges of the transponders. While searching the Internet for manufacturers and resellers, I got aware of three different types of RFID hardware: desktop and handheld equipment, small electronic components and lastly NFC devices. Desktop and handheld RFID readers were excluded since they are mainly used in the industry and are not very mobile. Among others, I found Bluetooth activated readers. However, the size and the price made them impractical. I also looked at some small electronic RFID components to consider if they could be used in a customized device, but that would be out of the scope for this thesis. NFC was proposed by my supervisor and is mainly developed and used by Nokia. NFC is subset of RFID and has a short reading range. A reader is integrated into the Nokia 6131 NFC phone.

I have also been in contact with representatives from Nokia and other experts that have experience

with RFID equipment. Among those people was Bård Myhre from SINTEF[1] and Timo Arnall from OSAD [2], whom guided me to websites offering compatible NFC Tags for the 6131 phone. Appendix D lists the pros and cons of different kind of tags and readers from a number of websites. I was in contact with some of the resellers for more information about their products and even though I did not get detailed answers, I felt that it was satisfactory. The equipments I ordered are shown in figure 1.1. These equipments were bought from Top Shop, a online shop in Finland. Top Shop has both mobile

Figure 1.1: Nokia 6131 NFC, a USB RFID reader and a development kit

phone integrated RFID reader and USB readers. There is also a great selection of Mifare tags.

1.8 Outline Of The Report

Chapter 2 presents background information about the technologies used and those relevant for this thesis. Topics such as wireless technologies, visual tags and object hyperlinking is presented. Section 2.7 gives an overview of related works. The overview is divided into two parts, the first will discuss RFID in campus environments and the second will discuss systems using a backend server.

Chapter 3 discusses the programming API's that allows mobile communication to RFID hardware through Java. This chapter presents the usage of a standardized data format and using a custom data format. Chapter three gives a detailed overview of how RFID tags are used.

[1] The Foundation for Scientific and Industrial Research
[2] Oslo School of Architecture and Design

Chapter 4 explains how the prototype is designed and discusses the design process of the prototype application. Scenarios and requirements are presented, including an analysis of the campus environment.

Chapter 5 is about the implementation of the prototype. This chapter will explain the data format, data model and the application data flow.

Chapter 6 gives an insight to how the system was tested and what methods and techniques that were used. The chapter describes the results of several tests done with multiple users in detail, as wells as the results from the user satisfaction questionnaires.

Chapter 7 discusses the findings, evaluates the prototype and gives a conclusion on the concept. Future work is also presented.

Finally; the references, a glossary of terms listing abbreviations and the appendices.

Chapter 2

Background

This chapter will describe relevant technologies and discuss previous research done on mobile wireless networks. However, the chapter only touches the surface of the technologies, as it is out the scope to give an in-depth description. Among the topics that will be discussed are wireless technologies, NFC and RFID and the importance of object hyper linking.

2.1 Wireless Communication

Wireless communication is a technical revolution that has been researched for several decades. As more and more people around the world use mobile phones, usability and functionality also becomes more important. By expanding the Internet bigger day by day, wired networking changed our everyday life. However, the trend has changed after the new millennium to also include wireless networks. Today, advanced mobile phones make it possible to combine mobile networking with wireless local area networks. This combination is possible through the use of cable replacement technologies, which enables pervasiveness in mobile devices.

There are currently five cable replacement technologies available. These are Bluetooth, ZigBee, RFID, Ultra Wide Band (UWB) and Infrared (IrDa). This section will describe Bluetooth, ZigBee and Near Field Communication, which is considered as the three most relevant for this thesis.

2.1.1 Bluetooth

Bluetooth is a short-range wireless technology intended to replace cables between portable or electrical devices. Bluetooth enables mobile phones, PDAs, digital cameras, printers and laptops to exchange information in an easy and secure way. The Bluetooth communication protocol is designed for low power and low cost transceiver microchips, which enables devices to connect to and share data when they are in proximity of each other. Once in proximity, devices can discover each other, explore and

use each others services. Services are also referred to as *profiles* in the Bluetooth specification. Some examples of services are file sharing, imaging, printing and serial connection to mention a few.

Bluetooth is available in three different range intervals. These three are listed in table 2.1. Several improvements has been done to the Bluetooth specification and the latest version allows transmission speed up to 3 Mbit per second. Bluetooth, or IEEE 802.15.1 WPAN[1] which is its technical name,

Class	Power	Range	Used by
Class 1	100 mW	100 meters	Industry
Class 2	2.5 mW	10 meters	End-users
Class 3	1 mW	1 meters	Small electronic devices

Table 2.1: Bluetooth ranges

operates in the unlicensed 2.4 GHz ISM[2] band. A Bluetooth WPAN can consist of up to seven devices in a temporary ad-hoc network.

Ericsson started developing the Bluetooth system in 1994 and afterwards founded the Bluetooth Special Interest Group (SIG) in 1998. The Bluetooth SIG is a voluntary organization that consists of volunteer staff from Ericsson, IBM, Intel, Microsoft, Motorola, Nokia and Toshiba. All of which provide experts that develop and enhance the Bluetooth specification. The cooperation of these companies, including software and hardware suppliers, has lead to the wide availability of the Bluetooth technology for end-users. Bluetooth is supported by Microsoft Windows and also by several Linux distributions, which enables developers to integrate Bluetooth support in their applications.

Please see [6] for more information about Bluetooth. [7] is also used a reference.

2.1.2 ZigBee

Zigbee is a wireless network technology and an alternative to Bluetooth and is meant to be easier and cheaper than other similar WPAN standards. ZigBee has several similarities with Bluetooth, but is developed to be used in very different applications than Bluetooth. ZigBee targets remote monitoring and control applications that require low data rate and long battery life. It operates in the ISM band and has a range up to 75 metres. Compared to Bluetooth, the data transmission speed is slow, but can be up to 250 KBits. Some usage areas of ZigBee is:

- Lighting controls
- Heating control
- Industrial, home and building automation
- Security systems

There is also a relationship between IEEE 802.11 (WiFi) standard and ZigBee. ZigBee supports star and peer-to-peer network topologies. The star topology is similar to a Bluetooth PAN and a little

[1] Wireless Personal Area Network
[2] Industrial, Scientific and Medical

RFID system where there is one device responsible for either initiating or controlling communication. A ZigBee network consist of a *coordinator* and several *end devices*[3]. A coordinator, or full function device (FFD), is a device that is able to initiate, terminate and route communication in the network. A ZigBee end device, also called an reduced function device (RFD), is a device that is capable of minimal computation and that can be implemented with minimal hardware resources. Examples of such devices are alarms and sensors.

For more information about Zigbee, please see [8], [9] and [10].

2.1.3 Near Field Communication

Near Field Communication or NFC in short, is the name of a short-range wireless communication technology build upon standards from RFID. NFC enables wireless connectivity between electronic and mobile devices in a standardized way, which makes it possible to use it in a number of different applications.

NFC is mainly directed for usage in mobile phones, where *touch-based interaction* is in focus. Since it is a short-range protocol, the range is only a few centimetres and devices must therefore be very close to each other. To activate communication, a NFC phone simply needs to touch another device or a NFC information carrier[4]. The list of advantages for using NFC in mobile phones is long, however the main purpose of NFC is effectiveness. NXP Philips[11] has listed areas where touch-based interaction can be used:

- consumer electronics
- mobile devices
- PCs
- smart objects
- payment purposes

NFC uses the freely available 13.56 MHz frequency band to communicate with high frequency RFID tags. NFC is compatible with existing contactless communication standards, which makes it adaptive and easy to implement. [12] and [13] mentions, for instance, initiation of a Bluetooth or WiFi connection. NFC supports the infrastructure of both Philips Mifare and Sony Felica contactless smart cards. Philips produces Mifare transponders and smart cards, which is one of the most popular and widespread contactless smart card platforms. On the other hand, Sony Felica smart cards are currently used in transport payment services in Japan[5].

What differences NFC from RFID, is the NFC protocol which enables a device to be both "Initiator" and "Target". However, not both can be enabled on the same time. NFC devices also support the same "active" and "passive" operation modes as RFID. The passive mode of communication enables devices to be operational, even if the device is turn off. However, the active mode of communication requires the

[3] It can be as many as 255
[4] The carriers are also called "tags" for short
[5] http://en.wikipedia.org/wiki/Suica

use of an internal power source. NFC enabled mobile phones can use their own battery to generate the power needed when operating in active mode. Ecma International[6], which approved NFC as a standard in 2004, describes the passive operation mode as a "power-saving mode" [12].

NFC was created in 2002 as a collaboration between Nokia[7], Philips[8] and Sony[9]. It was standardized by 2004 as a ISO (18092) and ECMA (340) standard. These companies have also cooperated in the development of compatible tags and phones. Nokia has produced two mobile phones that are able to read NFC tags, the hybrid Nokia 3220 and Nokia 6131 NFC. Nokia is one of the major contributors to the NFC community and is the leading company in the development of JSR-257. Java specification 257 is the description of the Java API that implements NFC on the J2ME platform.

2.1.4 Comparing NFC, Bluetooth And ZigBee

Table 2.2 compares the three discussed technologies and summerizes the specifications.

	NFC	Bluetooth	ZigBee
Network	Peer-to-peer	Point-to-multipoint	Peer-to-peer, Point-to-multipoint
Range	0.1m	10-100m	1-100m
Freq.	13.56MHz	2.4GHz	2.4GHz(Global), 915MHz(US), 868MHz(EU)
Speed	up to 424kb/s	721kb/s-3mbit/s	20-250kb/s
Security	Hardware, protocol	Protocol	Protocol
Modes	Active-passive, active-active	Active-active	Active-passive, active-active
Applications	payment, data exch.,	data exch., cable replacement,	sensor and control networks
Infrastructure	RFID	Mobile phones, PDAs, laptops	sensors, laptops
Usability	Touch based, easy to use, complicated API	Easy to use, long set-up time	Unmanned, low-power, complicated API
Cost	Low	Moderate	Low

Table 2.2: Comparing the NFC, Bluetooth and ZigBee

2.2 Radio Frequency Identification

This section will describe what radio frequency identification is about, its usage, standards and security and privacy concerns. This section will also give an overview of the components in a RFID system.

[6] www.ecma-international.org
[7] www.nokia.com
[8] www.nxp.com
[9] www.sony.net

2.2.1 What Is Radio Frequency Identification

Radio Frequency Identification, normally called RFID, is a wireless communication technology that enables objects to be automatically identified. This is achieved by labelling any type of physical object with an id. The labelling is referred to as *tagging* and the labels themselves are small radio transmitters called *transponders*, or simply *tags*. Tags can be identified by RFID readers from a few centimetres to several meters away. RFID has the advantage of being a *transparent* technology, meaning that tags can be read without having line-of-sight. The transparent nature of RFID has shown to be very popular lately.

RFID technology has existed since World War II and has been subject to research in many countries, mainly in the USA and in the Soviet Union. However, the technology has only been available for scientists and researchers by the seventies. RFID has similarities with existing systems like bar codes and smart cards with magnetic strips, which all are able to store data. RFID tags however, can hold more data and transfer its content faster. Unlike static bar codes, a RFID system could be used to rapidly link objects to a dynamic system. A full RFID system consists of many components. Three main components are tags, readers and software. The software is used to communicate with the readers and the tags are read by the RFID readers. Other optional components are sensors, external antennas and network gateway.

2.2.2 RFID Tags

RFID tags are able to store data, which can be either a long text string or a simple identification number. Data on the tags can be read by a RFID reader, when the tag is in proximity. RFID tags and readers exist in different shapes, sizes, ranges, features and capabilities. Tags are composed by two different parts, a microchip and an antenna. The microchip is used for storing and processing data. The antenna is used to transmit and receive modulated radio frequency signals. Figure 2.1 shows different types of tags. It exists three types of RFID tags; *passive*, *active* or *semi-passive*. Passive tags do not

Figure 2.1: Examples of RFID tags

have an internal power supply and is only operative when contacted by a reader. This means that passive tags "sleep" until they are woken up by the radio waves of a reader. The waves from a reader contain enough power for the tag to send a response.

Active tags have their own internal power source, which enables them to continuously broadcast data. Active tags have the advantage of being much more advanced then passive tags, since the power supplied from a battery is normally enough to operate sensors and perform calculations. However, active tags cost more and need maintenance for the power source, which eventually runs out. Semi-passive tags are battery-assisted tags and are a hybrid of active and passive type tags. The power is only used to run the microchip and not the broadcasting of the RF signal, which enables semi-passive tags to operate sensors and last longer than active tags.

2.2.3 Frequencies

Unlike NFC which only operates in the 13.56MHz frequency band, RFID operates in several frequency bands. The RFID frequency interval, which ranges from 9 kHz and up to 2.48 GHz, contains five sub-ranges. Each interval has its own special area of use. The reading ranges are also very different and depend on two factors: power and operating frequency. The size of the antenna can also affect the range in some cases. A bigger antenna within a frequency band can occasionally extend the reading range with a few centimetres. Table 2.3 lists the five frequency band that RFID uses. Low frequency (LF) RFID

Frequency Name	Frequency	Range
Low Frequency	9 - 135KHz	1 - 4 cm
High Frequency	13.56 - 15.6MHz	1 - 10 cm
Amateur Radio Band	430 - 440MHz	Up to 90 m
Ultra High Frequency	860 - 930MHz	10 m
Microwave Frequency	2.4 - 2.48GHz	3 - 4 m

Table 2.3: Frequency bands and their ranges

systems have the shortest reading range. LF RFID tags are very cheap and typical applications for it are animal and human identification. Small devices, like toys, are also known to make use of LF tags. RFID tags have different purposes and target applications depending on factors like frequency band and transponder type. Selecting the correct frequency system is therefore very challenging. LF RFID readers normally read only one tag at the time ([14]), which makes it suitable for access control systems. It will however not suit public systems (i.e. metro) where multiple tags are in range every second. High frequency systems (HF) have longer reading range and are more advanced than LF systems. HF RFID systems are widely used and have many different applications today, i.e. smart-cards, library, item tracking and laundries to name a few.

2.2.4 Security And Privacy

As with all new technology, RFID has also been a subject to heavy discussions regarding its security concerns. This section will give an overview of security and privacy issues found in RFID. When explaining to a person not familiar with RFID that future mobiles can be used for payment purposes, the very first question that emerges is: "Is it safe?". This is a natural question to ask, as most users are already familiar with security and privacy issues in wireless home networks and the medias large coverage of the topic[10].

RFID transparency was characterised as an advantage, but it is also a treat to consumer privacy. Tags and readers can easily be hidden, allowing retailers and companies to track people and items. In a worst case scenario, a company could collect so much information about a person, that habits and belongings are revealed. Companies with a desire for tracking their own products, easily turn into tracking people. Besides collecting habits and belongings, a person's movement can be tracked. This scenario might sound fictional, but the opponents of RFID have shown that it is easy to make it into reality. The founders of the CASPIAN[15] organisation, Katherine Albrecht and Liz McIntyre, have revealed several ways that RFID could be used for espionage:

- Drivers licence
- Credit cards
- Bus cards, metro tickets
- Passports
- Clothing, shoes

Representatives from the RFID industry say that "...they are not designed to be used as spy chips"[11] [16]. However, this is not easy to believe, when all of the tagged objects mentioned above exist. For example a RFID enabled drivers license was developed by NADRA [17], American Express developed what they call a "Contactless Credit Card"[18], the use of RFID tagged biometric passports[19]. In their book "Spychips" [20], Albrecht and McIntyre also explain that network enabled smart-appliances can identify your home and therefore make you a target of spam and junk mail. They also say that tagged-objects pose as much risk as the ones kept at home.

The RFID industry also uses the argument of short reading ranges. NFC uses high frequency tags, where reading ranges are approximately 30 centimetres, better in good conditions. Copying or stealing private information from a NFC phone requires one to be in close proximity to the source. Albrecht and McIntyre [20] point out that two tags have to use the same frequency and protocol to be able to communicate. In case of NFC, this means another NFC phone or a handheld reader.

Obviously, there are many examples of how RFID tags can lead to loss of personal information. Preventative measures are therefore needed. When talking about RFID security, the words "eavesdropping" and "jamming" comes up. Eavesdropping can be defined as a person listening in secrecy to

[10] http://www1.vg.no/teknologi/artikkel.php?artid=137883 and http://www1.vg.no/teknologi/artikkel.php?artid=158492
[11] Bill Allan, Texas Instruments

a private conversation. Jamming is a technique for intentionally interfering radio signals and a way of securing transmissions and communications. Researchers of RFID security have designed devices that prevent eavesdropping, by jamming the radio signals. The following is a description of three examples of such devices.

Soppera *et al.* [21] say that "privacy can be maintained if the tag output is indistinguishable from a value". An approach to this is to rewrite the content of the passive tags with random data and a secret key to a special active tag they call an "Acceptor Tag". To be able to read the contents of the passive tags, readers need to be authenticated by the acceptor tag. Another system is proposed by Juels and Brainard [22]. Their system consists of a software layer that works as a filter. Special tags have *classifications* that the software layer will use to determine whether to allow reading. Juels *et al.* [23] proposes a special tag called a "blocker tag", which can protect ordinary passive tags. When carried by a consumer, the blocker tag blocks RFID scanners by simulating all possible tag-identifications. A similar device is proposed by Rieback *et al.* [24]. This device is called "The RFID Guardian" and controls the communication between readers and tags. Readers used in homes or offices, have a software-backend that identifies the reader to the Guardian. The RFID Guardian is portable device that is able to protect tags, by jamming the radio signals. Access grated readers are able to read the data from the tags, through the integrated RFID reader in the Guardian.

2.3 Visual Tags

Visual tags are a collective term for images that are able to store large amount of data in a very small area. 1D and 2D tags are the two most commonly used visual tags today. Bar codes are widely used in the logistics industry and can be found on every item in grocery stores. 1D tags, or vertical bar codes, are also used in a lot of other different areas of modern civilization. 2D tags are being used for identification of small items such as integrated and printed circuit boards.

Different types of 1D tags and 2D tags exits. Visual tags can store different amount of data and different types of characters. They even come in different shapes. Visual tags are also called symbols and the mapping between the data and the symbol is therefore called *symbologies*. A symbology specifies how a symbol is to be encoded and decoded to and from a scanner. It also specifies the pattern of the symbol. A 1D symbol consist of parallel vertical bars. The width of a 1D symbol depend the length of the data. 2D symbols however, are normally shaped as squares or circles. 2D symbols have the advantage of being able to store much more data on smaller areas, and are even more robust against exposure from the environment. Table 2.4 lists the most used symbologies today. Figure 2.2 shows three of the most used tags. The left most symbol is an example of a EAN-13 tag. EAN (European Article Numbering) is an extension of the standard used in the USA, the UPC ('Universal Product Code'). The popularity of EAN codes is gaining increased interest globally. The right most image shows a tag called Data Matrix, which is a two dimensional barcode. The Data Matrix tag in

Symbol type	Country	Usage	Tag type
EAN 13	Europe	Grocery items	1D tags
UPC-A	US	Grocery items	1D tags
Code 128	Worldwide	Grocery items, Library	1D tags
Data Matrix	Worldwide	Electronics, small items	2D tags
PDF417	Worldwide	calibration and operating instructions	2D tags

Table 2.4: Visual tag symbologies

Figure 2.2: EAN-13, UPC-A and a Data Matrix symbol

figure 2.2 is a barcode encoding of the text *Mobile Applications Group - mobapp.hiof.no*. The Data Matrix image was created using the tag generator on the Semacode web page[12]. Semacode is also the creator of the freely available development kit for scanning and decoding Data Matrix symbols.

2.4 Comparing Tags

The different types of tags presented in the previous sections have different applications. For this reason, standards are developed to standardise how the technologies work. These standards are important, as they allow tags of different types to communicate with readers throughout the world. Regulatory standards ensure product characteristics, interoperability, reliability and safety [14]. Table 2.5 summarises these attributes and lists the ISO standards valid for each tag. Table 2.5 list two main

	Active RFID tags	Passive RFID tags	2D tags	1D tags
Standard	ISO 15693, ISO 18000-7	ISO 14443, ISO 15693	ISO 16022	UPC, EAN
Frequency	all	13.56MHz, 127/135KHz	camera, laser	camera, laser
Modes	active	passive, semi-passive	passive	passive
Range	70-100m	5cm-30cm	10-30cm	5-10cm
Security	software, protocol	software, protocol	software	none
Applications	0	0	0	0
Cost	high	low	very low	low

Table 2.5: Comparison of tags

[12] www.semacode.org

types of tags, RFID and barcodes. They are selected because they are much more easy to use, learn and operate. Other tags, such as Wifi and UWB are either time consuming to implement or simply too expensive. A barcode is simply a piece of paper with an image on it. Anyone with a printer can generate and make use of barcodes. An advantage barcodes have over RFID-solutions is the possibility to implement a barcode-system without having to pay a single penny. The only component of a barcode-system that is needed is a reader. Software development kits are freely available, which can be used to decode the image[13]. One of the major advantages that RFID have over barcodes is the ability to

	Barcodes	**RFID**
Tag reading	Single	Multiple
Range	10 meter	90 meter
Reusable	No	Yes
Visibility	Line of sight	Transparent
Implementation	Easy	Challenging
Cost	Cheap	Costly

Table 2.6: Comparing RFID and Barcodes

scan the RFID-tags without having visual sight. Depending on hardware, it is also possible to identify multiple tags on one scan. Combined with long reading ranges, RFID increases efficiency of business processes [21].While barcodes are simply images, RFID tags are dynamically changing. Table 2.6 shows pros and cons on both technologies, but RFID seems to have more advantages.

2.5 Mobile Technology

The technologies described in this chapter works quite well and new applications are emerging all the time. They are also well documented and have specifications that enable several different functions. However, these technologies are useless if the output from the devices cannot be read. For this reason the specifications include a software layer that enables laptops and other devices to read and decode the output. Software that communicates with the software layer is called a *middleware* and can be implemented in several different platforms. In this section, the Java 2 Micro Edition platform is descried.

Java 2 has been selected in this thesis since APIs for both Bluetooth and NFC exist and is under standardization. Programming languages like Python and C++ have also capabilities to access wireless technology, but is neither well documented nor standardized.

2.5.1 Java 2 Micro Edition

Java Micro Edition (J2ME) is a light weight and highly optimized version of Java 2 standard edition, which enables mobile phones, set-tops boxes, embedded systems, pagers and various consumer devices

[13] http://people.inf.ethz.ch/adelmanr/batoo/

to run Java application. Java has the advantage of being platform independent, which enables applications to run on many different systems. In the case of J2ME, applications can be executed on many different mobile phones and PDAs.

Cell phones, mobile phones and PDAs have different specifications, functions and available resources. For this reason, J2ME can be *configurations*. A configuration specifies the resources and available memory on a specific device. J2ME have two base configuration specifications, the CLDC and the CDC. CLDC is targeted for use in cell and mobile phones. CDC is targeted for PDA's and other devices that have more resources available, i.e. set-up boxes. No further details will be given on CDC. Readers interested in CDC are advised to read [25].

Just as *applets* are designed to execute on the Internet, *Midlets* are designed to run on mobile phones. Developers are able to create Midlets, by utilizing the API's of the MIDP package J2ME. CLDC provides the core Java functionality, where MIDP is the layer above it providing user interface and other useful tools. The J2ME platform also includes optional packages, such as support for file creation

With its portability and the richness of features, J2ME easies the implementation of ubiquitous computing. Today modern mobile phones are becoming more and more advanced and most of them support all of the optional packages of J2ME. There is more advantages to achieve ubiquitous computing with communication protocols such as GPRS in combination of Java Servlet technology. The Java 2 Micro Edition platform enables any server side technology to be used.

Developed midlets can be tested easily with a mobile phone emulator. Sun Microsystems provides a wireless development toolkit, which includes several phone emulators. Specific phone emulators, are also provided by the phone producers. In this thesis, the Nokia 6131 NFC emulator is used. A how-to guide to install the emulator onto the Eclipse Java development environment can be found in appendix C.

More information on J2ME can be found on [26]

2.6 Linking Tagged Objects To Virtual Resources

RFID enables any object in the real world to become an information carrier. In a setting were different objects are tagged, it will be important to read and use the data in an efficient way. Since the tags are designed to identify an object, they will be useless if they are not associated with a physical object. In the former sections, information carriers like RFID, NFC and visual tags were described and their ability to identify objects in different ways. Their differences, however, are the amount of data and the method the data is transformed. When the identification data is stored in a database, a link between the physical world and a virtual space is possible. The linkage is a key for using what is called *object hyperlinking*. Object hyperlinking aims to map objects in the physical world we are living in to objects in a virtual space. Sources like [27] defines object hyperlinking as "extending the Internet to objects

and locations in the real world". Even if this definition aims to Internet hyperlinks and resources, any type of electronic service or resource could be used. Want *et al.* [28] defines it as *blend the affordances and strengths of physically manipulatable objects with virtual environments or artefacts*.

Object hyperlinking is in itself just a description of what is what and what kind of relevant resources are available in the virtual world. It has, however, an important role in smart environments and in a technology called *ambient intelligence*[14] (AmI). AmI builds upon ubiquitous computing and combines several professions and special fields, i.e. electronics engineering, machine learning, context-aware and adaptive systems. As sensors and computers evolves into smaller and smaller devices, AmI will allow those devices to be able to communicate with each other [29]. Object hyperlinking in an advanced infrastructure like AmI, will link intelligent sensors to virtual resources and smart user-interfaces. Implementing RFID systems in human environments with off-the-self components and computer systems today, can give an understanding and preview on future AmI systems. AmI is out of the scope of this thesis, however, interested readers are advised to explore [30].

Objects can be tagged with the tags presented in the above sections, there is also one way of tagging not mentioned yet and that is "virtual tags". Virtual tags are geographical coordinates mapped against locations. Virtual tagging can be achieved with GPS-activated mobile devices. A conceptual model for tagging areas using Bluetooth is proposed by Uyuducu [[5]]. The conceptual model describes how Bluetooth access-points are used to identify the area in its proximity. Tagging an area instead of specific objects causes demanded information to be presented in a different way. There are some similarities between contactless tags and virtual tags, in the way that the computation is not seen. Its only the human-computer interaction interface which is visible. This leads to the vision of "the disappearing computer" mentioned by Bill Gates [31].

2.7 Related Work

This section will discuss some related work done with RFID and visual tags in human and campus environments.

2.7.1 RFID And Visual Tags In Campus Environments

Kim *et al.* [32] discusses students bad eating habits. A survey done at their campus, has shown a number of fact like rapidly changing schedule, lack of knowledge about cooking and lack of transportation to a grocery. Kim *et al.* purpose therefore a system that aims to improve students eating habits. This system consists of an intelligent basket, a backend server and a dining hall. The dining hall is a combination of a shopping, cooking and dining environment. The intelligent basket is build up by a RFID reader and a network enables display. When an item is put inside, the RFID reader scans the item and displays information about its healthiness. When the student has finished the shopping an automatic checkout

[14] It is also referred to as "Ambient Technology"

process takes place. The student is now ready to cook and later eat the food in the dining hall. Kim *et al.* also mention personlization and tracking.

Willis *et al.* [33] discusses the navigation and way finding problems that blind and visually impaired students have at an campus. A localised information/navigation system build on RFID is therefore proposed. A grid consisting of low-cost passive RFID tags is installed in hallways and rooms in the building, outdoor pathways. The tags store information about the surrounding environment and room inventories, which Willis claims will exclude the need for an centralised database and wireless infrastructure to handle connections. In their system, an RFID reader is integrated into a walking cane and a shoe. They also mention that tags can be installed under carpets and at the edge of stairs and sidewalks. All computation is done on a Bluetooth activated PDA. Willis *et al.* say that Huffman encoding can be used to compress the size of the data to be stored.

Klemola *et al.* [34] introduce a system for discovering location specific mobile web services. While the two previous systems used RFID reader carried by the users, this system uses fixed readers attached into the users environments. Their motivation for doing so, is the minimal availability of mobile RFID readers. A mobile phone with an attached RFID tag, is used to touch a fixed RFID reader. The reader, which is connected to a backend server then pushes services to the users device. A prototype called CampusTag is tested in a campus environment. Messages are pushed to the users devices using a WAP Push Proxy Gateway (PPG). Using PPG as infrastructure is not technically difficult, but it still requires configuration and it is very expensive for both users and service providers. Using GPRS for communication requires no configuration or maintenance, as it uses infrastructure of the mobile ISP. The CampusTag system also requires the users to register their phone numbers through a web interface, since the PPG service addresses the messages directly to the phone numbers.

A system that makes use of one dimensional visual codes is the ETHOC system developed by Rohs and Bohn [35]. A mobile phone with an integrated barcode reader, is used to read tagged objects in a campus environment. By reading the data from the physical objects and attaching online information and functionality to them, Rohs *et al.* mentions that there will be a natural interaction between the physical and the virtual environment. By following the *physical hyperlinks*[15], it is possible to trigger online functionality such as event notification and making reservations. One other interesting thing in the ETHOC system is the ability to track users and thereby provide a personal history for each user. The mobile used in this project had an attached barcode reader, which made the phone unnecessarily big. To handle this problem, researcher utilized the integrated camera and developed software to recognise both one and two dimensional barcodes[16]. Not only does it remove the need for an external reader, but it is much more efficient.

Chen [36] propose a generic and hardware independent framework, for a ubiquitous information system. He says that the generic nature of the framework, will provide an easy way to implement an RFID system in school settings. He also states that the hardware integration layer, makes the framework

[15] Physical hyperlinks are described in section 2.6

[16] http://people.inf.ethz.ch/adelmanr/batoo/

hardware independent. However, it is a challenge to make a completely hardware independent system with regard to the number of different manufacturers of RFID hardware.

Han *et al.* [37] from the Yonsei University in Korea, worked in 2004 on the UTOPIA project. Multiple departments was involved in the development of a context-aware computing system for a campus environment. The UTOPIA project consist of several modules, this section will however concentrate on the campus module called U-Campus. The system is based on colorized data matrixes and an application server for providing and controlling services. Han *et al.* discusses three implemented services, a service for viewing business cards and multimedia, a messaging service for scanning name plates and a "tour guide" service for displaying info about buildings. The application server is discussed more in next subsection.

2.7.2 Use of backend servers

F. Kitson [38] at Hewlett-Packard, developed a very relevant system named InHand that utilizes RFID. In his paper, Kitson discusses context-aware mobile services using RFID. "Touch" based interaction is discussed, which is the main focus. The InHand project uses a backend server that sends the users data relevant to their context. The PDA used to read the RFID tags, sends its own user id including the tag identification. Date, time and location is also sent. Based upon these contextual identifications, the backend server maps and selects the relevant services. The server is able to personalize the output from the services and Kitson even mention the use of multimedia. The use of user profiles are mentioned, but not explained in detail. One interesting thing that is mentioned by Kitson is the usage of *caching*. Caching can be used to store downloaded data for later usage and thereby saving amount of bandwidth. There are some other facts that should be mentioned. In his discussion, Kitson mentions that push/pull[17] technology is a challenge. He also says that it is difficult to find wide selection of content.

One other project that uses RFID embedded PDA is the mobile weblog system developed by Cheng *et al.* [39]. They discuss a system for blogging spontaneous life experiences from places visited or objects used. In their system, RFID is used to map weblogs against physical objects in our environment. Users are able to write a story about their experience and then upload it to a server. The tag-ID in that particular place or on an object is then used to link the place or object to the story. This done by storing a mapping of all the weblogs associated with an tag-ID in the centralized server. Then, by scanning an RFID-tag at some place, all the relevant weblogs can be found and browsed directly through the PDA.

Vuorinen [40] uses the J2EE platform as the backend server, in a field reporting tool. In contrast to the other projects, Vuorinen uses a mobile phone with an RFID reader integrated in its shell. The mobile client uses GPRS to communicate with the server. The server also accepts Short Message Service (SMS) connections, however it is regarded as a backup. This is an example of a typical architecture that is technically straight forward. The main relevance, however, is the server which implements the

[17] Whether you are "pushed" information or asking for information yourself

J2EE. J2EE is a powerful platform that enables endless possibilities, i.e. cross-platform communication, advanced context processors and filters. Vuorinen mention that this platform can work as a gateway to other backend servers.

The previous section discussed the U-Campus system developed by Han *et al.*. The backend server implemented uses two different databases, one that holds the data matrix codes and another one that holds the contextual content. An interesting functionality of the U-Campus system is the use of a Wifi network provided by the mobile carrier. This removes the costs generated by a GPRS connection and enables the PDA to access a lot of resources. The overall architecture is very similar to the prototype implemented in this thesis, however there are some major differences. The U-Campus system uses a PDA and Wifi connection to the backend server, this thesis uses typical mobile phones with GPRS connection to the backend server.

Sorce *et al.* [41], a research team from Italy, released recently a paper discussing the augmentation of campus environment by using a multimodal guidance system. The system architecture consist of two main components, a PDA and a backend server. The multimodality of the client, integrates voice and speech recognition. This allows users to communicate with the system by speaking to it. Commands to the system are executed by automatically recognising the audio input and sending the commands to the server. The server side, which has a artificial "ChatBot" implemented, is then able to respond to the commands. This two way communication results in an conversation between the user and the artificially intelligent chatbot.

Chapter 3

Technical Overview

This chapter will give an overview and describe the classes and interfaces that enables Java Midlets to access RFID tags. I will start from the top and describe the general purpose classes and down to the more manufacturer specific classes. This chapter is meant to be an easy to understand reference to how the different API's are related to each other. The main motivation for including this chapter, is the lack of documentation for the API's enabling contactless communication. This chapter will focus on the construction of the Mifare 1K tag and its memory organization, which is also very similar to the Mifare 4K tag.

3.1 A Jungle Of API's

There are two main API's that enables RFID connectivity, the *Java JSR-257* and its extension *Nokia 6131 NFC JSR-257 Implementation*. Java specification 257 specifies the Contactless Communication API, which is a collection of classes for accessing various types of RFID tags, visual codes and smart cards. The Nokia 6131 NFC Implementation is an extension to JSR-257, providing classes for accessing certain types of tags. Among these are Sony Felica, NXP Desfire, Mifare Ultralight and Mifare Standard. The extension also provides peer-to-peer functionality. Table 3.1 lists the packages Java JSR-257 Contactless Communication API is divided into, table 3.2 lists some of the extensions.

JSR-257 is a part of a bigger framework called *The Generic Connection Framework* (GCF), which is a set of API's providing support for many different types of connections. GCF is a hierarchy of interfaces and classes mainly in the javax.microedition.io and java.io packages. They are used to create connections such as HTTP, GPRS, Bluetooth and input/output streams. One of the advantages with GCF, is that it is generic. This means that connections can be created in very similar ways. The framework provides an extensible foundation allowing new types of connections to be created in the same manner as the existing connection types.

Name	Usage
javax.microedition.contactless	Provides classes for discovering tags
javax.microedition.contactless.ndef	Provides access to NDEF formatted data
javax.microedition.contactless.rf	Gives access to plain tags
javax.microedition.contactless.sc	Interface for accessing smart cards
javax.microedition.contactless.visual	Used for decoding bar codes

Table 3.1: JSR-257 Packages

Name	Usage
com.nokia.nfc.nxp.desfire	For accessing Phillips Mifare DESFire tags
com.nokia.nfc.nxp.mfstd	Interfaces for accessing Mifare Standard 1K and 4K
com.nokia.nfc.nxp.simpletag	Gives access to Mifare Ultralight tags
com.nokia.nfc.p2p	Interface for using peer-to-peer communication
com.sony.felica	A type of tag mostly used in Japan

Table 3.2: JSR-257 Extension Packages

3.2 Communicating With Tags Using NDEF

To be able to communicate with RFID tags, some key facts has to be known: what type of tag is targeted, where and how the data is stored and how it is authenticated. A communication protocol and a contactless communication link is then needed between the read-write device[1] (RWD) and the target. As JSR-257 builds on GCF, it contains classes and interfaces that extends the *Connection* interface of GCF. A class extending this interface, is also called a *protocol implementation class*. Packages listed in tables 3.1 and 3.2 contains such protocol implementations. Communicating and creating a communication link to RFID tags is therefore a matter of choosing the right protocol. Section 3.2.3 explains how to use the NDEF data format.

As mentioned above, nearly all types of connections are created in a common way. Creating a RFID communication channel has therefore some similarities to creating a Bluetooth connection. Just as with Bluetooth where devices must be *discovered*, tags have to be discovered by the RWD. The roughly outlined Java code below, shows how a Bluetooth connection can be achieved:

Code listing 1: Connecting to a Bluetooth device

```
1)  // start search
2)  // as new devices are found, place them in a list
3)  LocalDevice local = LocalDevice.getLocalDevice();
4)  DiscoveryAgent agent = local.getDiscoveryAgent();
5)  Vector devicesFound = null;
6)  agent.startInquiry(DiscoveryAgent.GIAC, this);
7)  devicesFound.addElement(foundDevice);
```

[1] The read-write device is the Nokia 6131 NFC phone in my case

```
8)
9)  // get a list of services supported services by the device.
10) RemoteDevice rd = devicesFound.firstElement();
11) ServiceRecord[] sr = null;
12) sr = rd.getServiceRecords();
13)
14) //now that we have a list of services, connect to the device
15) String connectionURL = sr.getConnectionURL(...);
16) StreamConnection  streamConnection =
17)         (StreamConnection) Connector.open(connectionURL);
```

Connecting to a RFID tag is very similar. The following code snippet shows roughly how to connect to a NDEF formatted RFID tag:

Code listing 2: Connecting to a RFID tag

```
1)  // signals the RFID hardware to listen to tags.
2)  DiscoveryManager dm = DiscoveryManager.getInstance();
3)  dm.addTargetListener(this, TargetType.NDEF_TAG);
4)
5)  // when a RFID tag is near the RWD, read its properties
6)  TargetProperties[] detectedTarget;
7)
8)  // connect to the tag and read NDEF formatted data
9)  String connectionURL = detectedTarget[0].getUrl();
10) NDEFTagConnection conn = (NDEFTagConnection) Connector.open(connectionURL);
11) NDEFMessage message = conn.readNDEF();
12) NDEFRecord[] records = message.getRecords();
13) print(new String(records[0].getId())+".\n");
14) print(new String(records[0].getPayload())+".\n");
```

In both code listings above, *Connector.open();* is used to open a connection to the remote device. However, how does the Connector interface differentiate between a Bluetooth and a RFID connection? It does that by using strings that identify and describe the type of connection to make. These strings are also called URLs. The "connectionURL" can have various forms, but the general sheme for the URLs is "scheme://host:urlpath;parameters". The "host" field describes the name of the host or device where the resource is located, "urlpath" is the path to the resource followed by optional parameters. The URL to a Bluetooth service located at a host called "ask" (could be a server or a Bluetooth dongle) would then be

"*btspp://ask:85b763c0d92411;name=ExampleService*"

similarly, the connection URL to a Mifare 1K tag would be

"*nfc://ndef;type=mf1k;uid=8a57d584*"

3.2.1 What Type Of Tags Is Targeted

What type of tags that can be discovered depends solely on the RFID reader hardware. Some readers identify several different types while others only support a few. The RFID reader integrated into the Nokia 6131 NFC phone is designed to identify tags based on ISO-14443-4 type A. This international standard defines the communication and transmission protocols for proximity smart cards. ISO-14443-4 is a half-duplex protocol that is divided in four different parts and two different types: type A and B. Their differences is on how power is transmitted to the tags and how the anti-collision[2] mechanism works. The Nokia 6131 NFC user guide [42], lists these supported tags:

- MIFARE®: Standard 1k, Standard 4k
- Mifare Ultralight / Type 2 tag
- Mifare DESFire / Type 4 tag
- Sony FeliCa (non-secure parts) / Type 3 tag
- Innovision Topaz and Jewel (read only) / Type 1 tag
- Cards based on ISO 14443-4 (with or without ISO 7816-4) / Type 4 tag
- NFCIP-1 Initiator

The tags listed above are categorized in *types*, however those types must not be confused with the types defined by ISO-14443-4. NFC Forum[43] is the organization whose work is aimed at standardizing the NFC Data Exchange Format (NDEF). NFC Forum defines four types of RFID tags, which are listed in table 3.3:

Name	Capability	Memory	Speed	ISO Standard
Type 1	Read, re-write and read-only	96 bytes to 2kb	106kbits/s	ISO-14443-A
Type 2	Read, re-write and read-only	48 bytes to 2kb	106kbits/s	ISO-14443-A
Type 3	Either read and re-write or read-only	Variable, 1MB max	212 or 424kbits/s	FeliCa (JIS) X 6319-4
Type 4	Either read and re-write or read-only	Variable	up to 424kbits/s	ISO-14443-A and B

Table 3.3: NFC Type tags

[2]Anti-collision is used to identify tags more precise when more than two tags is in proximity

The desired target can be registered to the *DiscoveryManager*, which invokes the RFID hardware. The DiscoveryManager will then notice the application when the targets is in proximity of the reader. More than one listener can be registered, but that depends on the application. Class *TargetType* defines four targets that is implemented:

RFID_TAG - aimed at tags that have customized communication protocols
NDEF_TAG - must be used when using tags containing NDEF tags. Is also able to read other types of RFID tags.
ISO14443_CARD - used to read smart cards (telephone cards and VISA cards for instance) using ARDU commands.
VISUAL_TAG - Not implemented in the Nokia NFC SDK

The Nokia NFC Forum says that applications should use the *NDEF_TAG* constant, if the targeted tags support and contain NDEF data. The *RFID_TAG* constant can be used for instance to listen to FeliCa tags or Mifare Ultralight tags. It can also be used when manipulating tag content directly through byte streams.

3.2.2 Where The Data Is Stored

To be able to know where the data is stored, one has to be familiar with the memory layouts of the targeted tags. Tags from different manufacturers, have different ways of organizing the data on their tag EEPROM[3]. This section will look at the memory design of the most commonly used RFID tags in the marked, the Phillips Mifare tag.

Phillips Mifare tags come mainly in 1 kilobytes (1K) and 4 kilobytes (4K) rewritable memory. The memory structure is divided into several different entities. These are sections, section trailers, data blocks, manufacturer block and value block. The memory of a Mifare tag consist of several sections, where each section consist of several blocks. Some sections contain more than one type of block. Figure 3.1 show the memory organization of the Mifare 1K tag. Sections can vary in size and may contain from 2 to 16 blocks. The Mifare 1K (1024 bytes) have 16 sectors and 64 blocks, each of 16 bytes. Each sector has 4 blocks, thus a total of 4*16*16 = 1024 bytes. The first sector has two data blocks, one sector trailer block and one manufacturer block. Rest of the sectors have 3 data blocks and one sector trailer. Table 3.4 describes the function of each type pf block.

In each of the sectors, there is one sector trailer block. This is a 16 byte long data area, able to control the access to a specific sector. The sector trailer, consist of three parts; *authentication key A*, *access bits* and *authentication key B*. Authentication key A and B are a 6 byte code and the access bits use 4 bytes. To be able to preform memory operations the RWD must authenticate itself. The

[3]Electrically-Erasable Programmable Read-Only Memory

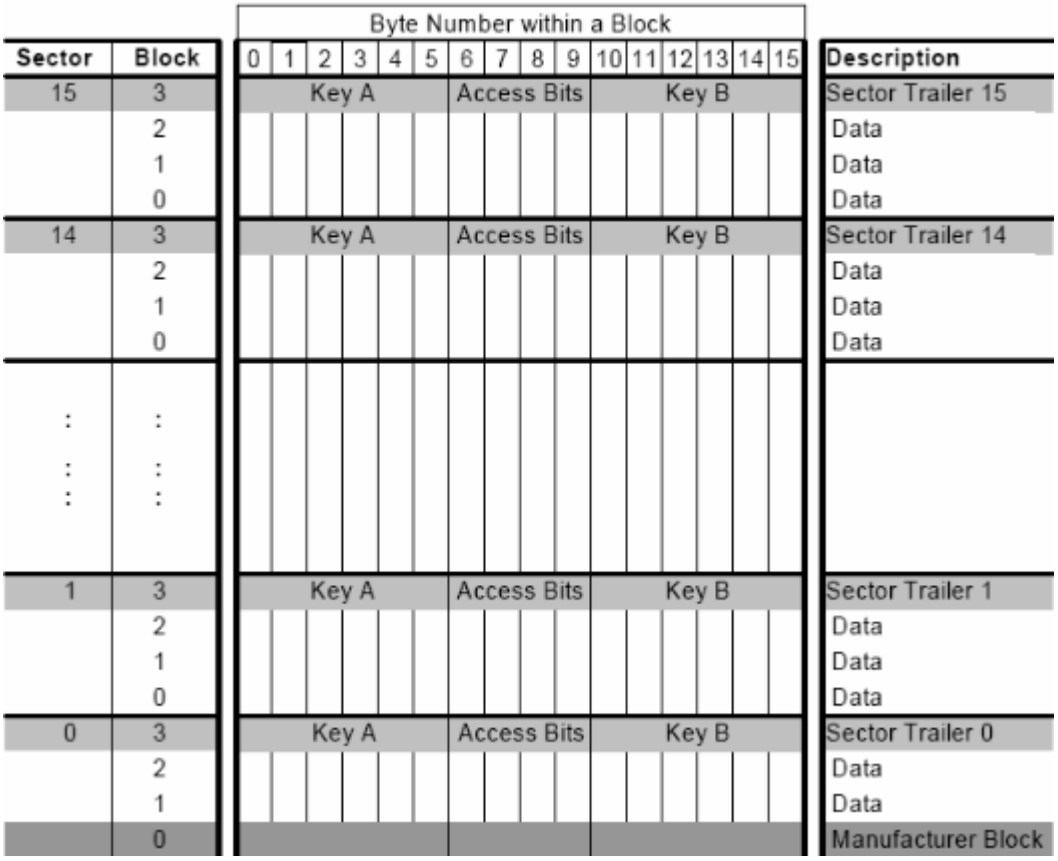

Figure 3.1: Mifare 1K memory organization

authentication process is carried out by providing the corrects keys. New Mifare tags are, however, configured to use a standard set of keys, which are the hexadecimal code 0xFF-0xFF-0xFF-0xFF-0xFF-0xFF. The standard key has to be changed by the RWD in order to write-protect the contents of the tag. The authentication process between the RWD and the tag is out of the scope of this thesis. Interested users are therefore recommended to read the Mifare 1K product sheet[44].

By skipping all the sector trailers, including the first sector, minimum

*15 sectors * 3 data blocks * 16 bytes* = 720 bytes

data blocks area available and can be used for user's data. 304 bytes are then left for security configuration and manufacturer data. One interesting thing I discovered while testing the Mifare tags, is that the integrated NFC transmission application in Nokia 6131 always writes the data starting from sector index 1 [4]. The transmission application does not check for existing data, which will therefore be

[4]Sector index 0 is used for manufacturer data

Name	Function
Section	A grouping of block.s
Sector trailer	Controls access permissions
Manufacturer block	16 bytes of one-time-writable data area for manufacturer specific data.
Data block	User data, rewritable. Sector trailer can be used to lock this data area.
Value block	This block type can used to directly manipulate stored values. Values can be incremented, decremented, read and written.

Table 3.4: Description of entities

overwritten every time new data is transmitted.

This was revealed by first using a customized midlet, to write 16 bytes of random data to sector index 1 and 4 bytes to the first data block in sector index 2. The integrated NFC application in Nokia 6131 was then used to write a NDEF text message to the same tag. An examination of the current memory status of the tag showed that the integrated NFC application wrote less than 16 bytes starting from sector index 1. The rest of the sector still contained parts of the previously written data, thus making it corrupted. The first data block in sector index 2, which corresponds to data block index 12, was not overwritten. This little test has, therefore, showed that data can be distributed over several data blocks and even sectors. There is no information about this in the Nokia 6131 NFC phone documentation. It is however mentioned with only one sentence in the Java documentation of class *ndef.NDEFTagConnection*.

Information from the memory of the Mifare 1K tag can be extracted in several was. However, the type of information that is extracted depends on how the application uses the Nokia NFC API. If the user application is set to read NDEF formatted data then only NDEF messages are read from the memory. Plain data written manually, that is, data that was not written as NDEF, must also be read manually by specifying the location of the data. The mixing of both NDEF data and manually written data poses some challenges for client applications. When writing custom applications, it is important to find out where the first empty byte is placed, how much data is needed to be written and how much data can be written before existing data is overwritten. A further discussion about handling the data area, is given in section 3.3.

3.2.3 How The Data Is Stored

This section will look at two ways of storing data on the memory area of the Mifare 1K and 4K tags. These two methods are: storing as NDEF formatted data and storing data in a custom format written directly to the tag. Instead of giving a description of the complicated specification of NDEF, I will give a roughly outlined overview of the NDEF components and relate it to the prototype that will be discussed in chapter 4 Design.

The previous section highlighted some key facts that a programmer of RFID tags needs to be aware of. How free space can be used optimally and how data should be treated all depends on how the data is stored. *NDEF messages* encapsulate one or more *NDEF Records*. There are several different types of NDEF Records, each of presenting a different kind of data. A record can for instance be a plain text message, an Internet link or an image. The two most used record types are *NFC Forum Well-Known-Type* (NFC-WKT) and *Media-Type*. NFC-WKT has three sub-types, these are *Smart Poster*, *Text* and *URI*. Table 3.5 describes these three types. It is an advantage in using NDEF, since it

Name	Function
Text	Plain text message
URI	Links such as *http*, *ftp*, *tel*, *mailto*
Smart Poster	As URI. Actions can be attached, ie. sending SMS, opening a web browser.

Table 3.5: Description of NDEF types.

is being standarized. Using a standarized format not only allows custom applications, but also future applications to be able to read the data. However, using NDEF also poses some technical challenges. NDEF Records can be chained and combined. An URI record could for instance be combined with a Text record for describing the resource. A NDEF Message can hold several records, but there is no API support for several Message-objects on a tag. This can be supported in a customized application, but then only one Message can be read by NFC applications. The easiest solution will be to store multiple records on a single Message and presume that they do not depend on each other. Multiple records allow a prototype application to use a FIFO stack, which places records in a queue. An example usage, is when several comments about a place is needed to be stored on a tag.

A Message is constructed by several small pieces which can be summarized as

[start] [header] [body] [ending]

The start field is a one byte identification marking the start of this Message. The header field is a series of bytes describing the body content. These bytes are hexadecimal protocol codes, which are described in detail in the NDEF specification [45]. How the header is built depends on the type of record and the length of the body. Ending is a one byte field marking the end of this Message. Here is an example of a Message with several records

[start] [header] [body] [header] [body] [ending]

Table 3.6 shows the structure of a NDEF Text Message holding the phrase "Hello, World!". The example is a reprint from the NDEF specification. Offsets zero - three, are the NDEF headers (as mentioned above). Offsets four and five, are optional payload identifications. These identifications can be used for internationalization purposes. The final offset is the content itself. UTF-8 is used for encoding the

content, which means that each ASCII letter uses 1 byte. Code listing 3 shows how to create such a Text message. The optional language identification is omitted.

Offset	Content	Explanation
0	N/A	IL flag = 0 (no ID field), SF=1 (Short format)
1	0x01	Length of the record name
2	0x10	The length of the payload data
3	"T"	The binary encoding of the name
4	0x02	Status byte: This is UTF-8, and has a two-byte language code
5	"en"	"en" is the ISO code for "English"
7	"Hello, world!"	UTF-8 string "Hello, world!"

Table 3.6: Structure of a NDEF Text Message

Code listing 3: Creating a NDEF Text Message

```
try {
      String url = detectedTarget[i].getUrl(Class.forName(
                  "javax.microedition.contactless.ndef.NDEFTagConnection"));

      conn = (NDEFTagConnection) Connector.open(url);

      NDEFRecordType myType = new NDEFRecordType(
                        NDEFRecordType.NFC_FORUM_RTD,"urn:nfc:wkt:T");
      NDEFRecord myRec;

        myRec = new NDEFRecord(myType, null, "Hello, World!".getBytes());

      NDEFRecord[] myRecArray = new NDEFRecord[] { myRec };
      NDEFMessage myMessage = new NDEFMessage(myRecArray);
      conn.writeNDEF(myMessage);
      screenText.setText("Data written");
      } catch (Exception e) {
        e.printStackTrace();
        screenText.setText("Error: " + e.getMessage());
      }
```

The same text in hexadecimal representation as written to the tag memory is shown below:

03 - 11D1010D54 - 48656C6C6F2C20576F726C6421 - FE

This data uses a total of 20 bytes of space on the tag. *0x03* and *0xFE* is the *start* and *ending* identification, which marks the start and end of the message. The header field follows almost the same pattern as in table 3.6. *0x11* and *0xD1* are called control markers, *0x01* equals to offset one, 0x0D equals to offset two and indicates that the size of the payload content is 13 bytes. Offset three, *0x54*,

is the hex code for the letter "T". The latter hex code is a binary encoding of what type of content the data is. In the same way, offset three would be *0x53-0x70* (the hex code for "Sp") if the content was a Smart Poster record and *0x55* if it was an URI record. However, the NDEF specification says that Smart Poster RTD (Record Type Definition) is an extension of the URI RTD. It is also possible to store Internet bookmarks and other type of URI's as a Smart Poster records. Finally, the hex code starting with *4865...6421*, are the hexadecimal representation of "Hello, World!".

3.2.4 How It Is Authenticated

Mifare tags provide advanced authentication and encryption algorithms. However, it is up to the user to activate these functions. There are no protection on the tags by default, so both reading and writing works without the need to activate the protection. I will for that reason, not describe authentication procedures. The reader is therefore advised to consult the Mifare 1K specification [44].

3.3 Using A Custom Data Format

Writing NDEF Messages using the Nokia 6131 NFC SDK implies the construction of a single NDEF Message. Most of the complicated work regarding writing is done behind the scenes by the SDK. Using a custom format contributes to enabling application specific functionalities, for instance a list or a queue. The objects can also be designed in a way that gives much more control, compared to the limitations on NDEF objects. This section proposes a simple format, which has some commonalities with the construction of NDEF.

The format uses few components and can be used in any type of service based information system. The format can be outlined as

[service type][size][[data field]...n]

The first field, *service type*, is a one byte service identifier. This will enable the identification of up to sixteen services. *Size* is the number of data fields. *data field* can be illustrated as

[start][data][end]

Each starts with a start marker and ends with a end marker. The data can be of any length within the boundaries of the memory size. Using a custom format means that writing has to be done manually. This involves finding out how many data blocks are needed. An example of how to do this is shown below:

```
1) convert string value to byte array
2) check that length of array is below total memory size
3) divide array length in 16 and round up to find total number of blocks
4) create a new array of length found in step 3
5) fill the new array with the data and fill the rest with 0
```

Even if the overall construction reminds of NDEF, there are a few differences. An NDEF Message can have multiple records, however, each record has its own header. The records must also be related to each other. The data fields in the customized format, do not have their own header and do not have to be related. Having no extra headers saves space, but limits the number of data types[5]. It can be advantageous to do so if the targeted application or system does not require multiple types of data. The type of data can however be service specific.

One of the main design goals for the structure of the format, is to enable data fields to be linked and queued in an easy way. To make a queue effective gaps have to be avoided and the number of data fields lowered. One has to remember that queues can be very effective when used in computer software. However, if the tag memory is full pushing and polling the queue on the tag memory can be a time consuming operation. If the number of data fields are kept low, another challenge can be avoided that is the new data field starts position. The structure of the format can then be changed into

[service type][size][block id...n][[data field]...n]

The structure above identifies the total data amount and the first available block id. *Service type* field and the *size* field is written to the first data block of sector two, each consuming 1 byte and 3 bytes. The remaining 12 bytes can be used to identify block id's where new data starts or simply the first free block. It is assumed that the queue is limited to maximum 10 objects, hence, 70 bytes pr. object.

3.4 Summary

My own tests have shown that a standard NFC software only reads the first Message object when multiple objects are stored on the tag. Writing and reading multiple Message objects, requires, therefore, advanced traversing of the memory. I also saw that new tags do not need authentication for writing to the memory. When writing to the tag using the Nokia software existing data is overwritten if the memory needed for the new data is more than its available memory. To prevent gaps and corrupted data on the tag memory data must be marked with start and end markers. If the user needs to write plain data to a tag then it is better to use the NDEF Text record type. Mixing plain data with NFC formatted data is not recommended, because of the risk of loosing data.

The reading speed, which was given to be 424 kbit/s, turned out to be not more than circa 360 kbit/s in my own tests. This chapter has also shown how to use JSR-257 to connect to and write a NDEF Message to a Mifare tag. This was done because of the poor documentation of the Java standard.

-

[5] Data types such as images and links

Chapter 4

Design

This chapter will describe the design process of this thesis. The prototype is named Proximity, as proximity is the key factor. The design process has been iterative and different design methodologies were combined. I heard about RFID the first time during my work on the Bluetooth Proximity project [4]. Me and my fellow project member, concluded at the time that RFID could ease implementation of a context aware information system. I then had informal talks with my supervisor, who guided me with resources. With no prior knowledge of radio frequencies or visual tags, I introduced myself to RFID by experimenting with the programming libraries. Agile design methodology allows designers to develop and implement systems according to the customers changing requirements. Wells [46] say that: "There are many small pieces. Individually the pieces make no sense, but when combined together a complete picture can be seen". This also applied for me as I researched the possibilities of RFID. The classical waterfall method was also used as each step of the design process was important for the next one. This chapter will first present scenarios, then an analysis of the campus and its existing services is discussed. Finally, requirements and the system architecture is explained.

4.1 Assumptions

Designing an information system for an environment like a campus is not a small task. By combining different technologies, information systems far bigger than the scope of this thesis can be developed. I have to limit the scope of the project and make assumptions that might not have been made in a production quality system. One of the most important limitations is on the graphical user interface side. The prototype that I want to create does not focus on human-computer interaction aspects and will be as simple as possible. The requirements for the prototype is mainly focused on the scenarios and I will try to identify them using different design techniques described in the following sections.

4.2 Scenarios

One of the first steps before starting the design process was to create scenarios for the possible use of RFID in augmented environments. Scenarios are helpful for showing future usages and to extract application requirements. This section will present four scenarios, which each will be described in the following sections.

4.2.1 Library

In today's modern libraries, there are quite a few ways to borrow interesting books. There is the library employees, who's job is to assist and lend you the books of your choice. There are Internet resources, for searching the locations of books and there is automated machines, which enables students to register books themselves. However, in an augmented library students use another way to register the books they want to borrow. A student finds the book of interest and then simply touches the book with a mobile device to register it as loan. By touching the book information about the book is shown on the mobile device. This information can be for example the duration of the loan and the books information.

4.2.2 Booking Rooms

Imagine a large university campus with several buildings spread around a city. It is not unusual for such universities that have many meeting and studying rooms and to have an electronic booking system. A student that needs a meeting room on one specific day searches through this electronic system and eventually reserves a room. In this case, the student knows the time and the room id to look up. On another day, a group of students unexpectedly need a room. While going past the rooms in the building, they finally find a room that fits. One of the students then uses his mobile phone to touch the name plate on the door to check if it is available. The mobile then returns information regarding the number of hours it is free and what type of room it is.

4.2.3 Information Boards

A lot of different types of information needs to reach the students of a University. The information is distributed through the mail system, via LCD screens, via forums and message boards. In all these channels the information is pushed to the users. A new type of message board has been developed and placed near the entrance at the campus. Then one day a student sees an announcement for a event on the board. The student then touches the announcement with the phone and instantly registers for the event. The event is also registered in the calendar so the user is reminded of the event.

4.3 Services Paradigms

This section analyse some common information service paradigms and relate them to Proxema. When developing and using services for mobile information systems, the services has to be presented in some form. There must be a way of interacting and communication to the user. Three paradigms that are identified and which are central in Proxema, are presented in the following subsections.

Pull

Pulling is the classical way of getting information. The user asks for a service, provides some criteria and gets the service or some information delivered back. Requesting more information about a book in the library is one example of pull. The services that will be provided by Proxema requires scanning a tag. Most of the services are therefore categorised as pull services, since this requirement is on the user.

Interact

Users are often able to directly communicate and interact with services and also change the contents of the service. An example of such a service is for instance in intelligent environments, where mobile equipment can be used to regulate temperature or lighting. In Proxema, one such service could for instance be room booking. As described above, this requires the user to scan a transponder attached to a room. The user initiates the service by first pulling the room id and then the service is changed into an interactive mode.

Publish

Publishing is a service that enables the users to make information available for the public view. Leaving a note about a location or a room using Proxema, is one such service.

4.4 Services Types

Point-of-need

Delivering information related to the users whereabouts, is called a point-of-need service. This type of service is for instance implemented in RFID augmented museums. Other examples are information about weather, traffic, restaurants or hotels near an unfamiliar place [47].

Path Description

Path description services could be used to find information about how to travel from the users location to another location. Such services exists for instance for cars and can guide the users to the location. An example of one such service using RFID in campus was the navigation system presented by Willis *et al.* [33].

4.5 Design Methods Used

A preliminary analysis was done by examining the phone and the RFID hardware. This analysis is documented in chapter 3. Finding the possibilities and limitations of the hardware is important for finding feasible requirements for the prototype. The next phase was an analysis of the users and the existing services. In this step, I had to investigate the existing services and the user patterns existing in the campus environment. Design methods had to be prepared and executed. For this project, I choose to use questionnaires and interviews for gathering data. These two approaches allows me to gather quantitative and qualitative data. Involving the users in the design process is important for any development project. I decided to do this by using design sessions. I call it a session, because of the limited time I had at the disposal rather than doing fully design workshops. Through the design process, I will merely focus on three areas of the campus which is outlined in the scenarios. These are reservation of groups rooms, library and the information boards. As part of the preliminary analysis, I prepared a design proposal. The proposal which is presented in section 4.6.3, was used as a basis in carrying out the design methods. The first design will be evaluated and modified in the next iteration, which will be the requirements analysis. This section will describe the design methods I have used, as well as how and why I did it.

4.5.1 Questionnaires

What: A questionnaire is used to get answers to specific questions from a large group of people. It is a challenge to develop good questionnaires, badly articulated questions can lead to misunderstandings and ambiguities. The questions can either be open or closed. Open questions suits best for exploration and often lead to qualitative answers. Closed questions are used when a researcher knows the possible answers. Choosing between open or closed questions depends on how much the researcher already knows about the topic of investigation. However, the questionnaire has to be designed according to the context, type of users and type of data needed [48].

Why: Frankly, radio frequency identification is unknown for most students. Compared to a city like Oslo where radio frequency identification is seldom used, I have the advantage of that the province of Østfold uses electronic bus cards which is in fact based on RFID. With this in my mind, I wanted to

investigate how students at Østfold University College felt about the introduction of mobile services using RFID. Among other reasons, I can reach a large group of students with little effort and give them opportunity of total anonymity. Using a questionnaire at this stage, can help me identify services that the users find useful.

How: I developed an online questionnaire with ten questions and a short introduction at the beginning. My primary mission in using the questionnaire, was to find out who were using the existing services at the campus. An email with a short description and link to the questionnaire, was sent to everybody at the school. All the user submissions are stored in a database. Statistics are then drawn out, which makes it easy to analyse patterns among the proposed services and user habits.

The questionnaire is divided into three parts. Part one collects information about the person filling out the form. Part two ranks RFID services and part three is asking about existing knowledge. Initially, I asked for the users gender and age range. The person is then asked about his/hers usage habit best pursuant to an existing service by using guided words. These selectable words were *never, sometimes, often* and *very often*. In part two, the person is asked to rank a number of example services possible with radio frequency identification. The options a numeric interval between 1 and 6, where 1 is bad and 6 is great. Then I chose to have two text areas, were the user can enter his/hers opinions about RFID augmented services. The questionnaire is ended with three statistical questions about the type of technologies and services the user is already familiar with.

4.5.2 Interviews

What: An interview can be seen as a conversation between two or more people. The conversation has a purpose and a topic that the interviewer has researched about. In short terms, an interview can be seen as asking users questions about certain things that the interviewer want more information about. There are different approaches in doing an interview. There are unstructured, structured, semi-structured and group interviews. Choosing the most appropriate approach depends on the purpose of the interview, the type of questions and the stage in the design process [48]. Interviewing is a demanding method, as the interviewer has to use both human and professional resources. Interviewers must map the situation of the interviewee and know how to speak to be understood correctly. Using an unstructured approach, allows an explorative conversation. Unstructured interviews are often very open and the interviewer does not have any expectations about the answers an interviewee can answer. It is therefore a good approach to gather opinions and impressions. Structured interviews on the other hand, results in conversations which are controlled by the interviewer. The questions are prepared before the interview with a set of possible answers. This approach resembles questionnaires. Semi-structured interviews are a combination of unstructured and structured approaches. Group interviews are considered when a department or organization has different sections.

Why: I used interviews since the prototype is meant to be used by a large number of people inside the campus. I had the choice of either doing one-to-one interviews or group interviews. I choose to use group interviews since the campus consist of a number of different sections and user types. I also used interviews as a qualitative method, for gathering user opinions. Another reason for using the group interviews is that the participants often use each other and, hence, raise diverse issues to be discussed. This was important as interviewees are different, some may talk a lot with a rich vocabulary and some may have difficulties expressing themselves.

How: I choose to use group interviews with an unstructured approach. Before interviewing the people I wanted to talk with, I made a plan for topics that I wanted to cover. Some open questions were also prepared and were used to start the conversations. I also wanted the interviewees to leave with good impression. To do this, I planned to divide the conversation into three stages. First I wanted to discuss how the interviewees were working and then concentrate on the existing services at the campus. In the final stage, I wanted to discuss my prototype and end the interview with the advantages of the planned prototype. I used a voice recording device throughout the interviews to be able to listen to the conversations later. The recordings were used to transcribe and analyse the interviews. The analysis was then used to identify requirements for the prototype.

4.5.3 Design Sessions

What: A design session can be described as a meeting were two or more people use different techniques to collaborate on the interface design of a product. Sharp *et al.* [48] mention high and low fidelity prototypes as being two techniques for involving users in the design process. Low fidelity prototypes such as storyboards, sketches and card based prototypes and high fidelity prototypes such as highly interactive visual user interfaces. A design session, also called a workshop, is a participatory design method that includes real users in the design process of graphical user interfaces. This is an important method, as real users often see things from an other perspective than the designers. The technique I used is based on a method called PICTIVE[49][50]. In this method, the user is given some props like pens, highlighters, paper, Post-It notes, stickers and labels. The user has also printed paper images of dialogs, windows, buttons and menus. These props are used to make a paper mock-up of a system.

Why: Design sessions are primarily used to involve the users in the development process and to give them an idea of how the prototype could look like during the interviews. I had sketches and drafts of the prototype before the interviews, however, I needed to discuss them with the users. Because of the short amount of time I had in disposal, I choose to do this right after the two interviews. It has to be mentioned that these sessions were the props were used as aids in the discussions.

How: I had two interviews with four different people. The first hour of the sessions were done as I described in subsection 4.5.2. In the last half hour, I gave the participants print outs of the Nokia 6131 NFC phone and the props. I then started the session by telling the participants how I planned to design the prototype. As the discussions continued the participants made comments about the drawings I had made in advance. Figure 4.1, shows an image of the props I had prepared for both of the sessions.

Figure 4.1: Preparation for design workshop

4.6 Analysis

4.6.1 Interview One

The first interview was done with an employee and a student. Person one has recently finished his master degree and is now working as a scientific assistant at Østfold University College. Person two is a master student in information technology and is finishing his master thesis. Person one and person two were chosen for this interview as expert users. Person one was especially chosen, since he is both an employee and a former student. His expertise combined with his experience of being a student at Østfold University College, was valuable.

In the first part of the interview we discussed how the interviewees worked and their usage pattern of the existing services. Their answers revealed few differences in the usage patterns either being a student or an employee. I asked person one if his usage patterns had changed after he became employed and to this question he answered: *Yes but I don't think that has to do anything about being an employee, but rather about what I am doing at all times*. He also mentioned that while writing his thesis he used the library services much more after than being employed. Person two also confirmed the same pattern.

Person one implies a dynamic workday and this also might be true for other employees of the campus, as several of the employees in scientific positions also do research alongside teaching. One interesting question then arises, does the everyday activities of a student also effect the usage pattern

of the campus services? As a college student there will always be an information need. The needed information in general, is more or less always connected to a course or the study program. However, making the students work day too uniform reduces the use of several services available at the campus. This is a challenge, whether or not the campus makes use of RFID.

In the second part of the interview, I concentrated more on the existing services. When I asked the interviewees about the LCD-screens inside the campus building, they answered that they never read the information on the screens because of the little relevance and interest to them. To my question about augmentation of existing services person one mentioned several facts. He said that the email system could also be seen as an information service and that many students simply do not read their mail. The interviewees agreed that there is too much of unnecessary information in the existing services and that an augmentation of services had to focus on relevant information for the user. Person one also pointed out his concerns about privacy in RFID augmented services.

After discussing the existing services, I presented the three main services that this thesis focuses on. Both of the interviewees gave positive feedback about my ideas, person one pointed out, however, the importance of services being augmented and not replacement. At this point in the interview, I asked the interviewees what type of functionality that can be connected with the services. The interviewees mentioned

- reservation of and information about a room
- get information about and references of a book
- adding notes about books on the phone
- connecting to a service like Amazon for reviews

The second part of the interview ended with a very important question to person one as an expert user: *Has RFID possibilities and a future in a campus, rather than only being used for logistics?* The answer from person one was:

Yes most probably, but that is something I have not gone into depth of considering yet. However, I am thinking more about the privacy issues. Whether or not a service is augmented, concerns also about how it is implemented. A service can be very useful if the service is implemented correctly. But if it is implemented incorrectly, it may spoil something that is already good. If a person has to click through several menus, just to for example reserve a group room, then it will be to complicated and few people will use it.

The answer above, indicates that developers should carefully approach both technical and visual aspects of the prototype. The privacy issues will be most important in a prioritized order. A number of countermeasures could be used to address the privacy issues. These will be mainly on the software side, as short range tags do not pose a threat in contrast to using long range tags. However, it is a necessity to *lock*[1] short range tags to hinder rewriting. Further, the software has to be automated to

[1] RFID tags can be configured to be write-once-read-many

hinder a deep menu structure and too many clicking. It is not only internal functionality that needs to be automated, but also the external environment of the software. In other words, the software always needs to be available and preferably running in the background of the phones operative system. In his answer, person one, also mentions correctly implemented services and poorly implemented services. One factor that decide the difference between them, is how well a good idea or valuable information is presented on the software.

4.6.1.1 Summary

Interview one revealed that the existing services has some drawbacks. Both interviewees confirmed that they never read the information screens placed around the school. The email system is an important information source, but too few students actually check their emails regularly. The usage pattern of the existing services is similar among employees and students, however I saw some differences. Which services that is being used and how much they are being used, depends on what an employee or a student is doing at all times. The employees are more conservative and use only the services that they need to use. The interviewees pointed out the importance of making the prototype simple, so the users do not have to click too many times.

4.6.2 Interview Two

The second interview was done with two representatives from my campus library. I will call them person three and person four. The main responsibilities of person three includes a wide range of tasks. He is an educated librarian and is the contact person for the IT department, maintains the library web pages and maintains Bibsys[2]. Person four is the general manager of the library and also a contact person for several study programs on the campus. Her main responsibilities are teaching and contacting with different departments at Østfold University College. This section will summarise and discuss the main topics of the interview.

Current Library Services

To be able to augment existing services, first, one has to understand the services and get an insight of the processes. In the first part of the interview, I asked, therefore, the interviewees about how they worked and tried to get an overview of the operation of the services. The interviewees provided me with detailed information that this section presents.

The librarians at the campus provide library services for the students and the employees. The services are mainly concentrated on books, which means that books have to be bought. When they arrive at the library, the books have to be classified and catalogued. Then necessary stickers have to be placed on to the books and at the end the book should be placed on the correct bookshelf. However,

[2]The global library web system for Norwegian universities

it is not only books that can be found in the library, magazines, papers, thesises and periodicals can be found too. The old periodicals have to be removed and replaced with the new ones. One of the librarians most important tasks is to help people find the information they need, printed or not. The librarians do this, by providing courses and user training on information search.

The library has also databases and web pages that need maintenance. One of the steps to do this is by keeping the library updated on what the departments do. A selection of employees at the library have therefore divided themselves among the departments, where each employee is a contact person. Contact with the departments are important, as they update the librarians about what type of literature and books they need.

The interviewees emphasize that the most important service, is still in front of the service counter. The librarians focus a lot on visibility and availability, as library services are organized as a support service in the organization chart. This means that there should always be an available librarian at the counter to actively support students and employees with their searches. Most often these searches are for books and lending books are therefore a common activity. This activity is taking place at the same time of doing the other tasks.

Usage Patterns

A campus library differs from an ordinary library in that it focuses more on the students. As a result the usage patterns are therefore naturally different. Required course reading materials are for instance lend out more often at the start and at the end of the semester. Usually, several students ask for the same materials.

The interviewees told me that the number of students using the library increases and that they follow-up the cooperation with both local and external departments closely. The library has, however, statistics concluding that large sections of the library are not being used. One might ask whether the number of students using the library affects the services. For that reason, I asked the interviewees if it could be possible to introduce augmentations that might attract more students. The immediate reply I got was that it depended much on economical factors. However, person four underline that the librarians could focus more on advanced information searches, if students were trained to do the simple searches themselves.

An example of an advanced information search, is for instance a problem statement for a masters thesis or complicated questions that has not been covered by any specific book. This type of information search is important and is a ability unique for librarians. In my opinion, human interaction is better when somebody is looking for non obvious information. Advanced information searching is therefore overlooked in this thesis.

The interviewees said that there should be a point in being at the library and that they wish that more people used the library as a workplace. Being at the library does not mean that one should be reading books for hours and hours. Very often, a book is not the answer to a specific question either.

It is, however, the acknowledgement of that it might be something you do not know about, but which can be useful. The interviewees pointed out that they try to remind students about this point and that library services could actually improve results.

Person three say that the general impression of the library is that it is so closely attached to *books* and not *information*. A large number of students do not know about online services even after three and four years of study. The importance of what person three is saying here will also have en impact on the design of my prototype and could apply to any type of service within the library. The awareness of the information value itself, is more important than practical services.

RFID In The Library

My impression from the beginning to the end of the interview was that the interviewees based their answers on the economical situation. To get an idea of how they would use RFID in the library, I asked the interviewees whether or not RFID would bring advantages. They replied that RFID would bring advantages and they also gave a few examples of what they could use it for. One example they mentioned, was remote scanning of books in opposite to the "touch" mentality. However, their ideas implied using active transponders with longer reading ranges. My thesis limits itself to the use of passive transponders and their ideas will, therefore, be possible in a future iteration.

Among other functions we discussed was how to use the phone for borrowing books. Person four mentioned using the phone as a library card. This is possible as the phone have a built-in RFID transponder which is writable. I proposed using the phone to actually registering a book as loan to the owner of the phone. The interviewees pointed however out a few problems with this type of service. In the current system a book has to be demagnetized. While the book can be registered, it cannot be demagnetized by the phone. Further, there is a authentication challenge. Obviously several safety issues arise without the authentication, so it has to be handled carefully.

Authentication details, including the personal profile of the owner, can be stored on the phone. This profile could then be used to filter contextual information to the user. As I told these ideas to the interviewees, they underlined one very important fact for me. This is the fact that filtering information can lead to removing information that could have been relevant to a given context. An example of this problem is for instance when the user's profile state that the user is an IT student. By only showing IT relevant books it is very easy to skip books that might have become useful in other academical areas. Person three mentioned that inputting relevant study areas in the user profile could be used as a countermeasure.

We also talked about how the shelves could be used since there are fewer shelves compared to the number of books. One idea that was mentioned was to identify the location of a book based on the id of the shelves. This idea is technically possible, but since the shelves already have describing signboards one might argue whether or not RFID really will provide augmentation. For that reason, I focused more on the books and asked for the type of information that should be linked to them. Person

three explained to me that the prototype could simply be connected to Bibsys. Doing this allows the application to have access to all the information that exists about a specific book. A nine digit code is the only requirement to finding information about a book in Bibsys. Attributes such as title, author, loan term and topic can viewed if the application should view information without connecting to Internet.

4.6.2.1 Summary

The second interview, helped me understand why the *information* is so important in an *information system* and not the technology itself. Focusing on the quality of the information, will increase the value of the prototype. I have found out that RFID can be used in several different ways to augment the work of the librarians. A lot of the uses I learned about in the second interview however, requires longer reading range than the transponders I use in this thesis. Another challenge is security inside the library and user authentication. Having these challenges in mind, I found out that it is better to focus on simple services with useful information rather than designing functionality with less value.

4.6.3 Design Sessions

This section will summarize the last part of interviews one and two, which can be seen as a small design session for the interface I am planning to develop. Figure 4.2, shows an image of a drawing that illustrates the start-up screen of the prototype. This image was used as an entry point to the discussions, as I started to explain the interviewees the design I had in mind. In addition to figure 4.2 which was made in advance, I showed the user a mind map. The mind map, shown in figure 4.3 was used to give the interviewees an idea of the type of functions available and a simple navigation pattern.

4.6.3.1 Design Session One

I started by explaining how the software will act upon scanning a tag, which was similar to the discussion in the last part of interview one. The interviewees immediate answers were positive, however, they suggested some additional functionality to the prototype. In addition to *adding* events, person one mentioned the possibility of adding custom notes about an event. Person two mentioned search possibilities, when and if the user has many events stored. Among other library services we had discussed, the interviewees proposed a functionality for collecting information such as references.

The last discussion I had with the interviewees was about the menus. In my first design of the menus and form layouts, I proposed to use one menu to hold various functions and one button on the opposite side. This button is for the most important action within the three main areas of the prototype. As an example, I said that this button could be *add* for the library service. The interviewees responded that there might be smarter solutions. The keypad could for instance be used for shortcuts, combined with the button. The action behind the button, has to be chosen carefully, since one action might not

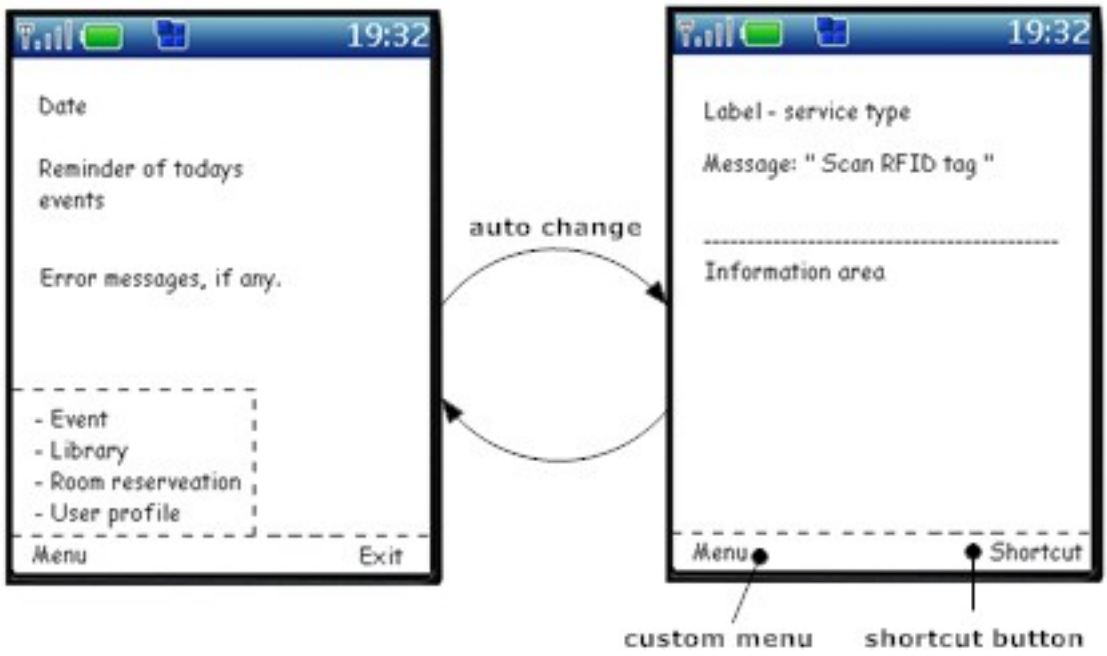

Figure 4.2: Illustration of prototype proposal

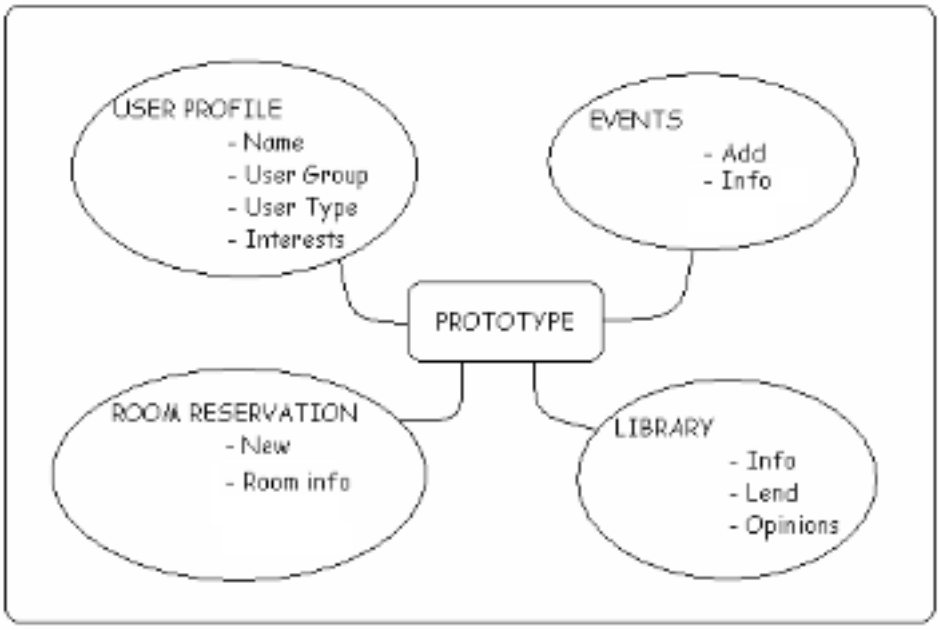

Figure 4.3: Mind map

be as important for one user compared to another user.

4.6.3.2 Design Session Two

As in design session one, I used figure 4.2 and figure 4.3 to start the discussions. In session two, I started by talking about how I was thinking to implement the services that this thesis focused on. I also said that the application could either be developed simple or advanced and that I planned to do it simple. When I told the interviewees that the prototype could be used to read other peoples opinions about books etc., I got a surprising answer from person four. She said the exact same thing as person one told me, which was the possibility to connect the prototype to Amazon.

The session continued with a discussion of how the menus could be designed. Just as in design session one, I proposed to use a menu on the left hand side and a button on the right side. As an example, I told the interviewees that the button on the right could be called "Borrow", however, the interviewees replied that "Info" would be better. The reason for their choice was that to know whether or not to borrow a book one needs information about the book to be able to decide if its interesting. They also underlined that presenting useful information was more important than actually lending the books.

While watching images of the prototype, person four got another idea of a functionality. She said that it could be nice to automatically extract reference information about the scanned book and then be able to transfer it to a reference list. This idea was also mentioned in design session one.

4.6.3.3 Summary

In both design sessions there was an agreement of using either only one menu in the user interface or one menu with a shortcut button on opposite side. The shortcut button should only show info, since other types of functionality might not fit for all user types. In design session one, it was mentioned functionality such as adding custom notes and search possibilities. Another common finding in both design sessions, was the possibility of extracting reference information from books. All the interviewees also agreed that the interface should be simple. In design session two, we also talked about that advanced interfaces should have a certain usage value.

4.6.4 Questionnaire

This section will present an analysis of the results from the online survey I had in my campus. I got a total of 225 replies to my online questionnaire. Among those, 112 was females and 113 males. As table 4.1 shows, the number of respondents decrease as the age increases. This pattern is linked to the number of students versus the number of employees. However, it is still interesting to that there is such a small amount the employees that actually respond. One should also remember that their might be students in higher age intervals.

When looking at the results of the hole survey, figure 4.4 show that the *Events* service is ranked best. It is, however, not so big difference in the rankings in of the services. The service that is ranked poorest is the *Publish user opinions* service. There is apparently not much interest in writing user opinions about books and different places of campus. Taking into account that it was possible to rank a service in the range of one to six, it is interesting to see that almost all the services scored well.

Table 4.5 is a list of how students and employees of different ages ranked the services. Figure 4.6 show the table as a two dimensional figure. People who rank the room booking service best, is between 36 and 45 years old. This might be because of the fact that many employees is in this age interval. However, there is very little difference among the age intervals. Another interesting pattern that is visible is that people between 46 and 55 years old rank the services poorly compared to the others.

I the questionnaire I also asked the users if they had heard about RFID. Of the 225 responses I got 115 said that they did not knew it, 40 people said that they had heard about it but did not know what is was and 70 people responded that they knew about it.

Age	Total
18-25	111
26-35	70
36-45	27
46-55	12
56-65	5

Table 4.1: Age distribution

In my questionnaire the users were able to comment other uses of a RFID phone and their requirements for the prototype. A lot of the comments says that it should be very cheap in use or totally free and the user interface should be very simple. Among the ideas for other uses of the phone I got was

- unlocking closed doors
- paying copy machines
- paying food in the canteen
- paying soda vending machine
- using the phone to navigate to a given location
- using the phone to vote in elections
- registering people in meetings
- removing reservations and undoing actions
- sharing events with friends
- the application should show receipt

Among the comments I got regarding requirements, was from a person saying that the services should be free because they use their private phone to use them. This is an important note that again, as person one mentioned, underlines the importance of privacy. A couple of the ideas mentioned above, such as showing receipt, will be included in the prototype.

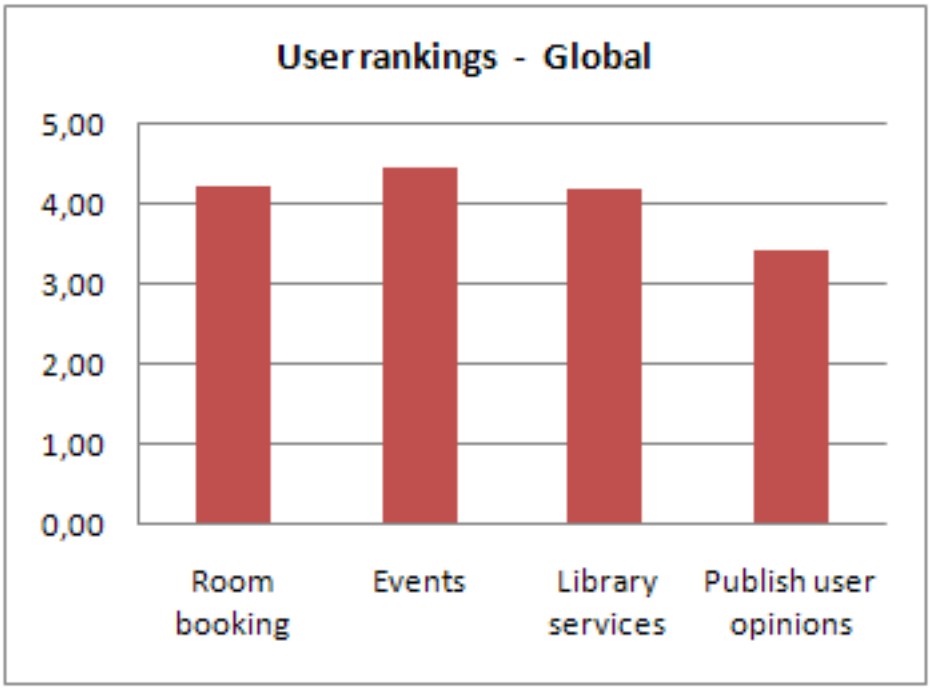

Figure 4.4: Global average of user rankings

Age	Room booking	Events	Library services	Publish user opinions
18-25	4,32	4,55	4,32	3,56
26-35	4,39	4,53	4,21	3,60
36-45	4,48	4,81	4,33	3,07
46-55	2,33	2,83	2,83	2,17
56-65	2,80	3,60	3,60	2,60

Figure 4.5: Global average, sorted by age intervals

4.7 Prototype Design

One of the focuses of this thesis is whether RFID can augment existing services in the campus. Existing services could be both electronic and services which require physical presence of the user. This section will outline augmenting services that the prototype should offer students and employees of a campus. The services are described according to my findings from the interviews, design sessions and the online questionnaire.

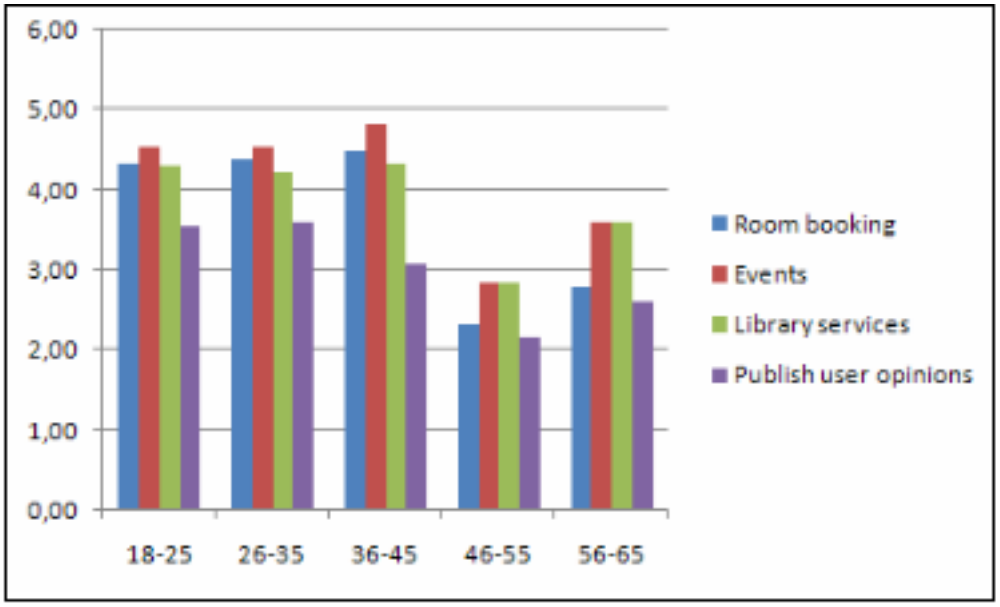

Figure 4.6: Global average, sorted by age intervals, shown as a 2D figure

4.7.1 Description

The prototype will have four main functions. These are "Events", "Room Reservation", "Library" and "User Profile". They will all, with the exception of the user profile, be connected to a RFID detection module. As discussed in previous sections, the prototype should not "trap" the user among several menus and unnecessary navigation. The prototype will therefore be automated in a way that enables auto-switching between screens. The user can then scan and identify any tag from any point of the application.

Manually choosing "Events" from the main menu, will switch the screen to the *reader* form. The reader form is a graphical user component, that will have to present the data received from the RFID detection module. For simplicity, the reader form should be able to deal with all of the three services. This means adding and removing necessary commands and clearing screens when a detection occurs. At the moment a detection occurs, the detection module will have to send the data read from the tag to the *library* module. The library module is a component for parsing and managing data read from the tags. The parsed data is then, as explained above, delivered to the reader component.

4.7.2 Modules

Figure 4.7 illustrates the prototype's system architecture. The following subsections will describe each of the modules.

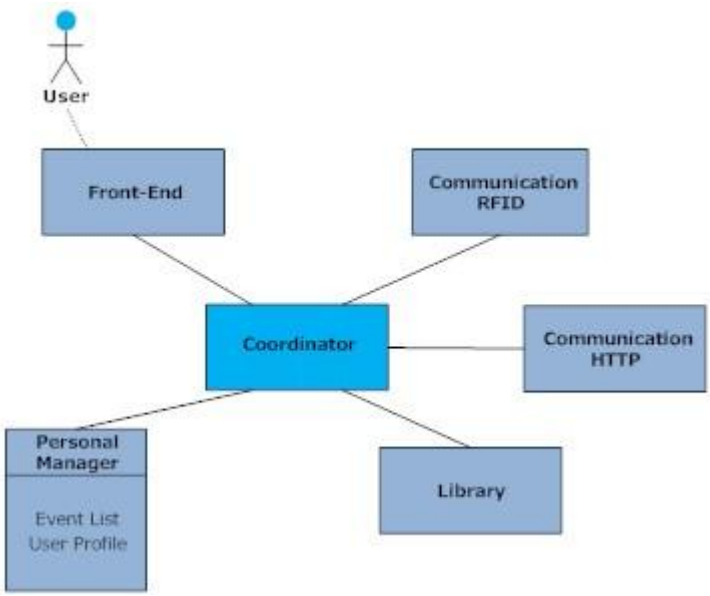

Figure 4.7: System architecture

Front-End

The front-end component is responsible for handling user actions and user input. It is a collection of graphical components that together make up the user interface of the mobile client. This component is also responsible for formatting and presenting the information in a good and appropriate manner for the user. The front-end components is directly connected to the Coordinator component, which in turn will call the necessary forms in proper order.

Communication - RFID

This component is responsible for handling all the communication with the RFID hardware. Besides reading and writing data to tags it returns output to the front-end components attached to it. Once registered it will stay activated throughout the life cycle of the application.

Communication - HTTP

This communication component handles all the operations against the server using the phones networking connecting. In future NFC phones, wireless infrastructure could also be used. The tasks of this component, besides managing the connection, could be everything from downloading media, updating databases or even connecting users.

Library

The main task of the library component is to have control over the information that resides in the system. The information is received by the communication components or from tags that are read. This could for instance be information about events, books and rooms.

Personal Manager

This component manages the user profile, which stores data about the user of the application. Persistent storage to a RMS database should also be possible. Another function of the Personal Manager, is to keep track and remind the user about events that has been added to the phones calendar.

Coordinator

The coordinator is the component that ties the application together. Its main task is to switch among the graphical user interfaces and report errors. Any process that is important for the life cycle of the application is controlled from here.

4.7.3 Main Functionality

4.7.3.1 Adding Events

This services makes it possible for students and employees at the campus to sign up and add events to the phones calendar, usually advertised on information boards placed around different locations. To sign up or add the event, the user scans the tag attached to ad. When completed, an confirmation message including date and time is viewed. Afterwords, the user can add custom notes about the newly added event. It should be possible to be remove an event.

4.7.3.2 Reservation Of Group Rooms

By touching a tag attached to an identification plate, a dialog showing date and time appears. Through this dialog, the user is able to reserve the room. The users profile is then used to update the external server, which keeps track of the reservations. Reservations should also be possible to remove.

4.7.3.3 Information About Books

There are already multiple libraries using RFID for tracking their inventory[3]. An augmentation allowing users to find information about books is possible in a campus library where books are tagged with RFID transponders. By scanning a book the phone shows information about the book and other relevant

[3] http://www.rfidgazette.org/libraries/index.html

information such as loan term. It should also be possible to collect reference information and a method to send to a desktop computer.

Chapter 5

Implementation

This chapter will present the implementation of the prototype, with emphasis on the more challenging parts. The chapter will not cover the implementation to the full extent and interested readers are advised to consult the Java documentation. The chapter starts first with an overview of the system and then continues with the data format and the data model.

5.1 System Overview

The basis for the prototype is several other previous work done on mobile context-aware systems. What differs my prototype from the related works reviewed in section 2.7, is the simplicity. The prototype is implemented using standarized API's available in most mobile phones today. It has therefore no dependencies other than a RFID reader/writer module. The complete system consists of three separate components, which are the client, the server and the database. Figure 5.1 illustrates how these three components are connected to each other. The solution is a typical client-server relation, with the client to the left and the server to the right (including databases). The clients have software that communicates with the server responsible for handling requests through the Internet or an nearby wireless networking system. The client send requests that identifies a service and necessary data required to process the request. The server uses this data to filter content from the database, which is sent back to the client.

Since I focus more on the usability of RFID based services, I chose to concentrate most on the client and the detection of different tags. Sending requests to Internet is therefore emulated in the current version of the prototype. A HTTP-connection module for the client and a server-side script to handle client-requests, is nevertheless developed and tested. This allows future improvements in an easy way. The main reason for adding HTTP support and the server-side script is to show how simple tools can be used to create service-based information systems.

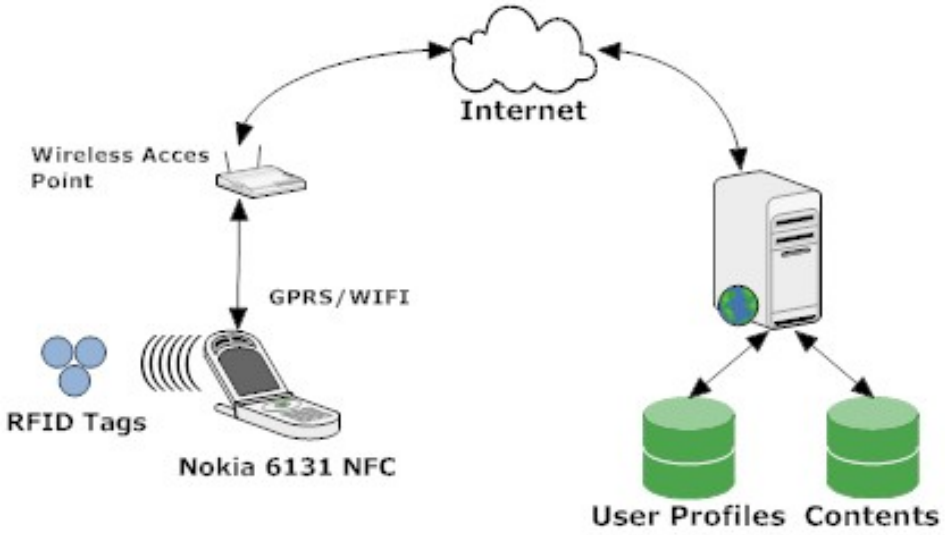

Figure 5.1: System overview

5.2 Data Format

In section 3.3, I discussed how a custom data format could be used to identify services in an information system. The outlined solution is meant to be an independent solution. For simplicity however, I chose to use a combination of NDEF and parts of my own data format. The NDEF format [1] I used was a simple Text message. This text message wraps a string for identifying a given service.

The initial data format I proposed in section 3.3 has the following format:

[service type][size][[data field]...n]

The implemented prototype uses this format to identify three service types. These three services are identified by the keywords *lib*, *evt* and *grp*. The data fields contain minimum amount of data required for either directly presenting information or as input for Internet based services. Room reservation is an example of an Internet based service. Data about a specific room is stored in a range of data fields. This minimum amount of data is presented to the user. In the next step, data about the user, including reservation data, is sent to the server over the Internet.

Examples of how the strings are formatted is as follows. Identifying a room can be done with the following string:

grp@roomid.roomtype.number-of-seats.availability

The data string for identifying an event can be in the same way:

[1] NDEF is being standarized to be a industry format

evt@date-from-to.title.description.needs-registration

The library service uses the following format:

lib@title.description.author.category.reference-data

The service type and the data fields are divided by the '@' symbol, which is the only commonality. The data fields can however use different patterns. An event string can for instance look like this:

evt@15062008100060.Advanced C++ lecture.Famous programmer lectures advanced C++

The "15062008100060" data fields can be divided into 15-06-2008-1000-60. This means 15th June 2008 from 10:00 to 11:00 (60 minutes). Another example of how the data fields are formatted is the following string to identify a book:

lib@Activity-Centered Design.A book about good design principles.Helene:Hembrooke.IT.##BOOK{hembrook,:author = {Helene Hembrooke},:title = {Activity-Centered Design},:year = {2004}}

The first two fields in the above string is the title and the description. The third field is the author name. Assuming that an author can have several middle names, the names are divided by a ':' symbol. The first and the last name is always the forename and surname. The last field is the reference information. The ':' symbol is used to mark line breaks. When the application pareses this data, it will present the reference information as Bibtex format:

@BOOK{hembrook,
author = {Helene Hembrooke},
title = {Activity-Centered Design},
year = {2004}
}

In the present prototype, the strings are written to the tags directly from the phone itself. This is again done for simplicity, but there is also possibility to make an external program using the desktop RFID reader included in the starter kit I bought from TopTuniste.

5.3 Data Model

This section will describe and illustrate the data structure present in the prototype. The data strings read from the tags, are parsed and systematized be a component called the "infomanager". In section 4.7.2, I have referenced to this component as the "Library" module. The infomanager is responsible for distributing data to the data structure, which in turn is used in necessary part of the application.

Figure 5.2: Data structure

Figure 5.2 shows the components of the data structure. Each time a RFID tag is detected, a *Message* object is created. The infomanager checks the content of the detected tag and then chooses the correct subtype of the Message object. The prototype uses only three subtypes of class Message, which are Book, Event and Reservation. The data structure contains however a subtype called StringMessage. This type is included in order to allow future expansions.

The application also supports complete persistence of the data structure. Each Message can be serialized separately or all at ones through the array lists, called respectively BookList, EventList and ReservationList.

5.4 Data Flow

This section will give a more detailed view of how the prototype is built. The prototype is multithreaded and has several *managers* that take care of different tasks. Two of the most important managers are the INFOManager and the RFIDManager, which run in separate threads. The RFIDManager runs continuously in the background and listens to the RFID hardware. When the phone touches a tag it will instantly send the contents of the tag to the INFOManager. The INFOManager determines the type of data and then registers a *message*[2] in the corresponding list.

After the INFOManager has registered a message it will notify the graphical user interface manager called *FrontEnd*. The FrontEnd manager coordinates the other GUI components and calls the correct component according to the type message created. A complete class diagram is shown in appendix B.

In the next subsections, I will explain how the main functions work.

Adding A New Event

The first thing that happens is that the user scans an advertisement with a RFID tag. The INFOManager detects the ad and sends the information to front end. The result is shown in the image series in figure 5.3. The images respectively show how the front end present the information from the tag, the newly added event and the same event added to the phones calendar.

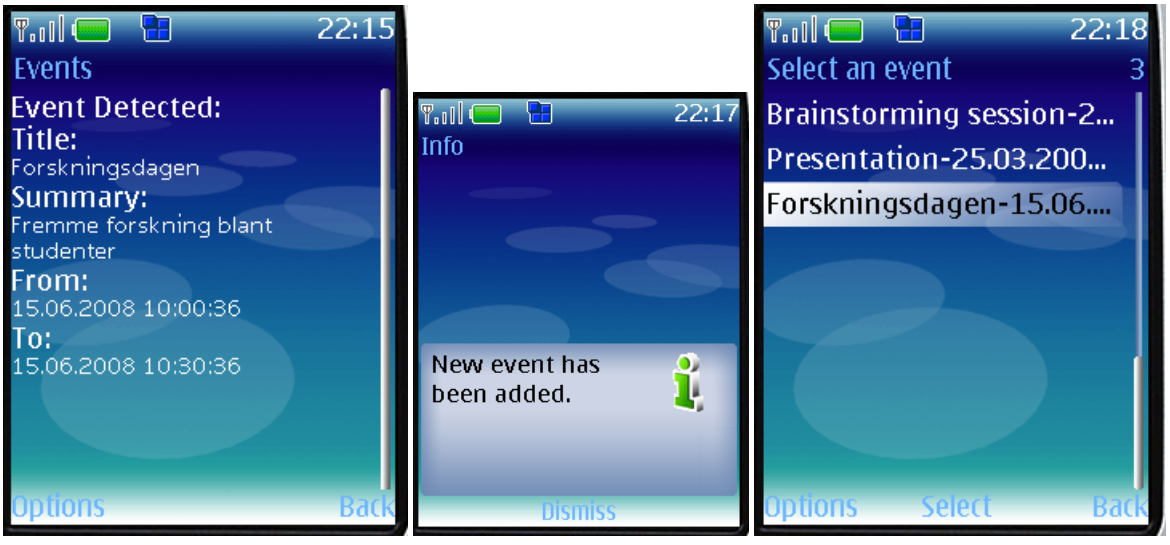

Figure 5.3: Adding an event to the phones calendar

[2]This is explained in section 5.3

Making A Room Reservation

To make a room reservation, the user first scans a tag attached to a room. Next, the front end shows the detected information about the room. The user can then make a reservation by using the menus. The process is illustrated in figure 5.4.

Figure 5.4: Making a new room reservation

Getting Information From A Book

The user scans a book or some other library object. The process is same as above. The information returned to the front end is showed in figure 5.5. Once a book is scanned and its information is displayed, the user can chose to see the reference data and eventually send it as mail.

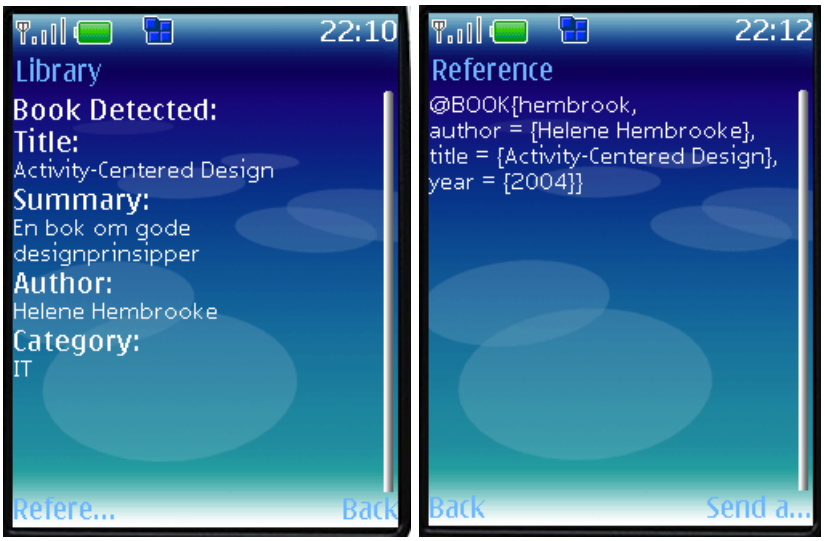

Figure 5.5: Getting reference info from books

Chapter 6

Testing

Testing is a key step for any development project of software targeted against a large group of people. This chapter will describe different tests that were performed on the prototype. These were both static system testing and dynamic system testing including real users. Several tests were carried out as a result of multiple iterations.

6.1 Method

In this section I will try to explain the different methods I used to test the prototype. I will not give a brief definition of the term *testing* and all the different types of activities that testing implies. Instead, I will explain why I chose the activities I used.

I have focused on using two central testing methods, which are used in different phases of a development project. These are component testing, also called module testing, and user testing. Component testing is a process were individual components and their functionality are tested for defects. This type of testing is usually preformed by the developer in the development process. The main goal of this process is to expose faults in small functions, methods, attributes and objects. Storey [51] explains that it is easy to remove faults detected at this stage because of the simplicity of the components.

The size and the complexity of the prototype allow me to out rule many testing techniques. If I had implemented all of the modules required on the server-side, I would have been obliged to do a more thorough testing of the whole system. Storey and Sommerville [52] define this testing as *system integration testing*. Integration tests are applied on incomplete systems composed of several modules and faults detected are more difficult to correct because to the increased complexity. Even though I have not used integration testing to the full extent, I did develop the infrastructure component and a skeleton of the server and databases.

User testing on other hand focuses on the users experience and the usability of the product. Sharp *et al.*[48] explains user testing as "a systematic approach to evaluating user performance in order to inform

and improve usability design". Usability in this context, is an evaluation paradigm called *usability testing* which are often applied on human-computer interfaces. This technique involves measuring user performance on tasks that are prepared by the evaluators. These tasks can for instance be to explore system functionality or to find specific information.

The prototype is not a complex application, but it is still not possible to answer the problem statement with only testing paradigm such as "quick and dirty" evaluations. A more systematic test with clearly defined tasks was therefore chosen. Component testing, with a little mixture of system testing, is also inevitable as I am the only developer of the project.

6.2 Component Testing

The component testing has been of particular interest because of several factors. The poorly documented API of the Nokia 6131 NFC phone and the old phone emulator which is totally different from the phone itself posed great challenges in finding the sources of some of the faults I found. Faults that turned into errors in the emulator were often the cause of system failure on the actual phone. An example of one such failure that went undetected through the emulator happened when trying to delete an event from the phones internal calendar. On the emulator, the following happened:

- User selects the "Event" menu option.
- User selects the "Show added events".
- User selects an event from the list and then selects "Remove" from the menu.
- The selected item disappears from the list and user thinks it is deleted.

However, when loaded into the phone, the following happened:

- User selects the "Event" menu option.
- User selects the "Show added events".
- User selects an event from the list and then selects "Remove" from the menu.
- <<system failure>>

The above failure caused the application to crash with a *PIMException* message. An analysis of the code revealed multiple cross-referenced data streams. This and other examples of similar bugs lead to the decision of not using the emulator for testing at all. A couple of things have to be noticed. The returned error message is not descriptive, so tracing the source using an old emulator makes it time confusing.

The faults I uncovered, was done using both static and dynamic testing methods. It has also been necessary to overlap these two methods. While most of the faults I found are corrected, some are not

because of limitations in the phones operating system. The test results, including faults found in the component and the system test, are listed in tables 6.1, 6.2 and 6.3.

Detect event	
Component test - Module name(s)	**Result**
ReaderForm, PIMManager, INFOManager EventsList	- Not possible to delete events - Menu options misplaced - Internal calendar not updated - ReaderForm missing "Scan tag" message
System test - Description	**Result**
Show event detection read form	OK
Detect tag, view contents	OK
Add event to calendar	OK
Delete event from calendar	OK
View calendar items	OK
Add note to event	OK

Table 6.1: Detection events

Detect room	
Component test - Module name(s)	**Result**
ReaderForm, INFOManager, HTTPManager ReservationForm, MessageBasket	- MessageBasket list contains duplicates - Not possible to delete reservations - Reservations missing descriptions - HTTPManager not able to connect to server
System test - Description	**Result**
Show room detection read form	OK
Detect tag, view contents	OK
View reservation form	OK
Add reservation	OK
Send reservation info to HTTP-server	OK
View receipt form from server	OK
View reservations	OK
Remove reservation	OK

Table 6.2: Detection rooms

Detect book	
Component test - Module name(s)	**Result**
ReaderForm, INFOManager, ReferenceForm	- INFOManager does not parse author middle name
System test - Description	**Result**
Show book detection read form	OK
Detect tag, view contents	OK
View reference information	OK
Send reference information as mail	OK

Table 6.3: Detection books

6.3 User Testing

Before the development of the prototype, I needed to get an idea of what the users expected and which functions they preferred most. The interviews and the questionnaire resulted in several requirements, which in turn resulted in a prototype. After several faults are removed and the prototype is validated, it is ready for user testing. The user testing will give me an indication on whether the users are able to perform the intended tasks. To find out if the prototype serves its purpose, a group of people is gathered and given a series of tasks.

The users that I chose had different technical background and were in different age ranges. The users are students at the campus and are randomly chosen. They were given the following tasks:

- Events:

- Detect event tag and view contents
- Add three events to calendar
- View calendar items
- Add note to one of the events
- Delete one event from calendar

- Room reservation:

- Detect room tag and view contents
- Add two different reservations
- View reservations list
- Remove one of the reservations

- Books:

- Detect book tag and view contents
- View reference information
- Send reference information as mail

6.3.1 Test Procedure

The first step in order to do the tests was to prepare a laboratory to reflect the real environment as much as possible. This involved preparing RFID tags, ads for the information board and tagging books and doors. The ads were marked with a yellow circle to indicate the area to scan. Next, I needed

to prepare and check the tasks that were going to be carried out by the users. I also prepared a questionnaire regarding their experiences from the tests, which was given to each participant at the end of the test.

When the participants arrived, I introduced the topics of my thesis and told them briefly about the purpose of the prototype. I also told the participants that they were going to do the tasks by themselves and without any help from me. The users had to do the test one by one since I only had one RFID enabled phone available. Each test was documented using video, audio, photography and logging.

The test started by me giving the first participant the phone. To make sure that I could observe as much as possible, I told the participant to start the application himself by pressing the launch button on the phone. I did this to check whether the user read the messages given at the start-up process of the application. As the participant worked his way through the tasks, I sat down and observed the user from a couple of meters away. However, it has to be said that an user test would have rather been done in a separately build user testing laboratory without the presence of scientists. The presence of anybody else than the person who is being testing will realize the Hawthorne-effect, which simply explained means affected test results. As I did not have a separate user testing laboratory with observation equipment, I chose to observe from a discrete corner without any contact with the user.

I noticed several things while I observed and taking notes of everything I saw and heard was therefore important for further analysis. After all the tasks were completed, I handed out the questionnaire. The user testing was ended with a informal talk, where I was able to ask additional questions regarding the application.

6.3.2 Data Collection

The first group to test the prototype consisted of people with different technical background, but they were mostly non technical. The second group consisted of expert users. The test results were quite different between the groups and they focused on two important areas of software design, the non technical on human-computer interface and the experts on technical aspects. The different usage patterns among the two groups revealed a numerous things and even an error that were not found in the system testing.

The first group consisted of three students and they used circa ten to twelve minutes each on the twelve tasks given them. Their usage patterns showed that they did not bother exploring the application, but rather followed the tasks in specified order. While doing the tests, the participants commented some common aspects of the prototype and also had some common mannerisms. The first thing I noticed was the way they waved the phone over the tags. Instead of *touching* the tags, they waved the lid several times in random directions until the phone detected the tag. However, when getting used to how to detect the tag, they scanned more controlled. One comment that all of the participants mentioned was the simplicity of the prototype.

The second group consisted of two expert users and they used circa seven and eleven minutes

respectively to complete all of the tasks. In opposite to the first group, the expert users showed more explorative usage pattern and did not follow the tasks in the specified order at all times. Still, they completed all the tasks. Both of the expert users also scanned the tags much more controlled than the participants of the first group. I did not notice any particular mannerism or any comments while they were doing the test, but gave a lot of important feedback on the questionnaire handed out.

The tasks were specified in an order that did not necessarily explore every function of the prototype, but in a way that implicitly forced the user to still find those functions on later stages. An example of this is when the application is started. At the very first start-up of the application, a message saying "Please update user profile" is displayed. The application is requesting the user to fill in name and id-number, which necessary for example when making a room reservation or sending reference information about a book as a mail. Figure 6.1 displays how the main screen look like. One of the things that I

Figure 6.1: The initial start-up screen

wanted to test was if the users read the initial messages displayed in the main screen. Of the three participants in the first group, only one user actually read the message about updating the user profile. The two other users, who did not read the message, got instead a warning message when trying to send reference information as mail. To be able to send the info, the users clicked their way back to the main screen and then updated the user profile. The expert users made also the same "mistake", to my surprise. One of the users commented that the application should instead show the profile and let the user update it immediately after the warning message, instead of clicking back to the main

screen. Better information about the missing profile data, can also be given at start-up. The latter was proposed by one of the expert users.

Another thing that several of the users commented, was the buttons placed at different locations. The term "button" also includes menu items. The location of the buttons and the menu items are dynamic. This is a consequence of the operating system on the phone and has to be taken into consideration by the developer. A "Back" button placed at the bottom right side, is for instance turned into "Clear" when the focus is placed on a text input field. When the focus is lost, the "Back" button reappears. A "Select" button without any function attached, appears on other locations. This odd behaviour seemed to confuse the users and have to be corrected in future iterations of the development.

After each test, I had some informal discussions with the participants. These discussions resulted in further ideas by the users. Among other comments I received, was:

- consider using Norwegian language
- increase font size in certain cases
- a confirmation box should be displayed when changing the user profile
- should be possible to "collect" scanned books
- implement a help function

6.3.3 User Satisfaction Questionnaire

A questionary was used to capture the users final impression of the application. The questionary that I used, consisted of the following questions:

- *How will you describe the usability of the application?*
- *What do you think about the graphical user interface?*
- *Which parts of the application was easy/difficult to use?*
- *What can/should be added/updated?*
- *What is your impression of RFID?*
- *How would you describe the functionality compared to the existing desktop services?*
- *Other comments*

The overall impression of the given feedback from the five users is that the application is simple and easy to use and also easy to learn. The comments to the first question says also that the interface is clear when a tag is scanned. Almost all of the comments to the second question, says that the user interface is easy to navigate through and that the interface is easy recognizable by people already using a mobile phone. A couple of the users also commented the above discussion about the misplaced buttons.

To my question about which part of the application that was difficult, three of five users respond that they did not have any problem handling the application. Three of five users also mentioned that it

should be an easier way to handle the user profile. About which parts that was easy to use, the users said that the application were overall user friendly and there did not mention anything specific.

The users commented that following should be added or updated: the possibility to collect books, better placement of buttons, help menu. One user also noted that a stronger RFID reader should be considered[1]

One of the most important questions is the users thoughts about the application compared to the existing desktop services. The opinions I received were divided. Three users say that the application is simpler and easier than the desktop services. Two users say that the existing services are good enough.

Figure 6.2: Two of the participants on the user testing

[1]The reading range is a limitation in the technology itself, but maybe better mobile readers will be available in the future.

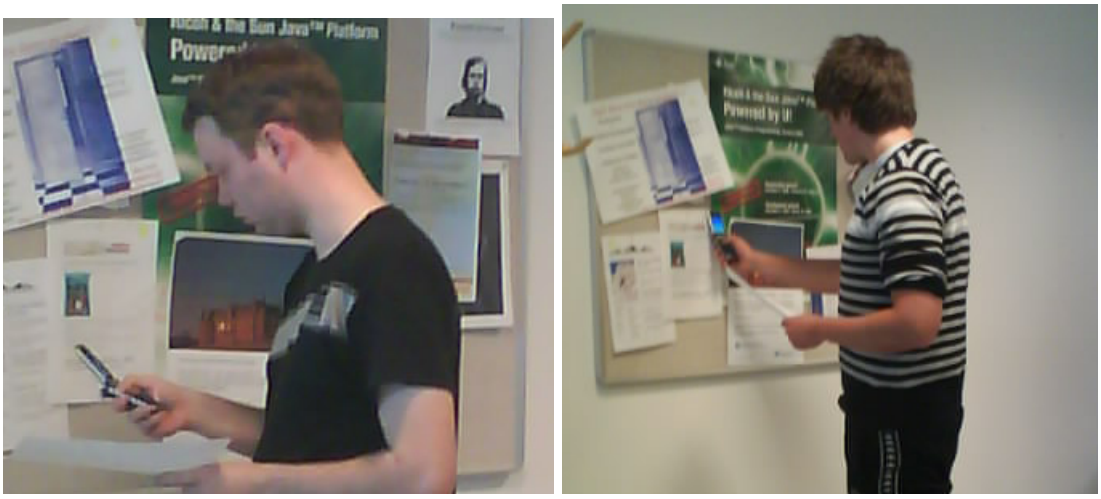

Figure 6.3: Images from the user testing

Chapter 7

Conclusions

This thesis investigated the usability of RFID augmented campus services. The thesis focused on whether RFID does augment existing services, how the new RFID programming interfaces could be used and customized and what the users thought about RFID based campus services. To answer these questions, I chose to design and implement a prototype using different design methods. Communication with the users has been one the most important techniques among the methods used. My results are therefore based on feedback I got from multiple questionnaires, interviews and user tests done by the users, presented in previous chapters. This chapter will present the results, as well as the contributions and proposals for future work.

7.1 Future Possibilities, Improvements And Use-Cases

Because of the automated nature, RFID has the potential to replace bar codes. However, some still look at RFID as an update to bar codes. Media focuses on a hypothesis that bar codes soon will come to an end, but this is not true. The extensive integration of bar codes all around the world will result in a coexistence of the two technologies. However, if the hypothesis is true, it will still take a couple of decades before RFID reaches the same level as bar codes have today.

There are already a lot of different applications of RFID, but the key point for this thesis is the possibilities of using RFID in ubiquitous mobile computing. Within the next two decades, home networks will incorporate sensors and intelligent spaces. The technologies described in the previous sections have proven the possibilities of this. RFID is still too expensive and will most probably be utilized with technologies like Bluetooth and ZigBee. Devices that in fact combine them exists, for instance [53]. Such devices are however costly and too big to avoid user interference.

The devices themselves are also an important factor to the users' experiences with the technology. Very often, the utilised technology is hidden behind the interface of the devices that are used by the users. For this reason, the obtrusiveness of the tags becomes important for the users impression of the

technology. Reading tiny RFID tags with the hybrid RFID-integrated mobile phones will get different responses compared to when using giant tags. It is the same with barcodes. When using very small tags, the expectations to the mobile device indirectly increase.

Canvas can be integrated into areas where more information is needed to be displayed or to simply layout information better. Using canvases instead of the standard components has the possibility to provide extensive GUI and HCI improvements. Among other proposes for the prototype is to layout dates where a specific room is busy, internationalization, help functions and more services. Example of other usage areas is for instance campus cafeterias, paying printouts, comment places and rooms and incorporating audio.

7.2 Claimed Results

The major contributions of this thesis are

- examination of the Java RFID programming interfaces
- a working prototype
- results of two different user questionnaires
- user experiences and testing results of the prototype

What differentiates this thesis from earlier work is first and foremost the Nokia 6131 NFC phone, with fully integrated RFID reader and writer on the lid. This phone was the only one to fully integrate RFID and at the end of this project period, even more NFC phones has emerged[1] [2]. The major contributors to the NFC community are still developing logistical applications such as ticketing and payment systems, but the community is growing rapidly and new ways of using NFC is emerging in a fast phase. For these reasons, there is no doubt that NFC will be a great part of peoples lives[3] and also influence their usage patterns of mobile devices.

The very first step towards an answer to my problem statement was to explore and master the usage of the programming API's. Doing so, made me realize that I had to make some important decisions. One of the decisions, was to not use my own methods for writing data as proposed in section 3.3. Instead, I used the standardized format called NDEF also discussed in the technical overview. Reasons for choosing NDEF was the already implemented interfaces for both reading and writing and the fact that NDEF is designed to carry short information strings, which is the case in my prototype. However, I did experiment with manual data writing and reading and did find that the processing time, when reading, to be a major obstacle. I also found that is not technically possible to combine both a custom format and NDEF objects in the same tag.

Knowing the technical possibilities of the hardware I purchased, helped me design the prototype. The users, meaning the students and employees, at the campus influenced the design process and made

[1] http://europe.nokia.com/A4991363
[2] http://www.cell-idea.com:8080/Virtual_Shop/benqT80.jsp
[3] We already see NFC speakers, NFC picture frames, NFC headsets (http://europe.nokia.com/A4785215) etc.

the implementation of the prototype possible. The first cycle, consisting of interviews and questionnaires, allowed me to collect measurable quantitative data. It has been interesting to compare the data against the user tests. The data can be divided into two subsets; user opinions of the services illustrated in chapter 4, and the impressions of the prototype usability discussed in chapter 6.

The users opinion about the illustrated services has been positive as far as I can see on the first questionnaire. However, there is also a number of comments that raise doubts about the services. I will mention them at random. The users are concerned about the cost. In a future version, the prototype must prioritize using the campus wireless network when Internet connection is needed. Privacy and safety was also mentioned several times by the users. The prototype only uses a users identification data in the current version, but it should be extended to also force a user to enter authentication data for the campus network. Because of the limited time I had on implementation, I did not focus on this.

Over fifty percent of the feedback, said that they did not know about RFID and that they only cared about how well the services were implemented. This is also exactly what the expert users told me. The prototype must have a certain value, in terms of information and effectiveness. The prototype provides both, but it will need improvements on the graphical user interface.

The users have always requirements and it is up to the developers to choose the technology that fulfils the requirements. Using RFID did augment the room reservation service to certain degree, which the test users did confirm, but it will need some improvements on the layout before I can conclude that it did in fact augment the service to the full extent. The test users who tested the prototype also confirmed that the prototype did augment the information board. My informal discussions with the test users showed that it was the possibility of inserting a reminder in the calendar that gave the event detection value. The book detection which is meant to offer the user library functions has its value in that is able to present reference information about the detected book. I did plan to implement functionality for borrowing books and other media, but this was not rated good enough nor is it possible because of security.

The services I implemented shows that RFID is useable in an environment like a campus. It is possible to implement more functions, but at the end there are questions to be asked. What is the value gain? Will RFID augment the information boards enough to make it more interesting to the few students that already look them? Augmenting information boards with RFID does increase the value gain, because the augmentation allows information to be saved digitally. Using RFID in the library has shown that there is value gain in terms of administrative tasks. Booking rooms using a mobile device is proven to be effective. However, RFID applications and services still needs development and innovation. A NFC application such as the prototype does also need integration into existing systems, how easy the integration is may vary among different campuses. One of the respondents of the first questionnaire said that he preferred "manual systems involving humans". The transformation of existing manual systems into automatic systems is however unmistakable.

References

[1] B.N. Schilit, N. Adams, R. Want, et al. *Context-aware Computing Applications*. Xerox Corp., Palo Alto Research Center, 1994.

[2] A.K. Dey and G.D. Abowd. Towards a Better Understanding of Context and Context-Awareness. *CHI 2000 Workshop on the What, Who, Where, When, and How of Context-Awareness*, 2000.

[3] Mark Weiser. The computer for the 21st century, 1991. `http://www.ubiq.com/hypertext/weiser/SciAmDraft3.html`.

[4] Ole-Christian Henriksen & Dervis Mansuroglu. Using bluetooth in a context aware mobile system (ubicams). May 2007.

[5] Dervis Mansuroglu. Proximity-based context awareness in smart homes. 2007.

[6] Wikipedia. Bluetooth, September 2007. `http://www.bluetooth.com/Bluetooth/Learn/`.

[7] Vern A. Dubendorf. *Wireless Data Technologies*. Wiley, 2003.

[8] Alan Bensky. *Short-range Wireless Communication*. Communications engineering. Newnes Publications, Burligton, USA, second edition edition, 2004.

[9] ZigBee Alliance. Faq, 2007. `http://www.zigbee.org/en/about/faq.asp`.

[10] Wikipedia. Zigbee. `http://en.wikipedia.org/wiki/ZigBee`.

[11] NXP Philips. Press release, 2004.

[12] Ecma International. Near field communication - white paper, 2004. `http://www.ecma-international.org/activities/Communications/2004tg19-001.pdf`.

[13] Wikipedia. Near field communication, September 17. 2007. `http://en.wikipedia.org/wiki/Near_Field_Communication`.

[14] Jerry Banks, David Hanny, Manuel Pachano, and Les Thompson. *RFID Applied*, chapter 1,2,3,5 and 7. Wiley, New Yersey, USA, first edition edition, 2007.

[15] Katherine Albrecht and Liz McIntyre. Spychips.org. http://www.spychips.com/.

[16] CBS 11 TV. Liz mcintyre on cbs 11 tv dallas. http://cbs11tv.com/video/?id=5117@ktvt.dayport.com.

[17] National Database and Registration Authority. Rfid on drivers licenses. http://www.nadra.gov.pk/site/395/default.aspx.

[18] American Express. Contactless credit cards. https://www124.americanexpress.com/cards/loyalty.do?page=expresspay.

[19] ICAO. Machine readable travel documents. http://mrtd.icao.int/content/view/18/199/.

[20] K. Albrecht and L. McIntyre. *Spychips: How Major Corporations and Government Plan to Track Your Every Move with RFID*. Nelson Current, 2005.

[21] Alan Steventon and Steve Wright. *Intellingent Spaces, The Applicaation of Pervasive ICT*. Springer, London, UK, 2006.

[22] Ari Juels and John Brainard. Soft blocking: flexible blocker tags on the cheap. In *WPES '04: Proceedings of the 2004 ACM workshop on Privacy in the electronic society*, pages 1–7, New York, NY, USA, 2004. ACM Press.

[23] Ari Juels, Ronald L. Rivest, and Michael Szydlo. The blocker tag: selective blocking of rfid tags for consumer privacy. In *CCS '03: Proceedings of the 10th ACM conference on Computer and communications security*, pages 103–111, New York, NY, USA, 2003. ACM Press.

[24] Melanie R. Rieback, Georgi N. Gaydadjiev, Bruno Crispo, Rutger F. H. Hofman, and Andrew S. Tanenbaum. A platform for rfid security and privacy administration. In *LISA'06: Proceedings of the 20th conference on Large Installation System Administration Conference*, pages 8–8, Berkeley, CA, USA, 2006. USENIX Association.

[25] Sun Microsystems. Connected device configuration. http://java.sun.com/products/cdc/.

[26] Sun Microsystems. Java 2 micro edition. http://java.sun.com/javame/technology/index.jsp.

[27] Wikipedia. Object hyperlinking. http://en.wikipedia.org/wiki/Object_hyperlinking.

[28] Roy Want, Kenneth P. Fishkin, Anuj Gujar, and Beverly L. Harrison. Bridging physical and virtual worlds with electronic tags. In *CHI '99: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 370–377, New York, NY, USA, 1999. ACM Press.

[29] Wikipedia. Ambient intelligence. http://en.wikipedia.org/wiki/Ambient_intelligence.

[30] AmbientIntelligence.org. Ambientintelligence.org. http://www.ambientintelligence.org/.

[31] Bill Gates. The disappearing computer by bill gates. http://www.microsoft.com/presspass/ofnote/11-02worldin2003.mspx.

[32] Eunhyung Kim, Benjamin Koh, Jennifer Ng, and Ray Su. mypyramid: increasing nutritional awareness. In *CHI '06: CHI '06 extended abstracts on Human factors in computing systems*, pages 1843–1848, New York, NY, USA, 2006. ACM Press.

[33] S. Willis and S. Helal. RFID information grid for blind navigation and wayfinding. *Wearable Computers, 2005. Proceedings. Ninth IEEE International Symposium on*, pages 34–37, 2005.

[34] M. Klemola, J. Korhonen, T. Ojala, and P. Vaananen. mtag - architecture for discovering location specific mobile web services using rfid and its evaluation with two case studies. *Telecommunications, 2005. AICT-ICIW '06. International Conference on Internet and Web Applications and Services/Advanced International Conference on*, 2006.

[35] Michael Rohs and Jürgen Bohn. Entry points into a smart campus environment – overview of the ethoc system. In *International Workshop on Smart Appliances and Wearable Computing (IWSAWC), Proc. 23rd International Conference on Distributed Computing Systems – Workshops (ICDCS 2003 Workshops)*, pages 260–266, Providence, Rhode Island, USA, May 2003.

[36] J.W. Chen. A Ubiquitous Information Technology Framework Using RFID to Support Students Learning. *Proceedings of the Fifth IEEE International Conference on Advanced Learning Technologies*, pages 95–97, 2005.

[37] T.D. Han, C. Cheong, J.W. Ann, J.Y. Kim, H.M. Yoon, C.S. Lee, H.G. Shin, Y.J. Lee, H.M. Yook, M.H. Jeon, et al. Implementation of new services to support ubiquitous computing for campus life. *Software Technologies for Future Embedded and Ubiquitous Systems, 2004. Proceedings. Second IEEE Workshop on*, pages 8–13, 2004.

[38] F. Kitson. Mobile media: making it a reality. *Queue*, 3(4):38–47, 2005.

[39] Y.M. Cheng, W. Yu, and T.C. Chou. Life is sharable: blogging life experience with RFID embedded mobile phones. *Proceedings of the 7th international conference on Human computer interaction with mobile devices & services*, pages 295–298, 2005.

[40] P. Vuorinen. Applying the RFID technology for field force solution. *Proceedings of the 2005 joint conference on Smart objects and ambient intelligence: innovative context-aware services: usages and technologies*, pages 39–41, 2005.

[41] Salvatore Sorce, Agnese Augello, Antonella Santangelo, Giovanni Pilato, Antonio Gentile, Alessandro Genco, and Salvatore Gaglio. A multimodal guide for the augmented campus. In *SIGUCCS '07: Proceedings of the 35th annual ACM SIGUCCS conference on User services*, pages 325–331, New York, NY, USA, 2007. ACM Press.

[42] Nokia Corporation. *Nokia 6131 NFC SDK: Users Guide*. Forum Nokia, 1.1 edition, July 2007.

[43] NFC Forum. Nfc forum issues specifications for four tag types, July 2007. http://www.nfc-forum.org/news/pr/view?item_key=2c0cb92de7d47bbbe7c99f13912b3307fc03c1c6.

[44] Phillips NXP Semiconductors. Mf1 ic s50 functional specification, mifare standard 1k tag. http://www.nxp.com/acrobat_download/other/identification/m001052.pdf.

[45] NFC ForumTM. Nfc data exchange format (ndef). Technical Specification NDEF 1.0, Nokia, 2006. www.nfc-forum.org.

[46] Don Wells. What is extreme programming? http://www.extremeprogramming.org/what.html.

[47] M. Dong and RG Qiu Sr. An approach to the design and development of an intelligent highway point-of-need information service system. *Intelligent Transportation Systems, 2004. Proceedings. The 7th International IEEE Conference on*, pages 673–678, 2004.

[48] H. Sharp, Y. Rogers, and J. Preece. *Interaction Design: Beyond Human Computer Interaction*. John Wiley & Sons, 2007.

[49] Wikipedia. Pictive, October 2007. http://en.wikipedia.org/wiki/Pictive.

[50] Meg Murray PhD. A closer look at pictive: A participatory design technique. http://www.larson-tech.com/567-Fall2004/pictive.htm.

[51] N.R. Storey. *Safety Critical Computer Systems*. Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 1996.

[52] I. Sommerville. *Software Engineering*. Addison-Wesley, 6 edition, 2004.

[53] TEK Industries. Tek freedom bare board, 2007. http://www.tekind.com/rfid/rfidfreedombareboard.htm.

List of Figures

List of Tables

Appendix A

Glossary Of Terms

MAG: Mobile Applications Group
RFID: Radio Frequency Identification
PDA: Personal Digital Assistant
WIFI: Wireless Fidelity
WPAN: Wireless Personal Area Network
Ad-Hoc Network: An ad-hoc network is an independent network that provides usually temporary peer-to-peer connectivity without relying on a complete network infrastructure, which includes one or more access points.
EAN: European Article Numbering
NFC: Near Field Communication
GCF: Generic Connection Framework
EEPROM: Electrically-Erasable Programmable Read-Only Memory. A small memory chip that retains data even without power

Appendix B

Class Diagram

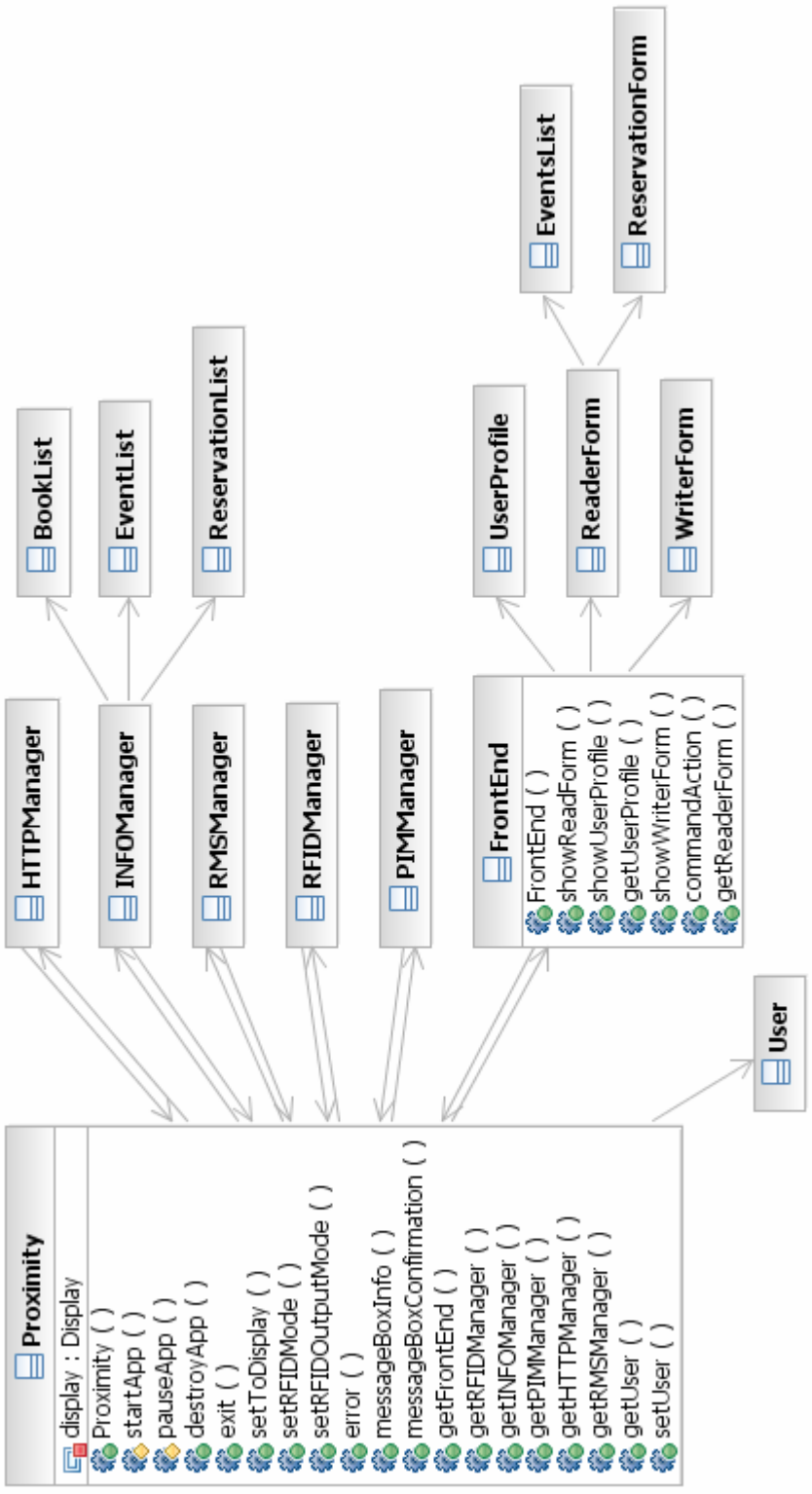

Figure B.1: Class diagram - Data flow

Appendix C

How to install Nokia 6131 NFC 1.1 Emulator into Eclipse

HOW TO INSTALL NOKIA 6131 NFC 1.1 EMULATOR PLUGIN INTO ECLIPSE

This little guide on how to install the Nokia 6131 NFC Emulator assumes that Eclipse is not installed. If you already have installed Eclipse and its version <= 3.2, then start from step 3.

You will need two applications to install the Nokia emulator on your computer, these are

- The Nokia 6131 NFC 1.1 - Download here
- Eclipse 3.2 or later - Download here

1. Install Eclipse
Start by downloading Eclipse. I am using Eclipse Europa EE, however both "Java" and "Classic" versions will work. As Eclipse doesn't need a separate installation process, you simply extract the downloaded Zip archive to a folder where you want to run Eclipse.

2. First run of Eclipse
After you have extracted the Zip archive, start the Eclipse. When running for the first time, Eclipse will ask you to select a "workspace". The workspace is the location where your project files will be saved. Accept the default location or another location of your choice. When done, make sure to select the checkbox "Use this as the default and do not ask again". By doing that, Eclipse will load faster and you will not have to select the workspace every time you load Eclipse. When Eclipse has loaded, you will see the welcome tab panel. Close it by clicking on the top left "x" symbol. Now close Eclipse again before starting the installation of the SDK.

3. Download the Nokia SDK
Click on the link given above to go to the Nokia resource page. Click on the "Nokia_6131_NFC_SDK_1_1.zip" link to and download the Zip archive. You have to log in or register yourself on the forum in order to download the file.

4. Install the Nokia SDK
Extract the contents of the Zip archive and execute the setup program. When you have accepted the end-user software agreement, you will get two choices. These are shown in the image below.

Fig 1

Select the option "SDK Integrated with Eclipse" and press "Next". Choose the root of Eclipse and press "Next". Press "Install" to start the installation.

PS: While installing, the Windows Firewall may give you warnings and automatically block some parts of the SDK. When this happens, the Windows Firewall asks if you want to unblock them. To make sure that the emulator work without problems, press "Unblock" on the warning dialogs to unblock.

5. Start Eclipse and check installation

Start Eclipse again and click "File"-"New"-"Project...". If Eclipse managed to initialize the Nokia SDK plug-in, you should see the "MIDP Project (Nokia SDK Plug-in)" and "Personal Profile Project (Nokia SDK)" wizards under "Java":

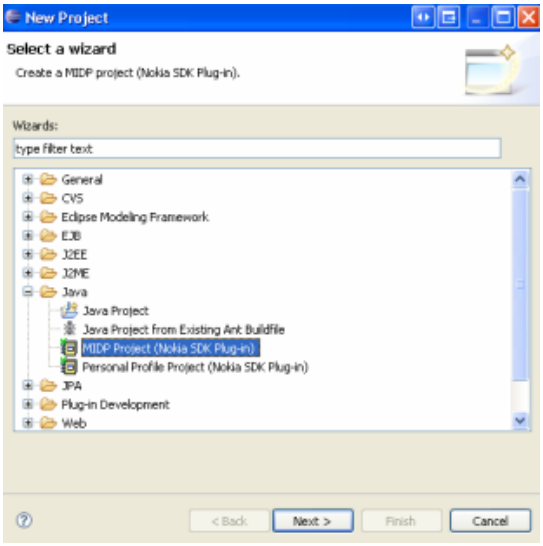

Fig 2

If you however don't see these two options, try installing the Nokia SDK plug-in as described in section 8.

6. Create a new midlet for the Nokia 6131 phone

Select "MIDP Project (Nokia SDK Plug-in)" in list of available wizards as shown in figure 2.

- Press "Next".
- Give the new project a name then press "Next"
- Check that "Nokia_6131_NFC_SDK_1_1" is selected in the drop down box, press "Next" and then "Finish"

7. Edit the project properties

In the project explorer tab, right click the newly created project and click "Properties".

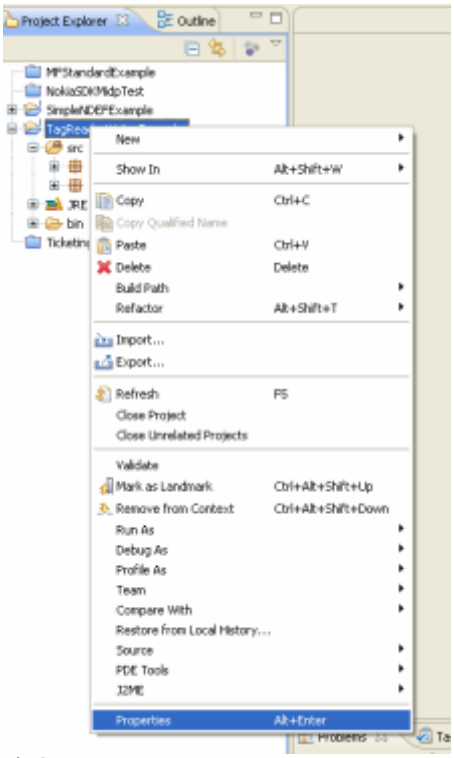

Fig 3

You will then see this dialog. Click "Java Compiler" and then select "Enable project specific settings". Then set the compiler compliance level to "1.4". Press the "Apply" button and then "OK" twice.

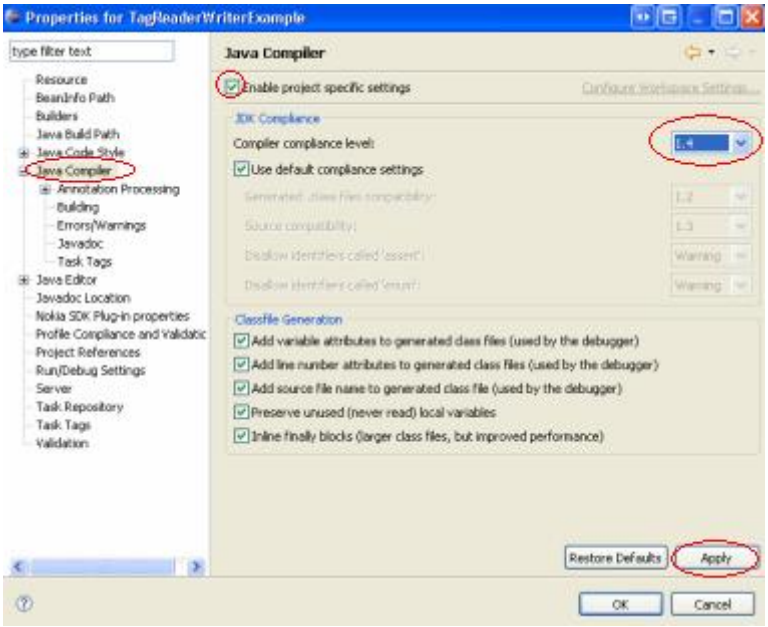

Fig 4

The emulator is now ready to go for your project. If you import some of the example projects found in the Nokia SDK directory, then remember to set the compiler compliance level as described.

8. Troubleshooting the installation

This section will be helpful if you have tried to install the Nokia SDK plug-in with the setup application and you don’t see the "MIDP Project (Nokia SDK Plug-in)" and "Personal Profile Project (Nokia SDK)" wizards as shown in figure 2.

Experience has shown that Eclipse is not always able to detect the Nokia SDK plug-in. In these cases, a manual installation of the plug-in might work. The Nokia SDK installer will create a folder called "com.nokia.phone.tools.sdk.plugin.eclipse_2.0.1" in both "plug-in" and "features" folders found in the location where Eclipse is installed. Check the existents of these folders.

If they exist, try restarting Eclipse. If it still doesn't work, you must install the Nokia SDK via the "Software Updates" function. Before doing that, there are a few steps we have to go through in order to make this work.

8.1 Tool you will need

You need:

jar.exe

- version 6.0.0.x
- resides in “C:\Program Files\Java\jdk1.6.0\bin”

jli.dll

- version 6.0.0.x
- resides in “C:\Program Files\Java\jdk1.6.0\bin”

8.2 Create the “update site”

There is three ways of installing a plug-in from the software updates function:

- from an external update website
- from an archived update site
- from a local update site

We are going to use the last of these three. Start by creating a folder named “com.nokia.phone.tools.sdk.plugin.eclipse_2.0.1-eclipse-site” in a location of your choice. I will call this folder “localUpdateDir” from this point.

Inside localUpdateDir, create a folder named “features” and a folder called “plugins”. The structure will now look like this:

. .
com.nokia.phone.tools.sdk.plugin.eclipse_2.0.1-eclipse-site
. . . features
. . . plugins

8.2 Package the folder-contents as Jar-archives

1. Create a folder called “jar” in C:
2. Create a folder called “classes” in C:\jar
3. Copy jar.exe and jli.dll to C:\jar\classes
4. Go to the location where Eclipse is installed and copy feature.xml, feature.properties and license.html from the folder called “com.nokia.phone.tools.sdk.plugin.eclipse_2.0.1-eclipse-site” inside [eclipse-dir]/features/. Paste these three files into C:\jar\classes\
5. Create a batch file (notepad-save as “bat”) in C:\jar\ and paste this code inside:

```
1  cd classes
   jar -cvf com.nokia.phone.tools.sdk.plugin.eclipse_2.0.1.jar  feature.xml
   feature.properties license.html
2  cd ..
3  Pause
```

6. Save the batch file and run the script. The script pauses so you can check that there aren’t any errors. Press enter to continue.

7. It should now be a Jar file inside C:\jar\classes with the name com.nokia.phone.tools.sdk.plugin.eclipse_2.0.1.jar
8. Copy the newly created Jar file and past it inside [localUpdateDir]\features
9. Delete all the files inside C:\jar\classes
10. Go to the location where Eclipse is installed and copy everything inside the folder called "com.nokia.phone.tools.sdk.plugin.eclipse_2.0.1-eclipse-site" in [eclipse-dir]/plugins/. Paste all the files into C:\jar\classes\

    The contents of the "classes" folder should now look like this

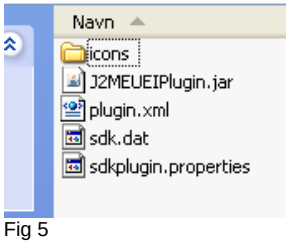

Fig 5

11. Go back to C:\jar and edit the batch script so it looks like this:

```
1  cd classes
2  jar -cvMf com.nokia.phone.tools.sdk.plugin.eclipse_2.0.1.jar  icons
   J2MEUEIPlugin.jar plugin.xml  sdk.dat  sdkplugin.properties
3  cd ..
4  pause
```

12. Save the batch file and run the script. The script pauses so you can check that there aren't any errors. Press enter to continue.
13. A new Jar file should now exists in C:\classes as explained in step 7.
14. Copy the newly created Jar file and past it inside [localUpdateDir]\plugins

You should now have to Jar files with the same name. The Jar file you pasted inside [localUpdateDir]\features should be around 7 Kb and the one inside [localUpdateDir]\plugins around 103-4 Kb.

8.3 Make a "site.xml" file

Create a file named "site.xml" in localUpdateDir. Past this text inside:

```
<?xml version="1.0" encoding="UTF-8"?>
  <site>
<feature
    url="features/com.nokia.phone.tools.sdk.plugin.eclipse_2.0.1.jar"
    id="com.nokia.phone.tools.sdk.plugin.eclipse"
    version="2.0.1">
<category name="NokiaSDK"/>
</feature>
<category-def name="NokiaSDK" label="NokiaSDK">
   <description>NokiaSDK</description>
</category-def>
</site>
```

Save the file and start Eclipse.

8.4 Using the "Software Updates" function

1. Go to Help-Software Updates-Find and Install…
2. Select "Search for new features to install"
3. Press "New Local Site…" button on the right hand side.
4. Find and select localUpdateDir. Press OK.
5. Press OK again. (Figure 6)

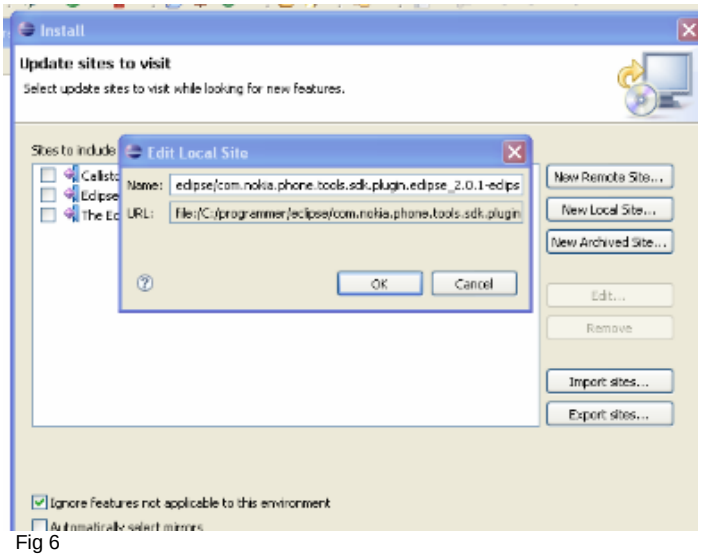

Fig 6

6. Press “Finish”.

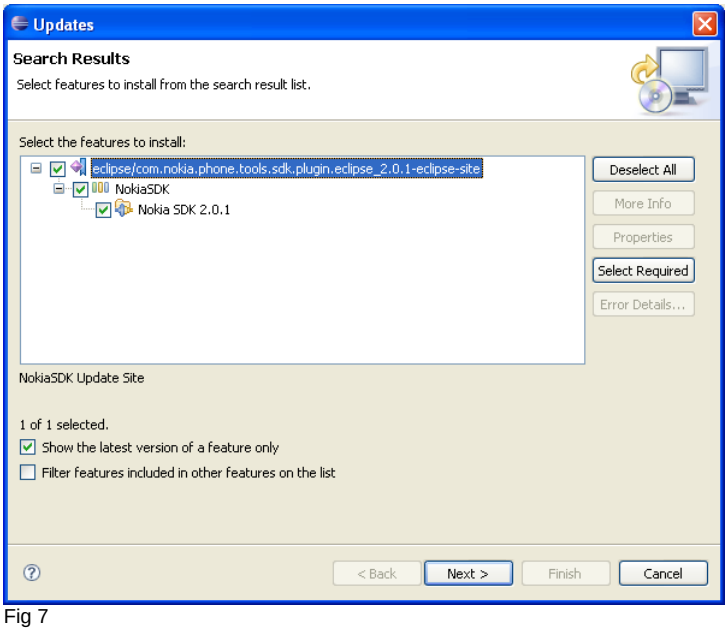

Fig 7

7. Select all. Press Next. (Figure 7)
8. Accept the license.
9. Press Finish. (Figure 8)

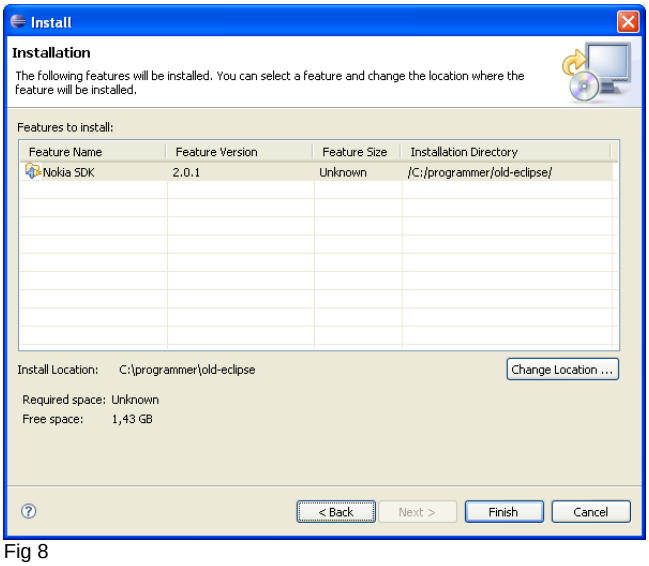

Fig 8

10. You will now see the verification dialog. Press “Install All”.
11. Press “Apply Changes” and then restart Eclipse.
12. You should now be able to see the two Nokia wizards as in figure 2.

Appendix D

Resellers Of Tags And Readers

Name	Site	+	-
Skyetek	www.skyetek.com	Small devices, tested by researchers	expensive
Readers			
Skyetek M1 Mini		Very small	Expensive, fits best in customized readers
Skyetek M1		Small	Expensive, not wireless
Tags			
None			

Table D.1:

Name	Site	+	-
RF Code	www.rfcode.com	Offers 433 MHz Tags	Expensive, active tags are not compatible with passive tag readers.
Readers			
M210 Mobile Reader		Small	PC-card for laptops
M220 Mobile Reader		Bluetooth activated, long range	Expensive, does not read passive tags
Tags			
Only active tags.			

Table D.2:

Name	Site	+	-
ToP Shop netstore	www.ti.com/rfid/shtml/products.shtml	Offers both tags and mobile readers, including development kits	Tag names have been changed
Readers			
Nokia 6131 NFC		Mobile phone with an integrated RFID reader	Bad documentation
NFC Encoder		USB Desktop reader, small	Not wireless
Tags			
A lot of different Phillips Mifare Tags			

Table D.3:

Name	Site	+	-
Texas Instruments	www.ti.com/rfid/shtml/products.shtml		Really bad web pages, bad selection of readers
Readers			
TRF7960/61			Not mobile
Tags			
Only Tag-It tags			

Table D.4:

Name	Site	+	-
CoreRFID	www.rfidshop.com	Great selection of mobile readers	No mobile phones with integrated RFID readers
Readers			
IDBlue		Bluetooth activated, small	Too expensive
Various CF Cards		RFID enables PDAs	Devices become bigger and heavier
Tags			
Only Tag-It tags			

Table D.5:

Appendix E

Questionary

The following questionary was executed to get feedback from students and employees about their knowledge of RFID.

SPØRREUNDERSØKELSE OM MOBILE TJENESTER I SKOLEN

Tenk deg en skole der du kunne lånt bøker bare ved å bruke mobiltelefonen, reservere grupperom bare ved å vifte med mobiltelefonen over romskiltet eller til og med være i stand til å "snakke" med informasjonstavlene. Alt dette og mye mer er mulig med dagens avanserte mobiltelefoner. Om noen år vil flere og flere mobiltelefoner bli levert med en såkalt RFID-scanner. En slik scanner vil være i stand til å lese RFID-kort helt trådløst bare ved å veive kortet over scanneren. De fleste som har brukt elektroniske busskort i Østfold eller en Autopass-brikke i bilen, vil kjenne seg igjen. Ved å scanne RFID-kort, vil mobiltelefonen være i stand til å automatisk laste ned mer informasjon om et produkt eller objektet som kortet er festet. Spørreundersøkelsens mål er å kartlegge studenters og ansattes synspunkter på hvordan RFID kan brukes i skolen og hvordan det kan forbedre eksisterende tjenester i skolen.

1. Kjønn

Mann Kvinne

2. Alder

18-25 26-35 36-45 46-55 56-65

3. Leser du informasjonstavlene ved hovedinngangen?

Aldri Noen ganger Jevnlig Veldig ofte

4. Har du noen gang brukt selvbetjeningsmaskinene (for å låne bøker) i biblioteket?

Aldri Noen ganger Jevnlig Veldig ofte

5. Synes du det er tilstekkelig informasjon om maten i kantinen?

Ja, men jeg vil gjerne vite mer om hva jeg kjøper

Ja, det er nok informasjon

Nei, men trengs ikke mer heller

Nei, jeg aner ikke hva jeg kjøper

6. Ranger mobiltjenestene, der 1 = lite interessant og 6 = veldig interessant

a) Reservere gruppe/møterom

Eksempel: En tilfeldig dag da du befinner deg på et annet sted i bygget, går du forbi et rom du vil registrere for et møte. Du tar frem telefonen og scanner rommets navneskilt (som har en ID-brikke montert). Du velger så dato og klokkeslett på kalenderen som vises å telefonskjermen og registerer møtet.

1 2 3 4 5 6

b) Meld deg på et arrangement og få en påminnelse i kalenderen

Eksempel: Du har akkuart kommet på skolen/jobben og ser en annonse for et populært

foredrag/hendelse. Du tar frem telefonen og scanner en ID-brikke festet til annonsen. Du blir nå automatisk registert som deltaker, samt at en påminnelse blir lagret i telefonens kalender.

○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6

c) Få informasjon om næringsinnhold og annet info om mat i kantinen
Eksempel: Maten i kantinen forandrer seg stadig. En dag ser du en ny rett og lurer på hva denne inneholder (f. eks glukemisk indeks, gluten osv). Du tar frem telefonen og scanner navneskiltet ved retten. Informasjon blir så vist på telefonen.

○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6

d) Låne bøker eller utstyr momentant ved å ta på objektet med telefonen
Eksempel: I det du går beveger deg rundt i biblioteket, registerer du bøker ved å ta på dem med telefonen.

○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6

e) Skrive og laste opp et notat om et rom/sted.
Eksempel: Du har besøkt et sted/rom på skolen/jobben og syntes rommet var dårlig ventilert. Du tar frem telefonen, skriver en melding og veiver telefonen over navneskiltet på rommet. Meldingen din blir så lastet opp til Internett, samtidig som du kan lese andres meldinger og opplevelser.

○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6

f) Andre ideer til hva mobiltelefonen kan brukes til på skolen:

7. Om slike tjenester (6a-f) skulle blitt tilgjengelig på mobiltelefoner, hva skal til for at du tar de i bruk? (eks: pris, layout)

8. Hvilke av følgende teknologier kjenner du? (kryss av alle du kjenner til)

- ☐ Bluetooth på mobilen
- ☐ Elektroniske busskort
- ☐ GPS navigasjon
- ☐ 3G mobilsamtaler

9. Hvilke av følgende mobiltjenester bruker du? (kryss av alle du kjenner til)

- ☐ Leser nyheter
- ☐ Sender e-post
- ☐ Søker etter info om steder, personer og lignende.

10. Har du hørt om RFID (Radiofrekvens Identifikasjon)?

- ○ Har aldri hørt det
- ○ Hørt om det, men vet ikke hva det er

Jeg vet hva det er

11. Andre kommentarer:

Skriv din mailadresse om du vil være med i trekning av en liten premie.
Vinnere får svar innen 1. mars 2008.

Send skjema

Appendix F

Interview One

Dervis: Hvilke tjenester er det du bruker ved skolen?

Øyvind: Biblioteket, nettsider, søking. Bruker nettsidene til bilioteket og selve bibliotekarene. Selvbetjeningsmaskinen bruker jeg ikke. I tillegg bruker jeg også studentweb.

Dervis: Hvor ofte får du bruk for tjenestene?
Øyvind: Biblioteket har jeg brukt i perioder, noen ganger har det vært hektisk som når jeg skrev oppgaven min. Andre ganger kan det går månder mellom hver gang.

Dervis: Har bruksmønsteret ditt forandret seg etter at du ble ansatt?
Øyvind: Ja men det tror jeg ikke har noe med at jeg er ansatt å gjøre (i alle fall ikke når det gjelder biblioteket), men mer med hva jeg har å gjøre til en hver tid. Akkurat nå skriver jeg ikke masteroppgave og da har jeg ikke behov for å bruke biblioteket.

Ole: Ja, det er som Øyvind. Til tider låner jeg mange bøker og spør bibliotekarer. Selvbetjeningsmaskinen bruker jeg aldri.

Dervis: Hva synes dere om LCD-skjermene og informasjonstavlene som er plassert rundt omkring på skolen?
Øyvind: Jeg ser ikke på de informasjonsskjermene fordi at jeg aldri har opplevd at jeg ikke har fått informasjonen gjennom andre veier. Det står heller aldri relevant der som jeg interesser meg for og da gidder ikke jeg å stå der å se om det kommer noe, samt de sekundene det tar å vente det tar å skifte bilde.

Ole: Nei jeg har samme oppfatning. Informasjonen som vises er lite informativ og skjermene kunne like godt ikke eksistert.

Dervis: Har du noen ide om disse eksisterende tjenestene trenger en forbedring, samt i form av nye tjenester?
Øyvind: Ja, nå kommer det jo også mye informasjon på epost, så man kan også se på det som en tjeneste. Man kunne ha laget system for å holde oversikt over hvilke studenter som er på hvilke kurs og sendt informasjon til telefonene deres. Et problem man har er at studenter ikke leser eposten sin og dermed blir det vanskelig å komme i kontakt med de.

Dervis: Sett det i lys av at man hadde innført et informasjonssystem her på skolen, hvilke elementer hadde da vært viktig. Hvilke tjenester hadde da vært viktig å holde i fokus med tanke på RFID?

Ole: Skal man innføre et informasjonssystem på en skole, må man passe på at det er noe som gir relevans for studentene og kun det de trenger og ikke all junken.

Dervis: Hvis man bare fokuserer på RFID baserte tjenester da? Som for eksempel en tjeneste der man kan lade opp print-kvote. Hva er deres syns punkter?
Øyvind: Jeg synes det er bra at det ikke er menneskene som er utstyrt med brikker, men all utstyret. Da ser jeg for meg at man kan lete etter relevant litteratur og lese av brikken og lagre dette på telefonen sin.

Ole: Jeg synes det hadde det vært bra at man kunne sentralisert noen tjenster på RFID-brikkene. En sensor på informasjonstavlene f.eks.
Øyvind: Men det har vel en personvernmessig side av det også. Har man en litt kraftig leser så kan man lese av brikkene på avstand.

Dervis: Jeg skal nå fokusere litt mer på de tre tjenestene som jeg har tenkt å implementere. Disse er biblioteket, informasjonstavelene ved inngangen og reservering av grupperom. I biblioktet er ideen at man kan identifisere hyller, bøker eller annet media for så å laste ned mer informasjon. Ideen med grupperommene er at man kan ha en brikke på dørene, for så å kunne reservere et rom ved å bare vifte med telefonen over romnummeret. Om informasjontavlene, kan man ha brikker montert på annonser osv, slik at man kan scanne og legge inn påminnelser inn på telefonens kalender.

Dervis: Hva synes dere om disse tre tjenestene?
Øyvind: Ja, disse tre synes jeg høres fine ut. Jeg tenkte det med å reservere et rom, er et mararitt hvis man ser litt gjennom der. Når man går forbi et rom og ønsker å reservere det, så man enten ha med seg en laptop eller finne en pc for å komme seg på nettet. Så bare det å kunne vifte telefonen og få beskjed om at det er resevert er jo kjekt.

Ole: Nei jeg synes jo også at reservering av rom er alreit.

Øyvind: Samtidig så må det være en supplemang til den eksisterende tjenesten. vil man reservere rommet en uke frem i tid, så vil man liksom ikke gå til rommet og vifte med telefonen.

Dervis: Har du noen ideer om hvilke funksjoner du ville hatt i forbinnelse om disse tjenestene?
Øyvind: For reservering av rom ville det vært resevering, informasjon om dette rommet fremover i tid. Biblioteket; hente ut informasjon og hente ut referanse. Lagre info om boken, som kan brukes til å søke informasjon på nettet.

Dervis: Hva synes du om at man kan få se hva andre synes om en bok man for eksempel finner på biblioteket?
Øyvind: Det vil være et problem dersom man bare baserer seg på dette biblioteket, siden det er for mange bøker i forhold til hvor mange bøker som blir lånt ut. Dersom en bruker for eksempel skriver en dårlig melding om en bok, så kan alle de andre også tro at boka var dårlig. Men dersom man samkjører det med andre biblioteker, så for man muligens flere brukere pr. bok. Man kan også koble det til Amazon f.eks.

Dervis: Tror de ansatte ville sett en nytteverdig i slike tjenester ?
Øyvind: Kanskje, men jeg tror ikke de ansatte er så interessert i hva andre synes om en bok. Når det gjelder grupperom så kan det f. eks være akutte situasjoner der det kan være nyttig.

Dervis: Hva synes du Ole, sett fra en students perspektiv. Vil studenter ha brukt slike tjenster, dvs at man kan lese andres synspunkter, melde seg på arrangementer osv.?
Øyvind: Ja, så absolutt. Det tror jeg ville vært veldig bra.

Dervis forklarer at RFID ikke er så fullt kjent blant folk og at Østfold bruker RFID i busskort.

Dervis: Har RFID muligheter på en skole og ikke bare logistisk sett?
Øyvind: Ja, det har det helt sikkert, men det er ikke en problemstilling jeg har tenkt på. Jeg ikke satt meg inn i RFID helt, Men jeg tenker mest på det personvern messige. Om en tjeneste er forbedret ovenfor brukerene eller ikke, tror jeg handler mye om hvordan det blir implementert. For hvis du implementerer en tjeneste på riktig måte så tror jeg at det kan gi veldig mye bra som du ikke kanskje klarer å forutsi på forhånd, men hvis du implementerer det feil kan man ødelegge noe som i utgangspunktet

er bra og som blir så håpløs at ingen gidder å bruke det. For eksempel hvis man skal ha en leser foran hvert grupperom å man vifter fremfor, hvis man da skal sjekke om et rom er opptatt fremover resten av dagen, så man starte et program og kanskje trykke på noe mer og da blir det straks uinteressant for brukeren og da har man kanskje lett igjennom en meny etter programmet. Da blir det så mye knot og da gidder man ikke å bruke.

–Dervis forklarer om auto-start funksjonen til Java Midleter på Nokia 6131. Dervis forklarer at prototypen skal være enkel og simpel og at det kun skal fokuseres på de tre hovedtjenestene.
–Dervis viser bilder av prototypen og forklarer hvordan grensesnittet er tenkt i prototypen.
–Dervis sier at brikkene blir identifisert ved bruk av et eget format.
–Dervis sier at grensesnittet må automatiseres

Øyvind sier at det høres bra og at han er enig.

–Dervis viser et navigasjonsmønster og spør om meninger

Øyvind: Det bør kan være mulighet for å legge til et notat og redigeringer på events. Det kan være hvis man ønsker å supplere det med noe informasjon. Det kan være greit å kunne legge til fritekst når man legger til en event.

Ole: Hvis man har mange events er det kanskje greit med en søkefunksjon.

Øyvind sier at det er bra at programmet ikke har en egen innebygget kalender.

–Dervis sier at layouten blir simpel.

Øyvind: En funksjon du kan implementere på biblioteket, kan være at man kan lagre alt av referanse informasjon, så kan man dumpe informasjonen fra telefonen til datan. Man kan for eksempel lagre det i bibtex format.

–Dervis nevner lånefunksjonalitet av bøker og dette gjøres ved hjelp av brukerprofilen på telefonen.

Øyvind: Men da må det i såfall være mulig å legge inn brukernavn og passord mot bibsys, for ellers kan man endre navnet og låne på andres navn.

–Dervis sier at man kan bruke en meny med funksjoner for en tjeneste, pluss en hurtig knapp/snarvei.

For biblioteket, kan man f.eks ha en meny, pluss en "add" knapp på hver sin side.

Øyvind: Den er litt skummel, for det som er viktigst for noen er ikke viktigst for andre. For noen som er strukturert og som finner riktig bok, kan "add" funksjonen være grei. Andre søker mer og trykker derfor kanskje på "info". Det er derfor litt skummelt anta hva som er best for brukerene. En annen mulighet er å bruke tastaturet og bruke taste som snarveier.
Interview ends here.

Appendix G

Interview Two

Dervis: Har dere vært borti RFID før?
kvinne: Ja, det er jo noen bibliotek som bruker, så jeg kjenner begrepet og vet som røflig hva det er. Men jeg har aldri vært borti det eller brukt det eller prøvd det, det har jeg ikke.

mann: vi kjenner det fordi det snakk om det på bibliotekt og om det skal innføres radiobrikker kan man vel kalle det.

Dervis: Det er flere bibiliotek i USA bruker det vet jeg
kvinne: det nye biblioteket i drammen bruker det, og BI har det vel.

Dervis forklarer hva RFID er for noe.
Dervis viser et busskort og sier at det egentlig er et rfid-kort.
Dervis forklarer hvor tynne brikkene være.

kvinne: det vi så i drammen, det det de limte bak bøkene, var ikke så tynt.
Dervis snakker om rekkevidde som er avhengig av størrelse på brikker.

kvinne: hvor lenge er levetiden på brikkene.

Dervis sier at levetiden er lenge på passive brikker.

kvinne:Det er jo noe som er interessant, det er en vanvittig jobb i sette en sånn i hver bok og det er en stor jobb og det koster mye penger for personer som skal ha lønn.

Dervis snakker om batterilevetiden på brikkene.

mann: du sa man kan lese trygg med dem, betyr dette folk må ha en kode for å lese det?

Dervis nevner kryptering og at man kan skrivebestytte brikkene. Dervis sier at noen brikker kun er laget for å inneholde en ID.

Dervis viser nokia telefonen og viser hvor RFID leseren er hen. Dervis demonstrer nå hvordan telefonen virker ved å scanne en som er tagget med en RFID brikke.

mann: hvordan for vi mer informasjon om bøkene da?

Dervis sier at det bare er en demonstrasjon med begrenset funksjonalitet.

kvinne: ja, for vi ville jo hatt helt andre, vi ville for eksempel hatt dokID nummer og klassifikasjonsnummer.

mann: hvor mye minne er det på disse?

Dervis sier at de klarer 720 tegn.

kvinne: ja, det holder i massevis for identifikasjon for oss.

Dervis minner igjen om den korte rekkevidden.

mann: må man ha et program når man har en sånn telefon, eller har du lagt inn et eget programvare nå.

Dervis forklarer at det er både et eget og en fra før.

——————— DEL 2 ———————

Dervis innleder del 2 av intervjuet, med å spørre om hvordan bibliotekarene jobber.

Dervis: Hva er det dere egentlig til vanlig, bortsett fra å låne bøker?

kvinne: Ja, asså det vi har ansvaret for det er jo at det er bibliotektjenester til ansatte og studenter. Bibliotektjenester det er bøker, dvs vi må kjøpe bøker og bestille fra bokhandel og når de kommer må de tas i mot å gjøres i stand og må katalogiseres, klassifiseres også skal det bestemmes hvor i biblioteket

de skal stå. Så må de gjøres i stand, dvs alle type klistre alle lapper som skal på og stemples også går de ut på hylle. Også har vi tidskrifter, som abonnementer bestilles og gjennomgås, hefter tas i mot ettersom det kommer med posten, settes i hyllene og sånt at gamle nummere sett i magasinet. også er det jo en viktig tjeneste det å hjelpe folk å finne den informasjonen de trenger, enten den er i trygt form eller den er på nettet. Og i den forbinnelse har vi mye brukeropplæring og kurs for studenter og ansatte i innformasjonssøking. ja, det er i grove trekk det vi gjør.

mann: så har vi databaser og nettsider som vi prøver å holde orden på. Også jobber vi mot avdelingene, jeg jobber for eksempel mot IT-avdelingen. Så har vi fordelt dette mellom oss, så hver er kontaktperson da mot hver avdeling, hvor vi prøver å få innpass, snakke med lærere, snakke med administrasjon også prøver vi å ha kontakt for å avholde kurs og fellesopplegg.

Kvinne: Og for å sikre at vi kjøper inn den riktige literaturen og abonnerer på de riktige tidsskriftene og at vi har tilgang til de viktigste basene. Kjøpe de riktige bøkene rett og slett, Vi har ganske store midler og ressurser som skal komme studenter og ansatte til gode, og da er det viktige at det gjør det. At vi ikke kjøper inn det vi tror trengs, men at vi kjøper inn det som trengs og da må vi ha kommunikasjon med studenter og ansatte for at være sikre på at det blir riktig. For det er jo ikke gitt at vi alltid forstår av oss selv.

Dervis: Hvordan ser hverdagen til en bibliotekar ut sett i forhold til antallet brukere som er i biblioteket? Dersom det nå er veldig mye å gjøre med å hjelpe brukere, åssen ser dere det i forhold til de andre bibliotekarfunksjoene. Dere har jo en rekke andre tjenester, websøk og diverse ting.

mann: selve websidene er vel relativt statiske. Men selve det også bruke database, prøver vi å gi veldig mye informasjon og kurs og åpne kurs for studenter og ansatte. Lærere har gjerne litt høy terskel ikkesant, tør ikke å vise at dem ikke kan så mye data. Da er det privattimer som gjerne passer best.

kvinne: Men det klart vi har et fokus nå på at den viktigste delen av arbeidet, den foregår nede ved skranken. Det skal være hovedfokus. Vi skal ikke føle at det vikigste jobben foregår her oppe hvor vi ikke synes. Den viktigste delen av jobben vår er å være tilgjengelige, være en støtte og vi du ser på organsisjonskartet over høgskolen, så er bibliotektjenesten organisert som en støttetjeneste. Og det er det vi skal være. Vi skal være en aktiv støttetjeneste for studentene og ansatte. Det betyr jo at det er i møte med studentene nede ved skranken eller i klasserommene det viktige arbeidet skjer. Det kan lett bli sånn det er det å være ved skranken, man har minst lyst til.

mann: det som er rutinen der nede, er at vi skal ta imot publikum. Og mye går med på dette med

utlån. Vi har ingen administrative ansatte som bare driver med utlån og tar imot bøker, men vi gjør dette samtidig som vi skal veilede og prøve å hjelpe å lære dem å søke. Og der har vi satt inn store ressurser for å være tilgjengelig, før så jobba vi mye mer i skranken og da var det vanskelig å komme vekk fra pcen for å ta imot en bruker. Derfor har vi satset på at vi skal være mest mulig fri og tilgjengelig når det først er noen der.

Dervis: Kan jeg spørre om hva det er brukerne spør etter?

mann: Stort sett er det enkle ting, at ikke finner boka. De fleste sånne trivielle er sånn at "hvor kan jeg finne boka, jeg finner ikke fram". De fleste tro at vi vet hvor bøkene står, men vi også må søke det opp.

kvinne: men vi gjør det selvfølgelig kjappere enn studentene. Også er det det at det i perioder er det mye spørsmål etter pensumbøker, helt natulig. Og det er klart når student nummer 40 kommer å spør etter det samme spørsmålet, men for student er det kanskje den første gang han/hun spør og da kan det virke imponerende for han/henne at vi kan si at det står eller der. Ellers så må vi gjerne søke.

Dervis: Føler at dere at bibliotekets tjenester blir brukt, eller hvor ofte?

kvinne: Det øker hvertfall, vi har en veldig hyggelig vekst på utlånet og en veldig hyggelig vekst på mengden i arbeidet foran i skranken.

mann: Vi har vekst i alle fronter. Vi er med i flere utvalg rundt omkring på skolen og prøver å være med på ting som skjer rundt hele skolen, både sentralt og lokalt her. Vi samarbeider med andre avdelinger ikkesant, så vi har egne grupper som tar for seg databaser og web som kommer fra de andre avdelingene.

kvinne: Men vi har stort vekstpotensiale. Det er jo en stor-ikkebrukergruppe. mann: Ja, samlingen blir jo også stående her og har statistikk på at de ikke blir brukt.

Dervis: Føler dere at noe kan forebederes, noe som kan innføres som gjør at det tiltrekker flere brukere?

mann: det er mye som kan gjøres, men det kommer litt ann på ressurser å personale rett og slett. Nå har vi hatt litt frafall, det er flere som har sluttet og vi har ikke fått nye inn. men vi har tanker om å gjøre mer.

kvinne: Sånn som vi har det nå har vi to stillinger som ikke er besatt, en pga en som har gått av med

pensjon, men så har vi ikke fått lov til å tilsette noen enda. Kanskje vi får lov, det vet vi ikke. Kommer litt ann på hva ledelsen sier og så er det en som er syk. Men det er klart det er mange som kan gjøres for å frigjøre vår tid, så vi kunne være til enda bedre hjelp for studentene. Man kunne se for seg at de ble flinkere på å søke etter de enkle tinga sjøl, så ikke vi trenger å bruke tid på å finne den elementære pensumlitteraturen. St de gjorde det sjølv og at vi kunne hjelpe til når det var større vanskeligere søk. I forbinnelse med oppgaver. At vi kunne fokusere på de større mer kompliserte søkene og at vi kunne fått en større bedre utlånsautomat så flere kunne låne bøkene sjøl, så vi kunne bruke mer tid på å veilede.

mann: Veiledning og markedsføring er vel det vi mangler mest. Men det er ikke vanlig å ha en markedsfører i et bibliotek. Det prøver vi å gjøre selv, vi er ikke gode på det for så vidt, vi har ikke kompetanse i det.

Du (kvinne) nevnte kompliserte søk, har du et eksempel på hva det kan være?

kvinne: Ja, det kan være type masterstudent, men også bachelorstudent som har en problemstilling som har søkt i basene og som ikke får til. Eller veldig spesifikke og kompliserte spørsmål som det ikke finnes noe på og som man ikke bare kan gå å peke ut en bok på. Og da har vi en kompentanse i å søke og kompetanse i hva slags baser finnes og som vi har tilgjengelig, sånn at vi kan hjelpe til med dypdykk.

kvinne: de fleste som søker, de blir veldig snevre i måten de søker på. Det gjelder alle studenter, også it-studenter.

Dervis: Jeg skal føye til to ting. Jeg er jo it-student jeg og, men det å være å i biblioteket, det er liksom en sånn; jeg er der, jeg trenger noe og jeg må finne det. Og jeg vet jeg har ressurser der inne og jeg vet jeg har pcer jeg kan søke på der inne. men det å være der som en del av min utdanning og som student, det er en spesiell følelse. Jeg vil heller være i biblioteket og søke der ting er også finne boken.

kvinne: ja, vi ønsker jo oss at flest mulig skal bruke biblioteket som en arbeidsplann, for det er jo det er, det er liksom ikke en lesestue. det er ikke et arkiv eller en oppbevaringsplass. Det er en arbeidsrom og ikke noe poeng i å sitte der hvis man ikke har noe der å gjøre. Målet må jo være at flere forstår at de ha nytte av biblioteket på flere punkter i utdanningsløpet sitt. Det er ikke målet at alle skal sitte der nede og lese febrilsk i bøker og veldig ofte er bok det riktige svaret. For de fleste er det jo sånn at det ikke er bok som er svaret. men erkjennelsen om at det kan være der nede som man ikke vet, det kan være noe der som man trenger. Det er jo den vi prøver selge. Erkjennelsen at det er en tjeneste som kan gjøre at kan forbedre resultater Ikke pga bibliotekaren sin skyld, det handler ikke om å gå på biblioteket for å glede oss. For det er noen som tror, "stakkars deg" sier dem til meg når jeg

har brukeropplæring og det ikke kommer noen. Asså jeg gjør det ikke for min egen skyld.. Det er ikke for å glede bibliotekarene at denne tjensten er her, det er for å være en støtte og for vi tror og mener helt bestemt at ved å bruke de ressurser som er stilt til rådighet , så kan studenter forbedre resultater og lære mer.

mann: Problemet er at biblioteket er så knyttet til boka, ikke til informasjon. Dvs at vi har masse ressurser på data og når folk har gått her i tre år og fortsatt ikke vet om Ordnett for eksempel og forsatt ikke vet at det finnes et digitalt oppslagsverk som går på synonymordbok som man kan koble rett inn i word. Det er sånn ting som overrasker meg at vi aldri klarer å få helt ut.

kvinne: istenden for å vi studenter som er frusterte for å de ikke for lov til å låne med seg ordbøker hjem, for de skal være i biblioteket til enhver tid. Så ryktet vårt henger ikke helt med det ryktet vi har. Og da har vi et problem med at så få studenter tror det er viktig å komme på brukeropplæringene som vi har. For de tenker selvfølgelig at de vet biblioteket er, og da for jeg to timer fri. Istendefor å tenke på at, kanskje biblioteket har noe annet. Vi sliter med å nå de studentene som kanskje hadde hatt mest bruk for oss. De studentene som kommer, er jo gjerne de som allerede har skjønt at biblioteket er nyttig.

Dervis nevner det gode poenget deres om bibliotek vs bok og informasjonsbegrepet.

kvinne: det er også det er bibliotekarer er utdannet for, å organisere kunnskap og informasjon uvhengig av fysisk form. Alle tror at vi er utdannet bibliotekarer, fordi vi er glad i å lese.

Dervis nevner spørreundersøkelsen og spørsmålet om selvbetjeningsmaskinen og at de fleste sa at de ikke brukte den.

Intervjuobjektene sier de forstår det godt og at studentene antagelig er redd for pipingen. De sier også at maskinen nærmest er håpløs.

Dervis: Hvis man ser litt bort fra kostnader og sånne ting, tror dere da at RFID hadde hatt en mulighet her, ville det brakt med seg noen fordeler?

kvinne: Ja, det ville det. det ville brakt den fordelen at vi kunne laget gode hyllekart. Som vi kunne ha koblet opp mot katalogen vår i ASK. Da kan man klikke på "hvor er boka" og få opp et lite kart som viser hvor boka er så slipper man å spørre.

mann: pluss at de antagelig vil kunne finne boka lettere, men da kommer ann på hvor avstanden er, for bøker forsvinner på hyllene, setter de dem litt feil.

kvinne: vi har mange gode hjelpere for å si det sånn, for folk tar ut en bok og ser på den ikkesant, også plukker de med seg bøker fra flere steder og glemmer de hvor så tenker de at de ikke vil jo ikke gjøre det rotet for oss, så istedenfor da å legge bøkene i en bunke slik vi vil, så setter de det på plass, men så blir det "på plass" ikkesant.. men det er klart at hvis vi hadde hatt en system som gjorde at hvis vi ikke finner boka at vi kunne gå rundt med en scanner og lett på hyllene istendenfor å lete også fått en form for beskjed.. det hadde vært en utrolig god hjelp.

Dervis forklarer at brikkene som brukes i prosjektet ikke tillater lengere rekkervidder, pga av begrensninger på telefon og teknologi. Dervis sier utviklingen går i retning av at flere telefoner for RFID lesere i fremtiden og at det må tilpasses brukere og tjenester man skal tilby brukerne. Dervis sier at telefonen er meget brukerorientert og tilbyr veldig enkle tjenester. Bruker tbane-billett eksempel og andre betaligstjenester.

kvinne: da man tenke seg at lånekortet er den der (dvs: telefonen)

Dervis sier telefonen har et innebygget minne.

kvinne: for det man kunne tenke seg da var at man på egenhånd kunne få beskjed på skjermen på mobilen hvor boka er, av typen kart. Det har også nevnt for it-avdelingen at vi ønsker et digitalt kart.

Dervis sier han forholder seg til kortdistanse brikker.

kvinne: da kunne man jo tenke seg at man søkte opp boka med mobilen, så kan man gå også vifte med mobilen over hylla også "jo, her er den", istenden for å skrive ned nummeret på en lapp og gå å lese.

Dervis sier han skal forske på brukbarheten av brikker med kort rekkevidde på HiØ, på bakgrunn av eksiterende tjenester, og om disse brikkene kan forbedre disse tjenestene. Dervis nevner de tre tjenestene som det skal fokuseres på.

Kvinne synes romreservering høres meget fornuftig ut.

Dervis nevner en rekke mulige tjenester man kan ha i biblioteket, som f.eks utlån.

Mann nevner at utlån kan bli et problem pga magnetiseringstrådene i bøkene.

Kvinne spør hvordan alarmen kan skrus av med en slik telefon.

Dervis sier han kommer til å ta hensyn til dette.

kvinne: asså vi har litt til og fra snakke tom radio brikke egntlig, men problemet er jo å få det gjordt. En ting er at det kan være besnærende, asså, sånne alarmstriper dem koster vel 1 krone stykke, sånn type brikke, så når vi har sett på sånne RFID så koster det mye mer enn 1 krone stykke. Dvs at i har ca 95000 objekter, det er 95000 kroner vi allerede har brukt, også blir det RFID brikkene. Det er jo en stor investering bare i brikkene.

kvinne understreker at det er pga prisen og etterarbeidet, man nøler med å ta teknologien i bruk.

Dervis: Det er færre hyller enn bøker, hva synes dere derfor om at man kunne tagget hyllene. Kunne dere da tenke dere en form for tjeneste som brukerne kunne utnytte da?

kvinne: Ja, det vil kunne være lettere form dem å finne riktig hylle. For selv om det for oss helt intiutivt og helt naturlig og logisk hvor bøkene står i huset, så er det ikke det for de som går der nede og skal bruke hyllene.

Kvinne forklarer i detalj hvordan bøkene er organisert i hyllene. Mann forklarer at hyllene er merket med skilt som man kan flytte rundt. Kvinne legger til at det ikke er noe i veien for at skiltene kan tagges.

Dervis sier at skal nå skal fokusere mer på oppgaven sin.

Dersom man antar at implementeringen er mulig og at det finnes et program på telefonen. Hva slags informasjonen ville dere sett for dere, ville vært mest logisk ved scanning av en bok?

Mann: Egentlig trenger den ikke mer enn et program som kobla informasjonen til posten på Bibsys, så ville man jo kunne få opp Bibsys rett opp på mobilen med all den informasjonen det er i katalogen. Den nisifrede strekkoden er mer enn nok til å koble boken til Bibsys. Men hvis det bare skulle være informasjon som ikke trenger å koble seg til internett for få inn, så må det være...

Kvinne: Tittel, forfatter, lånetid, emne

Dervis: Det som er interessant for meg, er at dere jobber på biblioteket og dere vet på en måte hva informasjon egentlig er og dere har såpass dyp innsikt. Hvis det skulle blitt innført et informasjonssystem på skolen, som skal tilby informasjonstjenester til studentene. Er det noe speielt dere tenker da i forhold til det da? Hva som f. eks ville vært viktig å hatt med og ikke hatt med. Brukere har f.eks

profiler i dette programmet jeg skal lage. Det gjør at jeg kan lage kontekst-sensitive tjenester. Om man skriver avdelingstilhørighet i profilen, da blir tjenestene som da tilbys i forhold avdelingen.

Mann: Da skulle han fått opplsyninger om meg , at jeg er kontaktperson for avdelingen hans og hvilken av tidskiftene som er interessant for han og de kunne få vite hvilke databaser som er interessant for han, innefor hvilke områder han søker. Hvilke bøker som er interessante, hvilke områder som er interessant for han. Kurser som er tilgjengelig for han.

Kvinne: Problemet med sånn type, som er utfordringen med alle slike brukersentrerte tjenester, er på en måte alle de som ikke skjønner de går glipp av. Alt man går glipp av, som kunne vært interessant for deg. Tenk deg IT-student da, du sitter å skal lage sånn type system som du snakker om nå, kan godt hende at du hadde hatt veldig god nytte av å lese pedagogikk. Men hvis har i profilen din at det som er interessant for deg er bøker som står på IT, så vil du aldri finne frAM til de hyllene. Det ligger jo alltid være sperrer når noen velger ut informasjon, så siler du jo vekk noe og det kan godt hende at det kunne vært veldig nyttig for deg det som blir silt vekk. Det er jo en ting som må tas med i betraktningen og reflektere over. Det er sånne ting jeg tenker over vi lager sånne skreddersydde opplegg. Det har alltid en pris og prisen er det man da går glipp av. Man kan miste overblikket og totalt bilde.

Mann: Man kan skrive i profilen at man jobber i IT, men jobber med relaterte fag f.eks pedagogikk.

kvinne: Når man går rundt der, kan man jo tenke seg at man kan få beskjed om å snakke med kontaktbibliotekaren min. Kanskje vedkommende har noe jeg ikke viste om, eller kanskje det kan være lurt å ikke gå direkte til hylla, men snoke litt rundt.

mann:det er vel gjerne at man har et tema, du har et tema som du jobber med som er relatert til teknologien, også relatert til pedagogikk og høgskolens organisering.

————— DEL 3 —————

Dervis snakker om tjeneste som skal forbedres og elementer ved dem som ikke er optimalt pr. idag. Dervis nevner også hvordan han har tenkt å implementere tjenester.

Dervis viser frem navigasjonsmønster og forhåndslagde skjermbilder av prototypen. Dervis sier også at programmet kan lages enten veldig enkelt eller veldig avansert, men at han ikke har tid til å gjøre den såpass avansert. Dervis forklarer hvordan den tenkte funksjonaliteten virker. Dervis nevner det at man kan koble prototypen til en felles base, slik at man kan laste ned andre meninger om diverse bøker.

kvinne: den basen finnes alt, den heter Amazon.com. Det finnes nemlig ingen verdensdatabase over bøker som finnes i bibliotek og det kommer ikke til å bli lagt heller. men Amazon er en tjeneste som en god del bibliotek lurer på om man skal koble opp mot katalogen sånn at man på en måte kan bruke 2.0 tilsnittet som man har i Amazon til å koble til bøker. Stockholms folkebiblioteket er et bibliotek som åpner for brukerkommentarer i bibliotekskatalogen.

Dervis spør om det er noe intervjuobjektene ville endret på i brukergrensesnittet. Dervis innleder til diskusjon ved å gi et eksempel om hvordan menyen kan utformes, hovedsakelig ang. en samlemeny og en snarvei på motsatt.

kvinne: Det er viktig å tenke på at folk ikke vil klikke så mange ganger, man gidder ikke å gå så dypt, det er jo et problem. Det er en utfordring for oss som samfund tror jeg, at vi er i ferd med å oppdra en generasjon som skal ha alt på et trykk. Sånn er jo ikke livet, det er jo ingenting som er på et trykk.

mann: vi ville satset her at det stå "Info" der da. De fleste brukere ville kanskje hatt "lån" der som du sier at det er mer interessant. men for å først få informasjon om en bok er interessant, som vi mener er riktig, så måtte det jo først blitt informasjon også lån. Da blir det fort 4-5 klikk. Vi er opptatt av at han/hun for informasjonen som er viktig for han, mer enn at han/hun fikk med seg en bok.

kvinne: målet er ikke nødvendigvis å låne ut bøker.

mann: Målet er at han for riktig informasjon fra databasene. Nå er det riktignok mye engelsk der og det som er problemet. Nå er vi et engelsktalende land, men folk vil helst lese norsk. Omsetningen på norske bøker er tre ganger høyere en engelske bøker.

Dervis sier at litt av poenget å vise at et enkelt RFID system med en enkel telefon kan tilby noe mere og det er informasjonen, selv om utlån ikke er det sentrale, så er informasjon det viktige. At man vet man har internet, bibsys, bibliotekarer osv, men hvis man i tilegg kan tilby en mobil tjeneste med informasjon. Det vil da være et tillegg.

kvinne: ja, det må være målet. At det er en tillegg. Ikke istendenfor.

Dervis demonstrer en tidlig prototype av programmet og hva slags informasjon som vises.

kvinne: det man kunne tenke seg vettu, når jeg sitter å ser på dette her. At man får opp opplysninger som man kan laste det videre inn i en litteraturliste, hvis det er en bok man har brukt. Så har man

alle opplysningene, så slipper man å....sånn at man bare scanner boka man har brukt og referer det i oppgaven.

Dervis presenterer en utgave av programmet som har bedre layout (Canvas versjon). Dervis sier at denne versjonen tar lengere tid å programmere, men at den til gjengjeld er mer brukervennlig, i motsetning til den enkle varianten.

mann: hvis man ser nytten av det, så er folk villig til å klikke tror jeg. men man må jo først få dem til å forstå det; at det er nyttig. Det er nytteverdien som blir viktigere da.

kvinne: hvordan man skal selge nytteverdien ja.

mann: og da må man ha en nytte inni der.

kvinne: ja, det kan jo ikke bare være et leketøy liksom, bare artig, det holder ikke at det er artig, at det går ann.

The interview ends here.