



OULUN YLIOPISTO
UNIVERSITY of OULU

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Henri Kytökangas

**UBILIBRARY: CONTEXT-AWARE LIBRARY
SERVICE ON A PUBLIC DISPLAY**

Master's Thesis
Degree Programme in Information Networks
May 2014

Kytökangas H. (2014) UbiLibrary: Context-aware library service on a public display. University of Oulu, Department of Computer Science and Engineering. Master's Thesis, 89p.

ABSTRACT

This thesis introduces UbiLibrary, a new context-aware library service developed for the Oulu City Library. The service content is aggregated from existing library databases and online services, refined with semantic and context information, and provided to library customers on a large public display that can be coupled with personal mobile phones. The requirements for the service are elicited by reviewing related literature, conducting a user study of the library's customers, interviewing the library staff, and analyzing the library's current digital services. The design and implementation of the service is done according to the agile software development model in close collaboration with the library staff.

The resulting functional prototype provides library customers with a range of applications from book recommendations to information on the library's services and events. The heart of the service is the metadata engine that maintains a database on the library's holdings, events and services. The metadata engine semantically enriches the information on the works in the OUTI database by crawling supplementary metadata from various online services such as KirjaSampo, GoodReads and LibraryThing. The database is dynamically browsed with a novel tag cloud based interface that also adapts book recommendations according to the age category and gender of the user as estimated by computer vision. An administration interface is provided to the library staff for maintaining the service.

The service is deployed in a real world setting on the UBI-hotspot located at the lobby of the Oulu City library. The service is assessed with a task-based user evaluation with 20 library customers, by feedback from eight professional librarians and by collecting statistics of real world usage over a period of 24 days. Overall, the service is found easy to use and presents a valuable addition to the library's existing digital services.

Keywords: pervasive computing, context-awareness, semantic information, web-services, system design

Kytökangas H. (2014) UbiLibrary: Kontekstitietoinen kirjastopalvelu julkisella näytöllä. Oulun yliopisto, Tietotekniikan osasto. Diplomityö, 89s.

TIIVISTELMÄ

Tämä diplomityö esittelee Oulun kaupunginkirjastoille kehitetyn kontekstitietoisien kirjastopalvelun nimeltään UbiLibrary. Palvelun sisältö kootaan kirjaston tietokannoista sekä julkisista verkkopalveluista ja sisältöä jalostetaan semanttisella sekä paikallisella tiedolla. Sisältöä tarjotaan käyttäjille suurella julkisella näytöllä, josta käyttöä voidaan jatkaa henkilökohtaisella älypuhelimella. Vaatimukset palvelulle tunnistetaan aiheeseen liittyvästä kirjallisuudesta, kirjaston asiakkaiden käyttäjätutkimuksesta, kirjastonhoitajien haastatteluista sekä tutkimalla kirjaston nykyisiä digitaalisia palveluita. Ohjelmistosuunnittelu sekä toteutus palvelulle tehdään ketterän ohjelmistokehityksen mukaisesti läheisessä yhteistyössä kirjaston henkilökunnan kanssa.

Työn tuloksena saatu prototyyppi tarjoaa kirjaston asiakkaille erilaisia sovelluksia aina kirjasuosituksista kirjaston palveluihin sekä tapahtumiin. Palvelun ydin on metatiedon käsittelyyn perustuva järjestelmä, joka tallentaa tietokantaansa tietoa kirjaston kokoelmista, tapahtumista sekä palveluista. Järjestelmä rikastaa semanttisin keinoin OUTI-tietokannassa olevaa kokoelmaa hakemalla lisätietoa erinäisistä verkkopalveluista, kuten KirjaSampo, GoodReads tai LibraryThing. Koottua rikasta tietokantaa voidaan dynaamisesti selata uudennlaisella avainsanapilveen perustuvalla käyttöliittymällä. Konenäön avulla tarjotaan käyttäjän iän sekä sukupuolen perusteella sisältösuosituksia. Kirjaston henkilökunnan tekemään järjestelmän ylläpitoon tarjotaan erillinen käyttöliittymä.

Palvelu otettiin käyttöön Oulun kaupunginkirjastossa sijaitsevalle UBI-näytölle. Palvelun toimivuutta arvioidaan tehtävä-pohjaisella käyttäjäarvioinnilla 20 kirjaston asiakkaan toimesta, keräämällä palautetta kahdeksalta kirjastonhoitajalta sekä nauhoittamalla kosketusstatistiikkaa reaali maailman käytöstä 24 vuorokauden ajan. Uusi palvelu todettiin helppokäyttöiseksi ja se tarjoaa arvokkaan lisän kirjaston digitaalisiin palveluihin.

Avainsanat: läsnä-äly, kontekstitietoisuus, semanttinen tieto, web-palvelut, järjestelmäsuunnittelu

TABLE OF CONTENTS

ABSTRACT

TIIVISTELMÄ

TABLE OF CONTENTS

FOREWORD

ABBREVIATIONS

1. INTRODUCTION.....	9
1.1 Motivation.....	9
1.2 Scope and objectives.....	10
1.3 Structure of the thesis	10
2. UBIQUITOUS COMPUTING ENVIRONMENT	12
2.1 Ubiquitous computing.....	12
2.2 Context-awareness	14
2.3 Large public displays	14
2.4 Examples of ubiquitous services in libraries	16
2.4.1 DigiMe - Ubiquitous Search and Browsing for Digital Libraries..	16
2.4.1 SmartLibrary	17
2.4.2 Personalized guidance in Intelligent Library	17
2.4.3 Interpretation.....	18
3. INFORMATION BEHAVIOR AND CONTENT DISCOVERY IN LIBRARIES	19
3.1 Information behavior in libraries	19
3.2 Semantic information.....	20
3.3 Examples of literature recommender systems	21
3.3.1 KirjaSampo	21
3.3.2 SemPub	22
4. ELICITATION OF REQUIREMENTS.....	23
4.1 UBI-hotspot	23
4.2 Requirements from the library staff.....	24
4.3 Analysis of the usage of the library's website	25
4.4 User study of library customers	26
4.5 Other studies of library customers	28
4.6 Observation of the hotspot usage.....	29
4.7 Essential (abstract) use cases	30
5. DESIGN	32

5.1	Objectives	32
5.1.1	Provide information about literature, library services and events to library customers.....	32
5.1.2	Create a digital advertising channel for the library.....	33
5.1.1	Explore the usage of contextual information in the hotspots.....	33
5.2	Design principles	33
5.2.1	User friendliness and responsiveness.....	33
5.2.2	Easy deployment, maintenance and error handling	34
5.2.3	Use of contextual information to stimulate interest.....	34
5.3	UbiLibrary's conceptual states	34
5.4	Usage scenarios and use cases	37
5.4.1	Administrator	37
5.4.2	User	38
5.5	Use case diagram	41
5.6	Technical requirements.....	43
5.7	Conceptual design.....	43
5.8	Content suggestion mappings	44
5.9	Graphical user interface design.....	45
6.	IMPLEMENTATION	47
6.1	System architecture.....	47
6.2	Client implementation.....	48
6.2.1	Applications	48
6.2.2	Overall architecture of the client.....	53
6.3	Server implementation	56
6.3.1	Overall architecture.....	56
6.3.2	Object types.....	57
6.3.3	Metadata engine	58
6.3.1	Online content providers.....	60
6.4	Data flow.....	61
6.5	Communication.....	62
7.	EVALUATION	64
7.1	Preliminary functional testing by library staff.....	64
7.2	Task-based user evaluation.....	64
7.3	Feedback from library staff.....	67
7.4	Statistics of real world use	68
8.	DISCUSSION	72

8.1	Attainment of objectives	73
8.1.1	Provide information about literature, library services and events to library customers.....	73
8.1.2	Create a digital advertising channel for the library.....	73
8.1.1	Explore the usage of contextual information in the hotspots.....	74
8.2	Future improvements	74
8.3	Summary of the development process	76
9.	CONCLUSION	78
10.	REFERENCES.....	79
11.	APPENDICES.....	85

FOREWORD

This master's thesis was written for the MediaTeam research group at the University of Oulu and it was part of the UbiMetrics research project. The focus of this project was on developing a new enticing digital service for Oulu City library. This thesis received inspiration from previous projects that have used the UBI-hotspots in innovative ways.

I wish to thank my supervisor, Professor Timo Ojala, for devising the work and all of the guidance he gave. I am also deeply grateful to Tommi Heikkinen for the advice and insight on everything technical. I would also wish to express my gratitude to Taru Tanska for helping me with the tedious task of user surveys.

My thanks also go to my dear family members and friends for supporting all these years. And thank you Helena for supporting me while writing this and in my life in general.

Oulu November 28, 2013

Henri Kytökangas

ABBREVIATIONS

AMQP	Advanced Message Queuing Protocol
API	Application Programming Interface
CSS	Cascading Style Sheet
DOM	Document Object Model
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
JSON	JavaScript Object Notation
JSONP	JavaScript Object Notation with Padding
NFC	Near Field Communication
OPAC	Online Public Access Catalog
OWL	Web Ontology Language
PHP	PHP: Hypertext Preprocessor
QR Code	Quick Response Code
RDF	Resource Description Framework
RFID	Radio Frequency Identification
SPARQL	SPARQL Protocol and RDF Query Language
SQL	Structured Query Language
STOMP	The Simple Text Oriented Messaging Protocol
UML	Unified Model Language
W3C	World Wide Web Consortium
XHTML	Extensible Hypertext Markup Language
XML	Extensible Markup Language

1. INTRODUCTION

1.1 Motivation

Traditionally libraries have been the primary source for people to search and locate information and knowledge. The advent of the Internet and the World Wide Web has drastically changed the ways in which people seek and utilize information. Furthermore our methods to access this huge information database are improving constantly. More and more people are relying on technology, like mobile devices, computers or embedded devices, to acquire information about the surrounding world. From this, it is very easy to observe that our information needs and information behavior have changed very rapidly in the past few decades. This in turn has changed the behavior of the customers of libraries. In a study conducted in Finland, 83% of the respondents said they sought knowledge from the Internet rather than any other means, including utilizing library services [1]. No longer are the libraries the primary source of factual knowledge, but rather a channel for fiction literature. This has a radical effect in designing the type of digital services that libraries provide. The services should focus on bringing meaningful information to the user, rather than a meager search of the library's database. The special nature of fiction necessitates a move from the old library indexing traditions, i.e. classifying and cataloguing books, to describing their content. Online bookstores such as Amazon are widely used to discover literature, because they accommodate users' search terms more accurately [2].

At the same time as this transition in our literature seeking behavior has changed, the contextual knowledge of literature is ever growing in the Internet. Hundreds upon hundreds of book review sites, literature services and online book stores collect information on all kinds of printed word. This is not limited only to literature, but to all knowledge. A popular way expressing this is describing it as big data. To tie all this unstructured data together, W3C has proposed to align all information with semantic links to a "web of data" [3]. This Semantic Web is often quoted as Web 3.0 and it will enable new possibilities to easily share and re-use data across application, community and enterprise boundaries. In Semantic Web the information is defined with an explicit meaning, which machines can understand, process and integrate without human intervention.

The impact of social networking on content creation should not be underestimated. One of the most intriguing elements of social information processing is the massive user generated pool of content and data made available online, which can be used in new innovative ways. Until recently this content has been mostly closed inside individual services with no methods to access it. The trend is, however, towards openness and allowing sharing of the information between different entities. This is commonly achieved with open protocols and application programming interfaces. Extending ubiquitous digital services with social networking services and community created content allows the creation of more entrancing and informative services. This social information processing can also be utilized in creating more personalized, targeted and informative content.

Ubiquitous and pervasive computing has been studied for some time now, and large research projects are devoted to discover the actual user needs in different environments. The future trend in big commercial projects, for example Google Now, is collecting characteristics about the user and the environment and providing useful

information and services specifically tailored for the situation. This thesis explores content creation and management, semantic information, context-awareness and social information processing in ubiquitous computing setting.

1.2 Scope and objectives

This thesis introduces the general ideology behind ubiquitous computing and public digital services. It will also discuss the information needs and information behavior of people arriving at a library. Based on the discovery of the importance in content management in ubiquitous services, semantic information management technologies are introduced. To discover what type of digital services are popular in library context, literature review of the current, state-of-the-art ubiquitous services deployed in libraries and literature recommender systems is conducted.

The thesis focuses on satisfying the information needs of library customers with ubiquitous computing technology, more specifically with a service that is mashed up from library databases and online content services, refined with semantic and context information, and provided to library customers on a large public display.

The thesis has three complementary objectives reflecting the needs of three different stakeholder groups. For library customers, the objective is to provide information on literature and library's services in a meaningful way. For library staff, the objective is to provide a digital channel for advertising the library's services to customers. For research organization, the objective is to explore the benefits of enhancing ubiquitous computing technology, particularly services provided on large interactive public displays, with contextual information.

These objectives are pursued by developing a functional prototype of a new context-aware library service dubbed UbiLibrary. The requirements for the service are elicited from the review of related literature, a user study of library's customers, the interview of the library staff, and the analysis of the library's existing digital services. The design and implementation of the service is done with an agile software development process in close collaboration with the library staff. The resulting prototype combines library databases and online content sources into a new library service that can be adapted with semantic and context information. The service is provided to library customers on a large public display than can be coupled with personal mobile phones. UbiLibrary is deployed for production use on the UBI-hotspot located at the lobby of the Oulu City Library. UbiLibrary is assessed with a task-based user evaluation, with an expert evaluation by eight librarians, and by logging real world usage. Based on the findings in evaluation, the attainment of the objects is discussed. These findings are also reflected to the original state of the library's digital services to discover the benefits of the new service.

1.3 Structure of the thesis

The remainder of the thesis is structured as follows. Chapter 2 provides an overview of ubiquitous computing, context-awareness and large public displays, with few examples of library specific applications. Chapter 3 discusses various aspects of information behavior and content discovery in libraries, and managing large amounts

of information with semantic systems. Chapter 4 reports the elicitation of the requirements for UbiLibrary. Chapter 5 describes the design process of UbiLibrary, followed by implementation in Chapter 6. The evaluation of UbiLibrary is presented in Chapter 7. Chapter 8 summarizes the work and provides ideas for future development. Chapter 9 concludes the thesis.

2. UBIQUITOUS COMPUTING ENVIRONMENT

Ubiquitous computing with its characteristics, pervasive computing, context-awareness and the purpose of ubiquitous services are discussed in this chapter. The most relevant issues and challenges are presented and analyzed based on literature. Three real life implementations of ubiquitous services in libraries are presented as examples.

2.1 Ubiquitous computing

Ubiquity is defined to mean the state or capacity of being everywhere, especially at the same time [4]. The definition is very similar to that of pervasiveness, state or quality of a particular thing or place [5], and they are often used to describe the same phenomenon. In the real world, for example a sound that can be heard within a particular room can be considered as pervasive from the room's perspective.

The radical technological development of computers and specifically of the microprocessor has enabled the wide spread usage of embedded devices. In recent decades the early massive computers have shrunk into compact devices that help us in our daily activities. These devices help us organize our lives, search for needed information or entertain among other numerous tasks. The next logical step from this is the phenomenon where computers will become embedded in our environment. Ubiquitous or pervasive computing means that the computation task is embodied in everyday objects instead of computers. In its broadest meaning, ubiquitous computing comprehends any number of mobile, wearable, distributed and context-aware computing applications [6]. When these applications become a part of the environment, they will also be a part of the everyday life. People can do activities like they normally do without really thinking or understanding that they are interacting with a ubiquitous service.

Ubiquitous computing integrates advances from mobile and pervasive computing [7]. All these terms are often used interchangeably, but they have distinct meanings as illustrated in Figure 1. These forms of computing are fundamentally different and they use different methods of managing the services that they provide.

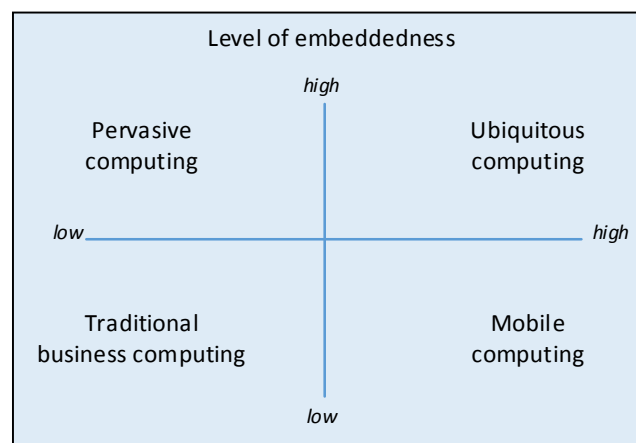


Figure 1. Computation models.

Mobility is also one of the common themes found in discussion of ubiquitous computing. Mobile computing is about increasing the availability to physically move the computing services with us. From a user's perspective mobility often means that the desired data and environment are available from any space, any time. This has the consequence that the computers are taken as ever-present device that expands our capabilities to remember, communicate and reason. The model of usage doesn't considerably change in mobile computing, because the computing device cannot obtain information about the context it is used in. [7]

Another dimension of embeddedness is pervasive computing, which implies that the computer is capable of obtaining information from the environment it is embedded in and builds dynamic models of computing based on the information. Pervasive computing environment can be defined as being saturated with computing and communicational capabilities, while still being invisible to the users. Pervasive computing services can be built by embedding models of specific locations into dedicated onsite computers or by building generic capabilities into computers to detect the surroundings and dynamically build models of their environments. The difference between ubiquitous and pervasive computing is the overall context of the computing. Generally, ubiquitous computing concentrates on devices embedded in the environment and pervasive computing usually centers on services that are used in a certain environment. Satyanarayanan *et al.* [8] identifies four research areas in pervasive computing:

1. Effective use of smart spaces - the software behaves according to the immediate surroundings varying its behavior.
2. Invisibility - minimizing the obstruction to the user and making the technology to disappear from user's consciousness.
3. Localized scalability - the ability to scale according to the complexity of the service, amount of users and the distance to users.
4. Masking uneven conditioning - reducing the variability of different environments and making sure that the user doesn't notice the difference.

The major challenge in ubiquitous computing lies in integrating mobile computing device with pervasive computing functionality. The ideal ubiquitous device, while moving with us, can build dynamic models of the surroundings and adjust itself accordingly. It will also be able to collect and communicate with the surroundings. Furthermore, it will remember the states it has operated and proactively set up services for the environments when the user re-enters them. [7]

Research into ubiquitous computing usually centers on how the information technology can be infused into everyday objects to improve people's lives [9] and what kind of benefits this introduces [10]. A lot of interest and research is going towards developing hardware and software applications for ubiquitous computing. In addition to these engineering and information science subjects, research on ubiquitous computing has continuing interest in human-computer interaction and computer supported cooperative work that draws on psychology, anthropology and sociology [6].

2.2 Context-awareness

Abowd *et al.* [11] define context awareness as follows: “a system is context-aware if it uses context to provide relevant information and/or services to the users, where relevancy depends on the user’s task”. Context-awareness refers to the abilities of services to react to changes in their environment and therefore it is a necessary technical element in creating a ubiquitous service environment.

Context is any information that can be considered relevant to the interaction between the application and user, including the user and application themselves [12]. Context information is used to characterize the situation and qualities of a person, place or object. There are three primary context types. User context refers to any information relating to the user, including dynamic information (e.g. location, current and previous activity, user’s emotion) and static information (e.g. social situation, personal information, habits, preferences). Physical context contains environment’s physical information (e.g. lighting, noise, temperature, humidity level) and device’s physical information (e.g. battery, memory, size of screen, input and output methods). Network context refers to the characteristics of the network the computing devices are connected to (e.g. network capacity, connectivity, and bandwidth). [13]

In general, there are three ways to acquire context information. Sensed context refers to information collected from environment and physical characteristics collected from the user that can be acquired by physical or software sensors. This includes user interaction habits which are collected mostly by software algorithms. Derived context refers to contextual information that can be computed on the fly from other collected information. Explicitly provided context information is received by directly querying it from the user. [14]

The central idea in ubiquitous technologies is that the services are capable of perceiving information, interpreting the meaning and reacting accordingly. From technological perspective, the methods of capturing information defined in pervasive computing can be used to enhance ubiquitous services. The purpose of context-aware computing is that the user’s environment can have an influence on the computing device. When the context of the user’s interaction is detected, the system can provide services or information appropriate to the interaction.

2.3 Large public displays

As the overall cost of digital displays has come down, they have become a more viable option to be used in large scale deployments. Research on how people react and interact with this new technology has blossomed. At the same time, mobile computing devices have grown in features and computational power. By merging these two elements, a new kind of environment with new interaction possibilities and applications is created. This section discusses the challenges and design considerations that have to be taken into account when creating applications in this new environment.

Large public displays differ from traditional displays, such as desktop monitors or mobile screens, in various ways. They offer an ambient large display that is visible from large distances and therefore capable of providing users with opportunistic information. Many academic and commercial real world deployments provide value to users even in a very passive mode of operation. From a public ubiquitous services

point of view, they can be used to display and provide more in depth and interactive applications.

The information processing power of the human brain is limited and the amount of sensory input arriving is greater than the brain can process in detail. The attention of a person selects which of these inputs are devoted the most resources. Visual attention is often modeled with a spotlight metaphor, where a certain region of visual field is selected for more attention. This attention is influenced by either bottom-up processes, for example a suddenly appearing error message, or top-down processes, like the goal of the user to discover a certain service [15]. Because of the limited amount of information processing resources and the massive amount of sensory input in lively city environments, people regularly suffer from phenomenon called information overload. One of the effects of information overload is the allocation of less time on each input and disregarding the ones with low priority. In a way people become more indifferent about their surroundings. Müller *et al.* [16] found out that people tend to disregard large public displays because they feel that the content is uninteresting. Possible users felt that the services in public displays don't bring other value than advertisement and novelty. This is a strong indication that the challenges of large public devices are in the content creation, engagement of users and overall user experience.

Researchers have been trying to figure out ways to inform people about the interactive affordances of large public displays. Interaction blindness is mainly caused by people not knowing that a large screen device can support interaction as most of them are not interactive [10]. Using a secondary object to initiate curiosity has been discovered to be an effective way of overcoming the interaction blindness [17]. It was discovered that that 76% of the participants that were attracted by the curiosity object also moved to the main display. Researchers have often noticed a phenomenon where a single user of a large public display also attracts a larger group of users. This Honeypot effect has been described by Brignull *et al.* [18] in the context of the public display that was demonstrated during a party.

Further challenges arise with the actual usage of large public displays. Many people find it socially awkward to utilize the displays in a public setting. Interacting with a large display is not seen as normal social behavior and this creates an additional barrier to interaction [19]. The perceived value of the service must be high enough for the users even to turn to such display. Furthermore, the location of a display plays a huge role in the success of deployed services. If the immediate surroundings of the display are too crowded, possible users wanting to utilize the service may decide not to, as they don't want to disturb the public or just feel too unsecure [20]. Privacy is a closely related to the issue to the usage shyness. In a setting where a service brings customized information based on the user for everyone to see, it is problematical to decide what information is considered too private to share [21]. People tend to perceive privacy differently. What is the socially correct amount of private information that can be shown on a display? Should the user be prompted about the possibility of displaying the information? It is also evident that even the fact of someone potentially observing what you are using a public display for can be problematic to someone.

Kaviani *et al.* [22] discovered different combinations of using large screen device and a personal mobile device to create a joined user interface. Their work supports the hypothesis that the user interface of large displays and small devices can be combined without a significant effect on user performance. Small device can be hugely beneficial

in displaying private information in this equation. Furthermore coupling a large display device with small device during the interaction helps to reduce the load of information presentation in the large display and it increases users' ability to manage content as the users are accustomed to their personal devices.

2.4 Examples of ubiquitous services in libraries

This chapter briefly discusses ubiquitous services deployed in library settings. Most of the scientific research done in this field focuses on information discovery and learning in a library setting. Many context aware services have been proposed for public settings such as museums and exhibition areas and they are used to aid navigation and offer information on exhibits nearby. However, the user needs are different in a library setting. Library customers are looking for a particular book or literature on a certain subject and providing meaningful information in this context is challenging.

Noh [23] conducted a study on next-generation digital library using context-aware ubiquitous services. Ubiquitous services deployed in libraries in real instances are few according to the results of the study. However, the study argues that future libraries will implement such new concepts as semantic retrieval of information, linked data, cloud computing and context-awareness. Context-aware technology that utilizes the semantic web is also currently an area of great interest. It is possible that in the future libraries contain a context-aware service that can recognize old and new users entering the library and provide optimal services to every situation. Based on the data collected with different sensors, the system can detect user's movement path, room temperature, loan status, user behavior, and so on and then cater to the specific needs of the customer.

2.4.1 *DigiMe - Ubiquitous Search and Browsing for Digital Libraries*

DigiMe is a content suggestion and filtering system deployed in a mobile setting. DigiMe concentrates on the problem of searching and browsing through vast source of information, such as a digital library, with limited resources especially in the ubiquitous environment. The purpose of the application was to provide users with essential library functionality, such as searching the database, managing friends and making bookmarks, in a mobile form. The researchers focused on how to cope with semantic information with limited computing power and applying the ubiquitous paradigm in a real setting. The main challenge of the application turned out to be the amount data transferred between the clients and the server and how long the received information should be stored. In order to minimize transferred data, the researchers implemented a filtering system to the search functionality using Social Semantic Collaborative Filtering (SSCF) [24], which filters the results based on user preferences and the preferences of the user's social group. [25]

UbiLibrary intends, in a similar fashion, to encourage the use of semantic information in a ubiquitous environment. Our approach is not affected by limitations in the amount of data that can be transferred. To achieve this, we promote the usage of already existing information services in conjunction with the library's database. We also aim to emphasize the importance of the social aspect of modern web services. In

our vision, the rich displayed content is affected by the community's opinion. For example, the results of a book search can be modified by the popularity and reviews the results have received.

2.4.1 SmartLibrary

Searching a specific book in large library can be a difficult task for novice library customers and providing personal guidance consumes library's resources. SmartLibrary was developed to help users to locate books by providing map-based guidance to books and collections on mobile devices. A search interface is used to submit a query to the library's OPAC (Online Public Access Catalog) and a list of books matching the query is returned as a result. Upon selecting a book, the user can see its metadata and an illustration of the library's floor plan indicating the location of the book. User positioning is achieved by WLAN fingerprinting. User evaluation confirmed that SmartLibrary aids the library customers in locating books better than conventional shelf classification [26]. The prototype had several limitations, however. The guidance application was slow and users had problems orienting themselves on maps due to low quality of graphics. The redesigned version of SmartLibrary was developed according to the web services paradigm and it supports a wider variety of devices. The improved version allows also creating fixed landmarks to the maps by the library staff. The map-based book positioning service is also used by the database search service to display the location of the book in the library. The user evaluation considered SmartLibrary most useful when used with public desktop terminals. [27]

UbiLibrary will provide a similar map-based location indicator as currently deployed in Oulu University Library. The location of the book is based on the library map and the knowledge of the alphabetical position of the records. However, user positioning will not be a part of UbiLibrary.

2.4.2 Personalized guidance in Intelligent Library

Ching-bang [28] described a ubiquitous system deployed to aid learning in a library environment. The system stands out due to location detection and analyzing the usage behavior of the clients. The system relies heavily on RFID tags embedded in books and users to locate and collect the information. The most interesting part is the Environmental Detecting Agent, which analyzes hidden data in the surrounding environment and provides users with real-time customized information. In order to provide this meaningful context information, the researchers developed three different agents or sensors monitoring the behavior of the clients in the library. The environmental agent is responsible for detecting and recording the movement of individual users. Together with previously analyzed routes it provides routing information to the main application. The book analysis agent collects book statistics, such as purchased quantity, loaning information and popularity. Preference analysis agent detects the user with a personal RFID tag and creates a personal portfolio for more customized recommendations. With these agents, the service is capable of analyzing the connection between the walking paths of the clients and actual physical content of the library to provide intelligent navigation in the library. The paper brings

up considerations for creating services with customized information. The service should, besides analyzing the user for customized recommendations, take into account the environmental factors in the library. Such factors are for example the progression paths of the readers, service locations of the library and the layout of the library.

UbiLibrary aims to be an informational service mainly accessed from a large public display. Therefore, the need to take into account the navigation paths of the customers in the building is not required. However, if the technology would allow, it would be interesting to create a service more directly linked to the physical world. The addition of sensors to the ubiquitous service, such as RFID tags, could be beneficial for user experience.

2.4.3 Interpretation

These experiments clearly demonstrate the need for content management and filtering in ubiquitous services, and especially in library environment where the amount of information is overwhelming. Among them, content itself is discussed thoroughly and the relevancy of the information to the user is constantly promoted. Poor content management reflects badly to the whole system. It is easy to understand that the quality of a digital content management service is dependent on the way it discovers, organizes and connects the discovered data. Therefore, a deeper analysis into methods that people and digital services use to discover and evaluate content is required.

3. INFORMATION BEHAVIOR AND CONTENT DISCOVERY IN LIBRARIES

This chapter discusses human information behavior and information discovery in a library setting. The basics of semantic information are also discussed and two content recommender systems based on semantic information are introduced.

3.1 Information behavior in libraries

The field of information science studies human information needs and behavior on fulfilling them. “Information behavior” is the preferred term used to describe the many ways in which human beings interact with information, in particular, the ways in which people seek and utilize information [29]. In information behavior research, information behavior models are graphical presentations used to describe the interaction with information.

Warner *et al.* [30] studied information needs in urban environments and presented a model identifying four distinct elements in urban information systems: individual citizens, their information needs, the information sources available to them, and the possible solutions to their problems. They identified five barriers in creating environments capable of fulfilling these needs:

1. Societal accessibility: information and the resources necessary must be available through individual’s social system.
2. Institutional accessibility: information sources must be capable to provide the needed information.
3. Physical access to information: the individual must be able to access the information he needs.
4. Psychological access to information: the individual must be psychologically willing to see his needs as information needs and accept that they can be fulfilled.
5. Intellectual accessibility: the individual must have training and ability that will allow him to process the information needs.

Information behavior in the library context can be seen as a distinct case of information behavior in urban environments. Studies show that libraries are used in many other ways than just for loaning books. Therefore, information needs are more complicated than those identified at first glance. Libraries in Finland are used to loan books, read magazines, work, study, and spend time among other activities [1]. The primary information need in libraries is to discover and identify literature. Secondary information needs include the locations of services, events taking place in library, practices in loaning and payments and so on. Information behavior in libraries is a well-studied subject, but most of it focuses on the actual retrieval of a desired book or a piece of information, rather than the more generic needs of library users.

Much of the research in information behavior focuses on how people locate information on school, work or research purposes. In reality, a notable part of the effort in information discovery is spent on normal everyday tasks, such as locating a taxi service number, discovering weather forecast or browsing advertisements for good

sales. In just few short years, thanks to Internet access becoming more prevalent, people are more likely to use digital information sources to reach this information. Due to the popularity of social networks, people also have constant connection to their friends and family and can utilize the collective knowledge of their personal network at any time. [31]

Using personal smart phones for information retrieval and the practicality of constant Internet access has stimulated research on information behavior in libraries. Some studies focus on adapting digital library services to smaller screens and providing mobile OPAC access [32] [33]. Other studies recommend libraries to take a more active stance by providing new equipment and tailored content [34].

Most people regard libraries as their main source of fiction books [1], but how much of an influence does the library environment have on their book choices? Ooi *et al.* [35] discovered that by the time people visited a public library, they tended to have a book title or an author already in mind. They were drawn from people's everyday information sources, such as family, friends, a book club and mass media, and modified by their personal preferences and circumstances. This suggests that the discovery of literature happens primarily outside the library. Library services should reflect this by incorporating information about the user to the services they provide. Some people also resorted to browsing at the public library, focusing their efforts on the "new books" display, the "book returns" section or shelves where specific fiction genres were located. However, premade recommendations made by librarians had minimal effect on the participants in the study.

3.2 Semantic information

User input is one of the primary challenges for content recommender systems. Generally, when users begin the interaction with such a system, they might have a very abstract notion of the type of the information they are looking for [36]. This causes a mismatch between the meaning of user queries and how the content recommender system interprets them. In some cases, the user may have a very high level of abstraction in mind, and is not sure how it should be entered as an input. User might enter a contextual keyword, the name of a similar work or even information described in the work. Conventional definitions of relationships between works usually produce mediocre results. In order to overcome these issues, a more in depth look into how data is organized and connected is required.

The Semantic Web uses standards and tools such as XML, XML Schema, RDF, RDF Schema and OWL. With these technologies it is possible to semantically map and link resources in the internet with each other [3]. For example, a single image element can be described to contain the context of the image, hence linking it to similar images in the internet. Rather than relying on indexing services to describe the content of the internet, the underlying structure of the web is modified to support indexing. With the aid of semantic methods, more complex search systems can be realized.

RDF is a metadata data model used as a general method for conceptual description of resources or modeling information relationships. Using this simple model, structured and semi-structured data can be mixed, exposed, and shared across different applications and domains. It is widely used in describing web resources and knowledge management applications. The basic building block of RDF is an object-attribute-value

triplet; A(O,V). Another way of thinking this relationship is a labeled edge A between the two nodes O and V. This linking structure forms a directed, labeled graph that can be used to discover relationships. [37]

3.3 Examples of literature recommender systems

This chapter focuses on two fundamentally different literature recommender systems that rely on semantic information. There are plenty of internet services dedicated to content recommender systems and it is a field of scientific interest.

3.3.1 *KirjaSampo*

KirjaSampo (“BookSampo”) [38] is a semantic portal for discovering literature that contains the metadata of practically all fiction literature in Finnish public libraries. The system employs semantic web technologies so that the underlying data model is based on content-centered metadata indexing paradigm using RDF. Linked Data principles are used for mapping the metadata in the national FinnONTO ontology infrastructure. The contents are also linked with the repository of a related cultural heritage portal KulttuuriSampo (“CultureSampo”) [39]. In fact, KirjaSampo actually uses KulttuuriSampo as a semantic web service, demonstrating the idea of reusing semantic information without the need for complicated parsing and modification. Most of the content is automatically obtained from existing databases, with the active help of librarians to annotate and correct the metadata. This means that KirjaSampo is independent and disruptive experiment not limited by existing library indexing systems. What makes the KirjaSampo different of similar book recommender systems is the richness of its content. Given a simple query for a book’s name, the system is capable of determining the context of the result, for example setting, author, time period, and characters. While developing the system, researchers discovered that the RDF data model was very beneficial due to its flexibility and the data model was modified multiple times. The major challenge was that the KulttuuriSampo search interfaces could only efficiently deal with text queries targeting a single resource, while the KirjaSampo model required much more complex queries. To resolve this, the search functionality of KulttuuriSampo was split into multiple stages using SPARQL queries. Basically, all SPARQL queries run a set of actions. First, “select” queries are run for every query word. Their results are then filtered with a “mapping” query to match the data types of KirjaSampo. “Filter” query is used to ensure that only abstract books and authors are returned. Finally, this result can be modified with “grouping” query to return an exact amount of results. [40]

Using SPARQL and the inherent amount and complexity of the data make querying the database very complex. From a developer point of view, the effective use of KirjaSampo requires a lot of time and effort to achieve a simple task compared to a public API provided by many web services. On the other hand, the flexibility of the semantic approach is unrivalled.

UbiLibrary uses essentially the same methods as KirjaSampo to collect and analyze data from KulttuuriSampo. Because of latencies and a simpler data model, UbiLibrary will not use the server dynamically like KirjaSampo, but rather collect and enrich

existing information with periodically executed searches. Technically UbiLibrary will use semantic web methods to retrieve information, but the information it will expose will not be designed to match the RDF data model.

3.3.2 *SemPub*

SemPub is a library system for semantic retrieval of medical articles. The system first extracts keywords from the bibliographic references of a medical article. The relevancy of each keyword is determined with the help of the Unified Medical Language System (UMLS) Ontology. The keywords are categorized into two groups. The keywords which have a high frequency in the considered article but are not common in the bibliography describe the novel idea of the article. The keywords which are present both in the article and in the bibliography describe the central theme of the text. The relevant words from various articles are used to determine the subject area of the article. The system functions in three major phases. First, the documental retrieval phase retrieves and parses the documents and saves the information in a database. The indexing phase analyses the metadata content of the articles. Based on the actual words, the article is reduced to a “bag of concepts” on which a semantic tree is built, thus the article is transformed into a graph. The nodes in the graph represent entities or concepts and edges show the relationship between them. Then the system applies a score to the graph based on the relevance of a concept, relation or a concept-relation-concept triplet in a given article. This phase populates the SemPub database, which will be used in the search phase. In the retrieval phase, a user defined search is transformed into a graph of concepts. This graph is then compared against the database and the most relevant articles are returned. This way a highly abstract and complex query of a complex object space can be achieved. [41]

UbiLibrary aims to use a similar strategy of prioritizing keywords that are most prevalent in a set of data. Efficient discovery and navigation of the database can be achieved by analyzing and organizing keywords related to books. The existing keywords in a library’s database can be used to describe the novel idea of a particular book. KulttuuriSampo provides additional keywords and concepts related to a book. These keywords can be treated to determine the central theme in a book. The overall process of SemPub - collect, index and retrieve - will be used as the fundamental approach in UbiLibrary’s search and discovery function.

Both SemPub and UbiLibrary use semantic information to achieve more complete search functionality. This thesis argues that this kind of rich data will prove valuable in enticing library customers and providing a more interesting user experience with UbiLibrary.

4. ELICITATION OF REQUIREMENTS

This chapter describes the different activities conducted to elicit requirements for the new library service.

4.1 UBI-hotspot

UBI-hotspots (hotspots from now on) are large, public, interactive displays installed in several locations in downtown Oulu. An indoor hotspot is one sided with 57” or 65” LCD touchscreen panel, while an outdoor hotspot has two panels back to back. Each unit is equipped with a control computer, a local hard drive, two cameras, an NCF/RFID reader and a loudspeaker. In addition, a display unit may also contain access points for panOULU WLAN and panOULU BT networks. The UbiLibrary service is going to be deployed on the hotspot located in the lobby of the Oulu City library, where every customer practically passes it (Figure 2). [42] [10]



Figure 2. Hotspot in Oulu City library.

The software of the hotspots is designed to allow each hotspot to function autonomously based on its proximity context. The hotspots are networked in a loosely-coupled fashion via event-based communication overlay, which allows hotspots to publish and subscribe to events related to their context. Every hotspot has a local resource manager, which communicates with various software components via the local communication bus. It takes inputs from different context wrappers and instructs the layout manager according to configuration policies. The layout manager is responsible for the dynamic portioning of the screen into website sandboxes using HTML iframes. All of this allows the system have a certain degree of self-organizing in order to avoid complete system initializations due to changing device configurations. [10]

The interaction model of the hotspots is based on three operative states: a passive broadcast mode, a subtle mode and an interactive mode. In the passive broadcast mode the whole screen is allocated to a digital signage service dubbed UBI-channel, where a playlist of advertisements is shown. The playlist is configurable on per hotspot basis. Large public devices are often used in this mode of one-way communication. While in broadcast mode, the system monitors people passing the hotspot. To entice interaction with the hotspot, a visible cue is displayed on the screen when a face is detected. This is referred as subtle state. The purpose of this state is to advertise the interactivity of the hotspot, for example by displaying an animation, current information or a greeting text. The information provided can be personal, but it should be something that the user is willing to share publicly. The threshold of starting the interaction with a public display can be high.

The transition to interactive state is triggered when a user touches the screen. When hotspot is in interactive mode, user is able to interact with the services provided through hotspot's UBI-portal. It is a web portal of various information and leisure services. It is possible to include any web site in public Internet to the portal. It can also be configured on a hotspot-basis to bring location specific content for the users. UBI-portal contains various services created by university research groups and third party developers such as interactive maps of Oulu, up-to-date news, events and weather information, various games, social photo sharing service and interactive art galleries to name a few. In the interactive state the UBI-channel is reduced to occupy the upper left quadrant of the screen. [10]

4.2 Requirements from the library staff

The development of the UbiLibrary service was initiated, because the library staff wanted to remove the hotspot from the library's premises. The staff found the hotspot as an unnecessary nuisance rather than as a value creating service. The services provided by the hotspot did not reflect the services of the library. The staff had observed the usage of the hotspot to be minimal, mainly children playing games and random people trying out the hotspot for the first time. These observations reflect the challenges related to public displays that were discussed in Chapter 2.3.

Our goal was to actively involve the library staff as the customer in the design process. Their requirements and wishes for the UbiLibrary service were collected in a series of meetings and via rapid prototyping of incremental versions. The staff's initial requirements were clearly driven by the library's existing information services and activities conducted on different channels such as the library's website and social media sites. The most desired feature was to use the hotspot as a public channel for this information, including book suggestions and events organized by the library and third parties. Effortless updating and publishing of this information on the hotspot was of high priority. A single publishing portal for all different information channels was seen as desirable, but nearly impossible to implement.

4.3 Analysis of the usage of the library's website

The website of the library contains a comprehensive description of the digital services provided by the library. The website had six different categories of services. “*Book suggestions*” provide a collected list of books for different target audiences. “*Opening hours*” contains information about the opening hours, locations and services available in all of the libraries in Oulu area. “*Services*” provide more detailed information about the services and activities of the library. “*Services regarding payments*” explains the payments and rules there related to book loans. “*Information retrieval*” helps users to locate information from databases and internet at large. “*Library database search*” is a separate service from the library's website, but it is taken into account in analyzing the usage of the website.

The usage data of the website was collected with Google Analytics engine between 1.1.2008 and 30.4.2013. The analysis was constrained to page hit counts and the time spent on a certain page, because those are the most relevant factors for the thesis. The page hit counts illustrated in Figure 3 report the main index page as the most accessed page. The index page contains general information from multiple categories, like current news, book suggestions and library opening hours. It is not directly comparable to other pages as it is the portal page accessed in most sessions. Comparing its hit count to the second most used page (library database search), it is clear that most users use the index page to access the database search. The third most used page is the opening hours while the usage of other pages is diminishable.

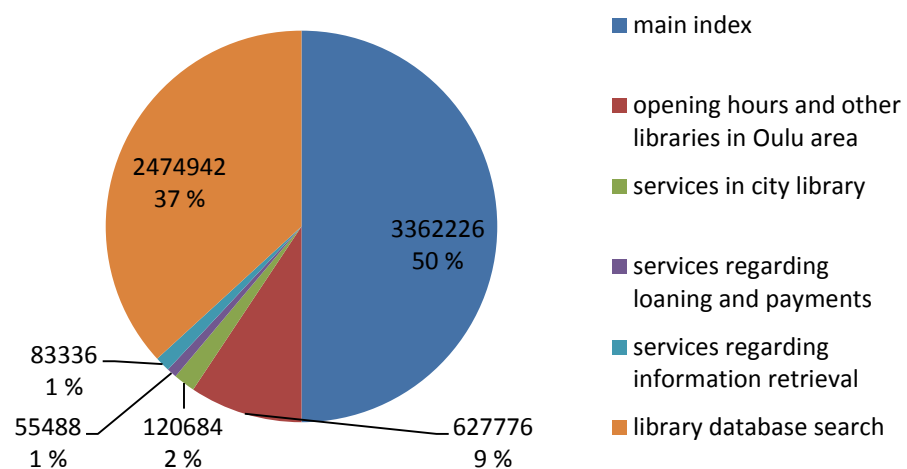


Figure 3. Page hit counts of different pages at library's website.

The time spent on a particular page is an important factor in analyzing the usage and the quality of the design of a website. Figure 4 shows that the users spent a lot of time in the main index. This suggests that the index page is too complex, or the page is accessed and left as a background tab while browsing the internet. The users also spent a lot of time in information retrieval, indicating that this category holds some valuable information. While this page is not accessed very often, the average time spent there is large. Note that the time spent on the library database search is not available, as it is a separate service.

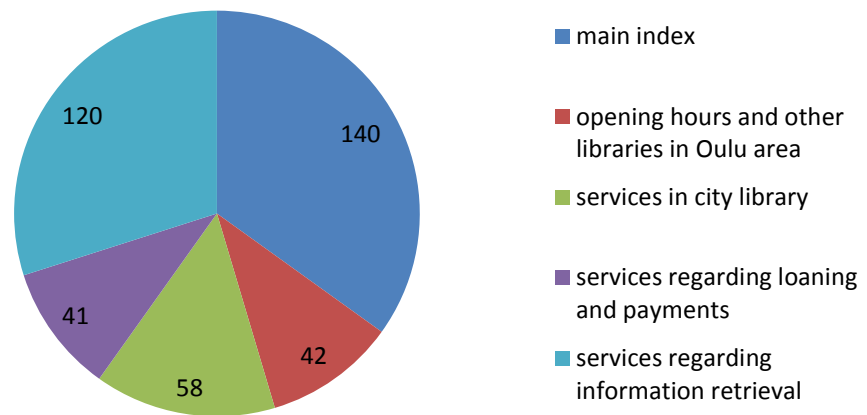


Figure 4. The average time spent on a page in seconds.

4.4 User study of library customers

The needs and practices of library customers were surveyeded with a user study conducted in the lobby of the library. 91 customers participated in the questionnaire study, 55 females and 36 males, whose age distribution is shown in Figure 5. The study focused on the services provided in the city library, not on the current services of the hotspot. The study also explored how people discover content and how they locate it in the library. The questionnaire used in the study is presented in Appendix 1.

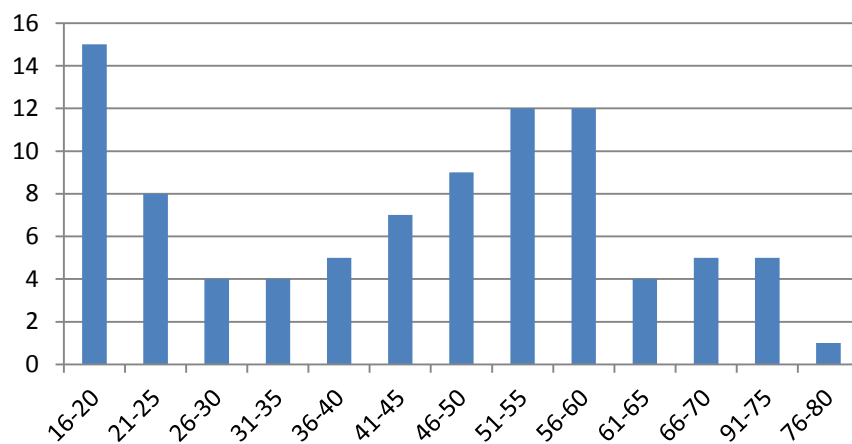


Figure 5. Age distribution of user study participants.

Different service locations and their usage by different respondent age groups are illustrated in Figure 6. The actions taken by respondents in these locations are detailed in Figure 7. The literature department and the newspaper hall are the most visited places, while some locations remain almost unused. This poses the question if people are aware of all services in the library? As expected, the visited places differ a bit between age groups. When asked what they do in these locations, respondents were either reading (12%), spending time (20%) or studying (21%), depending on the age group. A popular activity in every group was collecting reservations and spending

time, while the actual reading had a smaller importance. Young people used the library as a quiet place to study. New literature releases (32%) and movies available for loan (21%) were reported as two most common types of information that the respondents always accessed when visiting the library.

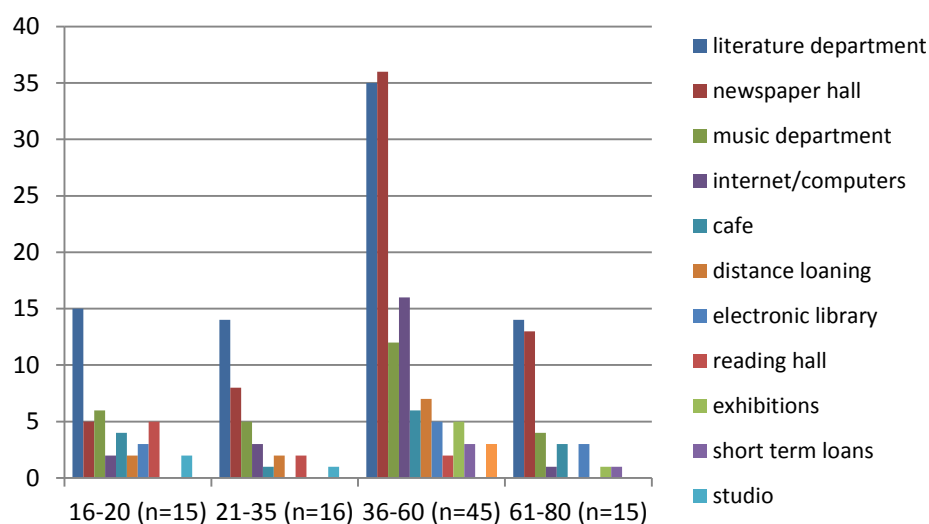


Figure 6. The usage of library's service locations.

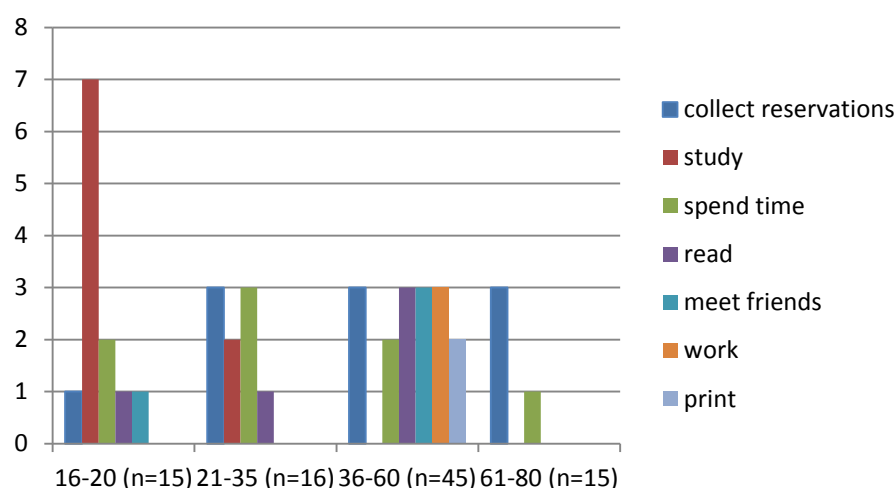


Figure 7. The purpose of visit to the library.

45% of the respondents already knew the piece they were looking for when they arrived at the library, while 35% did not know. Some respondents were not necessarily looking for literature. 23% reported media and 21% suggestions from friend as their method of discovering new reading. Rather surprisingly, specific needs for information (9.9%) or own interest (8.7%) played a significantly smaller role, which again underlines the change of information behavior in libraries. The library's database search engine was the most common (48%) tool for finding the location of a book in the library. Other less common methods were exploring the library (22%) and asking advice from the staff (19%). Four participants suggested that the library's database search engine should accept mistyped words as inputs and two people found it unpractical.

To get an idea what services the customers would want to have on the hotspot, the participants were asked to rank the following services in the order of desirability (5=most desirable, 1=least desirable):

1. Services in library, displays information about different locations and services provided in them.
2. Events in library, displays events arranged by the library.
3. Suggest books from library choices, displays book suggestion from a list updated by the library staff.
4. Suggest books from inputs, displays book suggestion from user inputs such as genre, writer or book name.
5. Locate a book in library, displays a certain book's position inside the city library.

The rankings illustrated in Figure 8 clearly indicate that locating a certain book was the most desired service, while information on library services was the least desired. Book suggestions from inputs were surprisingly popular. It is clear that the respondents were more interested in information relating to books than information relating to services in the library. Multiple suggestions on providing information on events around Oulu, not just in the library, were presented.

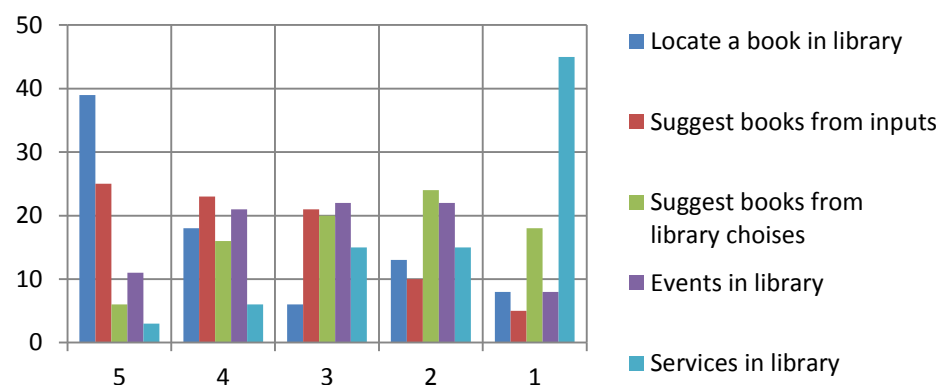


Figure 8. Desirability of candidate services on the hotspot (5=most desirable).

4.5 Other studies of library customers

“En osaa kuvitella elämää ilman kirjastoa!” is a customer study conducted by the national library of Finland in 2013. Although the final report is not yet published, the raw data of the study was available. In the data related to the libraries in the Oulu region, most respondents were female (75%), had a higher education (50%) and were 19-34 years in age (35%). The respondents reported loaning services (96%), reading magazines (62%) and searching information (30%) as their regular activities in the library. The fact that less than third used the library to search information underlines the transformation of library as a service. According to the data, the library is seen as a place where people come to spend time, read or loan books or study, rather than search for factual information. Evidently, most respondents value the loaning service to be the most valuable function. Compared to other libraries in Finland, the customers

of Oulu City library participated more actively in events, spent more time in the library generally and used more other services such as the newspaper hall.

The National Survey of Reading, Buying and Borrowing Books conducted in Australia provides many valuable insights on reading and reading preferences. The most obvious finding is that different people have different reading needs. Males are more likely to read for learning, information, knowledge and interest, while females focus more on relaxation and escape found through reading. Interest in fiction is typical to younger generations, but as readers mature they seem to read more widely both fiction and non-fiction. Among fiction readers, crime and thrillers are the most popular types of reading. Males showed more interest in science fiction, fantasy, war and westerns, while females found romances more appealing. On the other hand, females read a wider variety of fiction than males. Interest in historical novels and war novels increased with age, while younger people preferred science fiction and fantasy. The most popular type of non-fiction is biography, followed by history, cooking, gardening and hobbies. Pleasure and enjoyment became increasingly important as respondents aged and older readers also used reading to fill in time. The role of reading also varied with age and stage of life. While a wide range of factors influenced book selections, the most important factor was word to mouth. Other significant factors were book cover, reviews in media, previous works from the author, the topic of the work, and the good reputation of the author. [41]

4.6 Observation of the hotspot usage

The daily click counts of the library's hotspot before the launch of the UbiLibrary service ranged from zero to about one hundred, demonstrating the rather minimal usage of the hotspot. It is impossible to accurately estimate how many users these click counts translate to, but there are certainly many potential users in the library. Roughly 2500 faces are detected from the video feed of the overhead camera of the hotspot camera every day. The most popular services were map, games and the postcard service. In our view, the content is the most important motivation for people to interact with the display. The services provided must bring a certain value to the customer. According to observations made in [10], people were quite inaccurate in predicting what type of services they find useful. This makes the design process more challenging as we cannot solely rely on customer opinions and desires or the popularity of other hotspot services.

The behavior of the hotspot users was also observed covertly. The motivation to begin the interaction with the hotspot seemed to be mostly curiosity, experimentation or the need to be entertained. In most cases, the interaction began after the user had been standing close to the hotspot for a short amount of time. Most users were not aware of the services of the hotspot and spent some time discovering different services. Usually, the length of a session was short, ranging from few clicks to launching few services. Because the sessions were so short, any kind of lagging or waiting must be minimized in the design phase. Some users perceived the touch input to be so inaccurate that they deemed the hotspot to be broken. This highlights the importance of constantly working software and hardware. Any kind of abnormal behavior deviating from expected behavior quickly drives the users away from the service.

4.7 Essential (abstract) use cases

Based on the above studies, it can be deduced that library customers mainly use three types of services: loaning service, events organized in the library, and other services provided by the library, such as the newspaper room, study room or cafe. When approaching the hotspot in an urban setting, the user may have a variety of needs discussed in Chapter 3.1. However, the focus of this thesis was to design a service for the library so that the hotspot creates value for library customers. A library customer has different information needs discussed in studies detailed earlier. Taken these two definitions of information needs in a public setting into account, this thesis suggest that the potential user for the hotspot has three basic information needs to be satisfied:

- a. Information about the content of the library, such as books, movies, magazines etc.
- b. Information about the services of library, such as opening hours, terms of loaning, etc.
- c. Information about the events and works promoted by the library.

The content displayed on the hotspot should reflect these basic customer information needs. When the user initiates interaction with the hotspot, most of the time the motivation for the interaction is uncertain. The user study of library customers revealed that they are not always certain of the service they are looking for when arriving at the library. This uncertainty of the purpose must be taken into account in the design. To clarify user intention, the basic use cases for the hotspot can be divided into three categories:

1. Determined use cases, where the user already knows the type of information she/he is looking for when approaching the hotspot.
2. Unsure use cases, where the user is not sure of exact information, service or activity he wants, but is willing to find out.
3. Administrative use cases, where the staff of the library wants to update the content of the hotspot.

Table 1 shows the essential use cases for the UbiLibrary service, providing a clear focus on end user needs when designing and implementing the service.

Table 1. Essential use cases

User intention	System responsibility
Locate a certain book in the library (input: book name) (a) (1)	Accept user input Search the library database for the book location Map the location information of the book to a pre-determined position in the map Display position on a map
Locate certain event in the library (input: event category, event) (c) (1) (2)	Load events from the database Display events categorized in a calendar view Accept user input

	Display certain event's details
Receive information from certain books filtered by user inputs (input: book name, category, keywords) (a) (2)	Accept user input Search the database with input Display selected books and their information
Provide feedback (input: textual or choice) (b) (1) (2)	Load feedback questions from the database Accept user input Request confirmation Display acknowledgment
Receive a suggestion for a certain book (input: choice) (a) (1)	Load suggested books from the database Deduct the suggestions for the user Display suggestions categorized Accept user input Display details about the choice
Locate a certain service (input: choice) (c) (1) (2)	Display a map of the library and titles of the locations Accept user input Display detailed information about the location
Display advertising to user (b) (c) (2)	Load and display advertising from the database
Change content in the hotspot (input: content) (b) (c) (3)	Load and display previous content from the database Accept user input Update database with new content

5. DESIGN

The software design process of the UbiLibrary service roughly follows the agile software development model. Agile software development methods evolved as a reaction against the incremental waterfall methods that were regarded as inflexible, micromanaged and heavily regulated. The driving principles for agile software development are described as follows [43]:

- Individuals and interactions – self-organization and motivation are important, as are interactions like co-location and pair programming.
- Working software – documentation is not as important as working software to the client.
- Customer collaboration – constant interaction with the client is mandatory as the requirements cannot be fully understood in the beginning of the project.
- Responding to change – development should be focused on quickly responding to requirement changes. Continuous development of the software is the key.

The main reason for choosing this development model over others was the need to obtain instant feedback from the library staff. It was in our best interest to provide a service that the staff of the library would find valuable. We wanted quickly to achieve prototyping phase for each feature of the service, so that the feedback from the librarians could be incorporated in the final product. Furthermore, the technical limitations of the hotspot made it difficult to know in advance which features would (not) work, hence an adaptive design paradigm would be beneficial.

During the development the library staff had online access to a functional prototype of the system. This proved to be a functional way to receive constant and valuable feedback from the staff. It must be pointed out that the feedback was mostly reactive comments for minor improvements to the prototype, rather than proactive suggestions for fundamental changes in functionality.

5.1 Objectives

The objectives are derived from the needs of three different stakeholder groups: library customers, library staff and research organization.

5.1.1 Provide information about literature, library services and events to library customers

The customers of the library have varying information needs as discussed in Chapter 4. One of the main challenges in information retrieval is the successful input and analysis of search criteria. The objective is to create a service that can collect information about the customer, literature and library services, analyze this data and present it to the user in a meaningful way. Some information about the user can be obtained from the hotspot. Comprehensive information on literature can be collected from the Internet. This thesis develops the hotspots towards an information kiosk, where relevant information is presented in a more compact form in comparison to the

actual digital services of the library to facilitate quick interaction. The driving idea of the service is that when a customer arrives to the library, he can quickly and effortlessly see what kind of content provided by the library is popular at the moment. This content can be for example literature recommendations, events or opening hours. The service should cater for both new and old customers of the library. It should also be easy enough to use so that all different customer groups would be able to use it. Eventually, the information kiosk should have constantly returning users.

5.1.2 Create a digital advertising channel for the library

The main desire of the library staff was to use the hotspot to advertise library's services, events and general notifications to customers. In addition, the service must be able to provide tools to collect feedback from the customers. The staff of the library should be able to update the content of the service with ease – changing feedback questions, collecting feedback and updating event notifications should be implemented as administrative tasks. To avoid additional maintenance, the upkeep of general notifications and events should be integrated with a pre-existing content management system.

5.1.1 Explore the usage of contextual information in the hotspots

All hotspots are contextual in the sense that they are situated in a particular real-world context. The needs of the users of the library's hotspot are specific to the library setting, and possibly radically different from those of other hotspots. The UbiLibrary service explores how the contextual information available on the users and the library's information sources could be exploited. These findings inform the design of future context-aware services for the hotspots.

5.2 Design principles

The following designing principles were identified for the UbiLibrary in the initial phase. These principles have been enforced when designing the architecture and functionality of the service.

5.2.1 User friendliness and responsiveness

The hotspot in the city library is installed in a very central location; hence the UbiLibrary service should blend with its environment and not disturb potential users. If the hotspot notifies its existence to passers-by, it should do it in a socially acceptable manner. Potential users can have a wide range of technical skills. While some people might find interacting with the touch screen natural, others may not understand the purpose of the hotspot at all. Prior training should not be required to use the UbiLibrary efficiently. The information provided by the service must be easily comprehensible and the interactivity clearly demonstrated. While the service may not be immediately

understandable, users should be able to discover meaning through subtle interaction. Graphical design should promote exploration and trying out different features provided by the service. If some interaction is difficult to understand, then the service should support the user with a guide. [44]

One of the main objectives is to achieve returning users. If a user perceives the service sluggish or dysfunctional, he is very unlikely to return. All core functionalities should be always accessible. Even when processing and indicating loading, the service should be usable and the user should be able to navigate where he wishes. If the perceived value of the UbiLibrary service is lower than any alternate service accessible with personal mobile phone, the user is likely to use the alternate service. In other words, to justify its existence, the UbiLibrary service should be dramatically faster and simpler to use “on-the-go” in the library. Interaction design should be designed to support ad hoc interaction where initiating and ending a session should be seamless, without the need to log in or register.

5.2.2 Easy deployment, maintenance and error handling

UbiLibrary is targeted to be deployed in real world setting at the library where it is going to be maintained by the library staff. The maintenance of the service content must be simple so that a person with limited technical knowledge can do it. This means that administrative tasks must be designed and programmed as features of the service, rather than only providing access to the database to modify content. UbiLibrary should be easy to deploy on a wide range of suitable devices. The client should require only a standard web browser with JavaScript support, which is found in practically every browser nowadays. The server on the other hand needs a database and a webserver to function. It should be possible to only deploy a single server that all clients use. Finally, error handling must be robust so that the service is capable of reverting to usable state, if there is an error.

5.2.3 Use of contextual information to stimulate interest

UbiLibrary aims to entice user interaction over longer periods by providing something proven effective, i.e. user created content. Most content such as literature reviews is meant to be collected from social literature services. Book covers and descriptions should be pulled from various third party services. Library information should be parsed from library’s existing digital services or created by the library staff. The service should utilize the context information provided by the hotspot, in particular the gender and age of the user, in adapting the information provided to the user. This way, users are provided new personalized information instead of static “one size fits all” content.

5.3 UbiLibrary’s conceptual states

The default interaction model of the hotspot can have three distinct states: broadcast, subtle and interactive [44] [45]. UbiLibrary introduces three additional states:

distributed state, user specific state and group specific state. These states are presented in abstract form, so that all different inputs and factors are not taken into account.

Broadcast state

This is the default state of the hotspot in the passive (non-interactive) mode, where the screen is dedicated to the UBI-channel digital signage service. The playlist of the UBI-channel is updated periodically. This state supports the read'n'go type of interaction [46]. In the library hotspot this state can be used to promote different events or services of the library. One of the main requirements from the library was to enable advertising in the hotspot. The simplest way of achieving this is to use the existing UBI-channel and allow the staff of the library to update the playlist.

Subtle state

The hotspot changes from broadcast state to subtle state when a face is detected from the video feed of the overhead camera. In this state the hotspot is waiting for the user to begin the interaction by touching the screen. The interactivity of the hotspot is communicated to the user displaying an animation. Further, suitable information or a greeting can be showed in the subtle state. If the user does not start interacting with the hotspot within a predefined period of time, the hotspot reverts back to the broadcast state. In the library hotspot, the subtle state could be harnessed to show book covers or event notifications, for example.

Interactive state

In the interactive state the user stands in close proximity to the hotspot and uses the services of the hotspot typically via the touch screen. The ability to interact with the screen creates new possibilities for creating more complex services. The screen has a lot of real estate for presenting user interfaces. Therefore, it is imperative that the screen is divided to variable sized parts which contain different types of information and interactivity. In the library hotspot this state can present information about the library and its services.

Distributed state

In the distributed state the interface is divided between the screen of the hotspot and a mobile device [47]. This requires pairing the mobile device with the hotspot which can be achieved with a QR code, Bluetooth or NFC. Kavani *et al.* [22] suggested using a small device (SD), usually user's mobile phone, as an input to large display (LD), i.e. a large public display. In the library hotspot, distributed state can be employed to save information discovered in the hotspot into the user's personal mobile phone, for example the title and location of a book the user has discovered.

Group specific state

Market segmentation is a widely used concept in economics and marketing. A market segment is a subset of a perceived market made up of people or organizations with one or more characteristics that cause them to demand similar product or services based on qualities of those products such as price or function [48]. Every segment is different, so different customers have different ideas of what they expect from the product and the content should reflect that.

The face detection software in the hotspots allows detecting the gender and the age group of the user. This information can be employed to create a group specific state where the content of the hotspot is customized for the group. The library staff took a negative stance towards the idea. According to them, while really broad literature recommendations can be made according to age, it is generally discouraged. Another challenge is privacy, as some people might find the suggestions embarrassing. Nevertheless, they allowed the service to be adapted according to the face detection data, if it is done in a way that it does not annoy the users and user has the ability to remove the customizations and predetermined content made by the system.

User specific state

In the user specific state the service is assumed to know the identity and preferences of the user. This requires identification of the user, which can be coupled with the pairing of the user's device with the hotspot with the preferred method. Alternatively, the traditional combination of a username and a password can be used. After identification the system tracks all choices made by the user, which allows it to adapt its own behavior and to provide improved literature recommendations.

The library card would be the most natural identification method in the library. It would allow fetching the loaning history of the user from the library database which in turn would facilitate more accurate literature recommendations. Because of technical limitations UbiLibrary does not use this state, but it is mentioned here as a possible future improvement.

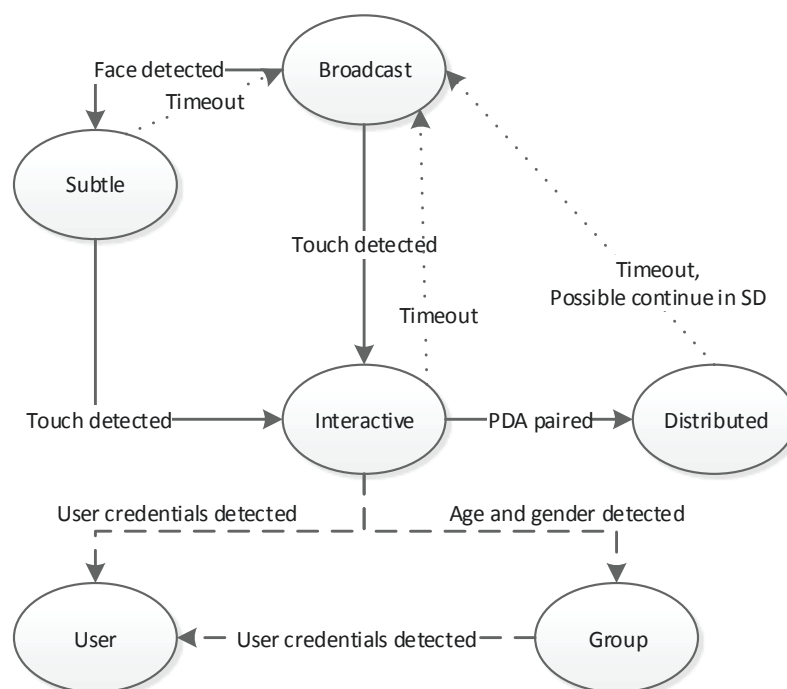


Figure 9. Conceptual states.

State machine

Figure 9 shows the relations of the different states and the required interactions between different states. The group and user specific states are descendants of the interactive state, as they provide modifications to the default interactive state, rather than changing the fundamental idea of the interactive state.

5.4 Usage scenarios and use cases

More detailed requirements for UbiLibrary are gathered from usage scenarios which are then detailed into use cases. Only those use cases that involve a system component are included. These usage scenarios and use cases are derived from the abstract use cases and requirements presented in Chapter 4.

5.4.1 Administrator

Scenario: Update the content of the event feed

Amy, a library employee, has received a new announcement she would like to add to the hotspot. She fills required information about the announcement to the system. She also needs to see a preview of the new event announcement before updating it. This announcement is automatically removed once the duration is exceeded.

A1. Create event

Administrator fills in information – title, description, picture and the end date – and the sample preview of the event is displayed. The title field is mandatory. The end date will be selected from displayed calendar. Upon pressing update, the system will respond with the event title and confirmation.

A2. Remove event

Administrator selects the desired title of event from a list of all events to be removed. Upon pressing update, the system will respond with the removed event's title and confirmation.

Scenario: Review and update the user questionnaire

Amy, a library employee, wishes to review the answers in a questionnaire. After seeing the results, Amy decides to add a new question and remove a question that no one answered and add a better question. After filling in the details about the new question she receives a confirmation of successfully creating a question. She then selects the question to be deleted and receives a confirmation. Other feedback saved in the system was not that interesting, so Amy decides to remove the gathered feedback from the system.

A3. Create question

Administrator fills in question title, question type and possible choices. Possible question types are freeform text, multiple choices and a single choice. Possible choices for answers are defined in freeform text fields. Upon pressing update, the system will respond with the question's title and confirmation.

A4. Remove question

Administrator selects the desired title of question from a list of all questions to be removed. Upon pressing update, the system will respond with the removed question's title and confirmation.

A5. Remove answers

Administrator selects all of the feedback results to be removed. The system responds with confirmation.

A6. Show answers

Administrator selects results to be viewed. The system responds with list of answers containing the question, answer and time of the action.

5.4.2 User***Scenario: Find a book suggestion***

John, a customer of the library, begins the interaction with the hotspot in order to find something to read. He started the interaction, because he saw an advertisement of a certain book in the broadcast channel. He navigates to a section where different suggestions are listed for different user groups. After selecting a certain category, John receives a list of matching books. After selecting a desired book from this list, John still needs more information about it. First he quickly checks to see if the book really is present in the library. The message displayed indicates that two copies are present in the city library. Then he proceeds to read the summary and details about the books. Still not convinced, he searches for reviews that other people have left about the selected book. John wishes now to locate this new found book in the library. A map of the library is displayed to him, and the approximate location of the book displayed in them map.

U1. List book suggestions

User selects the application to display suggestions. Categories of different suggestions are displayed. After selecting a certain category, a list of books is displayed in other part of the application. Book suggestions are fetched from various social book services available in the internet.

U2. Show additional information for a book

User selects the title of the book he wishes to acquire additional information. The system displays a short description, publisher, length of the book, class of the book and scoring. Reviews, availability, keywords are displayed as additional interactive components. Additional information relating to the book is fetched from various 3rd party web services and saved locally.

U3. Show availability for a book

After selecting to see the availability of a book, the system will fetch the availability of a book from library database. The availability of the book will be shown for the city library and all libraries in Oulu area. Availability will not be saved locally, as it is dynamic information. The system also creates a QR-code that contains a link to the

Outi-system. From this link, it is possible for the user to make a reservation using his mobile phone.

U4. Show reviews for a book

After selecting to see reviews for a book, the system will fetch reviews for this book from various web services and display them to user. This information will not be stored locally.

U5. Locate book

User selects the class definition of a certain book. System displays a map of the second floor of the library. In this map, the approximate location of the book in the library is displayed. The approximation should be at a bookshelf level.

Scenario: Locate a certain event in the library or Oulu area

Anna, a tourist visiting the city library, starts the interaction with the hotspot in order to find something interesting happening in Oulu area. She navigates to events section, where she sees the current events in the library. She is more interested in theater plays. She selects the corresponding category and the events are displayed for her in a calendar view. She selects some of these events, until finding the most suitable. She then wants to see all of the details relating to the event and the original event notification is displayed for her.

U6. Show events

By default, all of the library's events are displayed in the calendar. When user selects a certain event, title, a sort description of event, price and location are displayed in other part of the application. It is possible to change the calendar to month, week or day views. The category of the events is selectable from the content part. The events are fetched from 3rd party services and saved locally.

U7. Show event details

After selecting the details of an event, system retrieves all of the information specified in the original event announcement and displays it to user. This information is not stored locally.

Scenario: Give feedback

Mary, a regular customer of the library, is displeased in the current state of ordered newspapers. She navigates to a freeform text input and writes the feedback with virtual keyboard. She would also like to receive an answer to the given feedback by providing her e-mail address. At the same time she answers some of the questions in the library questionnaire.

U8. Give feedback

User navigates to feedback application and fills in the answers. Answering to questions is optional; user can choose witch to respond to. The questions displayed are dynamic, and modifiable by the library staff. The possible question types are multi choice, single choice and free textual input. After sending the feedback, the answers are saved on question basis and user receives a confirmation of the received feedback.

Scenario: Receive a content suggestion

While filling the feedback, the system was capable of detecting Mary's characteristics. A visible cue is displayed in the screen and text appears to indicate the content of the cue. Mary presses the cue, and receives a list of books that the application has determined interesting for her. Also there is an advertisement of a local event and a service in the library. Mary selects the event notification, and the system navigates to U6. Display events.

U9. Suggest content

System detects the gender and age of the user with the camera detection software. It shows a visible animated cue and waits for user to interact with the cue.

U10. Suggest books

System displays a list of books it recommends to user based on the characteristics of the user. These recommendations are received from analyzing different collections in 3rd party services.

U11. Suggest events

Different user groups are interested of different events. System suggests an event based on the characteristics of the user. This suggestion contains the title and the location of the event.

U12. Suggest services

Different user groups of the library use different services. System suggests a service located in the library based on the characteristics of the user. This suggestion contains the title and small description of the service.

Scenario: Search for a certain book

John returns to the hotspot in order to locate a specific book in the library. He inputs the authors name and a list of books from the author and keywords relating to these books are displayed. John then selects the book he is looking for, but the score given by the application to the book is low. John then selects a keyword relating to the books displayed and removes the author's name from the inputs. A list of books corresponding to the keyword is displayed. John then compares some of books, and decides to add a keyword specific to a certain book. He opens the details of a book and selects a certain keyword specific only to this book. The list of books and keywords are updated again. John now finds a highly recommended book and decides to borrow it.

U10. Search by input

User inputs a text with the virtual keyboard to the system. System searches this text from all the possible values that the books stored in the local database has. The possible hits are displayed as a list in other part of the application. The selected inputs are displayed in a list close to input field. User can remove selected inputs from this list.

U11. Search by keyword

A collection of keywords, selected keywords and the free text input is displayed in the application. User can now update the books list by adding more keywords by selecting

them from the collection of keywords, or by removing them from the selected keywords list.

U12. Show book keywords

When user opens the details of the book, he can select the keywords specific to the book to be show. The system displays the keywords, book name and author name as interactive items. When any of these items is pressed the system updates. This allows navigating the system with author names, keywords or searching different editions from a certain book.

Scenario: locate a certain service or library in Oulu area

Peter, a new customer to the library, wishes to locate some music notes in the library. Being a new customer he has no clue where they are located. He begins the interaction and navigates to the services of the library. A map of the building and list of departments is displayed to him. Peter selects the music department from the map and a description of the department is shown. Peter now selects the displayed “music notes” and the location is highlighted in the map. Peter also wants to know which the closest library to his home in Oulu area. He navigates to other libraries section and selects the closest library location to his home shown in the map of Oulu. The general information about this library is also shown.

U13. Show library departments

By default, system displays the map of the first floor of the library building. User can now select the desired department from the map or from the list of departments. The selected department is highlighted on the map. The content of the department is displayed in other part of the application.

U14. Show service details

When user has selected a department to be shown, he can also select any of the services in it. By selecting a service, the location of the service is highlighted on the map and a description of the service is shown.

U15. Locate a library

A map of Oulu containing library locations is displayed in the application. By selecting a certain library from the map, the system will show the details of the library in other part of the application. User can also select the desired library from the list of libraries. Then the system will show the details and highlight the position on the map.

5.5 Use case diagram

The relations of the use cases are illustrated in Figure 10. From these use cases we can observe the functionality required by particular interactions.

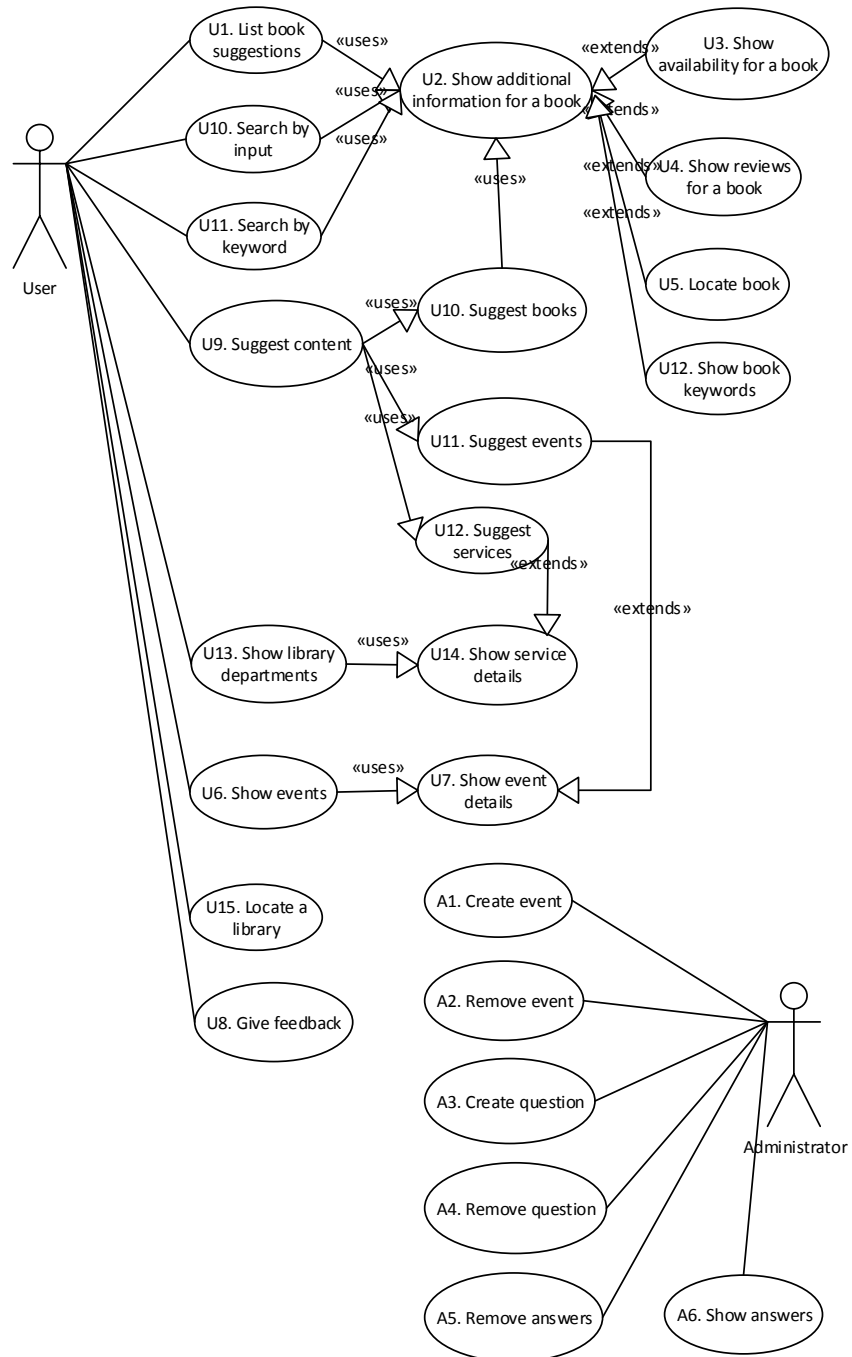


Figure 10. Use case diagram.

5.6 Technical requirements

Table 2 lists the technical requirements for UbiLibrary. While the user stories are the specification in agile software development, these requirements are derived directly from the use cases.

Table 2. Technical requirements

ID	Requirement	Use cases
T1	Server: permanent database	functional requirement
T2	Server: web server	functional requirement
T3	Simple JSON based messaging protocol	U1,U10,U11,U13,U6,U15
T4	Hotspot event collector	U9
T5	Functional map of Oulu	U15
T6	Access to library's database	U3, U2
T7	Access to additional information services	U4, U10

5.7 Conceptual design

Figure 11 shows the conceptual design of UbiLibrary. The key functional components include a public display showing the client interface, library databases, and a content server hosting the client software and other functionality such as the content recommender engine that utilizes online content providers.

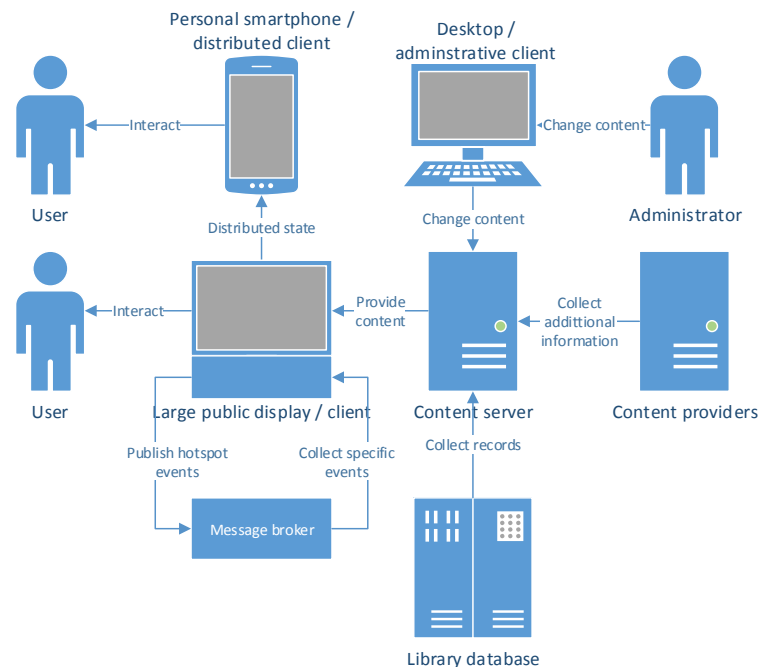


Figure 11. Conceptual design.

UbiLibrary follows the client-server architectural style, where server components serve the requests of the clients. Content server contains the web server hosting the

client and server core that collects information from various online sources to a persistent storage. It publishes the information it provides via simple messaging protocol. When the Content server is given queries from the client, it will perform intelligent searches on the content. Content server is designed to handle all of the possible clients in the service.

Library database contains all of the records that the Content server makes suggestions on. These records are enriched with additional metadata collected from online content providers that often have public APIs to facilitate easy access.

Clients are touch enabled devices that are equipped with a web browser. To be able to use the advanced functionalities and states – subtle, distributed, broadcast and group – the client needs to be deployed with the UBI-hotspot software. The client fills its temporal data storage by communicating with the server at the startup of the service. This temporal data is used to display the default content of the client. It is enhanced gradually with more information from the server as the user navigates in the client software. Distributed client is an implementation matching the distributed state, where the user initiates a session in his personal mobile device. The session is initiated by pairing the client software with distributed client. Administrative clients can change some of the content in Content server with predetermined tools.

The UBI-hotspot software is included in every hotspot. It provides the execution framework for all services and applications. The software manages the states of the hotspot, events and the communication with the hotspot. Message broker is a component acting as a gateway between two communicating applications. The purpose of the broker is to receive messages, perform some actions on them and send them to the intended receiver. The hotspot software publishes events about its state and context (e.g. state transitions and camera events) to the message broker. The clients attach themselves to the broker by subscribing to certain events. After subscription, the message broker makes sure the correct messages get passed to the correct clients. Based on the information of these events, the client dynamically modifies the content shown to the user.

5.8 Content suggestion mappings

Based on the findings presented in Chapter 4.5, the used topmost categories of books in the database of Oulu libraries were mapped to different age and gender groups as shown in Table 3. These mappings are later utilized in creation of content suggestions.

Table 3. The mapping of book categories to age categories

Female child	children's books, picture books
Male child	children's books, picture books
Female teen	young-adult fiction
Male teen	young-adult fiction, games, fantasy
Female young adult	psychological thrillers, fantasy, romance novels, romantic suspense books, entertainment literature, young adult's books
Male young adult	fantasy, science fiction, adventure, horror, young adult's books, thriller

Female adult	biographical novels, women's literature, development novels, cook books, gardening books, romantic suspense books, entertainment literature, historical novels, thriller
Male adult	wars, biographical novels , short stories, historical novels, thriller, crime
Female senior	memoirs, biographical novels, romance novels, historical novels, thriller
Male senior	memoirs, war literature, nature stories, historical novels, thriller

5.9 Graphical user interface design

One of the key challenges in designing user interfaces for large screens is the placement of interface elements. Large-scale displays can be used to display large quantities of information, but some of it may be located outside the user's visual field or reach [49]. Consequently, space and layout management on large displays is different from that of the conventional desktop environment. For example, the traditional way of grouping applications in windows, and dividing and overlapping them across the whole screen is problematic and in some cases irrelevant for large screen devices. Bezerianos *et al.* [50] proposed to divide the screen to focal and portal areas. The focal area is the representation of the canvas within which interaction occurs while the portal area is the remote part of the main display canvas on which the portal area's view is centered. This allows managing a larger number of applications in a particular location of the screen.

This thesis adopts the general idea of focus and portal areas for managing the contents of the UbiLibrary service, but in a more static manner. The primary goal is to divide the virtual canvas from one to three focal areas where the main interactivity takes place and the most relevant information is displayed. As illustrated in Figure 12 left, in the default minimized state all applications are minimized and displayed in a single focal area, where they are easily accessible. Non-interactive information can be displayed outside the focal area. When a single application is launched, the system moves to maximized state, where the canvas is divided into three focal areas. This expanded layout can be thought as an application menu tree, where the leftmost focal area is the index of applications ("main menu"), the central focal area is the application menu window ("menu window") and the rightmost focal area is the application content window ("content window").

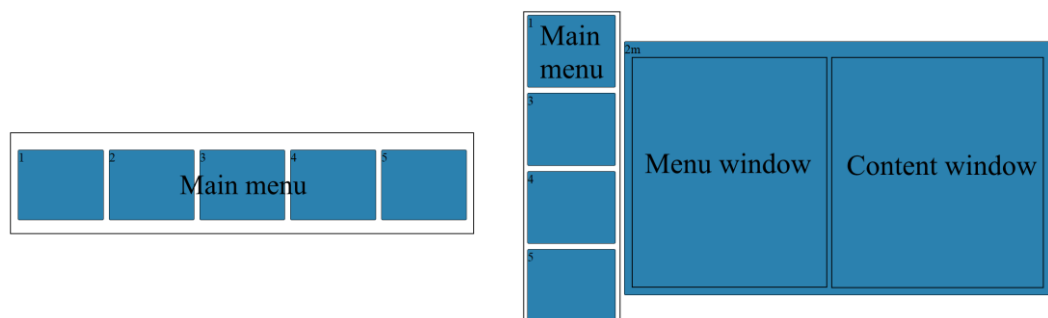


Figure 12. UbiLibrary's user interface minimized (left) and maximized (right).

Another challenge for the interface design is presented by the technical limitations of the touch screen foil of the hotspot. Empirical tests revealed that the spatial accuracy of touch events was about 1 cm, which translates to about 40 pixels. In the context of layout design, this means that even the smallest visual interactive element would have to be large compared to the actual screen size. This adds another dilemma to the design of the user interface design: how do we present textual content and interactive elements in a way that does not clutter the screen with many relatively small elements? The solution is to add content and interactivity to as many elements as possible as demonstrated in Figure 13. For example, a button in a default state may contain a small subset of information and display full information when clicked.



Figure 13. Minimized and maximized UI component.

Writing text using a virtual keyboard shown on a large touch screen is complicated for two reasons. First, spatially inaccurate touch detection can lead to many false key presses. Second, on a single-touch screen typing must be done one character at a time. These problems can be alleviated to a certain extent with text prediction with a set of valid prediction words that are derived from the context of the input field. This thesis proposes to use tag clouds as an alternative input method, where the user has to select a single input from a cloud of candidate keywords. Tag clouds are particularly useful when browsing for non-specific information or as a visual interpretation of a database [51]. In case of displaying information that has an order of precedence, the most preferred choice can be made larger than other choices, which aids in understanding the content. Tag clouds are not particularly suitable for searching specific information.

6. IMPLEMENTATION

This section describes the process for creating the client and server implementation. After the system architecture has been presented, more in depth solutions for client applications are presented. Last, this section introduces different server components, data structures and communication protocols.

6.1 System architecture

UbiLibrary is composed of three main logical parts illustrated in Figure 14: the client interface shown on some screen, the web server hosting the client software, and the server core. The *client* software implements most of the service logic including user interaction, graphical user interface and application state flow. The hotspot publishes the events it gathers from its devices via AMQP messaging protocol and those events are collected in the client software with WEB-STOMP adapter. The *web server* provides the client interface to different physical clients, which in this thesis correspond to the hotspot located in the library. The *server core* provides a permanent storage for service content that is published to the clients via a simple messaging system. The server core employs various content crawlers to populate the storage. The crawlers use different public APIs to discover and fetch content from content providers in the internet. The server core occasionally runs a set of scripts to update the content storage.

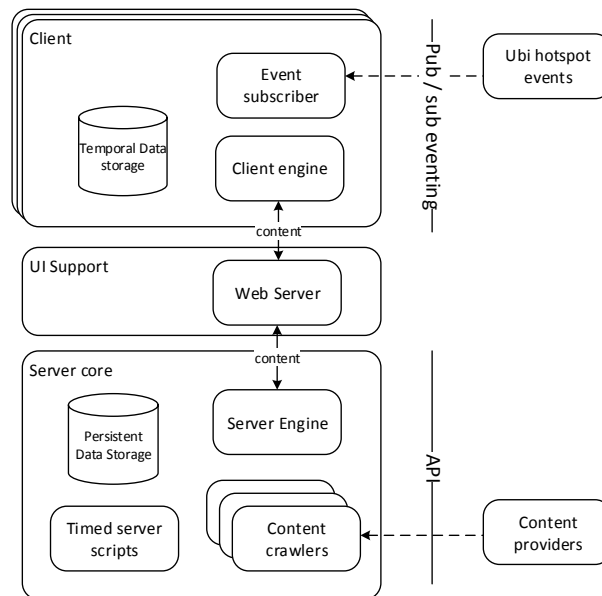


Figure 14. Overall architecture of the UbiLibrary service.

6.2 Client implementation

6.2.1 Applications

Figure 15 shows the tree structure of the interface, which helps in understanding the relationships between the main and the complementary applications.

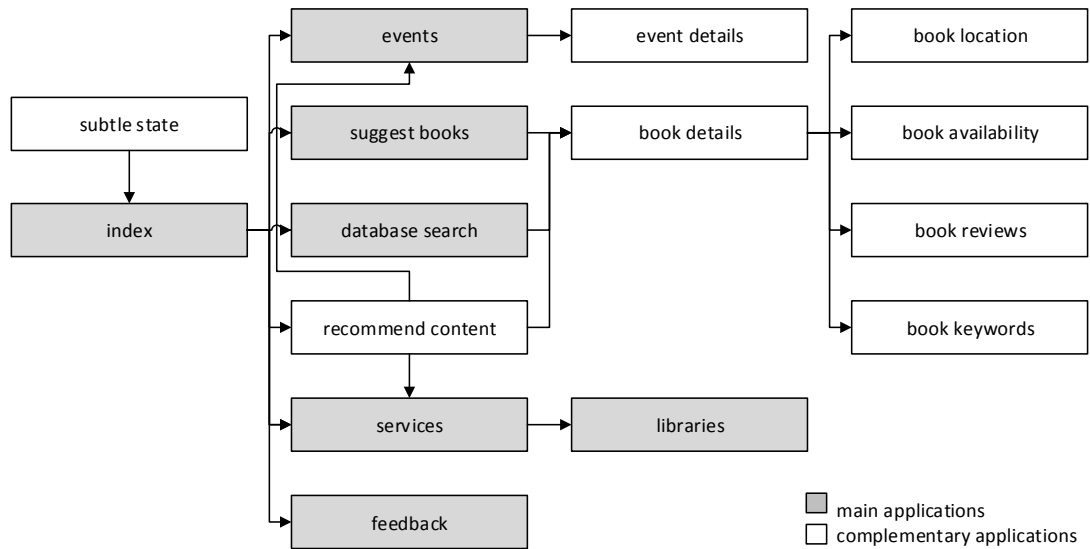


Figure 15. Tree structure of the user interface.

The use cases described in Section 5 are realized by a set of applications. Six of the main applications are always visible in the main menu, one main applications can be accessed from another application and eight complementary applications are launched when required. Some of the applications share common functionality. An administrative application is provided for the library staff for maintaining the service. At the bottom of the canvas, a toolbar displays time and date, required logos of the third party services, theme changer, main menu button and UBI-portal button. Selected screenshots of the applications are shown in Figure 16. Most applications are provided in the interactive state, but the subtle and group specific states are also used.

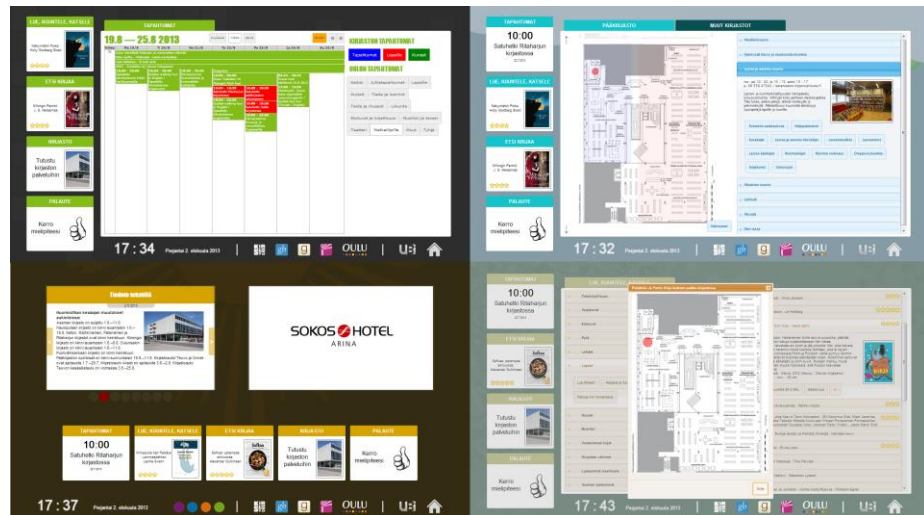


Figure 16. Example screenshots of the applications.

6.2.1.1 Main applications

The seven main applications are index, events, suggest books, feedback, services, libraries and database search.

The purpose of the *index* application (Figure 17) is to provide an easy to find starting point for user interaction. It is the default view, when the hotspot moves to interactive state. Index contains all main applications in the minimized state of the user interface. Index holds the main menu in a focal area centered on the screen. The different menu items advertise their content in the form of a picture or a textual description. The content can also be dynamic. For example, the events application advertises the next event in the library in its menu item. The *announcement channel* is implemented as a part of the index application as a secondary area. This channel contains the announcements created and maintained by the library staff with the administrative application. The *UBI-channel* containing library's advertisement is minimized to the index application, when the hotspot moves from the broadcast or subtle state to the interactive state.



Figure 17. Index application.

The *events* application (Figure 18) contains a calendar that the user can modify to show events of a particular month, week or day in the menu window. The event data is fetched by the server from the City of Oulu’s event calendar [52] and the library’s calendars. Different event calendar categories can be switched on or off from the content window. When a single event is selected in the calendar view, the event’s details are displayed in the content window. Here it is also possible to show the original web page of the announcement in a popup window.

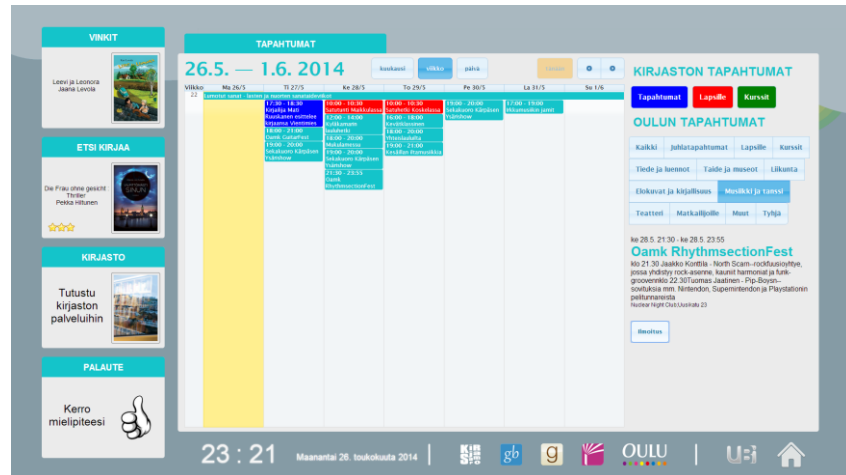


Figure 18. Events application.

The *suggest books* application (Figure 19) displays a list of suggestion categories in the menu window. These categories are collected from various third party services and analyzed by the metadata engine. The most loaned books in the library is an example of a category. When a category is selected, a list of matching books is displayed in the content window.

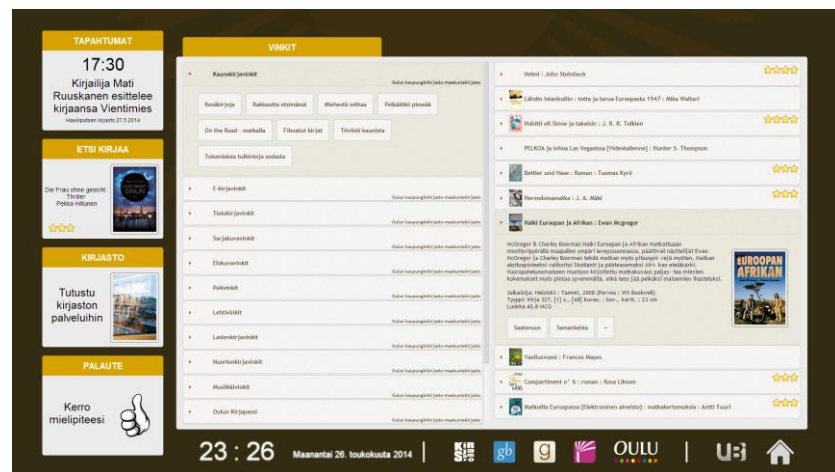


Figure 19. Suggest books application.

The *feedback* application (Figure 20) lists the questions specified by the library staff in the menu window. Users can also leave a freeform text feedback in the content window.

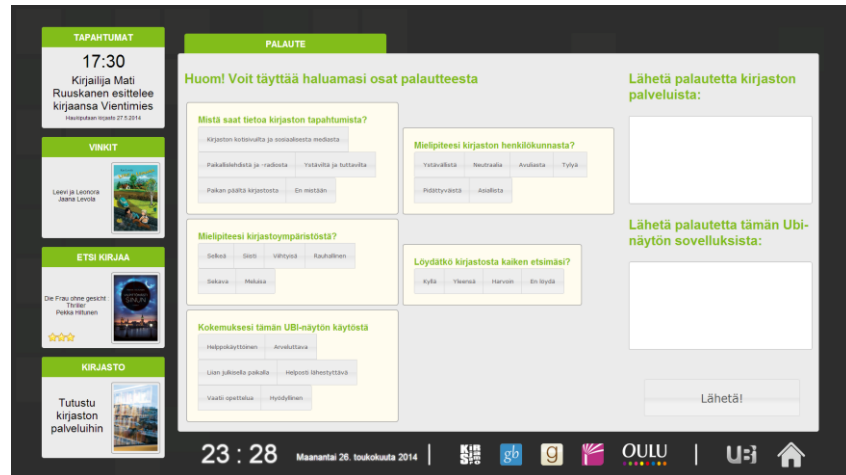


Figure 20. Feedback application.

The *services* application (Figure 21) displays the floor maps of the library in the menu window. When a particular department is selected from the map, its description is opened in the main window. This explanation contains a brief description of the department, including opening hours and links to all services in the department. When a service is selected, its location is highlighted on the map.

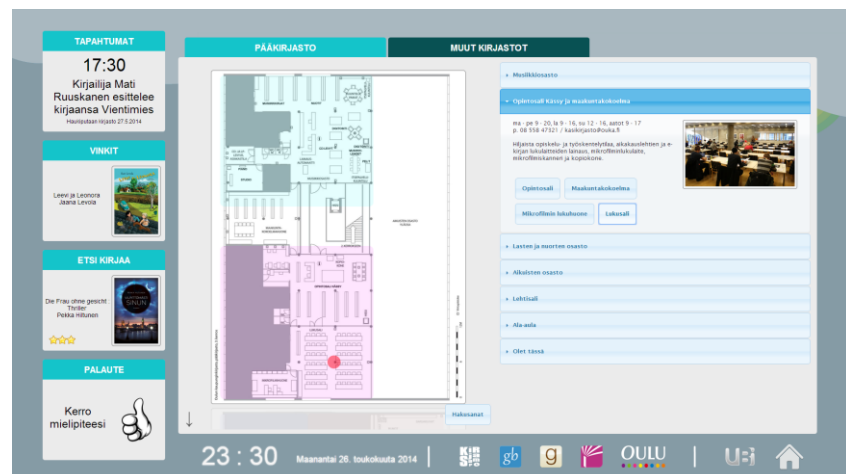


Figure 21. Services application.

The *libraries* application (Figure 22) shows in the menu window a map of Oulu where the locations of the libraries are highlighted. When a library is selected from the map, a short description, address and the opening hours of the selected library are shown in the content window. The interactive map is realized with the Google Maps API [53].

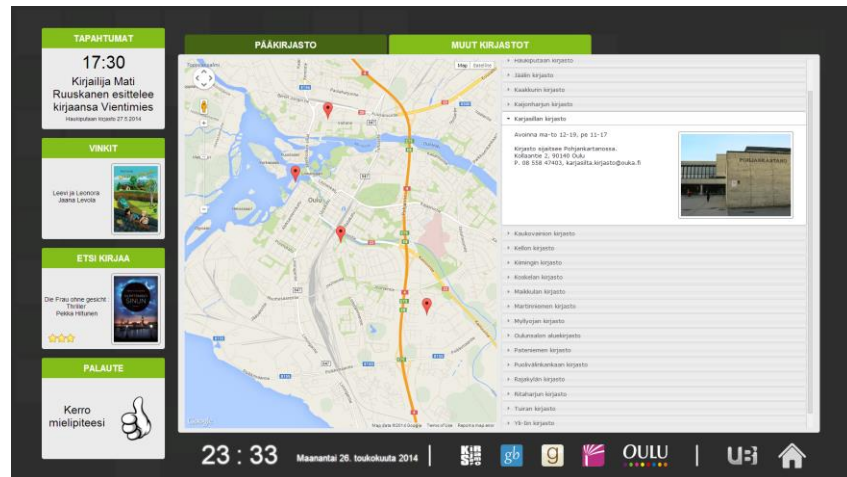


Figure 22. Libraries application.

The *database search* application is illustrated in Figure 23. When one or more keywords are selected from the tag cloud shown in the menu window, the client queries the server for the selected keyword(s). The server returns matched results sorted by their review rating and keyword matches. 12 results are displayed on a single response list with the possibility to load more. The user can also define a keyword by writing it to an input field. This input can be a keyword or the name of an author or a book or a part of those. The user can also remove a keyword by deselecting it from the keyword list. These actions allow users to navigate the library's database with moderate ease. The search functionality works in two ways. It allows the users to discover books by selecting keywords of interest and by displaying related book details in the content window. It also allows users to search the database for exact matches. The database search application relies heavily on the metadata engine described in the server implementation. The tag cloud displayed in the menu portion of the application gets updated every time the search is limited with a keyword. The new shown keywords are collected from the results and the words with most occurrences are written on a bigger font. Further related keywords that are not visible can be loaded to the cloud by pressing a button.

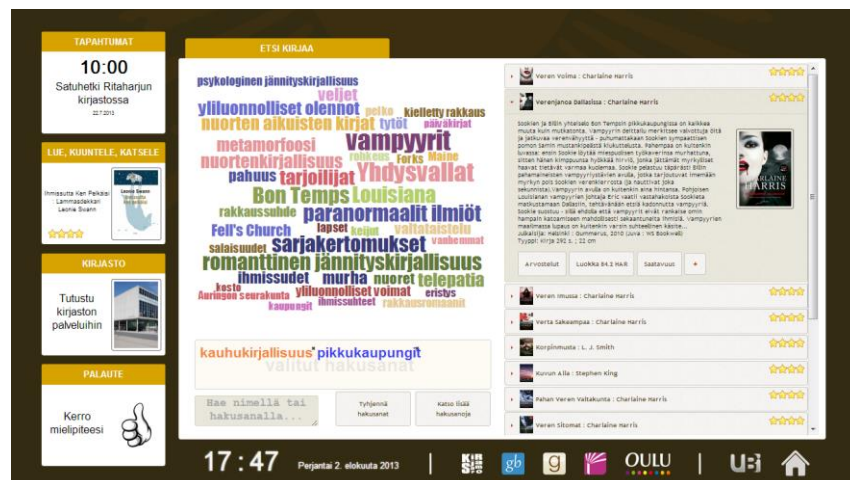


Figure 23. Database search application.

The *administration* application is used to update the service content of the hotspot. It can be used to change questions in the feedback application, view the answers to these questions and change the content of the announcement channel application.

6.2.1.2 Complementary applications

The *book details* application can be launched when a list of books is displayed. This application provides the cover, short description and publishing information about the book. Additionally, the user can launch the *book location* application to show the location of the book on the map of the library, the *book reviews* application to display the reviews of the book, the *book availability* application to find out whether the book is available in the city library and the *book keywords* to display keywords associated with the book. When viewing the availability of a book, the user has a possibility of initiating a limited distributed state by reading a QR code with a mobile phone. By scanning this code, the user navigates to the Outi loaning system and is capable of making reservations. By selecting a keyword in the *book keywords* application, the system navigates to the *database search* application with selected keyword. This allows users quickly to find out similar books or alternative releases of the selected book. The *book location* application is limited to the second floor of the library. It uses the knowledge of keywords, authors name and map information to discover the location. Based on these clues, it makes the best possible guess of where the desired book may be located. The application does not take into account that a book may be processed by the physical loaning service, as there is no technology for discovering it. A similar prototype of determining book locations was once deployed at the Oulu University Library [26].

The hotspot detects the gender and age category of the user from the video feed of the overhead camera and sends them to the client. There are five distinct age categories: child, teen, young adult, adult and senior. The *recommend content* application receives these attributes and sends them to the metadata engine. The engine returns books, locations and events that might be of interest to the user. Then the application notifies the user with a visible cue. When the application is opened, a list of the books is displayed in the content window, while the keywords related to the books, and events and locations are displayed in the menu window. This application utilizes the group specific state.

The subtle state is enabled, when a potential user approaches the hotspot and the resulting face detection event is received from the hotspot. The related *subtle state* application displays atop the default advertising channel an overlay that contains a welcoming greeting and the covers of different books. The purpose is to overcome interaction blindness by communicating that the hotspot is interactive and to entice the user to start an interactive session by showing that the hotspot is customized for library content.

6.2.2 Overall architecture of the client

The architecture of the client follows the principles of interactive web site design. Robbins [54] divides the structure of web sites into three parts: structural layer,

presentation layer and behavioral layer. The purpose of the structural layer is to describe the structure of the document, not to provide instructions on how it should look or behave. It is generally written in HTML or with stricter rules containing XHTML. The presentation layer describes how the elements in the structure layer look like, their position, size, color and so on. The style sheets in CSS format are used to provide this information. The behavioral layer adds interactivity and dynamic content to the application. The document object model allows scripts to access and modify the structural layer, making it possible to change any of the content, style or structure of a document by modifying the attributes of different objects. The W3C is developing a standardized version of the web scripting language called ECMAScript. In practice, this web scripting standard implementation is referred as JavaScript. The major browsers have slightly different ways of implementing the ECMAScript specification, which makes it difficult to create web applications that work exactly the same way in every browser.

The overall architecture of the behavioral layer is illustrated in Figure 24. The behavioral layer is programmed with JavaScript. The UML class structure notation is difficult to translate into web design terms, as the structure and appearance are separated from the implementation part. Only the most important functions and attributes of the classes are illustrated in the class diagram. A package is a logical container for classes of similar sort. In practice it is a folder in the file system. An instance of object in this model presents an object that is only initiated once, and is held in memory as a single object during execution. The class notation is slightly different from the original UML notation. A class can be initiated to multiple objects. Some of the classes, mostly the class *functions*, are used only in static context to simplify the architecture. The library notation is a code library created by a 3rd party and used to speed up the development process. Library packages and classes are used to speed up the development by implementing required functionality. In the client, jQuery and jQuery-UI are the most important libraries. They are mainly used to create different components of the user interface. Additional jQuery plugins are used in the calendar, event feed and virtual keyboard applications. The d3.js library is primarily used in the creation and animation of the word cloud. The stomp.js library is a JavaScript implementation of the STOMP client specification and it is used to communicate with the hotspot interface. The Google Maps JavaScript library is used by the libraries application to display the map of Oulu.

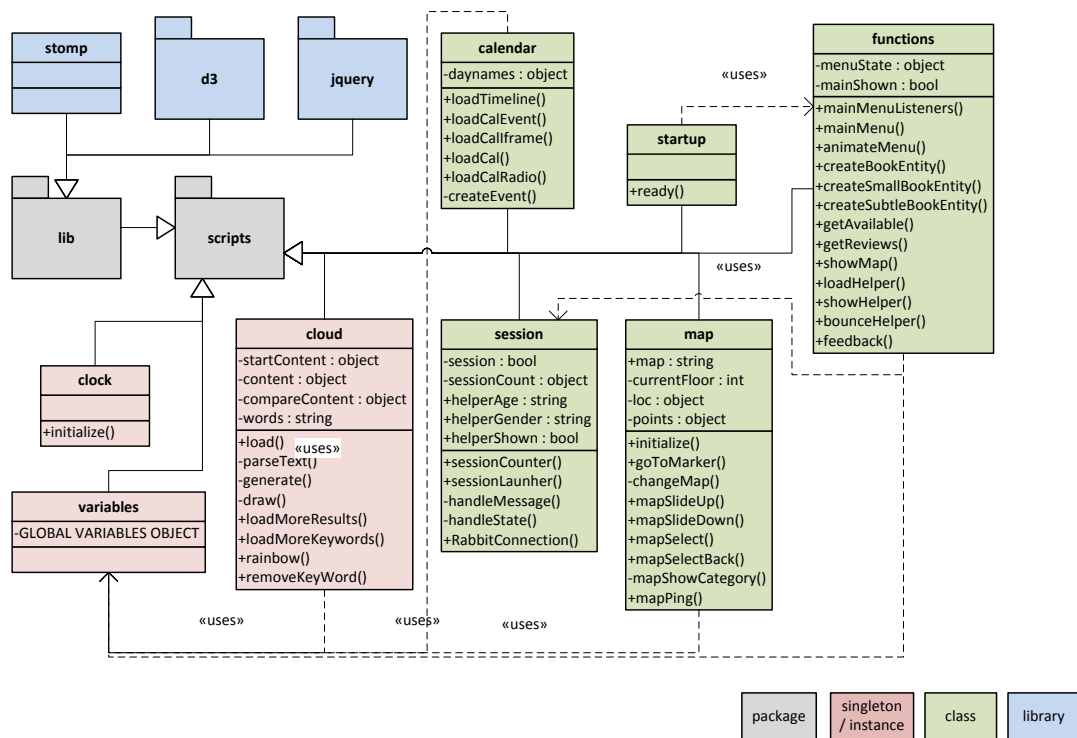


Figure 24. Client's class model.

The layout management depends on the number of different applications; therefore it requires the calculation of application sizes. This calculation is performed on the startup and saved in the *variables* instance. It also holds most of the size definitions for main elements for easier modification of the layout and general variables used all around the program. The content inside these main elements is defined to fill to the size of the element in the CSS. Most of the content is created in the class *startup* that is executed upon loading the document. It performs multiple calls to server to load the dynamic content, and initiates the different applications to their default views. The *functions* class holds general functions that all applications use. The *cloud* instance holds the implementation of the library database search and the word cloud associated to it. The *map* class holds the implementation for both the library services and libraries applications. The *calendar* and the *clock* classes handle the different date and event management used by multiple applications. The *session* class is used to determine the actual state of the hotspot and to receive events from the camera. It is also used to save the session data.

6.3 Server implementation

6.3.1 Overall architecture

Figure 25 shows the general architecture and class relations of the Content server software. The *connection* class is responsible for receiving and returning messages received from the clients. The *variables* interface holds the implementations of the data types and general settings used in the server. The *parser* class implements the content crawlers used by different classes. The *db* class provides methods to access the database. The database is accessed via a singleton object to minimize data corruption. The *run* package holds the server scripts that are used to update and maintain the database. The *QueryPath* library is used in the parsing of websites and the *phpqrcode* library is used in the creation of QR codes. The server is implemented with PHP programming language. It uses the MYSQL database software to implement the permanent data storage requirement.

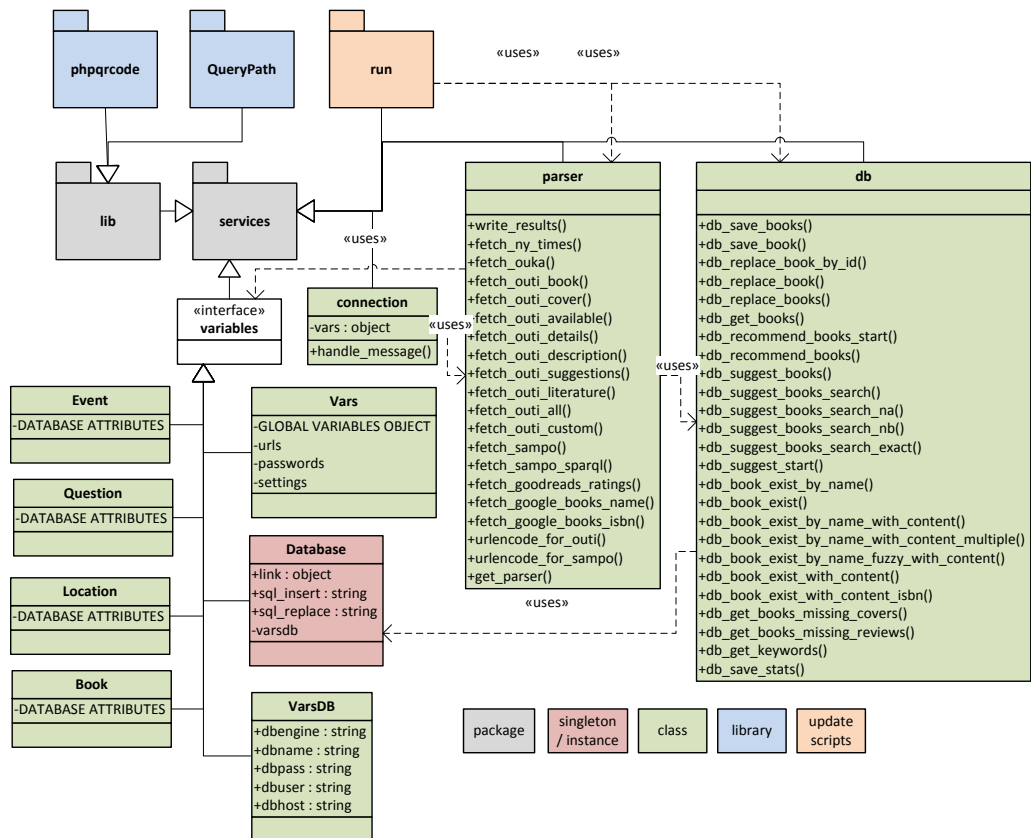


Figure 25. Server’s class model.

6.3.2 Object types

The database model is presented in Figure 26. The purpose of the database is to save objects that are usable in the whole system, including the clients and the server. Every table in the database holds a set of objects with variables defined as column headers.

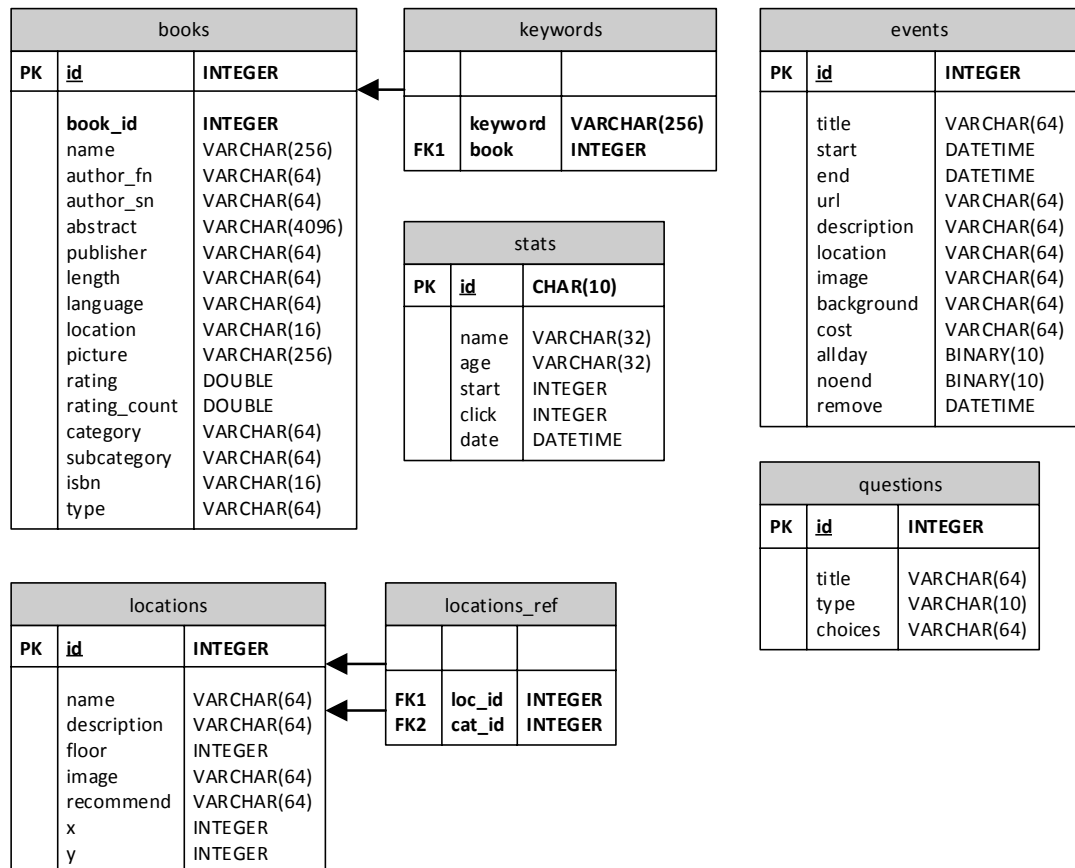


Figure 26. Database model.

There are five types of persistent objects in the system: Book, Location, Stat, Event and Question. The *books* table holds records of all works located in the library database. The keywords for books are separated in their own table with the foreign key principle, to speed up keyword based searches. The book discovery application relies heavily on the speed of keyword searches, and searching for specific keywords can be made with a query to the *keywords* table. This is a lot faster than making a full text search for a single field containing all keywords for every single book in the database. *Events* table holds the descriptions of events parsed from the library's calendars. *Question* table holds the questions used in the feedback application. The client application logs all touch events and application launches and sends them to the server to be stored in the *stats* table. The log data will be used to analyze the usage and popularity of different applications. *Locations* table contains the descriptions of different services and specific locations in the library. Locations are also used to describe the libraries in the Oulu area. Because every Location object can contain other

Location objects, the *locations_ref* table is used to map these locations together. Table 4 lists the object types and application using them.

Table 1. Object types used by applications

Object	Used by application
Book	suggest books, database search, recommend content
Event	events, announcement channel, recommend content
Location	services, libraries, recommend content
Question	feedback
Stat	client (used to store log data)

6.3.3 Metadata engine

The implementation of the metadata engine proved to be the most challenging part of the server. The purpose of the engine is to fill the first objective of the UbiLibrary – to collect information about the customer, literature and library and to analyze and sort this collected data in a meaningful way to provide contextually rich information.

The database contains three object types - books, events and location - that are actively used by the engine. Library's record information is constantly collected and updated from the library's database as book objects and despite its name, the data type also contains other types of media. Movies, magazines, music and books are all under this object. They can be separated to different objects with the type variable, if future development so requires. Event information is regularly updated from online calendars. Location information is static and persistently stored in the database. These basic data types are later augmented and used in content recommendations. Currently, the database only holds five different data types, but it is straightforward to extend the schema if needed. For example, the database could also hold information on authors.

The information related to literature is the most enhanced object type. Other data types can be appended to contain keyword functionality or additional information to facilitate better content suggestions. For example, collected events could be appended with keywords by parsing event descriptions as they currently only hold the category of the event. The database contains all records located in the library's database and the 3rd party services are used to enrich this information. The engine uses different parsers to obtain the additional information, which is then appended to objects and stored in the database. These parsers are mainly content crawlers of 3rd party services with public APIs that are described in Section 6.3.4. The engine only accepts information related to works that can be found in the library's database. This way, the system never recommends a work that does not exist in the library. To discover missing information on records, the engine runs daily maintenance scripts.

Keywords are the most essential data in creating good literature recommendations. The records are "tagged" with keywords that are relevant to their content and can be used as metadata in information retrieval. Later on, these files may be searched using keywords, which can make finding files much easier. For example, landscaping guides can be tagged with words such as "nature", "trees", "flowers" or "landscape". Keywords in the system are classified as primary keywords retrieved from the library's database, which describe the novel idea of a record and additional keywords retrieved from other services that describe context of the record. Primary keywords have a

higher priority in creating content suggestions. Based on the number of occurrences, keywords are also assigned weights that are utilized in the content retrieval phase. The engine uses semantic web methods to collect the keywords, but it does not use the semantic way of saving them (i.e. RDF triplets). For example, a particular book in the database can be augmented with more information by semantically searching KirjaSampo by the name of the book and then by refining the search on received information in a single query.

Content retrieval from the metadata engine happens in two distinct ways. The *database search* application uses the simple method of retrieving records based on inputs. This method takes a list of user selected inputs and selects the most relevant results based on the number of occurrences of the keywords and the matching of the inputs with the other fields of the records (book's title, author's name or a portion of these). These results are further organized based on ratings collected from social recommendation sites. By design, this method of searching is exact and does not provide additional results that are out of the bounds of the inputs.

The *recommend content* application uses the metadata engine in a more complete manner. The method of creating recommendations differs a bit depending on the object type. In case of location and event recommendations, the engine takes the age category and gender of the user as input. For locations, the most used place(s) reported for the age category of the user in the user study are selected. For events, the gender, age category and starting time of the events affect the suggestions. The algorithm prioritizes events starting in immediate future and nearby the library. The algorithm to determine book recommendations is more complex. Based on the age and gender of the user, the engine uses the suggestions made by librarians for the target age/gender group. The algorithm also uses some very broad definitions of literature genres to provide more interesting recommendations based on the survey [41] discussed in Section 4.4. For example, aged men get more recommendations of war themed books and teen females get more romantic novels. By collecting related keywords based on librarians' suggestions and inspecting the most popular books of novel themed keywords, a network of keywords is collected. By examining what novel keywords are popular in this network, the most frequent keywords are selected and a search to the database is performed. Generally this works rather well, as a single publication holds many keywords related to the content and the librarians' suggestions have a higher value. By design, this method of database search is fuzzy. The idea is that the engine will return almost an infinite number of suggestions, so that the user will never run out of content to discover. This is achieved by relaxing input parameters, for example by adding suggestions from other age categories, when more suggestions are needed. Finally, the generated recommendations are ordered with ratings collected from social recommendation sites.

The general idea of information flow in the metadata engine is illustrated in Figure 27. The server uses online services to collect information provided by users. One of the original ideas was to allow library customers to vote and review works, but it was excluded from the final implementation as an unnecessary feature.

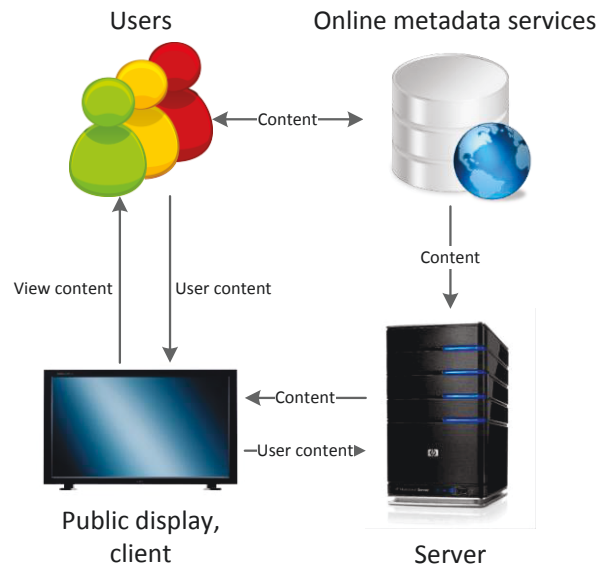


Figure 27. Information flow in the metadata engine.

6.3.1 Online content providers

One of the most empowering features of the success of internet is the capability to “mash up” new services from existing online (web) services. The Programmable Web lists 66 different APIs designed to share information about literature [55]. They are divided in three categories: book reference services, social books services and library book services. They were carefully examined and the most useful ones were included in the UbiLibrary system.

The OUTI web library is a portal for the database of library collections [56]. Since the beginning of 2012 the collections of libraries in Oulu, Ii, Kempele, Liminka, Lumijoki, Muhos, Siikajoki, Tyrnävä and Vihanti have been available in OUTI and customers can borrow material from all OUTI libraries. OUTI provided the starting point for the UbiLibrary database, as well. Whatever book is retrieved from any other service, it is always compared to the OUTI database. The major drawback is that the OUTI library does not provide a public web API.

The library publishes book suggestions at its website [57]. These suggestions are divided for different target groups, and therefore can be used directly in book recommendations for different user groups. One of the original requirements was to exploit the already existing book suggestions, so that UbiLibrary would not increase the workload of the library staff. A public API to the underlying publication system called LifeRay was not available thus the content had to be parsed.

GoodReads [58] is an online service that provides opinions, reviews and grades about literature. It has a flexible API that allows developers to program applications capable of using most of the features of the GoodReads website. The website also has book collections referred as book shelves that contain suggestions for a particular theme. With this information it is possible to map certain types of literature to interests of user groups. Also, the amount of reviews and grades that a certain book has received is a strong indication of the popularity of the book.

KirjaSampo [38] is joint project of the Semantic Computing Research Group (SECO) of the Aalto University and the Kirjastot.fi organization. With their generous help, we were able to use the database. The KirjaSampo portal uses semantic web language to retrieve required information from a vast database. The most fascinating feature of the database is the mapping of the information with more comprehensive keywords. KirjaSampo also lists keywords based on the content of the book, i.e. locations, characters and years. This in turn allowed creating a deeper tag cloud from the keywords. KirjaSampo proved to be a valuable resource on Finnish fiction literature.

LibraryThing [59] is a simple service that takes an ISBN or a title and returns a list of ISBNs from the same "work", i.e. other editions and translations. It is used when a book is received from a content provider but not found in the database. This way we can find and map the result to a matching version in the library's database.

Google Books [60] is a new service that not only lists the metadata of books but attempts to archive the contents of the books as well. The metadata engine uses Google Books as a fallback source for missing information such as cover images, authors, publishers, etc.

The New York Times Best Sellers List [61] collects statistics about copies sold in different book form factors. It provides a simple interface to identify the most sold book in different categories that can be mapped to those specified for UbiLibrary. While the list itself is oriented towards United States, it gives a good starting point to understand the current popular reading across the world.

Kirjakauppaliiitto [62] keeps monthly records of most sold books in Finland, which are a valid source of popular Finnish literature. The website has no public API available, so the page has to be parsed. The problem with the most sold books is that they are often sparsely available in the library.

TasteKid [63] provides suggestions based on titles of works. It is not limited to literature, but lists also games, movies, TV shows and music. This service is particularly useful for the suggest books application. Also, the library holds in its collections all kinds of media content that could be more actively promoted. TasteKid API does not provide any exact identification information (e.g. ISBN), thus it has to be used in conjunction with LibraryThing.

6.4 Data flow

From the beginning it was very obvious that the client program would contain most of the core logic of the service while the server in turn provides centralized functions such as database access. The example sequence diagram in Figure 28 illustrates data flow between the main applications in the client, server and online content providers. It should be noted how the server does not make any decisions on how to behave when an empty set is returned from the database. Instead of searching automatically for more from content providers, it returns an empty set. This information is processed in the client and corresponding information is displayed for the user. The reason for this design is to keep the client responsive and the user in charge of the behavior of the service. As discussed before, the interaction sessions with the hotspots are often short and the users are prone to navigate elsewhere or lose their interest in case of delayed responses. Navigating away does not break the event flow between the client

applications and the server. After navigating away, user may launch a new application and a different query is sent to the server that launches a new thread to handle the query. When the client receives the responses from the server, it compares the id of the query to the id of the latest response and acts accordingly. If the ids do not match, the response is simply ignored.

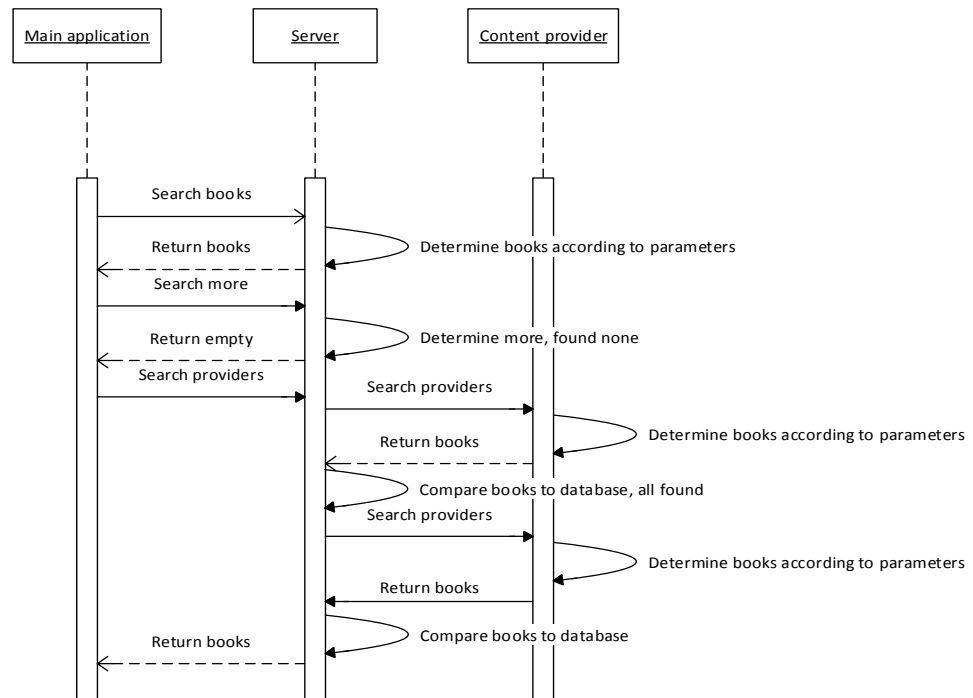


Figure 28. Example sequence diagram.

6.5 Communication

HTTP is the underlying messaging protocol of all communication between applications, server and online content providers. In computing, the same-origin policy is an important concept in the web application security model and it disallows cross-domain requests in JavaScript XMLHttpRequest. Because of this limitation, website parsing and information collection from online content providers is performed on the server. Services with public API's provide content usually in JSONP format that allows web browsers to request data from a server in a different domain.

The message received from the server always contains a list of the desired object type in the body of the HTTP GET response. This list is encoded in JSON, because it can be directly read as an array of objects in the client. The possible error code or required information is encoded in the header of the HTTP GET response. This simplifies implementation, because there is no need for additional parsers or messaging specification, and server requests are usable everywhere in the client code. This is similar to communication with the public APIs of web services. The different message types are defined in Table 5.

Table 2. Message types

Message (header) (type)	Message content (body) (variables)	Response message (header)	Response content (body)
events, message id	String calendar	message id, error code, error message	array[Event]
books, message id		message id, error code, error message	array[Book]
suggest, message id	array[String] keywords, array[String] searchwords, Boolean more	message id, error code, error message	array[Book]
location, message id		message id, error code, error message	array[Location]
question, message id			array[Question]
next_event, message id			array[Event]
ads, message id		message id, error code, error message	array[Event]
stats, message_id	array[Stat]	message id, error code, error message	

7. EVALUATION

Evaluation was performed on three different aspects; task-based user evaluation, feedback from the library staff and statistics collected in real world usage. Task-based user evaluation and statistics measure the performance of the implemented client applications in a real world setting, while the feedback from library assess the successfulness of the service from professionals point of view.

7.1 Preliminary functional testing by library staff

Before deploying UbiLibrary in a real world setting at the library, functional testing of a full prototype was conducted by library staff. They were asked to perform some basic tasks and their interaction with the service was video recorded, followed by an open ended discussion. The purpose was to expose any major functional flaws in the service, but none were discovered. All required features were implemented and functional. However, many minor bugs were identified and corrected before the actual launch of the service. The analysis of the video recording revealed that the staff had some problems with basic usage such as touching the elements, navigating the menu and so on. This demonstrates the importance of this type of functional testing by external testers before the actual launch of the system, as the usage can become too self-evident for the developer.

7.2 Task-based user evaluation

The UbiLibrary service was evaluated in form of a task-based user evaluation conducted in the lobby of the library. Each participant was asked to complete following six tasks using UbiLibrary:

1. Locate an interesting event from the service.
2. Find a thriller book you would be interested in.
3. Find reviews for the book *Jari Tervo : Troikka*.
4. Find the opening hours of the music department.
5. Find the address of the library in *Myllyoja*.
6. Locate the most read book in Finland in the library.

After completing the task, each participant was asked to complete the questionnaire shown in Appendix 2, followed by a short interview. Overall, the evaluation took around 20 minutes to complete. Each participant was rewarded with a free coffee and a snack.

In total 20 people participated in the evaluation, 12 males and eight females. They were grouped into five age categories similar to the face detection software: children (0-12 years, 0 participants), teens (13-18 years, 6), young adults (19-26 years, 3), adults (27-55 years, 7) and seniors (56-100 years, 4). Unfortunately, no children could be recruited into the evaluation. The gender of the participants was used in the valuation of the book suggestions.

Figure 29 shows the average task completion times in minutes for each task and age category. It seems that completing the tasks took slightly longer the older the users were. This is not surprising as younger generations are more accustomed to use technical devices, touch screens in particular. Surprisingly, the average time that the tasks took to complete is not that different across the age categories – the maximum variance in the task completion time is close to one minute in task 3. This is a rather good result, considering the complexity of the tasks and the fact that all participants were novices in using the UbiLibrary service. For example, completing task 3 took 1 minute 20 seconds on average. The tasks requires first finding and launching the *database search* application, writing the name of the book or the author, and locating the review functionality from the results. The fastest participant completed the task in 13 seconds by just writing a part of the author’s surname and locating the book and reviews from the results. The task involves some of the most important features of the service. Overall, the task completion times show that there are no major defects in the overall design of the service.

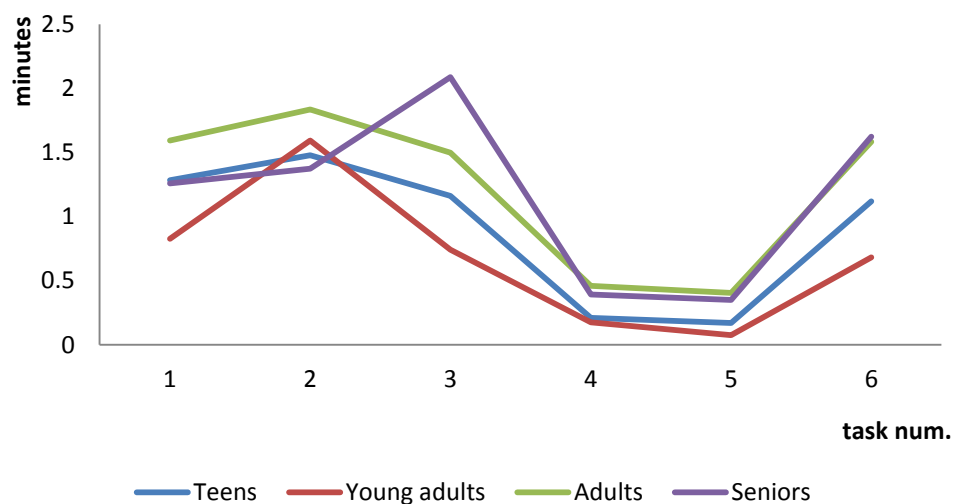


Figure 29. Average task completion times for different age categories.

Most participants found the new service to be useful (80%) and complementary to the existing digital services. When asked if the participant would use the service again, 55% answered yes, 20% answered maybe and 25% answered no. The fact that 75% could be returning users is encouraging.

The participants did not report any missing features in the interviews, but made two suggestions for future enhancements. One participant requested more information about the authors, which could be implemented in the *book details* application. Another participant requested for a tutorial on using the service, but that would contradict the original design rationale of the service.

The *database search* application was the most useful service according to the participants. In the very early design it was proposed to the library staff as limited book suggestion functionality, receiving a lot of interest. The collected ideas were eventually refined into an application that gathers information from the internet and the library databases. This demonstrates the importance of iterative development with the customer. The participants perceived the *book location* as the second most useful application, followed by the *events* and *suggest books* applications.

The *recommend content* application recommends books, events and locations based on the age category and gender provided by the face detection software. As a pleasant surprise, 70% of the participants were pleased with the recommendations they received. Two of the interviewees even named it as the most useful service. Only the young adults reported that the recommendations were useless to them. It seems that the recommendations could be a lot more useful functionality than originally anticipated. This is a strong indication that the whole hotspot service portfolio could benefit from personalized content. However, no conclusive deductions should be drawn from these results, as the data is so small. Also, we cannot be certain that all participants were correctly profiled by the face detection software.

A particular difficulty in the usage of the new applications was identified and dubbed as “comprehension challenge”. Some participants appeared to be overwhelmed by the information overload provided by the combination of the large size of the screen, the different positions of interactive elements and the complexity of the application. The users concentrated on elements and focal points that were not relevant for desired interaction. The user started wondering aimlessly around the application after they lost the intended interaction sequence. Three participants had severe comprehension problems and they regularly diverged from the designed interaction path. Two users specifically complained that there is too much text on the screen and it is hard to know what is important to read. This emphasizes that applications targeted for large screens should clearly indicate the location of the relevant content and intended interaction sequence. The software developer might be tempted to fill the large screen with content and maximize the canvas in a way that is not possible in a desktop. Then the danger is that the user no longer understands the intended usage of the application. In the maximized application mode UbiLibrary has only three focal areas: main menu, application menu and application content. This thesis suggests that three focal areas are the maximum number presented to the user in a single state.

The participants had mixed success in using the tag cloud as a navigational tool. The concept of adding and removing keywords to filter the database was the most demanding part. Some did not understand either the purpose or the interactivity, some understood the interactivity but not the purpose, while some understood both the purpose and the interactivity. While about half of the twenty participants instantly understood the concept, the other half had varying difficulties with it. In task 2 that was specifically designed to test finding books with the tag cloud, about a third of the participants ignored the cloud entirely and instead entered a textual query. Some participants clearly detested the idea of entering keywords to search the database, while others found it to be the best feature. Typing with the virtual keyboard on the touch screen proved very challenging for some users and fast typing is difficult for anyone. The usage of the virtual keyboard should be discouraged, unless it is mandatory. All in all, the utility of the tag cloud to UbiLibrary remained debatable.

The evaluation exposed multiple little bugs to which the author of the thesis (i.e. the developer) himself had become blind. For example, by only changing the title of the *suggest books* application, the users performed a lot better in task 6 after the evaluation. A lot of the problems were specific to the large screen and numerous tweaks to the sizes and the positions of the elements were made based on the findings of the evaluation.

7.3 Feedback from library staff

To assess how well the design and implementation of UbiLibrary succeeded from the librarians' professional point of view, we asked eight members of the library staff involved with digital library services to use UbiLibrary, answer a set of questions and provide general feedback. This can be regarded as a so-called expert evaluation.

1. The original ambition of the library staff was to use the hotspot for promotional advertisement. Does the implementation fulfill the desired needs?

According to the responses, the currently shown advertisements change too rapidly to allow viewers to understand the content. The maintenance and updating of the advertisements was found easy.

2. Does the new service support the library staff in their work in any way? What could have been made better?

Overall, UbiLibrary was seen as a beneficial for introducing the library for new customers. The library staff can use the service to show the facilities and literature suggestions to customers. The location and privacy of the hotspot was raised in almost all answers. Three of the staff members directly suggested moving the hotspot to a more private place in the second floor of the library. This would promote longer sessions and it would be simpler to direct customers to find out content from UbiLibrary. It is not feasible to send customers back to the first floor where the hotspot is currently located. The actual library content of the library is far away from the hotspot. On the other hand, moving the hotspot would be against the design objective of using the hotspot and the UbiLibrary service as the first information point when arriving at the library.

3. The service is meant to complement the current digital services of the library. In your opinion, how well (coverage, ease of use, maintenance) were these goals achieved?

The majority of the staff was positively surprised about the quality and coverage of the new service. The usage was perceived as simple and there was enough coherent content for the customers. Specifically, the *database search* and *suggest books* applications were seen as successful implementations. UbiLibrary was perceived more of a replacement than a supplement to the current web site.

Discovering the interactivity of the hotspot was identified as a major challenge in multiple responses. The library staff agreed almost in unison that the interactivity of the hotspot will constantly be missed by library customers. They suggested raising the awareness of the interactivity among customers with advertisements, stickers and rewards. Some kind of a hook to allure the customers to interact with the hotspot was seen pivotal for the success of the new service.

4. The application uses rich information collected from the internet to create more meaningful content and therefore competes with similar internet

services. Is the application capable of challenging the existing services (i.e. KirjaSampo) and how this information could be used in the library space in a more preferable way?

This question received the smallest number of responses. Generally, UbiLibrary was seen as a valid competitor for similar services, but only in the library space. UbiLibrary was not seen as a viable primary option to other similar content recommender systems. This is very common in web business where generally few popular services capture the majority of the users.

Other similar services are clearly designed to be used from home desktops and there is definitely a need for a custom service designed for public library spaces. According to a staff member, not everyone is willing to ask advice from the staff and UbiLibrary could be even essential in these cases. The librarians agreed in that the usage of UbiLibrary in information retrieval is more practical than searching similar information with a smart phone.

5. Do you believe that digital services similar to this (metadata of books, reviews or recommendations) will gain popularity specifically in the library space?

The range of information in the library space was seen as real possibility if the information is presented in a way that interests the users. There is no real need to implement complicated systems if the users ignore the service. Raising the interest is especially difficult in the case of regular customers with established usage patterns.

7.4 Statistics of real world use

UbiLibrary was deployed on the hotspot in the lobby of the library to be used by the real customers of the library. Detailed statistics on the usage of the service were collected over 24-day period from 9.8.2013 till 2.9.2013. The number of starts and clicks were logged for each application, together with the age category and gender of the users detected by the face detection software. As reported before, it assigns each detected user into one of five distinct age categories: children (0-12 years), teens (13-18 years), young adults (19-26 years), adults (27-55 years) and seniors (56-100 years).

Figure 30 shows the total amount of starts and clicks every day. The daily average number of started applications was 50 and the daily average number of clicks was 240. During the data collection period there were 209 distinct usage sessions in total, which corresponds to around nine sessions per day. We see that the usage of the service was least active at weekends, obviously due to the shorter opening hours of the library.

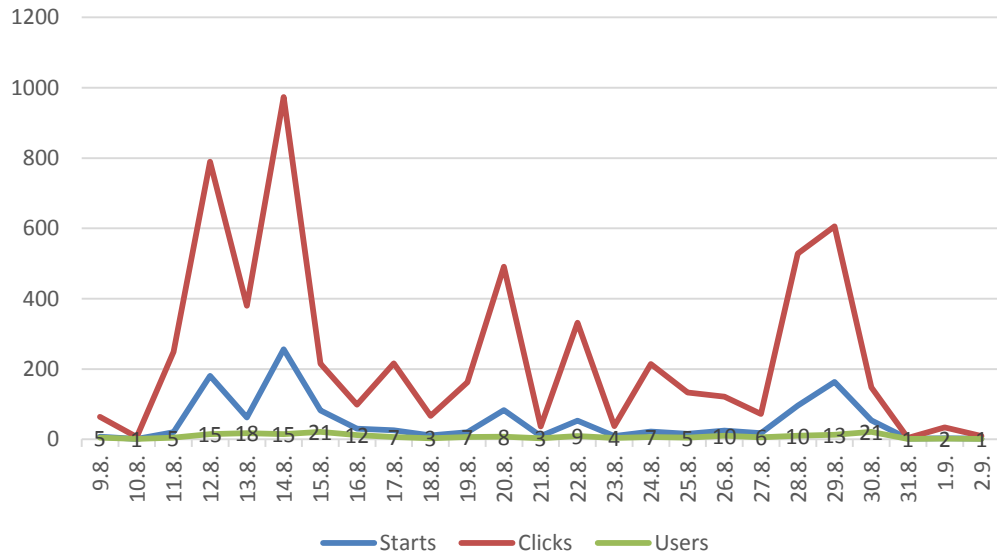


Figure 30. Usage statistics during the data collection period.

Figure 31 shows the distribution of the starts and clicks over different age groups and genders. The optimal situation here is when the amount of clicks is high compared to the amount of started applications. When this ratio is high, we know that the users are accustomed with the service and spent their active time in a portion of the application they need rather than trying to search the functionality they need. Rather surprisingly, the most active group was adults. It is possible that the average age of a library customer is higher than that of typical users of other hotspots. It seems that adult males could be the ideal target group for UbiLibrary. They have the longest sessions and their click count for application launch ratio is the highest. It is also apparent that the service is unpopular among young females. While the primary reason remains unknown, one possibility is that the recognition software erroneously recognizes them as males. Of course, it could be that young women just not are interested in the service, similar to the seniors who are not accustomed to use information technology services.

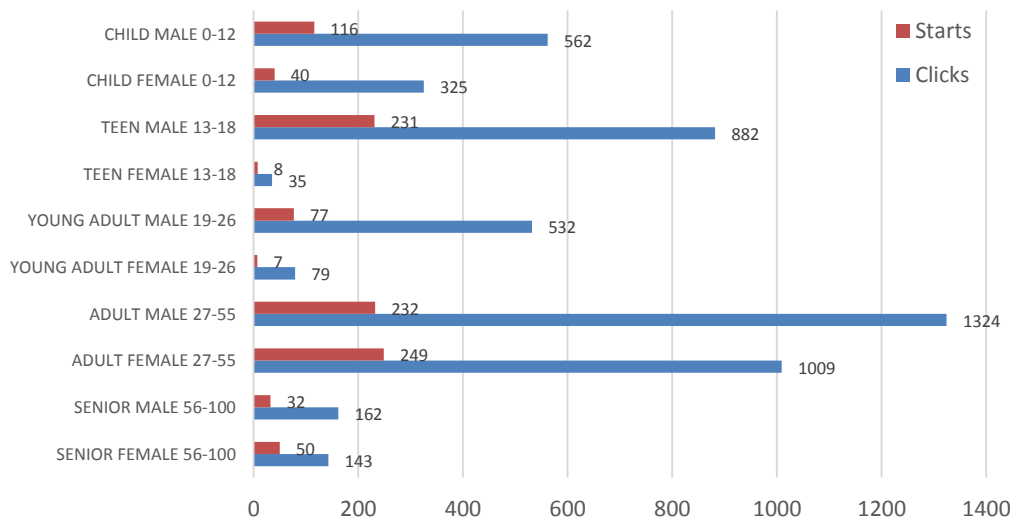


Figure 31. Usage statistics for gender and age categories.

Figure 32 shows the starts and clicks of each application that reflect their relative popularity. The bottom six entries correspond to the primary applications. The most popular service turned out to be the *database search* application, followed by the combination of *services* and *libraries* applications. The *suggest books* application was less popular than anticipated - it is possible that the user interface of the application or the suggestions provided by it were not successful.

The average clicks/start ratio is very similar for all main applications, around six clicks per application start. This is a rather low ratio, as some applications are rather complex and require multiple clicks to perform correctly. Obviously, this ratio is skewed by some users quickly testing out the new service with few clicks. The average click count per session is 28.7 clicks, which appears sufficient for discovering new content from the service.

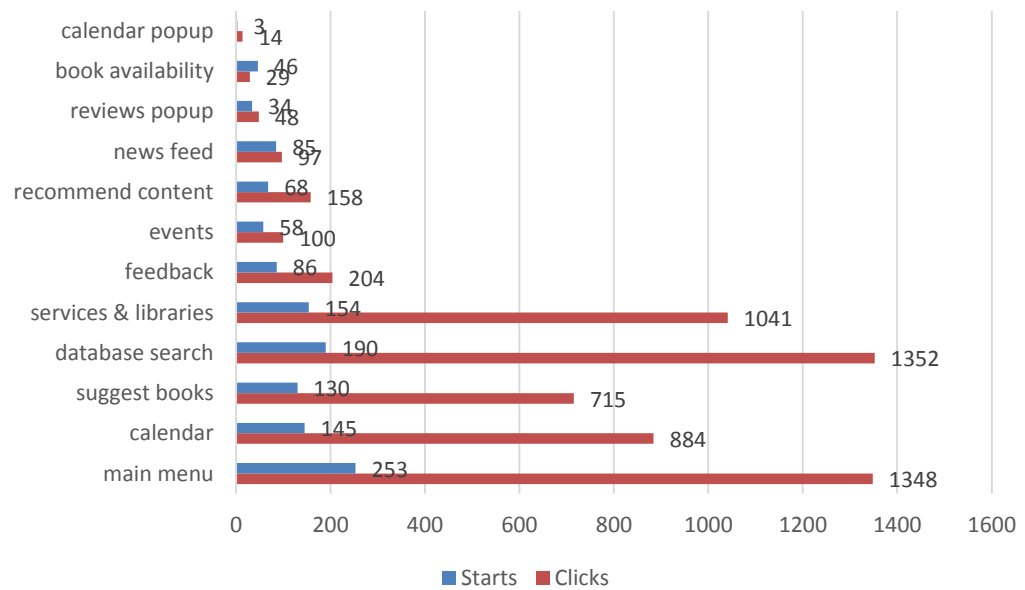


Figure 32. Usage statistics of applications.

The user sessions were divided into three groups based on their activity and click count. The first group has at most ten clicks in a session, reflecting the user just starting the interaction and then quickly leaving the hotspot. These sessions have been referred to as curiosity launches [64]. The user may have been interested enough in the service to begin the interaction, but did not actively use the service to discover its functionality. The user might have even started an application or two, but did not continue the interaction any further. This group of curiosity launches had almost half of the sessions (49.7%), which clearly shows that more effort must be placed on convincing the users to remain active after launching the service. This could be achieved by carefully designing the main menu in such a way that instantly leads the users to a popular application.

The second group uses the hotspot for an extended period of time and clicks the screen 11-70 times. This group had 77 (36.8%) sessions. It is clear that this group is using the service to discover new content. This group contains returning user making a specific action, such as searching the availability of certain book, and new users

trying to discover content or the purpose of the service. This is the most important group for the continuation of the service.

The third group can be described as the power users with more than 70 clicks. There was in total 28 (13.4%) power users. They use the service efficiently and comprehensively.

In the future, the success of UbiLibrary as a library service can be directly measured from the change in the average session length and the proportions of users in groups two and three.

8. DISCUSSION

As the technology around us is progressing in a rapid pace, so are the digital services and applications we use in our daily lives. The emergence of semantic web technologies will benefit ubiquitous computing services by providing semantically more relevant information. This semantic data, if utilized correctly, can be used in a public setting to enrich the user experience and create new environments where the boundary between the digital and physical worlds is diffused.

This thesis presented the UbiLibrary service designed to provide ubiquitous digital services in a library setting and implemented using the client-service computing paradigm. The amount of client machines supported by the server is practically infinite, so the system can be deployed on multiple locations around the library. The client is intended to provide a portal to the digital services of the library. The services are designed to satisfy the information needs of library customers identified in the user study. The client interface experiments with various interaction and user interface design paradigms on large public displays. UbiLibrary is a new kind service in utilizing a rich set of technologies and devices. It employs various set online services to gather a rich set of information. The main strength of UbiLibrary against similar services is the locality or the context-awareness of the service - the service won't recommend anything that is not present in the library or close to it.

The content management and recommendation system is a key component of the service. For successful creation of recommendations, three important data types were identified for the library: literature, events and services. Given user inputs, the recommendation algorithm creates sophisticated suggestions for the user. Content is filtered with contextual information, i.e. the age category and gender of the user detected with computer vision technology. More detailed information, for example the loaning history of the user, could be used to create more accurate recommendations. The thesis also explored social content filtering, as the recommendations provided by the system are affected by reviews and ratings in online user communities. These content recommendations received surprisingly good feedback from the participants in the user evaluation. This way, UbiLibrary demonstrates the benefits of enhancing ubiquitous services with rich context data.

UbiLibrary could be utilized as a content recommendation system in different settings. The system itself was not designed with interoperability in mind, so the transformation to different environments can be challenging. However, the content crawling engines and the underlying data model are designed to be extensible.

When UbiLibrary is compared to the original services in the hotspot, the benefits to the customers and the staff of the library are clear. Both the librarians and the library customers were more motivated by the library specific content as can be observed from the usage statistics and the user evaluation. Comparing the new service to the current website of the library is troublesome as the user experience is radically different. One major achievement is that the information is in more compact and accessible format. UbiLibrary can also be used as a library's database search tool and its interface is simpler and more fluid compared to the interface of the OUTI web library. The filtering of the results with social recommendations also aids in promoting the more popular books. The tag (keyword) cloud provides a new dynamic way to browse the library database. To understand the real effect of the new service on the usage of the library's hotspot, much longer period of usage statistics have to be studied.

8.1 Attainment of objectives

The attainment of the original objectives set for UbiLibrary is analyzed in the following.

8.1.1 Provide information about literature, library services and events to library customers

This objective was set from the perspective of the library customers. UbiLibrary achieves this by utilizing various online services. The service is tightly integrated to the existing library's digital services, and remains up to date with minimal maintenance. This content includes the events in the Oulu City library and in Oulu area in general, services in the library and the locations of other libraries in Oulu. By utilizing the library's database, the service content of the library is provided to the customers. The content of the library database is enriched with information received from online services. To make this information more interesting to the user, the application employs various methods in filtering the content. Based on the characteristics of the user, different content is displayed for the user. UbiLibrary also uses social book review services to filter the results and display reviews of the books.

The service is meant to be the starting point of a library visit, sort of an information kiosk. From the collected feedback, it can be seen that majority of the users find the service useful and are willing to use the service again. There are no technical methods currently implemented to identify users, so it is impossible to measure the amount of returning users. The only meaningful way of analyzing if the hotspot is used as an information kiosk is to compare the length of the sessions. Full interactive session is regarded to be achieved when the user touches the screen more than ten times. It is sufficient for navigation in the service and at least seeing information relating to the content of the library. From the statistics collected during the first month of real world usage, we estimated that roughly half of the sessions were full interactive sessions. In order to discover whether the service has returning users and it is used as intended, the statistics must be analyzed for a longer time period. If the statistics indicate that there is an increase in the amount of users and, more importantly, increase in session length, the service is working as intended.

8.1.2 Create a digital advertising channel for the library

The library staff's desire was to modify the hotspot to advertise the library's services and event notifications, and to collect feedback from the customers. Currently, event notifications and general library information can be updated to the UbiLibrary's announcement channel and library's advertisement to the UBI-channel. UbiLibrary provides an administration application to maintain and update the content of the service. The content created for UBI-channel is managed with different software. According to the feedback from the library staff, the maintenance and updating of the content was found easy.

8.1.1 Explore the usage of contextual information in the hotspots

The interest from the research group was to discover what kind of benefits bringing more context specific information to the hotspots would provide. The case of library specific information was an ideal opportunity to discover what kind of effect location specific services has on the users. It is too early to draw conclusions from the usage statistics whether context specific information entices the library customers to use the hotspot. The findings of the user evaluation indicated that location specific applications are beneficial.

A participant in the user evaluation expressed an interesting idea. He debated that all new services are already present in the library and using their digital versions does not offer much added value. Also, there is the additional requirement to learn to use the new service. A service specific to a particular location could be more beneficial when deployed in remote locations and devices. This contradicts our value proposition of providing more context specific information to library customers. However, the idea of remote deployment is intriguing. First of all, the urban space contains many different services and it is difficult to predict what type of service is needed in a certain situation. Secondly, when a person enters a particular location, he will have the motivation to discover information related to that location. On the other hand, deploying location specific applications to all hotspots would allow the users to discover content from other distant locations. For example, the user could see at the UbiLibrary service deployed on a distant hotspot if a certain book is available in the library and modify his plans accordingly.

The context specific service must be regarded complementary to the existing services in a certain location rather than as a complete replacement. According to the findings of this thesis, the hotspots would benefit from providing information related to their immediate surroundings, as there is a strong indication that this would lead to more regular users. However, other services must still remain visible and easily accessible at all hotspots.

The research presented in this thesis provides evidence on the benefits of context specific information in ubiquitous setting. However, further research is required to determine exactly what environments would benefit from more complex context-aware applications. The feasibility of the content recommendations must be valued individually at different environments. One obvious result of this thesis is that the hotspots would benefit from more dynamic approach to service provisioning. The service content shown in the default view of the hotspot should be more personalized based on the specific features of the location and the characteristics of the user.

8.2 Future improvements

Although the actual functionality of the UbiLibrary service expanded from the originally envisioned simple book suggestions and advertising channel during the development process, there are still many potential improvements to the current service. Some prominent ideas that were not included in the final prototype are discussed in the following.

Currently, the *index* application is a simple menu with two advertising channels. The *suggest content* application receiving so much positive feedback promotes an

alternate, more complex design for the index application. The improved index application could take into account the user's age and gender. It could be a combination of *suggest content* application and main menu as illustrated in Figure 33. The new *index* application could show content from all applications and it could update itself regularly based on the data gathered from contextual sensors. For example, the widget for the *events* application could hold multiple event announcements that are selected based on the age and gender attributes. This way, a lot more lively and attractive starting point for the interaction could be achieved. This would also aid in turning curiosity launches to interactive sessions.

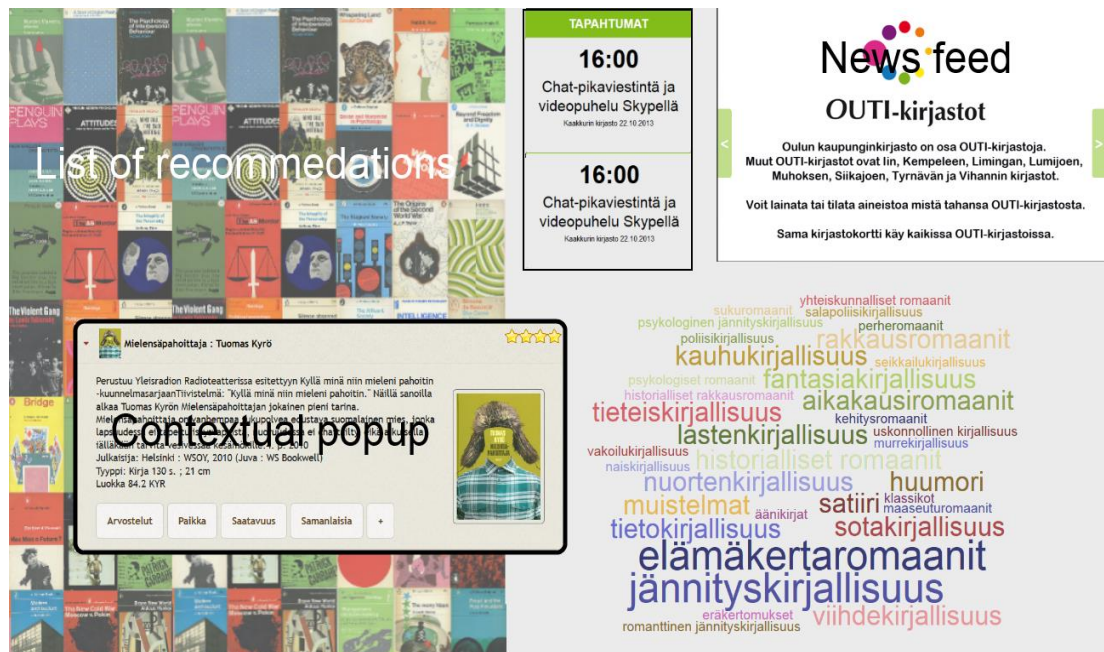


Figure 33. Proposal for the alternative *index* application.

The recommendation algorithm can be improved in various ways. First of all, there are a plenty of other sources for literature related information that could be used to improve the algorithm [53]. It is trivial to implement new content crawlers for these content sources that would enrich the information saved in the database. Adding new types of parameters to the recommendation algorithm on the other hand is more complex as the whole algorithm must be modified to take into account these new variables. Secondly, the relevance between different data types is not handled at all. The recommendation system could for example take into account the content of a certain book and based on that information, make suggestions about current events happening in the city.

TasteKid is a recommender service that is only used for book suggestions. It also holds an extensive database of movies, TV-shows, music and games and could be used to generate suggestions for other media types, not just books. The obvious location for such functionality would be the *book details* application. TasteKid could be used in junction with UbiLibrary, but the results from TasteKid should be then compared against the OUTI database to find the matching articles present in the library.

A participant in the user study requested more detailed information about authors, genres and Finnish culture in general. This could be achieved by enhancing the

KirjaSampo content crawler. As far as the books and authors in KirjaSampo are concerned, they are also automatically integrated to KulttuuriSampo. This makes it possible to approach the entire Finnish culture from a thematic point of view. For example, it is possible to retrieve data of museum objects, photographs, paintings, contemporary newspaper articles using keywords. By using the semantic commands, it is relatively easy to create searches that gather more in depth information about the subject matter in general. The most natural location for this information would be the *book details* application, but it could also be a section or an application of its own. [40]

The issue of social embarrassment of using the service in public setting was brought up several times in the user studies. One customer even sent us feedback about the hotspot being in too public place to be used as a search tool. This issue could be circumvented by adding additional UbiLibrary clients around the library. These clients could have a different form factor, ranging from a tablet to larger screens.

The commercialization of the application has also been discussed. It will remain to be seen how much positive interest the application gathers among library professionals. As UbiLibrary is mainly designed to be used as a public service, there is no incentive for it to create revenue.

8.3 Summary of the development process

Figure 34 summarizes the development process used in this thesis. The work started with interviews of the library staff to gain insight into typical library customers, their needs and information seeking strategies. At the same time, the library staff reported their needs for the new service and made suggestions on the type of services the hotspot could provide. To gain more in depth knowledge about the library customers and their information needs, a user study of the people arriving at the Oulu City library was conducted. Given this information, the abstract functionalities to be supported by the new service were identified. These essential use cases helped in defining the actual objectives for the development process. These objectives also constrained the types of services provided. We collected information on similar services from literature and analyzed the best practices used in content recommender systems. In the implementation phase, more detailed usage scenarios and use cases were specified. The actual implementation was performed using rapid prototyping according to the feedback from the library staff. The evaluation was initiated with a preliminary testing by the library staff to validate the functionality of the new service. A task-based user evaluation was organized with library customers soon after the service was deployed on the hotspot located at the lobby of the library. An expert evaluation with the librarians was conducted and statistics of the usage of the service by real customers were collected.

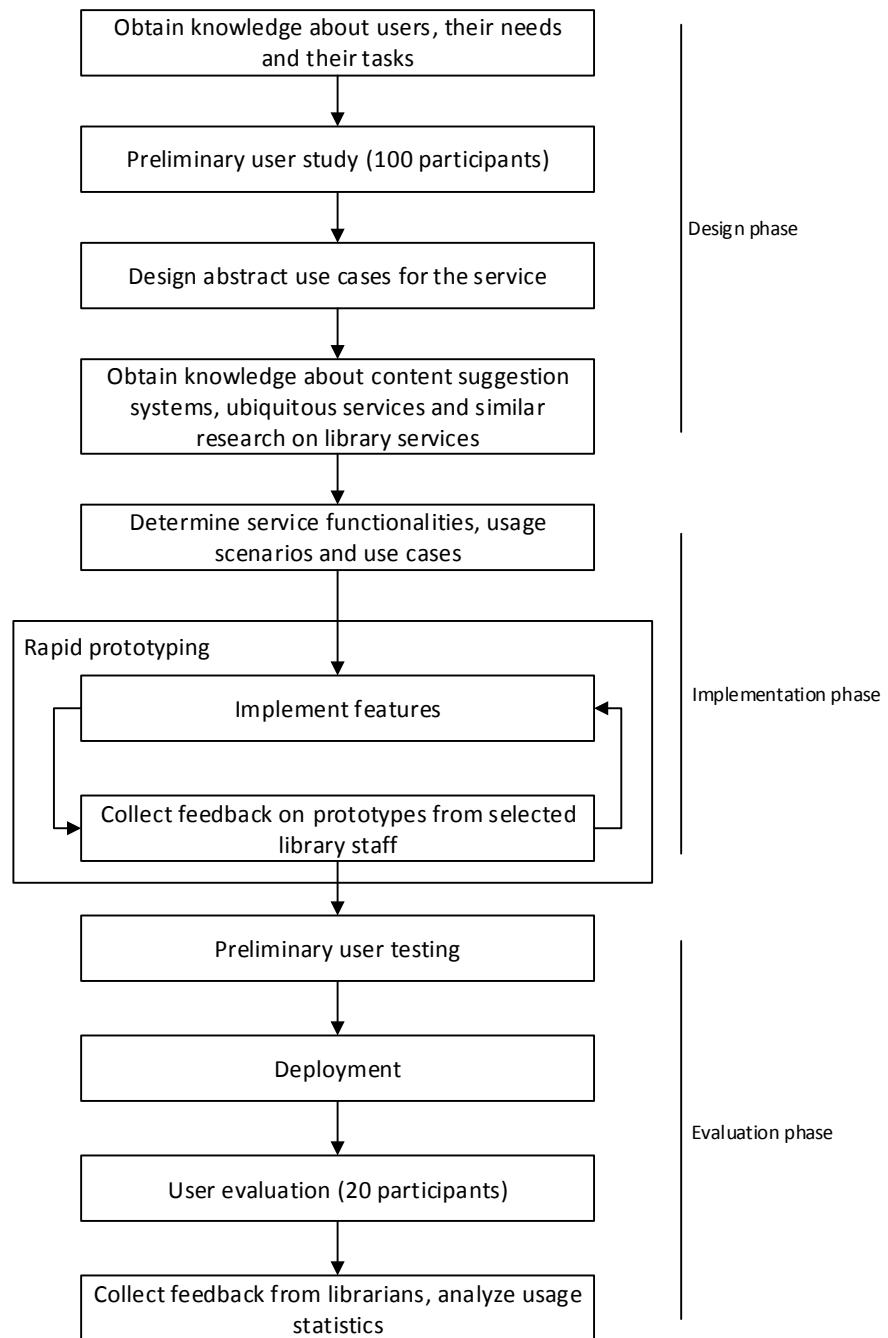


Figure 34. Development process.

9. CONCLUSION

This thesis was motivated by the fact that the staff of the Oulu City Library was not pleased with the default service portfolio of the hotspot located at the lobby of the library. The thesis was tasked to design and implement a new service for the hotspot that would serve better the needs of the library's staff and customers. The requirements for the new service were elicited by reviewing related literature, conducting a user study of the library's customers, interviewing the library staff, and analyzing the library's existing digital services. The design and implementation of the service was done with agile software development methods and rapid prototyping, in close collaboration with the library staff.

The resulting UbiLibrary service dynamically aggregates content relevant for the library from the library's own databases and several online services. Given the content, UbiLibrary provides library customers with a range of applications from book recommendations to information on the library's services and events. The key component of the system is the metadata engine that maintains a database on the library's holdings, events and services. The metadata engine enriches and filters the content with semantic and contextual information. For example, the metadata on the works contained in the OUTI database is semantically enriched by crawling supplementary metadata from various online services. The resulting metadata database can be dynamically browsed with a novel tag cloud based interface that filters book recommendations according to the age category and gender of the user as estimated by computer vision. An administration interface is provided to the library staff for maintaining the service.

The service was deployed for production use on the UBI-hotspot located at the lobby of the Oulu City library. The service was assessed with a task-based user evaluation with library customers, by professional librarians and by collecting statistics of real world usage. Overall, UbiLibrary was found easy to use and presented a valuable addition to the library's existing digital services. However, longer term follow-up of the usage is needed to draw further conclusions on the success of the service.

10. REFERENCES

- [1] Serola S & Vakkari P (2011) Yleinen kirjasto kuntalaisten toimissa; Tutkimus kirjastojen hyödyistä kuntalaisten arkielämässä. Finnish Ministry of Education and Culture.
- [2] Rothbauer P (2004) The Internet in the Reading Accounts of Lesbian and Queer Young Women: Failed Searches and Unsanctioned Reading. *Canadian Journal of Information & Library Sciences* 28(4): 89–110.
- [3] W3C Semantic Web Activity (accessed 20.04.2013) W3C. URL: <http://www.w3.org/2001/sw/>.
- [4] Ubiquity - Dictionary.com (accessed 01.06.2013) Dictionary.com, LLC. URL: <http://dictionary.reference.com/browse/ubiquity>.
- [5] Pervasive - Dictionary.com (accessed 01.06.2013) Dictionary.com, LLC. URL: <http://dictionary.reference.com/browse/pervasive>.
- [6] Galloway A (2004) Intimations of everyday life: Ubiquitous computing and the city. *Cultural Studies* 18(2–3): 384–408. DOI: <http://dx.doi.org/10.1080/0950238042000201572>
- [7] Lyytinen K & Yoo Y (2002) Ubiquitous computing. *Communications of the ACM* 45(12): 63.
- [8] Satyanarayanan M (2001) Pervasive computing: Vision and challenges. *IEEE Personal Communications* 8(4): 10–17. DOI: <http://dx.doi.org/10.1109/98.943998>
- [9] Hosio S, Kostakos V, Kukka H, Jurmu M, Riekkari J & Ojala T (2012) From school food to skate parks in a few clicks: using public displays to bootstrap civic engagement of the young. Proc. 2012 10th International Conference on Pervasive Computing, Newcastle, UK, 425–442. DOI: http://dx.doi.org/10.1007/978-3-642-31205-2_26
- [10] Ojala T, Kostakos V, Kukka H, Heikkinen T, Linden T, Jurmu M, Hosio S, Kruger F, Zanni D (2012) Multipurpose Interactive Public Displays in the Wild: Three Years Later. *Computer* 45(5): 42–49. DOI: <http://dx.doi.org/10.1109/MC.2012.115>
- [11] Abowd G, Dey A, Brown P, Davies N, Smith M & Steggles P (1999) Towards a better understanding of context and context-awareness. Proc. 1999 First International Symposium on Handheld and Ubiquitous Computing, Karlsruhe, Germany, 304–307. DOI: http://dx.doi.org/10.1007/3-540-48157-5_29
- [12] Dey A (2001) Understanding and using context. *Personal and ubiquitous computing* 5(1): 4–7. DOI: <http://dx.doi.org/10.1007/s007790170019>

- [13] Liu W, Li X & Huang D (2011) A survey on context awareness. Proc. 2011 International Conference on Computer Science and Service System (CSSS), Nanjing, China, 144–147. DOI: <http://dx.doi.org/10.1109/CSSS.2011.5972040>
- [14] Mostefaoui G, Pasquier-Rocha J & Brezillon P (2004) Context-aware computing: a guide for the pervasive computing community. Proc. 2004 IEEE/ACS International Conference on Pervasive Services, Beirut, Lebanon, 39–48. DOI: <http://dx.doi.org/10.1109/PERSER.2004.1356763>
- [15] Müller J, Alt F, Michelis D & Schmidt A (2010) Requirements and design space for interactive public displays. Proc. 2010 International Conference on Multimedia, Firenze, Italy, 1285–1294. DOI: <http://dx.doi.org/10.1145/1873951.1874203>
- [16] Müller J, Wilmsmann D, Exeler J, Buzeck M, Schmidt A, Jay T & Krüger A (2009) Display blindness: The effect of expectations on attention towards digital signage. Proc. 2009 7th International Conference on Pervasive Computing, Nara, Japan, 1–8. DOI: http://dx.doi.org/10.1007/978-3-642-01516-8_1
- [17] Houben S & Weichel C (2013) Overcoming interaction blindness through curiosity objects. Proc. 2013 Extended Abstracts on Human Factors in Computing Systems, Paris, France, 1539–1544. DOI: <http://dx.doi.org/10.1145/2468356.2468631>
- [18] Brignull H & Rogers Y (2003) Enticing people to interact with large public displays in public spaces. Proc. 2003 IFIP International Conference on Human-Computer Interaction, Zurich, Switzerland, 17–24.
- [19] Huang M, Koster A & Borchers J (2008) Overcoming assumptions and uncovering practices: When does the public really look at public displays? Proc. 2008 6th International Conference on Pervasive Computing, Sydney, Australia, 228–243. DOI: http://dx.doi.org/10.1007/978-3-540-79576-6_14
- [20] Müller J, Jentsch M, Kray C & Krüger A (2008) Exploring factors that influence the combined use of mobile devices and public displays for pedestrian navigation. Proc. 2008 5th Nordic Conference on Human-computer Interaction: building bridges, Lund, Sweden, 308–317. DOI: <http://dx.doi.org/10.1145/1463160.1463194>
- [21] Tan S & Czerwinski M (2003) Information voyeurism: Social impact of physically large displays on information privacy. Proc. 2003 Extended Abstracts on Human Factors in Computing Systems, CHI '03, Florida, USA, 748–749. DOI: <http://dx.doi.org/10.1145/765891.765967>
- [22] Kaviani N, Finke M, Lea R & Fels S (2012) Investigating a Design Space for Multi-Device Environments. *International Journal of Human-Computer Interaction* 28(11): 722-729. DOI: <http://dx.doi.org/10.1080/10447318.2012.715535>

- [23] Noh Y (2013) A study on next-generation digital library using context-awareness technology, *Library Hi Tech* 31(2): 236–253. DOI: <http://dx.doi.org/10.1108/07378831311329031>
- [24] Kruk S & Decker S (2005) Semantic Social Collaborative Filtering with FOAFRealm. Proc. 2005 Semantic Desktop Workshop in conjunction with ISWC, Galway, Ireland
- [25] Kruk S, Grzonkowski S, Gzella A & Cygan M (2006) Ubiquitous Search and Browsing for Digital Libraries. Proc. 2006 7th International Conference on Mobile Data Management, Nara, Japan, 84. DOI: <http://dx.doi.org/10.1109/MDM.2006.71>
- [26] Aittola M, Ryhänen T & Ojala T (2003) SmartLibrary- Location-Aware Mobile Library Service. Proc. 2003 5th International Symposium Human-Computer Interaction with Mobile Devices and Services, Udine, Italy, 411–416. DOI: http://dx.doi.org/10.1007/978-3-540-45233-1_38
- [27] Aittola, M, Parhi P, Vieruaho, M & Ojala T (2004) Comparison of Mobile and Fixed Use of SmartLibrary. Proc. 2004 6th International Symposium Mobile Human-Computer Interaction, Glasgow, UK, 383–387. DOI: http://dx.doi.org/10.1007/978-3-540-28637-0_42
- [28] Ching-bang Y (2001) Personalized guidance and ubiquitous learning in intelligent library with multi-agent. Proc. 2010 2nd International Conference on Computer and Automation Engineering, Singapore, Singapore, 26-28. DOI: <http://dx.doi.org/10.1109/ICCAE.2010.5451351>
- [29] Larson R (1998) *Encyclopedia of Library and Information Sciences Third Edition*. Berkeley, Taylor & Francis. DOI: <http://dx.doi.org/10.1081/E-ELIS3>
- [30] Warner E, Murray A, Palmour V (1973) *Information needs of urban residents*. US Dept. of Health Education and Welfare Bureau of Libraries and Learning Resources, Washington
- [31] Kukka H (2012) *Case studies in human information behavior in smart urban spaces*. Doctoral thesis. University of Oulu Graduate School; University of Oulu, Faculty of Technology, Department of Computer Science and Engineering
- [32] Mills K (2010) *M-Libraries: Information use on the move in M-libraries 2: A Virtual Library in Everyone's Pocket*, London, Facet Publishing, 235–244
- [33] Jones M, Rieger R, Treadwell P. & Gay G.K (2000) Live from the stacks: user feedback on mobile computers and wireless tools for library patrons. Proc. 200 Fifth ACM Conference on Digital Libraries, Denver, Colorado, 95–102. DOI: <http://dx.doi.org/10.1145/336597.336646>

- [34] Lippincott J. (2008) Mobile technologies, mobile users: implications for academic libraries, ARL Bi-monthly Report 261.
- [35] Ooi K & Liew C (2011) Selecting fiction as part of everyday life information seeking. *Journal of Documentation* 67(5): 748–772. DOI: <http://dx.doi.org/10.1108/00220411111164655>
- [36] Loganantharaj R & Narayan V (2006) SemPub: An Ontology Based Semantic Literature Retrieval System. Proc. 2006 19th IEEE International Symposium on Computer-Based Medical Systems, Salt Lake City, Utah, 875–880. DOI: <http://dx.doi.org/10.1109/CBMS.2006.144>
- [37] Decker S, Melnik S, van Harmelen F, Fensel D, Klein M, Broekstra J, Erdmann M & Horrocks I (2000) The Semantic Web: the roles of XML and RDF. *IEEE Internet Computing* 4(5): 63–73. DOI: <http://dx.doi.org/10.1109/4236.877487>
- [38] KirjaSampo (accessed 12.06.2013) Kirjastot.fi. URL: <http://www.kirjasampo.fi>.
- [39] KulttuuriSampo (accessed 12.06.2013) Semantic Computing Research Group. URL: <http://www.kulttuurisampo.fi/>.
- [40] Mäkelä E, Hypén K & Hyvönen E (2011) BookSampo—Lessons Learned in Creating a Semantic Portal for Fiction Literature. Proc. 2011 10th International Semantic Web Conference, Boston, USA, 173–188. DOI: http://dx.doi.org/10.1007/978-3-642-25093-4_12
- [41] Nielsen A (2001) A National Survey of Reading, Buying and Borrowing Books for Pleasure: conducted for Books Alive, Canberra, A. C. Nielsen
- [42] UBIOulu mediamyynti (accessed 01.06.2013) Oulun yliopisto. URL: <http://www.ubioulu.fi/katugalleria>.
- [43] Beck K, Beedle M, van Bennekum A, Cockburn A, Cunningham W, Fowler M, Grenning J, Highsmith J, Hunt A, Jeffries R, Kern J, Marick B, Martin R, Mellor S, Schwaber K, Sutherland J & Thomas D (2001) Manifesto for agile software development. URL: <http://agilemanifesto.org/>
- [44] Vogel D & Balakrishnan R (2004) Interactive public ambient displays: transitioning from implicit to explicit, public to personal, interaction with multiple users. Proc. 2004 17th Annual ACM Symposium on User interface Software and Technology, Santa Fe, USA, 137–146. DOI: <http://dx.doi.org/10.1145/1029632.1029656>
- [45] Heikkinen T, Lindén T, Jurmu M, Kukka H & Ojala T (2011) Declarative XML-based layout state encoding for managing screen real estate of interactive public displays. Proc. 2011 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops), Seattle USA, 82–87. DOI: <http://dx.doi.org/10.1109/PERCOMW.2011.5766976>

- [46] Memarovic N, Langheinrich M, Alt F, Elhart I, Hosio S. & Rubegni E (2012) Using public displays to stimulate passive engagement, active engagement, and discovery in public spaces. Proc. 2012 4th Media Architecture Biennale Conference, Aarhus, Denmark, 55–64. DOI: <http://dx.doi.org/10.1145/2421076.2421086>
- [47] Hosio S, Jurmu M, Kukka H, Riekkilä J & Ojala T (2010) Supporting distributed private and public user interfaces in urban environments. Proc. Eleventh Workshop on Mobile Computing Systems and Applications (HotMobile 2010), Annapolis, MD, USA, 25-30. DOI: <http://dx.doi.org/10.1145/1734583.1734590>.
- [48] Dibb S & Simkin L (1996) The market segmentation workbook: Target marketing for marketing managers. London, Routledge.
- [49] George Robertson G, Czerwinski M, Baudisch P, Meyers B, Robbins D, Smith G & Tan D (2005) The Large-Display User Experience. IEEE Computer Graphics and Applications 25(4): 44–51. DOI: <http://dx.doi.org/10.1109/MCG.2005.88>
- [50] Bezerianos A & Balakrishnan R (2005) View and space management on large displays. IEEE Computer Graphics and Applications 25(4): 34–43. DOI: <http://dx.doi.org/10.1109/MCG.2005.92>
- [51] Sinclair J & Cardew-Hall M (2008) The folksonomy tag cloud: when is it useful? Journal of Information Science 34(1): 15–29. DOI: <http://dx.doi.org/10.1177/0165551506078083>
- [52] Oulun tapahtumakalenteri (accessed 12.08.2013) Oulun kaupunki. URL: <http://tapahtumakalenteri.ouka.fi>.
- [53] Google Maps Developer API (accessed 20.06.2013) Google Inc. URL: <https://developers.google.com/maps/documentation/javascript/>.
- [54] Niederst J & Robbins J (2001) Web design in a nutshell: A desktop quick reference vol. 2. O'Reilly Media.
- [55] Programmable Web (accessed 05.04.2013) ProgrammableWeb Inc. URL: <http://www.programmableweb.com/apitag/library>.
- [56] Outi verkkokirjastot (accessed 03.04.2013) Oulun kaupunki. URL: <http://oukasrv6.ouka.fi:8001/Intro?formid=findf>.
- [57] Oulun kaupunki - kirjasto (accessed 03.04.2013) Oulun kaupunki. URL: <http://www.ouka.fi/oulu/kirjasto/etusivu>.
- [58] GoodReads API (accessed: 12.06.2013) Goodreads Inc. URL: <http://www.goodreads.com/api>.

- [59] LibraryThing (accessed 12.07.2013) LibraryThing. URL: <http://www.librarything.com/>.
- [60] Google Books (accessed 02.06.2013) Google Inc. URL: <http://books.google.com/>.
- [61] The New York Times Best Sellers List API (accessed 01.07.2013) The New York Times Company. URL: http://developer.nytimes.com/docs/best_sellers_api.
- [62] Mitä Suomi lukee - Kirjakauppaliitto (accessed 01.05.2013) Kirjakauppaliitto ry. URL: <http://kirjakauppaliitto.fi/ratings>.
- [63] TasteKid API (accessed 13.07.2013) Tastekid. URL: <http://www.tastekid.com/page/api>.
- [64] Hosio S, Kukka H, Jurmu M, Ojala T & Riekkö J (2010) Enhancing interactive public displays with social networking services. Proc. 2010 9th International Conference on Mobile and Ubiquitous Multimedia, Limassol, Cyprus, Article No 23. DOI: <http://dx.doi.org/10.1145/1899475.1899498>

11. APPENDICES

Appendix 1 Library customer questionnaire

Appendix 2 User evaluation questionnaire

APPENDIX 1: LIBRARY CUSTOMER QUESTIONNAIRE

Esitiedot

Haastateltava nro _____

Ikä: 10-15__ 16-20__ 21-25__ 26-30__ 31-35__ 36-40__ 41-45__

46-50__ 51-55__ 56-60__ 61-65__ 66-70__ 71-75__ 76-80__

80+__

Sukupuoli: mies__ nainen__

Ammatti: opiskelija__ töissä__ eläkkeellä__ työtön__ muu, mikä? ____

Alasi, jos olet opiskelija tai töissä?

Kotikunta?

Kauanko olet asunut Oulussa?

Pääasiallinen kirjasto, jota käytät?

Kuinka usein käytät kyseistä kirjastoa kuukaudessa?

Kuinka usein käytät pääkirjastossa?

Haastattelu

1. a. Kuvaile tavallinen kirjastokäyntisi: missä käyt, mitkä asiat kiinnittävät huomiosi (uutuushylly, näyttelyt jne.)
b. Eroaako tämän kertainen kirjastokäyntisi tavallisesta?
2. Mitä odotuksia sinulla on, kun tulet kirjastoon? Miten kirjasto vastaa odotuksiisi?
3. Mitä kirjaston palveluita käytät? (lainaus (nuoret, aikuiset, musiikki, pelit), lehtisali, lukusali, studio, kaukopalvelu, satutunnit, maakuntakokoelma, kurssit, tietopalvelu)
4. Tiedätkö jo valmiiksi mitä kirjaa/teosta tulet hakemaan?
a. Ei
i. Tiedätkö jo lajityypin tai kirjailijan?
ii. Mikä olisi helpoin tapa löytää haluamasi kirja?
b. Kyllä
i. Mistä lähteestä (internet/kaverit/vinkit) sait idean?
ii. Löydätkö aina kirjasi tästä lähteestä?
5. Mitä välinettä aiot käyttää kirjan löytämiseen kirjastosta?
a. (Hakukone) Onko OUTI-verkkokirjaston haussa parannettavaa?
b. (Kirjastonhoitaja) Pyydätkö aina ensiksi apua työntekijältä?
c. (Etsiminen) Tiedätkö suunnilleen kirjan sijainnin vai joudutko etsimään sitä? Turhauttaako etsiminen?
6. Oletko käyttänyt UBI-näyttöä?
a. Ei
i. Miksi et ole käyttänyt
ii. Mitä oletat UBI:n sisältävän?
iii. Millainen sisältö palvelisi tarpeitasi
1. Yleisesti
2. Kirjaston käyttöön liittyen
b. Kyllä
i. Miksi käytit? Miten käyttö sujui? (oliko käytössä ongelmia?)
ii. Mitä sisältöä käytät ja kuinka usein?
iii. Mitä mieltä olet UBI:sta?
iv. Kaipaanko jotain lisää UBI-näyttöön? Yleisesti ja kirjastoon liittyen?

c. Olisitko valmis tunnistautumaan kirjastokortilla sekä käyttäjätunnuksella UBI:in? Miksi olisit / et olisi?

7. Laita nämä sovellusehdotukset tärkeysjärjestykseen
 1. Kirjan nimi näppäillään ja kirjan paikka näkyy kartalla
 2. Sovellus ehdottaa kirjoja käyttäjän valitsemista syötteistä (genre, kirjailija, jne.)
 3. Sovellus ehdottaa kirjaston valitsemia kirjoja
 4. Sovellus esittelee eri kirjaston eri tapahtumia
 5. Sovellus esittelee kirjaston eri tiloja, joissa voi viettää aikaa

Prerequisites

Interviewee number _____

Age 10-15__ 16-20__ 21-25__ 26-30__ 31-35__ 36-40__ 41-45__

46-50__ 51-55__ 56-60__ 61-65__ 66-70__ 71-75__ 76-80__

80+__

Gender male__ female__

Occupation: student__ employee__ pensioner__ unemployed__ other, what? ____

Your field of occupation?

Municipality of Residence?

How long have you been living in Oulu?

The library you frequent most?

How many times you visit a library in a month?

How many times you visit the Oulu City library in a month?

Interview

1. a. Describe your regular library visit: where do you go, what things raise your interest?
 - b. Does this visit differ from the normal visitation?
2. What expectations do you have when arriving at the library? How does the library meet your demands?
3. What services of the library do you use? (Loaning, reading room, newspapers, studio, remote service, story hours, Provincial Collection, courses, Information)
4. Do you know in advance what book/production you will pick up?
 - a. No
 - i. Have you decided on genre or author?
 - ii. What would be the easiest route to discover the piece you want?
 - b. Yes
 - i. What source (internet/friends/media) did you get the idea for the loan?
 - ii. Do you always use the same source in discovering the piece?
5. What tool will you utilize in discovering the location of the book in the library?
 - a. (Search machine) Is there anything to improve in OUTI- Web Library search?
 - b. (Librarian) Do you always ask help from a librarian?
 - c. (Searching) Do you know the approximate location of the book or do you have to look for it? Does it frustrate you?
6. Have you used UBI-hotspot before?
 - a. No
 - i. Why haven't you tried?
 - ii. What do you expect it to contain?
 - iii. What type of content would serve your needs?

1. in general
 2. in library context
- b. Yes
- i. Why did you use it? How was the usability? (Were there issues in the use?)
 - ii. What content do you use and how often?
 - iii. What do you think of UBI?
 - iv. Do you need something more in UBI-hotspot? In general or library specific?
- c. Would you be willing to log in with library card or user identification to the UBI? Why/Why not?
7. Put the following service proposals in the order of importance
1. Name of the book is inserted and the location of the book is displayed on a map
 2. Service that suggests books from user inputs (genre, author, etc.)
 3. Service that suggests books selected by librarians
 4. Service that suggests different events in library
 5. Service that demonstrates different departments around the library

APPENDIX 2: USER EVALUATION QUESTIONNAIRE

Esitiedot

Haastateltava nro.: ____

Ikä: lapsi__ teini-ikäinen__ nuori aikuinen__ aikuinen__ eläkeläinen__

Sukupuoli: mies__ nainen__

Tehtävät

1. Etsi sinua kiinnostava tapahtuma
2. Etsi sinua kiinnostava kirja palvelusta
3. Etsi *Jari Tervo: Troikka* arvostelut
4. Etsi musiikkiosaston aukioloajat
5. Etsi Myllyojan kirjaston osoite
6. Etsi Suomen luetuimman kaunokirjallisuusteoksen paikka

Kysymykset

7. Onko palvelu hyödyllinen?
8. Mitä palvelusta puuttuu?
9. Käyttäisitkö sovellusta uudelleen?
10. Huomasitko suosituksia? Ovatko suositukset hyviä?
11. Mikä oli hyödyllisin sovellus?

Prerequisites

Interviewee number _____

Age: child__ teen__ young adult__ adult__ pensioner__

Gender: male__ female__

Tasks

1. Discover an event you find interesting
2. Discover a book or production you find interesting
3. Locate reviews for *Jari Tervo: Troikka*
4. Locate the opening hours of the music department
5. Locate the address of the library in Myllyoja
6. Find the location of the most read fiction book in Finland in the library

Questions

7. Is the service useful?
8. What is missing from the service?
9. Would you use the service again?
10. Did you notice the recommendations? Were they any good?
11. What was the most useful application?