UNIVERSITY OF BERGEN

Geographical Information System to support location critical vaccine trials

Master thesis

by
Hans Terje Møller

The Department of Information Science and Media Studies
hans.moller@student.uib.no

08.06.2010

ABSTRACT

OpenXdata is an open-source community project where the aim is to eliminate the use of paper in data collecting of e.g. patients in the healthcare sector in low-income countries. This is done by a generic approach in data collection and management using mobile phones or handheld computers. As the current system stands, the users are only able to view the collected data in simple forms. With a Geographical Information System the user will be able to view the data in relation to geographical locations on a map where the data is collected. In this thesis we will have a look at what possibilities there are with using a Geographical Information System as a support system for location based data collections in low-income countries in the health care sector. This is achieved by developing a Geographical Information System application prototype. Its functionality and usability are thereafter evaluated by 10 users. The evaluation of the prototype shows that it is necessary to have such an application and it is an important addition to OpenXdata, which enable users to easily understand the collected data.

ACKNOWLEDGEMENT

There are a lot of people that I would like to thank for helping me and inspiring me to work with this thesis. Not all of them can be mentioned, but they know who they are.

I would first of all like to thank my supervisor Weiqin Chen who has been very helpful and understand during the progress of this thesis. Things did not always go as planned, but with her support it turned out not so bad after all.

The next person I would like to thank, Jørn Klungsøyr, have been a great co-supervisor. First of all, it was he who made this thesis possible. I would like to thank him for the great support during the development phase of this thesis. And I would especially like to thank him for the everlasting patience during the difficult times of the development. And last but not least, I would like to thank him for giving me the opportunity to travel to Uganda and meet the developers of OpenXdata, it will never be forgotten.

I also thank my fellow students for the good times in the reading room and for their support and comments to my work.

And last but not least, I would like to thank my family and friends for the continuously encouragement and support.

Bergen, June 2010

Hans Terje Møller

TABLE OF CONTENTS

1. INTRODUCTION

OpenXdata is an open-source community project at the Center of International Health at the University of Bergen where the aim is to eliminate the use of paper in data collecting of e.g. patients in the healthcare sector in low-income countries. This is done by a generic approach in data collection and management using mobile phones or handheld computers. With these devices the users can directly enter the data into an electronic form without the use of papers. The aim for this approach is to create cost-effective technologies for data collection, management and dissemination in low-income countries, in this case patient data. With this technology, we can eliminate some of the existing challenges with the use of pen-and-paper. Some of these challenges are; no explicit data on who wrote the form, where the form was written and when it was written. Along with this there are challenges with double data entry, validation, error correction, logistics, security, delay, etc. By using mobile data capture and management we are allowed to use any type of form to be handled digitally in an organization. It also ensures consistency of data, fewer steps where human error can be introduced, tracking of data, reduced time spent on documentation, instant feedback, reduced manpower and instant availability of data (Studentportal.uib.no, 2009). Along with these advantages we have the possibility to expand the usefulness by for instance adding a decision support system, synchronization and automatic generated reports.

The OpenXdata project is based on the open-source development method. The reason for this is to make it as low-cost as possible and allow more interoperability between the different partners in the project. Along with this, open-source ensures openness, free usage and collaboration across commercial and non-commercial entities. *"The promise of open source is better quality, higher reliability, more flexibility, lower cost, and an end to predatory vendor lock-in."* (Opensource.org, 2007).

This thesis is about developing and evaluating a Geographical Information System – GIS, for short – with the intention to function as a support system along with visualization of data. The approach to this thesis is to develop a software application prototyp as a module to OpenXdata. With this GIS application the users can locate information on a map from OpenXdata.

GIS is used to register, model, save, retrieve, manipulate, analyze and present geographical data. The definitions for a GIS may vary depending on the type of application. A more general definition would be *"a digital information system whose records are somehow geographically referenced"* (Medeiros & Pires, 1994).

1.1 THE GOAL OF THE THESIS

The main goal of this thesis is to develop and evaluate an application that will function as a support system by for instance track personnel collecting data in the field or by visualizing the collected data. As earlier mentioned, this is going to be done by developing a module for the already existing OpenXdata software. By developing this application, the end-users of OpenXdata can get a complete overview of all the collected data and the approximate location of the field workers. If then a field worker needs some sort of assistance, they can call in to the headquarters and ask the personnel if there are any colleagues close to them in the field. The field worker can then use this information to contact the closest colleague and ask for assistance. For this to be doable, all the handheld devices need to have Global Positioning System – GPS, for short – support. The device can then add the coordinates of the location to the form which they fill in and by this be able to show up as a marker on the map. An alternative to capture the location if GPS is not supported is by triangulation, although because of the lack of stable mobile connections and lack of mobile antennas in most of the low-income countries, this is probably not a reliable solution. Further on, we want to test if a GIS application can function as a support system in e.g. a vaccine trial by visualizing the collected data.

In short, with this application we are looking for the possibilities of using a GIS to function as a support system.

1.1.1 PREVIOUS RESEARCH ON GIS

The reason for choosing a GIS as an information system was because of its superiority over other information systems in combination with databases and its unique geographical analysis and visualizing. For instance, if someone changes some data in the database we will automatically update the GIS application.

Many will probably ask themselves; “Why not use a Global Positioning System for this purpose?” Surely a GPS-application could be used in many circumstances, but GPS has too many limitations for a project like this. *“Due to their reduced computational power and their autonomy, many actual navigation systems have some limitations. The geographical information they can use covers only limited areas, and its precision may be insufficient in some situations. The autonomy of the system complicates data updates and access to distant information sources, especially those providing dynamic information.”* (Stockus, Bouju, Bertrand, & Boursier, 1999). Stockus et al. also claims that the use of the Internet and a Web browser for accessing, processing and visualizing spatial information is one of the aspects that actually give strong interest in the field of research and development of GIS.

We will look more into research and articles about GIS in the next chapter of this thesis.

1.2 How does this solution differ from other existing solutions

Many would probably argue that this thesis and the GIS application that are developed is not something new and inventing, and that is probably true. GIS has been around for a long time and have possibly been used for creating support systems of different areas many times. However, this does differ from these systems since we are here dealing with creating a module for the already existing software OpenXdata. We will now step-by-step go through what makes this GIS application “special and important” in contrast to other existing GIS applications.

1.2.1 Vaccine trials

OpenXdata is developed in such a way that it is extremely flexible. With OpenXdata the user can create almost any imaginable research forms for collections of data. One of these research areas could be vaccine trials. *“A vaccine trial is a clinical trial that aims at establishing the safety and efficacy of a vaccine prior to it being licensed”* (Wikipedia, 2009). With OpenXdata the user can create a research form for collecting data about the vaccine trial. And with a GIS application the user is able to create geographical data (spatial data) of the data collected from the research. With this spatial data the user can compare the collected data with different elements, e.g. health conditions and population in certain areas. With this visualization the user may be able to better understand the data from the research. This is just one of the support functions of a GIS application.

1.2.2 Low infrastructure

In many cases where OpenXdata is applied are countries or areas with low infrastructure. This makes it challenging for the field workers collecting the data to have a decent overview of where they have been and what areas that remains. This is a challenge that can be overcome by implementing a GIS application for registration of areas which have been visited and which have not. In other words, the user can with a GIS application plot in the households that have been visited and by this get a good overview of which households that are remaining.

1.2.3 Visualization

Probably the most essential feature of a GIS application is the ability to visualize the collected data. This can be any kind of data, e.g. health monitoring, epidemics control, route optimization and resource administration. Considering this thesis, we visualize patient information. The users of the system can then search on a patient within the system and are then able to view the patient’s data and location. However, if we think on a bigger scale, the user can also be able to view if there are any health hazards for that given patient in the area by viewing other collected data on the map. This may help the user to treat the patient more correctly based on information from the GIS application.

1.2.4 Open-source and restrictions

So far we have not really seen any clear reason to not use existing GIS solutions; however, if we take open-source into consideration along with the restrictions of OpenXdata, we get another story. Since OpenXdata is an open-source community project, it is not possible to adapt any type of existing GIS solutions out of the box. As will be mentioned later in the thesis, the development of the GIS application started out with using Google Earth as the framework, except since this is restricted by licenses, the work was discarded. Instead of using Google Earth it was decided to uncover libraries that were completely open-source and free of charge. This is one of the reasons for developing a new GIS application for OpenXdata.

In addition to this, OpenXdata treats its data with an exact method, that is, OpenXdata has its own customized database for managing the data in which the GIS application has to be customized in order to retrieve data. When creating a report[1] in OpenXdata it generates a query statement to retrieve the results from a given research form created in OpenXdata. This leads us to customize the database connection in the GIS application to be able to handle this query and its result.

Lastly, since this GIS application is intended as a module to OpenXdata it is necessary to incorporate these two applications together as much as possible. The inspiration is to be able to access the GIS application within the OpenXdata application. It is by this not possible to take an already existing solution and just integrate it with OpenXdata. The functions that are intended in the OpenXdata GIS application will most likely in many cases be unique compared to other GIS systems. Along with this, it is highly unlikely that any other existing GIS application contains all of the functions intended for this GIS application in the future.

1.2.5 Other solutions

Some time was spent trying to find existing solutions that contained the intended functionality that this thesis comprehends. There are several solutions that incorporate a tracking functionality by retrieving the GPS coordinates from a mobile device. However, few of the solutions we came across were non-profit or open-source and contained the necessary functionality to make it work alongside OpenXdata.

1.3 Research question

The aim of this thesis is as mentioned to develop and evaluate a GIS with the purpose of giving the user visualization of data and support to field workers. The approach to achieve this is by developing a web application that will be incorporated with OpenXdata.

The research question for this thesis is:

Can a Geographical Information System function as a support system for location based data collections?

[1] A report in this context is a dataset of selected fields from a research form

1.4 Evaluation method

The application has been tested by two types of evaluators. The first group is a selective group of fellow students which have experience in usability testing. This group were given tasks concerning the user interface and handed a questionnaire for evaluating the user interface. The second group is some of the people working with or using OpenXdata. These evaluators were given tasks to test the functionalities of the application and answer corresponding questions.

The next step was to interpret the collected data and draw conclusions based on the answers on the different questions.

1.5 Organization of the thesis

The first chapter, this chapter, is a brief introduction to the thesis along with justification of why this thesis is being written. Chapter two gives a more in-depth view of literature used to write this paper along with an in-depth description of terms and tools used in this thesis. The third chapter contains a detailed description of the process in developing the prototype. Chapter four is the process of evaluating the applications along with its findings. The fifth and last chapter contains the conclusion of the thesis along with a section about future work.

2. LITERATURE REVIEW

In this section we will be looking at existing research on GIS and mapping of data in general and the general understanding of what support systems is. We will try to get an understanding of what has previously been done and how this may correspond to the work done in this thesis. Besides this, we will have a look at the research and development methodology adapted in this thesis. In the end of the chapter we will have a look at the evaluation methodology.

2.1 RELATED FIELDS

In this part of the chapter we will take a closer look at what GIS are and its usage areas along with an understanding of what a support system is.

2.1.1 GEOGRAPHICAL INFORMATION SYSTEM

A generic definition of GIS is: *"GIS is a special-purpose digital database in which a common spatial coordinate system is the primary means of reference."* (Fazal, 2008). With a GIS the users can apply any kind of map and import data on to this map by receiving data from a database. This can be any kind of data, e.g. weather information, earthquake hazards, population, building information, and so on. GIS has been around for decades, however, it has evolved rapidly the last decade or so. That is, it has been revolutionized from being a highly specialized niche to a technology that affects nearly every aspect of our lives, i.e. from finding driving directions to managing natural disasters (Neteler & Mitasova, 2008). The level of detail of a map in a GIS may vary a great deal depending on the intention of the system. One may be presented with a rather fuzzy map showing only small details, or instead, a map of high detail of a certain area. A more descriptive explanation of this can be viewed in the figures below.

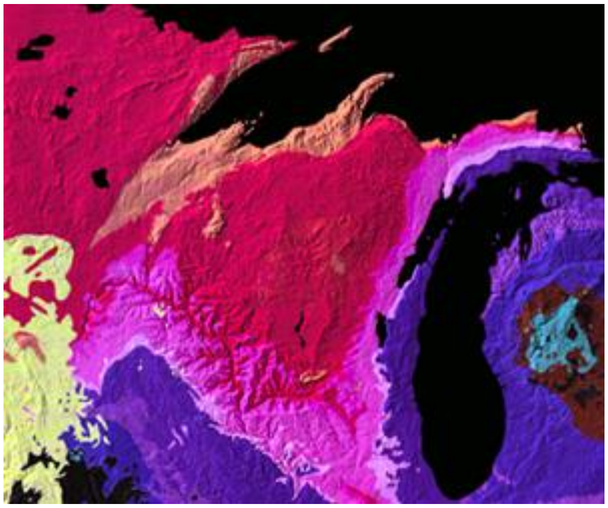


FIGURE 1 A FUZZY GIS MAP (GEOGRAPHIC INFORMATION SYSTEMS, 2007)

In Figure 1 we can see a rather abstract overview showing some landforms and geology.

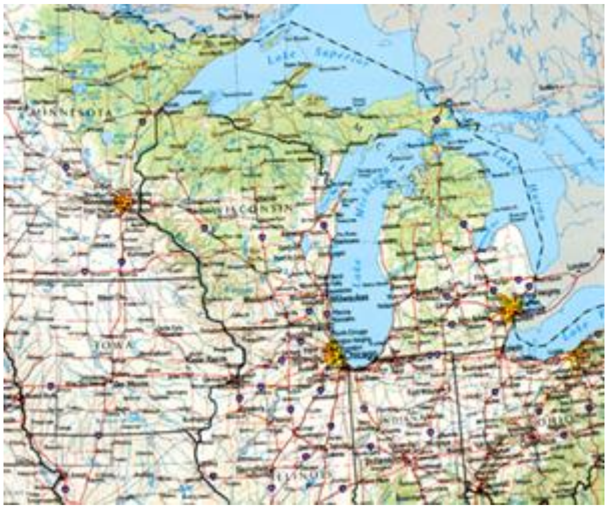


FIGURE 2 A DETAILED GIS MAP (GEOGRAPHIC INFORMATION SYSTEMS, 2007)

In Figure 2 we get a more detailed impression of that same area as shown in Figure 1.

A comprehensive GIS require means of (Fazal, 2008):

1. Data input, from maps, aerial photos, satellites, surveys, and other sources.
2. Data storage, retrieval, and query.
3. Data transformation, analysis, and modeling, including spatial statistics.
4. Data reporting, such as maps, reports and plans.

Fazal (2008) also states that three observations should be made about the previously mentioned definition:

*“**First**, GIS are related to other database applications, but with an important difference. All information in a GIS is linked to a spatial reference. Other databases may contain locational information (such as street addresses, or zip codes), but a GIS database uses geo-references as the primary means of storing and accessing information.”*

*“**Second**, GIS integrates technology. Whereas other technologies might be used only to analyze aerial photographs and satellite images, to create statistical models, or to draft maps, these capabilities are all offered together within a comprehensive GIS.”*

*“**Third**, GIS, with its array of functions, should be viewed as a process rather than merely software or hardware. GIS are for making decisions. The way in which data is entered, stored, and analyzed within a GIS must mirror the way information will be used for a specific research or decision – making task. To see GIS as merely a software or hardware system is to miss the crucial role it can play in a comprehensive decision-making process.”*

So what this tells us is that applying a GIS will let us process data in a more untraditional manner. But why should we apply GIS in the first place? One could say that the previous statement is reason enough to apply a GIS, but let us dig a little deeper to actually see why a GIS is important. First off, GIS is extremely dynamical in the sense that it can be customized to handle almost all types of data and information. A GIS is a rational way to present and juxtapose information and thematically data. *“By putting maps and other kinds of spatial information into digital form, GIS allows us to manipulate and display geographical knowledge in new and exciting ways.”* (Fazal, 2008). Fazal (2008) also writes: *“looking at data geographically can often suggest new insights, explanations.”*

A GIS is also a great framework for cooperation across organizations and communities. Since many types of GIS data are costly to collect, cooperation between organizations involved in e.g. environmental protection and such can cooperate using the same system. By defining a standard for the data, the organizations can exchange digital information between each other.

Since we are here dealing with a GIS module for an open-source project, it is only a matter of course to include how a GIS behave in such an environment. Since development of maps and gathering of spatial data is most commonly an expensive affair, many of the GIS application developers and contributors demand some sort of fee to use their product. Nevertheless, it exist several initiatives that strive to make it free and available to all users. *"Over the past few years a growing number of Open Source GIS, Web Mapping, and GPS projects has been established with different goals."* (Neteler & Mitasova, 2008). Further on in the article Neteler & Mitasova (2008) states the following about open-source GIS: *"The overall idea differs significantly from the strategies used in the proprietary GIS development industries."* What they mean by this is, in the proprietary GIS development industries, the functionality are in general more specific and customized for its purpose, however, with open-source GIS the idea is to generalize the functionality to function for several types of scenarios. From this article we get a clear indication that there is a wide range of free and available GIS software on the internet. However, the article also states that many of these GIS applications are more or less specialized on certain areas, even though this is something the developers are trying to avoid, and it is by this not uncommon that developers choose to design and develop their own GIS application. But sometimes it is also possible to evolve already existing solution in order to meet our own requirements. The question is then, what is the easiest, to understand already existing code or to start over with one's own code? One could say it depends on how far from your requirements the existing application are and your level of expertise in that area.

One thing is to have a free and available GIS framework, but it is also necessary to include a map in order to call it a GIS. In many cases it is not necessary to have a map of a high detail level over large areas; in those cases one might be able to create your own maps. But in those cases where one would need a high detailed world map it may be hard to come by. Most of the world map providers such as Google Maps, Yahoo! Maps, and so on, demands a fee if they were to be used for commercial purposes. Nevertheless, there is one open-source community that has taken this into their own hands. This map is named OpenStreetMap and is the map version of Wikipedia[2]. The ideology behind OpenStreetMap is to create a detailed world map which everyone can use. Normally this would be comprehensive and expensive

[2] Wikipedia is an online encyclopedia where "everyone" can contribute.

task, but the OpenStreetMap community has solved this by giving everyone access to contribute.

2.1.1.1 A HISTORICAL APPROACH TO GIS

"Development of GIS was influenced by key groups, companies and individuals along with timely development of key concepts. The idea of portraying different layers of data on a series of base maps, and relating things geographically, has been around much longer than computers." (Fazal, 2008). In Table 1 one can see the stages in the GIS development according to Fazal (2008).

TABLE 1 THE STAGES OF GIS DEVELOPMENT (FAZAL, 2008)

Stage	Period	Description	Characteristics
The Era of Beginning	1960 – 1973	Pioneering	• Individual personalities important • Mainframe-based systems dominant
The Era of Innovation	1975 – 1980	Experiment and practice	• Local experimentation and action • GIS fostered by national agencies • Much duplication of efforts
The Era of Commercialization	1980 – 2000	Commercial dominance	• Increasing range of vendors • Workstation and PC systems becoming common • Emergence of GIS consultancies
The Era of Exploitation	2000 – onwards	User dominance Vendor competition	• Embryonic standardization • Increasing use of PC and networked systems • Systems available for all hardware platforms • Internet mapping launched

As we can understand from the previous quotation, the digital GIS are just one of the many stages in the GIS history. Being able to use cartography to generate knowledge and information has been around for decades. An example of this is how John Snow, who was a British physician with an interest in cholera and anesthesia (McLeod, 2000), by using a map was able to locate an outbreak of cholera in Broad Street, Golden Square, London (Snow, 1849).

2.1.2 SUPPORT SYSTEMS

Support is a broad term depending on the context where it is used. In this case we think of support as the ability to help the user reach their goals in a better way. That is, to give the user a tool that can improve their working environments. We may call this type of support for "technical support"; *"In general, technical support takes the form of advice to users on changes to the product and/or how it is used for maximum benefit."* (Das, 2003). Further on in the article, Das (2003) states: *"an unsupported product is hardly considered a product at all"* In many cases, support is a necessity in order to make the software fully functional and give the user all of the potential abilities to work the system.

2.2 RESEARCH AND DEVELOPMENT METHODOLOGY

We will now have a look at what research and development methodologies that were used throughout this thesis. We start at looking at the research methodology and afterwards we move on to the development methodology.

2.2.1 RESEARCH METHODOLOGY

When initiating the research for this thesis, no framework for research was set in advance. The research was basically accomplished by doing a step-by-step research methodology where we first find what we want to work with, then we try to get it working, and at the end we make it work. This was rather unintentionally, but the research for this thesis has followed a research methodology known as "The research journey: an eight-step model". One can see the steps of this journey in Figure 3.

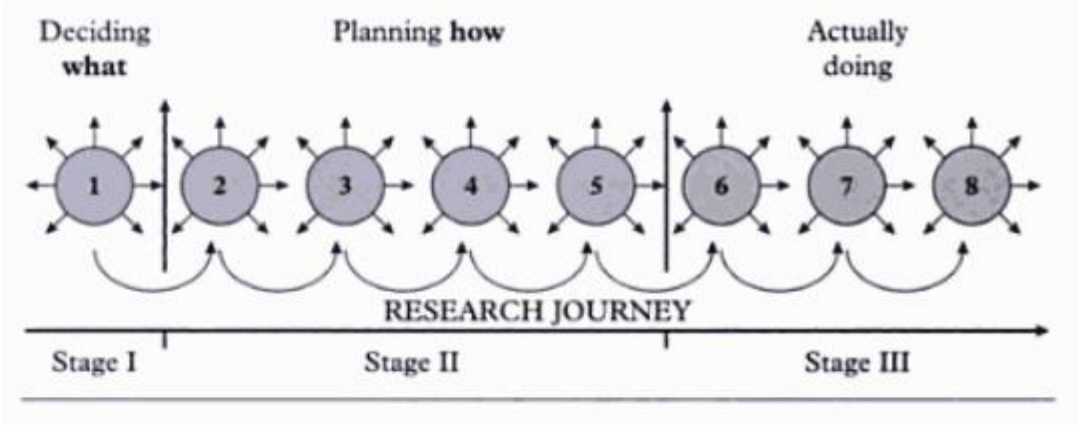


FIGURE 3 THE RESEARCH JOURNEY (KUMAR, 2005)

We can from Figure 3 see that the first stage consists of deciding what we want to research. The next four stages consist of planning how to achieve this research and the remaining three consists of completing the research. Even though this research methodology were used unintentionally, this was the idea from the beginning where we first find what we want to research, then make a prototype of this and at the end evaluate the prototype.

In order to not duplicate someone else's work, it was done a thorough research to figure out if it exist any similar solutions. This was basically done by searching the web for GIS that had the intention to work as a support system. Many different types of GIS were found, but none of them included the functionality that was made for this prototype and very few of them were open-source.

2.2.2 DEVELOPMENT METHODOLOGY

It has already been mentioned several times, to test this thesis's research question it was decided to develop and evaluate a prototype application. *"It is often said that users can't tell you what they want, but when they see something and get to use it, they soon know what they don't want."* (Preece, Rogers, & Sharp, 2002). The content from this quotation is exactly the reason for developing this prototype, in order to know if it can be used as a support system or not, the users need to understand how it will work and if it will work. So what is a prototype? *"A prototype can be anything from a paper-based storyboard through to a complex piece of software, and from a cardboard to a molded or pressed piece of metal. A prototype allows stakeholders to interact with an envisioned product, to gain some experience of using it in a realistic setting, and to explore imagined uses."* (Preece, Rogers, & Sharp, 2002). In other words, the idea with this prototype is to give the user the ability to actually envision the usage areas of a GIS application working alongside OpenXdata. This being a prototype, much of the possible functionality will be absent, however, it will give the user an indication of how it can be used. The prototype made for this thesis is a so called high-fidelity prototype. *"High-fidelity prototyping uses materials that you would expect to be in the final product and produces prototype that looks much more like the final thing."* (Preece, Rogers, & Sharp, 2002). The advantages of a high-fidelity prototype is that it gives the user a clear indication of how the system will work and the users themselves are able to test the functionality instead of thinking how the functionality would be. This prototype is in many ways a proof-of-concept design; however, this is a disadvantage of high-fidelity prototypes since it is commonly inefficient for this. Nevertheless, this being a disadvantage in general high-fidelity prototyping, it does not seem to affect this prototype in a high level.

2.3 Evaluation

In this section of the thesis we will look at how the evaluation was accomplished and which techniques that were used.

2.3.1 Why do we evaluate?

First of all, in order to actually understand if this prototype will work as a support system or not, it is a compulsory task to understand the needs of the user. This is the key reason for carrying through this evaluation in the first place. But we also need to evaluate for other reasons. One of the biggest challenges for a user interface developer is to put himself/herself in the position of the user. One of the biggest "enemies" to the developer is to presume that the users will understand the features and user interface. This is in most cases not the case. We therefore need to understand how the user interface work from a users perspective. *"Just as designers shouldn't assume that everyone is like them, they also shouldn't presume that following design guidelines guarantees good usability. Evaluation is needed to check that users can use the product and like it."* (Preece, Rogers, & Sharp, 2002).

2.3.2 Interview and questionnaire

"If your research problem lends itself to a qualitative mode of inquiry, you are most likely to use the unstructured interview or observation as your method for data collection." (Kumar,

2005). This corresponds almost exactly to the methods used for data collecting in this thesis. With only five usability evaluators and five functionality evaluators, we are constrained to carry through a qualitative research. But instead of applying an unstructured interview as mentioned in the quotation, the data collecting procedure for this thesis involved a structured interview. Along with this, observation was also included in the data collecting procedure. This was a non-participant observation, this is when one: *"as a researcher, do not get involved in the activities of the group but remains a passive observer, watching and listening to its activities and drawing conclusion from this."* (Kumar, 2005). This was not a comprehensive observation; it was done more or less to see how the evaluator handled the user interface and the functionality.

The interview used in this thesis is as mentioned a structured interview. *"In a structured interview the researcher asks a predetermined set of questions, using the same wording and order of questions as specified in the interview schedule."* (Kumar, 2005). But instead of an interview schedule this is an interview based on a questionnaire. "*A questionnaire is a written list of questions, the answers to which are recorded by respondents. In a questionnaire respondents read the questions, interpret what is expected and then write down the answers."* (Kumar, 2005).

When selecting the sources for collecting data it was decided to include one group to evaluate the functionality of the application (primary source) and another group to evaluate the user interface (secondary source). The primary source is people working with or using OpenXdata. The secondary source is the people with experience of usability evaluation.

3. Design and Development

Design and development is a vast part of this thesis, much of the time has been spent on research of different technologies and development methods. Since this software is a module of already existing open-source software, much consideration was taken in form of using technologies that are open-source or free in form of license. This resulted in a lot of tests and failures before the final decisions were made.

3.1 Requirements

This being a module of the OpenXdata project many of these requirements can be cross-referenced with the requirements for OpenXdata.

3.1.1 Overall description

This part of the requirements includes what is intended with this project, which the clients and customers are, who the users are, and the general constraints of the project.

3.1.1.1 Project perspective

The purpose of this GIS application is to expand the usage areas of the OpenXdata software. That is, to give it a new usability element by including a graphical representation of the data collected. Along with this, the intention is to give the user a new level of support, both for themselves and the field workers.

3.1.1.2 Client, Customer and other Stakeholders

The intended clients, customers and stakeholders are in general whoever uses the OpenXdata software for data collections. That can be small or big health organizations, hospitals, doctor offices, countries or international health organizations such as the World Health Organization (WHO) [3].

3.1.1.3 Users of the Application

There will most likely not be any constraints of who the user can be. This application shall use easy and understandable word and expressions. Because of this anyone can use the application without the need of any comprehensive education or training.

[3] http://www.who.int/en/

3.1.1.4 General Constraints

The most obvious recognizable constraints for this application are the following:

1. Open-source
   - OpenXdata is an open-source community project. It is therefore necessary to develop this application accordingly.
2. Implemented into already existing software
   - Since this is an intended module for already existing software we have to follow the constraints alongside OpenXdata.
3. Programming languages and tools
   - Since OpenXdata has been developed in a certain way, it is essential to follow those constraints in order to not meet any unnecessary challenges. That is, to use the same programming language or a programming language that can be used alongside the code of OpenXdata. And use the same tools which were used for developing OpenXdata in order to stay away from compatibility issues.
4. Usage areas
   - Since we are customizing this application to work alongside OpenXdata we are constraining the usage areas to whatever the usage area is for OpenXdata.

3.1.1.5 The Scope of the Work

At the moment the scope is rather diminutive considering the unsecure utility value of the application. The intention is to create a simple version of the application with the data visualization functionality and then to test this on the developers/users of OpenXdata to find out if this is something they may want to implement in the future. Hopefully as the project goes along, more people get involved and more features gets thought of and included. The vision is to create a starting point for an application that may include more developers over time and result in a fully working application at some point.

3.1.1.6 The Scope of the Application

A GIS with built-in visualization functionality is nothing new in the sense of technology. But considering the new possibilities in a system such as OpenXdata this will possibly extend the scope of the application drastically. As mentioned in the section above, the thesis is starting out as a small prototype with few features. But if more features get implemented and finalized, the scope of this GIS application will probably extend much further than first intended. At some point this application will hopefully be well integrated with the existing OpenXdata application and by this offer a vast selection of possibilities for the user. This can for instance be to offer the user a tracking functionality, statistical data of areas, workflow management, and so on. And all this is done more or less in a graphical manner, that is, to represent the data collected in a more graphical and visual manner.

3.1.2 Functional Requirements

In this section we will mostly be looking at the visualization functionality of the system since that is the scope the thesis.

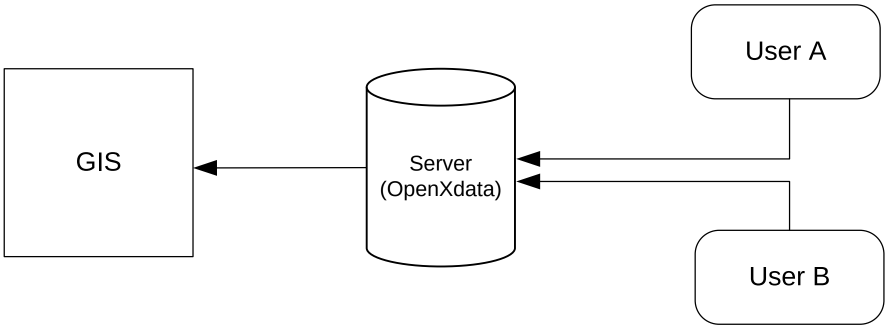


FIGURE 4 REPRESENTATION OF THE SYSTEM CONTEXT

In Figure 4 one can see a coarse representation of the context of the system. The “GIS” represents the Geographical Information System application. Then we have the OpenXdata server, and “User A” and “User B” represents the fieldworkers. The course of events here is as follows; the users send their reports from the field also containing a longitude and latitude to determine their location. The report is then stored in the OpenXdata server. After this is accomplished the GIS application connects to the server at a prompt of the user and the collects all the reports. The reports are then placed on the map as markers by using the latitude and longitude which came along with the reports from the field. By clicking these markers we can then see what they contain.

Below one can see the functional requirements of the system:

1. Lets the user navigate the map (panning and zooming).
2. Lets the user choose maps.
3. Gives the user the option of retrieving reports from the database.
4. Gives the user the option of selecting the desired report in a menu.
5. Lets the user see the selected report as markers on the map.
6. Gives the user the option of selecting a report as a marker and show the report content in a pop-up.

3.1.3 Non-functional Requirements

The non-functional requirements are the requirements that are non-specific, that is, instead of giving an indication of what the system shall *do*, it gives an indication of how the system should *be*.

3.1.3.1 Look and Feel Requirements

One of the main focus areas in developing this software is to make it as simple and dynamical as possible. One of the challenges in making it feel simple and dynamical is to make a seamless integration of the database connectivity. With this function the user shall be able to retrieve all the saved reports from OpenXdata. After the reports have been retrieved the user shall be able to click one of the reports in a list and get all the patients in that report displayed on the map as markers. Then the user shall be able to click that patient marker to get the information collected about that patient. In this way the database request are done dynamically and the information retrieved are depending on the content of the report. The challenge is to get the user to understand this feature without explaining to the user how the software works. This can be done with giving the user obvious choices and also giving the user restrictions and feedback if the user does something wrong.

Another challenge is to make an easy understandable user interface. The user interface should be familiar with already existing map applications so that the user with that type of experience could easily recognize the features. And of course, users with a general understanding of conventions should be able to understand the user interface in this application. The use of easy understandable icons and descriptive text labels are therefore an important requirement to this application.

3.1.3.2 Usability Requirements

Give the user obvious choices and help the user to interact correctly with the application. This is done by not adding or highlighting options and choices to the users unless it can be used. By doing this, the user will be able to handle their choices in a correct manner. The different choices must also be limited to their objective and not contain unnecessary options which may confuse the user.

The application should be efficient to use and give the user proper feedback of their actions. This meaning, the user can efficiently reach their goals with few mouse clicks. Another optional feature goal is to implement keyboard shortcuts for more experienced users.

Along with ease of use and efficiency, the applications should be easy to learn. It is important to not think of the user as an advanced user, at least for the basic functionality. Since this application probably will be used by a large diversity of users, the application must as earlier mentioned give the user obvious choices for the basic functionality. This meaning, the options to the user should be explained with generalized icons or explainable text. This could of course be a problem since this application is likely to be used over several continents whereas the conventions may differ from each other. The application will also contain advance choices for more advance users. These choices must also be easy to learn and understandable. Again, this is done by making the options self-explanatory with use of either well known icons or explanatory text.

3.1.3.3 Performance Requirements

As this application are highly likely to be used in underdeveloped countries with low internet bandwidth and older technology, it is important to focus on the performance aspect of this software. To do this, we need to support local map rendering and local database support. By doing this, the application is not constrained to have an internet connection at all times. This application also has to be connected to a database in order to receive data to display on the map. The amount of data to be downloaded will probably vary a lot; the application should then support the feature of downloading only the necessary data at each given moment.

Since this application will be developed in Google Web Toolkit (GWT)[4] and will basically act as a web page, it is probably not an important factor to focus on the performance when it comes to resource usage from the machine hardware. Though, it might require some bandwidth usage when loading the different elements of the page. This could be solved with low-bandwidth style sheets.

3.1.3.4 Maintainability and Portability Requirements

The application will run on a local or external server and it will by this not be necessary to install the software more than once on one computer in order to use it. The user should be able to open a web browser and go to the address where the application runs and access the application. The application will also be maintained by developers in the sense that they will continuously improve the software and release updates.

3.1.3.5 Security Requirements

Security is a very important factor for this application. Since we with this application are handling patient records, it is important to maintain the doctor-patient confidentiality. But since this application will be a built-on module to the OpenXdata framework, we can rely on the security implementation from OpenXdata. That is, the user will have to log in to OpenXdata in order to use this application.

3.1.3.6 Legal Requirements

The application will not use any other frameworks than those who are open-source and free of charge and usage. However, the users may themselves choose to use third party software and such, e.g. Google Maps.

3.2 Design

The first stage in designing this application was to figure out each step in order to retrieve the data from the field workers in order to display them as markers on the map. The initial idea was to create a system that would track the personnel in near real-time by tracking their built-

[4] Details about GWT can be found under 3.3.1 Tools

in GPS in their mobile devices. However, this was discarded early in the process since it involved too many technical challenges. The process of developing this application has been evolutionary in the sense that new elements have been added or changed as the project went along.

3.2.1 Initial system architecture

In Figure 5 one can see the initial system architecture. As can be seen later in the chapter, it has not really changed much since the start; however, the technical solutions of how to achieve this have changed more or less completely. As mentioned above, the initial idea was to track the personnel in real-time. It was soon decided to discard this idea and focus on retrieving the data from the database as soon as it got uploaded. By doing this it was possible to get a fairly accurate update on where the field workers were located. Another idea was to generate statistical information based on the data collected, but this was also discarded mostly because of time being the enemy and the technical difficulties.

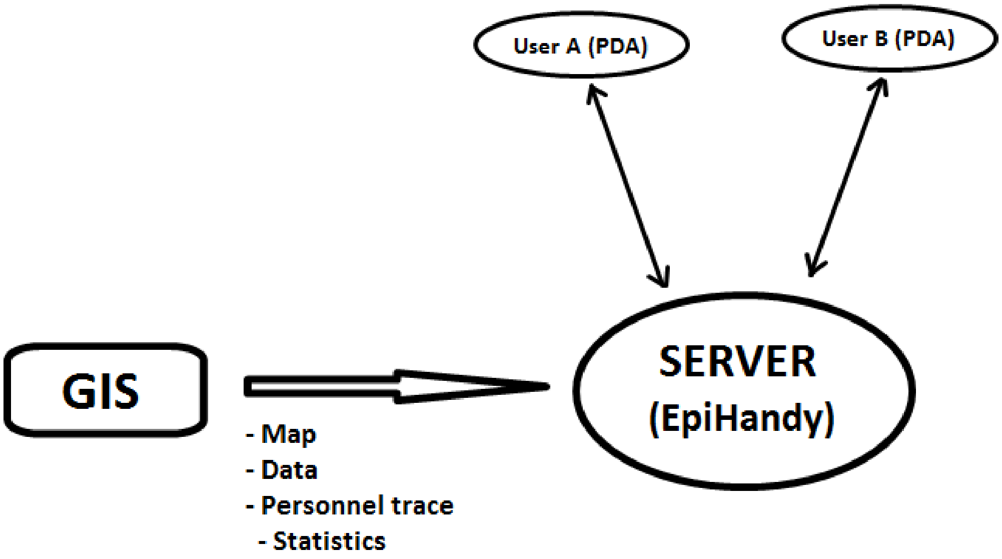
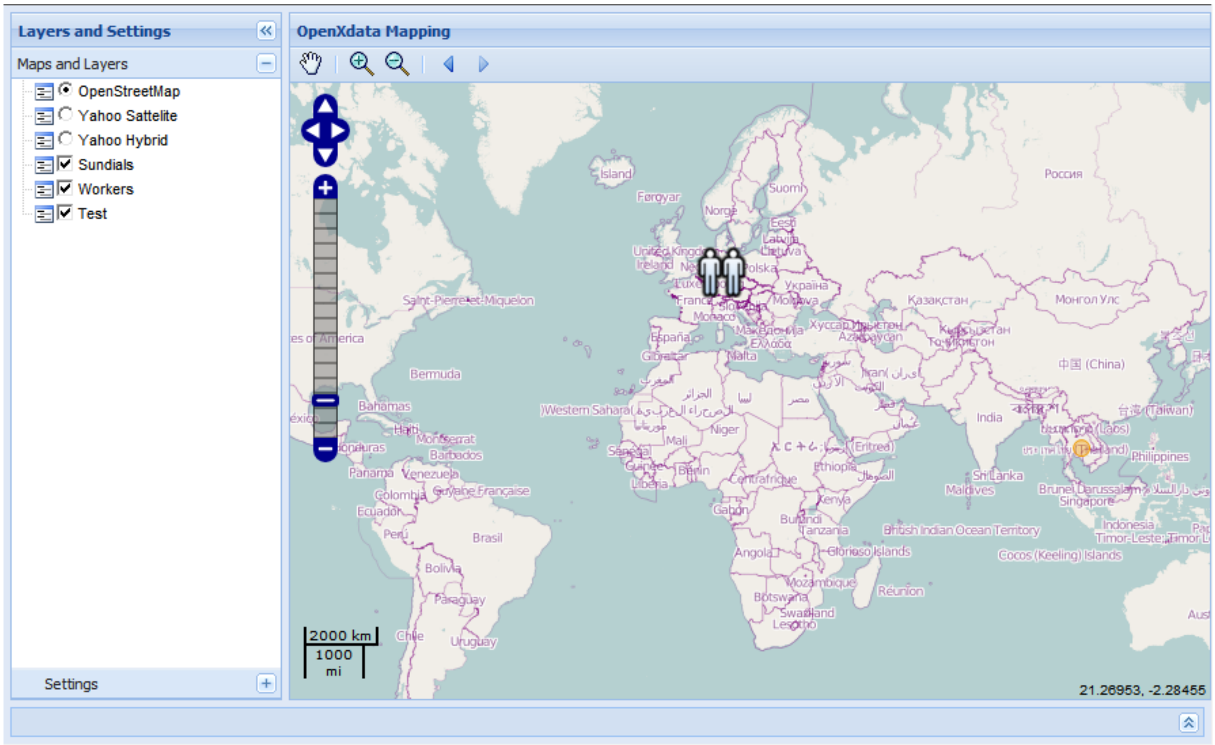


FIGURE 5 INITIAL SYSTEM ARCHITECTURE

3.2.2 Prototype I

The first prototype, Figure 6, was made using MapFish and OpenLayers as the frameworks (more on this later in the chapter). This prototype only included the basic functionality and not the intended functionality for viewing the data from the OpenXdata database. Work did at an early stage start on making a prototype by using Google Earth; however, this never had a representative user interface and is therefore not included here.



Layers and Settings

Maps and Layers

● OpenStreetMap

○ Yahoo Sattelite

○ Yahoo Hybrid

☑ Sundials

☑ Workers

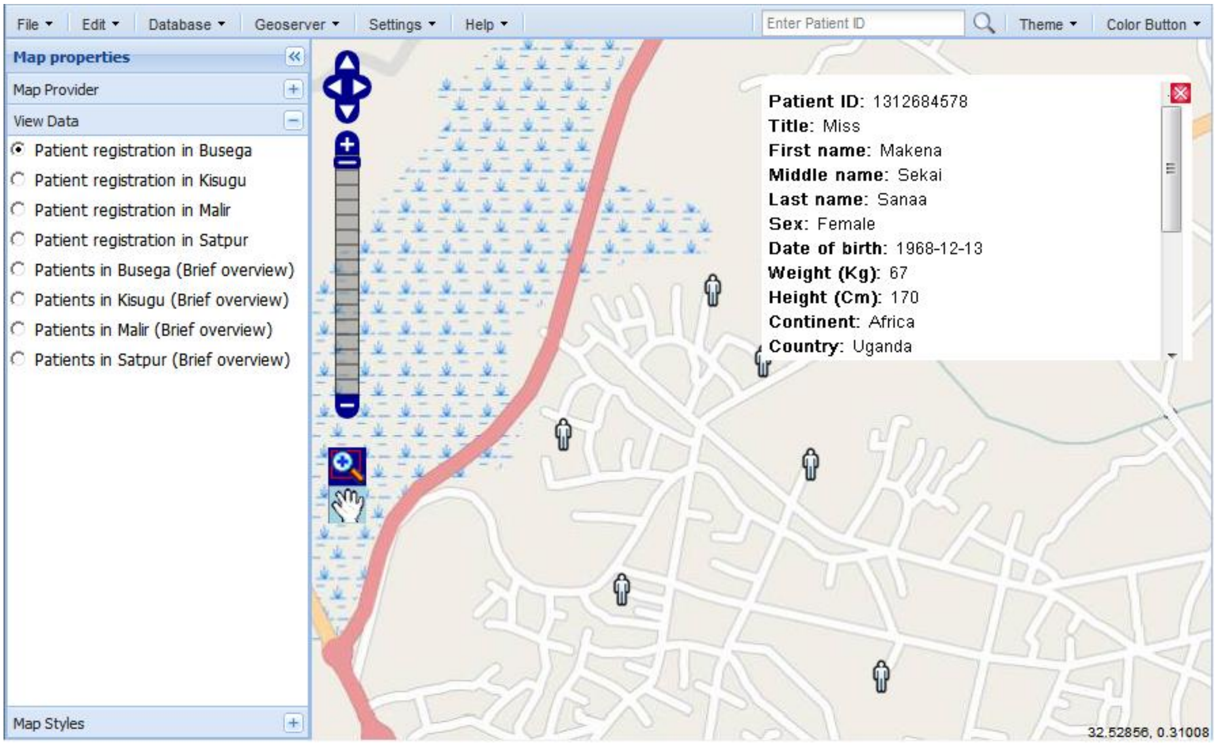
☑ Test

Settings

FIGURE 6 SCREENSHOT OF THE FIRST PROTOTYPE

3.2.3 Prototype II

The second prototype is also the current prototype. As can be seen in Figure 7, it looks almost the same as the first prototype (Figure 6). But this prototype is built using other libraries such as GWT-OpenLayers and Ext GWT (more on this later in the chapter). In this prototype we were able to integrate the functionality for gathering data from the OpenXdata database and display it on the map.


File
Edit
Database
Geoserver
Settings
Help
Enter Patient ID
Theme
Color Button
Map properties
Map Provider
View Data
● Patient registration in Busega
○ Patient registration in Kisugu
○ Patient registration in Malir
○ Patient registration in Satpur
○ Patients in Busega (Brief overview)
○ Patients in Kisugu (Brief overview)
○ Patients in Malir (Brief overview)
○ Patients in Satpur (Brief overview)
Map Styles
32.52856, 0.31008

Patient ID: 1312684578
Title: Miss
First name: Makena
Middle name: Sekai
Last name: Sanaa
Sex: Female
Date of birth: 1968-12-13
Weight (Kg): 67
Height (Cm): 170
Continent: Africa
Country: Uganda

FIGURE 7 SCREENSHOT OF THE SECOND AND LAST PROTOTYPE

3.2.4 Class diagram

Figure 8 is the class diagram of the GIS application.

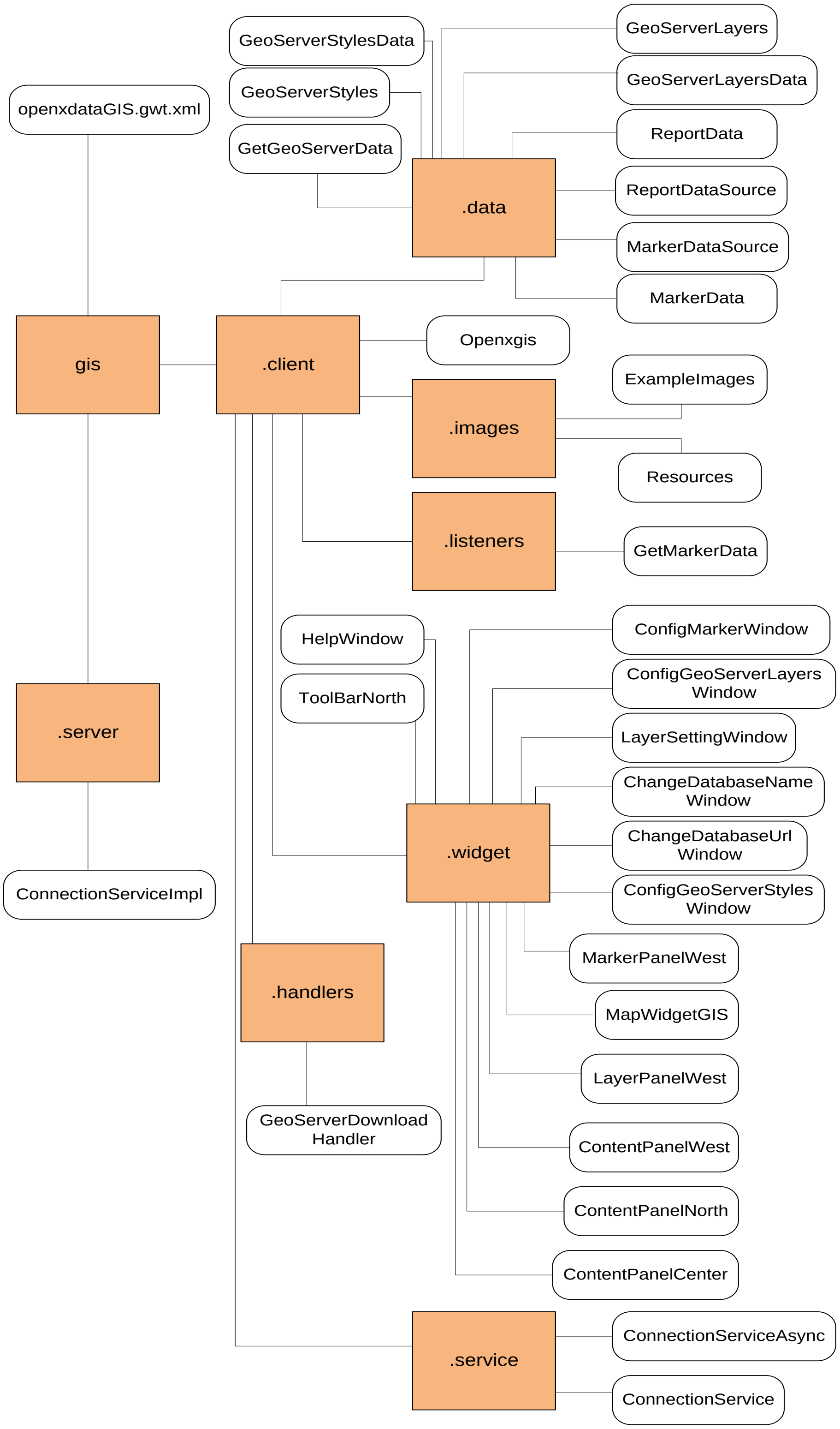


FIGURE 8 CLASS DIAGRAM

3.2.5 System architecture

Figure 9 is the system architecture of the GIS application.

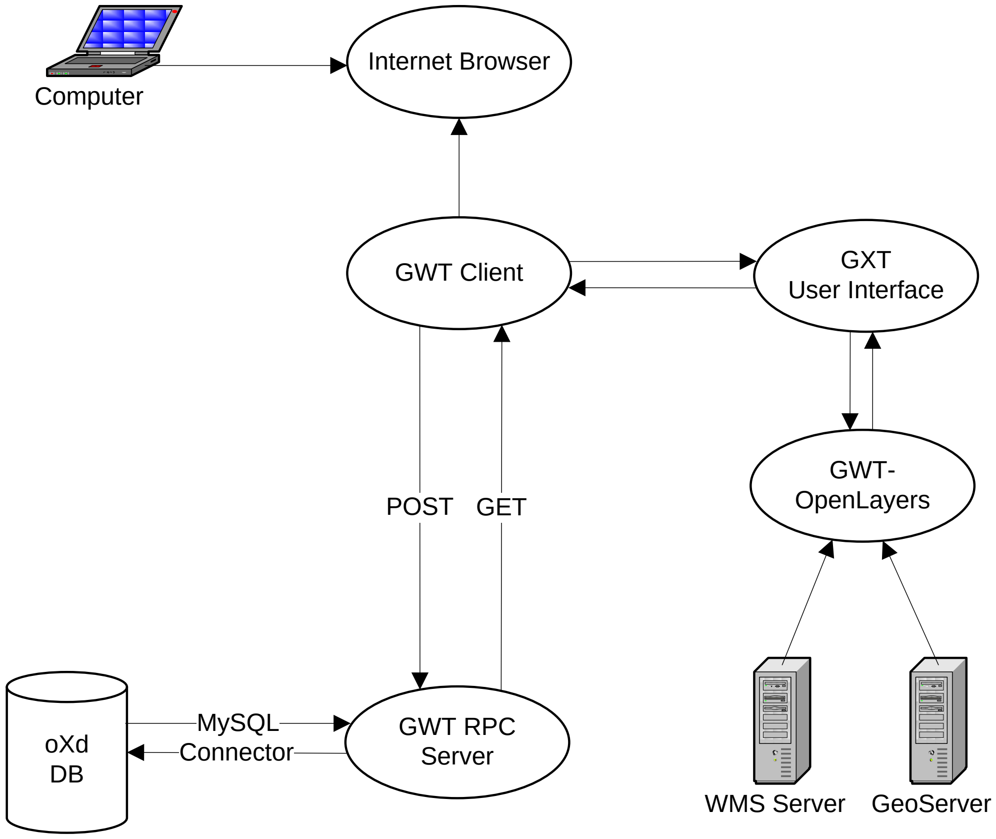


FIGURE 9 SYSTEM ARCHITECTURE

Acronym explanation[5]:

- GWT: Google Web Toolkit
- RPC: Remote Procedure Call
- GXT: Also known as Ext GWT
- GWT-OpenLayers**:** Java library for the OpenLayers JavaScript library
- oXd DB: OpenXdata database
- WMS: Web Mapping Service

[5] For a more detailed explanation of some of these acronyms, look at 3.3.1 Tools.

3.3 Development

The development of the GIS application has definitely been the most frustrating and time consuming element of this thesis. Due to many wrong turns early in the project, a lot of dead ends were hit and we had to turn around and try again. This became extremely time-consuming and a lot of work ended up being in vain. This can be blamed on a lack of quality research and maybe ignorance, however, the choices was made in the belief that it could be done and accomplished. Had this been a stand-alone project were there was not any constraints following licensing and such, it probably would have had a different outcome, but since we are here dealing with a module for a open-source community project it was necessary to take precautions when these issues surfaced. We will have a deeper look at this later in this chapter.

3.3.1 Tools

In order to make this application it was necessary to include a wide range of tools. All these tools work together to make the application to what it is. Some tools are also included for the collaboration with other developers. Most of the tools in use are generally known and has good support and documentation.

3.3.1.1 OpenXdata

This is the software made from the developers in cooperation with the Center of International Health. OpenXdata[6] is an application for creating forms for data collecting, in this case mainly health specific data. The user can create a form using OpenXdata on a local computer and then export this form with its defined questions to a mobile device. The user can then collect data on the mobile device using the form from anywhere they like. After the data is collected the user can send their results to a database for storage. With this tool the user can easily customize their own surveys and studies for data collecting. OpenXdata is developed using Google Web Toolkit which allows the user to open the application in their internet browser either locally or from a server. This software is also open-source and free of charge.

[6] http://www.openxdata.org/

3.3.1.2 JAVA

Java[7] is the programming language used for developing both OpenXdata and the GIS application. Java is an object oriented programming language which is well documented and well known. The benefits of using Java as the programming language is that it has a wide support over several platforms which allows all kinds of users to take advantage of it.

3.3.1.3 ECLIPSE

Eclipse[8] is a development platform with the possibilities to employ different frameworks through plug-ins. This platform is open-source and a not-for-profit project community. In this relation the platform uses Google Web Toolkit as the framework which uses Java as the programming language for development.

3.3.1.4 APACHE TOMCAT

Apache Tomcat[9] is an open-source servlet container as a result of Java Servlet and JavaServer Pages (JSP). Java Servlet is a class within the Java library which lets a Java class request a Hyper Text Transfer Protocol (HTTP). And JSP is a Java technology that provides the users with support for Web development. In other words, JSP is a high-level abstraction of Java servlets which allows Java code to be compiled on the server to deliver for instance a Hyper Text Markup Language (HTML) document. With Apache Tomcat one can deploy a Web Archive which lets the user run an application directly from the Apache Tomcat server. In this context Apache Tomcat is used to run OpenXdata, GeoServer and phpMyAdmin.

3.3.1.5 GEOSERVER

GeoServer[10] is an open-source community-driven project. GeoServer is a software server written in Java which lets the user add and edit geospatial data to their own server. In other words, the user can add their own maps/layers to the server and import them into their own mapping applications. The maps/layers may also contain data about them which the user can request from GeoServer through their application. GeoServer serve support for among other things Web Feature Service (WFS), Web Coverage Service (WCS) and Web Map Service

[7] http://java.sun.com/javase/

[8] http://www.eclipse.org/org/

[9] http://tomcat.apache.org/

[10] http://geoserver.org/display/GEOS/What+is+GeoServer

(WMS). GeoServer also lets the user integrate GeoServer with existing mapping APIs, e.g. Google Maps and Yahoo Maps.

3.3.1.6 EXT GWT (GXT)

Ext GWT[11] is a Java library for constructing internet applications with the use of Google Web Toolkit. With the Ext GWT library the user can create a customizable user interface for own purposes. Ext GWT supports a wide range of different widgets which gives the user several options for customization. Ext GWT is an open-source community and free of charge.

3.3.1.7 GWT-OPENLAYERS (GWT-OL)

GWT-OpenLayers[12] is a small developer community which aims to integrate OpenLayers into the Google Web Toolkit framework. GWT-OpenLayers is used as a plug-in or external library which the user can import into their project. With this library the user can create a mapping application like they would with OpenLayers, but instead of using JavaScript as the programming language; the user can develop it using Java. The GWT-OpenLayers still lack some of the features from the OpenLayers library but at the time being it covers the most basic features.

3.3.1.8 GOOGLE WEB TOOLKIT (GWT)

Google Web Toolkit[13] (GWT) is a toolkit for development of browser-based applications. GWT lets the user create complex JavaScript front-end applications in Java. In other words, GWT is for the most part Java source which converts to JavaScript when compiled. GWT is Open Source and free of charge.

3.3.1.9 PHPMYADMIN (MYSQL)

OpenXdata uses among others phpMyAdmin[14] (MySQL) to manage their database. phpMyAdmin is a free software tool to handle MySQL. MySQL is a relational database management system (RDBMS). With phpMyAdmin the user can create databases and its content. In this context the GIS-application calls on the OpenXdata database in phpMyAdmin and retrieve the data from the database and add it in the GIS-application.

[11] http://www.extjs.com/

[12] http://gwt-openlayers.sourceforge.net/

[13] http://code.google.com/intl/no/webtoolkit/overview.html

[14] http://www.phpmyadmin.net/home_page/index.php

3.3.1.10 TORTOISESVN

TortoiseSVN[15] is software for version control. With TortoiseSVN several users can connect to a repository where the source code is located and individually work on the content there. TortoiseSVN handles all the commits to the repository with version control. By using this tool the users can cooperate through a joint repository and by this have the latest source code at all times. TortoiseSVN is also free to use.

3.3.2 PROGRESS

As mentioned early in this chapter there has been a lot of back and forth when developing this application. This was a result of developing for an open-source community; thereby the application could not apply a product which was bound to any kind of licenses. We will now step-by-step go through the different stages for developing the GIS application.

3.3.2.1 ITERATION 1 (AUGUST – OCTOBER 2009)

The initial plan was to develop the GIS application under the software named EpiHandy developed at Center for International Health. However, this was early in the process discarded and the choice went over to develop under the OpenXdata software which is a result of the EpiHandy software. Along with this the plan was to use Google Earth as the framework for the GIS application. Being that it was early in the process it was necessary to perform some research on how OpenXdata worked and how it could be used to develop the GIS application. Some time was spent on this by downloading the source code of OpenXdata and analyzing how it was built.

After getting the basic understanding of how OpenXdata was set up, work started on developing the GIS application using Google Earth. This was also a new experience, so a lot of time went in to figuring out how to use Google Earth and how to implement it alongside OpenXdata. Since Google Earth has a powerful API the development went rather fast and seemed to be progressing rather well.

When reaching around the middle of October it was discovered that we could not use Google Maps due to licensing issues[16]. If we were to use Google Maps for commercial purposes we

[15] http://tortoisesvn.tigris.org/

[16] http://earth.google.com/intl/en-US/licensepro.html

would have to pay a fee[17] for each user of the system which led to too high expenses. This was a major setback since a lot of time and work had been put in to figuring out how to develop with Google Earth. Soon after work started on finding a new framework for developing the GIS application which were open-source and not bound by any kind of license.

After about a week of research, it was decided to try out MapFish along with OpenLayers. This involved a wide range of new challenges, mainly because both of these frameworks are developed in JavaScript.

3.3.2.2 Iteration 2 (November – January 2009/2010)

Creating the basic user interface with MapFish was mostly a straight forward task since it exist a diverse range of examples on the web. However, when it came to creating the functionality, i.e. marker functionality, database connection and so on, the challenge of using JavaScript as a programming language came clear. Because of the lack of experience using JavaScript, much time was spent on figuring out how to complete the different challenges. One of the biggest challenges was how to retrieve the data from OpenXdata through the database using JavaScript. The support of connecting to a database through JavaScript seemed very difficult since JavaScript does not support this kind of functionality on the client-side of the code. It was therefore necessary to create a server-side for handling these actions. The basic idea was to create a server-side using hibernate to retrieve the data from the database and send it to the client-side. Unfortunately it seemed easier in theory than what it was to actually perform this in practice.

Hibernate is library for the Java language which provides a framework for mapping Java classes to database tables. This relieves the developer from manually enter the SQL statements. Mapping the Java classes is done in a XML-file. By then using this XML-file hibernate can maintain the database schema which defines the different database statements.

However, as earlier mentioned, this was easier said than done. The challenge was to figure out how to use hibernate with JavaScript which did not seem to be a widely adopted practice as far as examples and tutorials on the web were.

[17] http://earth.google.com/support/bin/answer.py?hl=en&answer=21406

6. November 2009 – 21. November the students working on their thesis alongside the OpenXdata project were invited to attend the OpenXdata conference and launch in Kampala, Uganda. Since documentation of OpenXdata was rather absent at that time this trip gave us a great opportunity to get a deeper understanding of how OpenXdata was built. And also what sections of the code we should focus on. Many questions got answered and helpful guidance was acquired from the developers working on OpenXdata. However, many of the challenges still stood strong after this trip.

3.3.2.3 ITERATION 3 (FEBRUARY – MAY 2010)

In late January it was discovered that a small open-source community were working on creating a Java library of the OpenLayers JavaScript library named GWT-OpenLayers. After contacting one of the developers in this community it was confirmed that this project was still under development and new features were added frequently. Since progress with MapFish and OpenLayers went rather slowly, it was soon decided to stop development on this and start over with the GWT-OpenLayers library. By changing the libraries it was possible to do all the coding in Java with the use of Google Web Toolkit as the framework for the application. Along with this it was decided to use GXT (Ext GWT) as the library for the user interface. This is a Java library based on the JavaScript library named Ext JS. With this library it was possible to make an almost identical replica of the MapFish user interface. Along with these changes other developers were involved in the project to develop other features than the marker feature which this thesis involves. This made it easier to solve issues as they came along.

Changing the libraries seemed to be a step in the right direction, development speeded up and a coarse prototype was soon after completed. However, there were still challenges on how to retrieve the data from OpenXdata and displaying them on the map dynamically. The idea was yet to use hibernate for this, but because of the lack of experience using this tool, it was soon decided to use the built-in tool in Google Web Toolkit named Remote Procedure Call (RPC). With this tool we could easily create the database connection on the server-side and make calls to the server-side from the client-side with only small fragments of code.

After understanding how to retrieve the data, the next challenge was to handle this data in the right manner. This seemed easy in theory but since the reports from OpenXdata vary in content, the data retrieved had to be handled dynamically. In other words, the code had to handle different types (text, numbers, and dates) and quantity of data. Along with this the data had to correspond to the coordinates it was given. One of the issues concerning this is that the user can name the coordinate fields in the form to whatever they want. This was resolved by looking for the most obvious field names like ‘Longitude’, ‘Latitude’, ‘Lat’, and ‘Lon’ and so on. If the users have named the coordinate fields in the form for something that the application cannot understand, the users get prompted to write in the name of the those fields from the form in order to show the markers. After the markers have been placed, it was necessary to generate a popup from the marker with its corresponding data. This went rather well to complete, but because of the lack of documentation of the GWT-OpenLayers library, it was a bit more challenging than necessary.

Around tenth of May the prototype with the marker functionality was finished and ready to be tested. Surely there was still a lot of features that should be included, like statistical data, more accurate search methods, and better indication of last registered patient and so on, but this had to be disregarded in order to be able to finish the thesis in time.

As can be clearly seen, a lot of time was in vain owing to bad choices early in the process. However, much erudition was taken from this process and fortunately it ended rather well.

3.3.3 Sample codes of important classes

We will now have a look at the most central classes and its code that make this application work. We will mainly focus on code from the marker functionality, the connection, and the entry point where we load all of the widgets.

3.3.3.1 Entry point

In this section of the entry point we add the different panels and add them to the root panel. The north panel contains the tool bar, the west panel contains the maps and markers, and the center panel contains the map.

```
// Adding the different panels to the view port
panel.add(contentPanelNorth.getContentPanelNorth(),
      contentPanelNorth.getNorthData());
panel.add(contentPanelWest.getContentPanelWest(),
      contentPanelWest.getWestData());
panel.add(contentPanelCenter.getContentPanelCenter(),
      contentPanelCenter.getCenterData());

// Adding the map widget to the center panel
contentPanelCenter.getContentPanelCenter().add(mapWidgetGIS.getMapWidget();

// Adding tool bar and menu bar to the northern content panel of border
layout
contentPanelNorth.getContentPanelNorth().add(toolBarNorth.getToolBar());

// Adding tool bar to content panel west
contentPanelWest.getContentPanelWest().add(layerPanelWest.
      getLayerContentsPanel());
contentPanelWest.getContentPanelWest().add(MarkerPanelWest.
      getMarkerContentPanel());
contentPanelWest.getContentPanelWest().add(mapStylesPanelWest.
      getLayerContentsPanel());

// Adds the view port to the root panel
RootPanel.get().add(panel);
```

3.3.3.2 MAP WIDGET

In the code example below one can see the sources for the different map types.

```
public static final String Metacarta = "http://labs.metacarta.com/wms/vmap0";
public static final String GeoServerWMS = "http://localhost:8080/geoserver/wms";
public static final String OpenStreetMap =
      "http://data1.geoposer.com:443/server/services/request.php?jname=/wms";
public static final String BingSatellite =
      "http://data1.geoposer.com:443/server/services/request.php?jname=/wms";
public static final String BingHybrid =
      "http://data1.geoposer.com:443/server/services/request.php?jname=/wms";
public static final String BlueMarble = "http://labs.metacarta.com/wms-c/Basic.py";
```

This is the code for creating the map options and creating the map widget.

```
/**
 * This method defines the map options
 */
private void setMapOptions() {
	mapOptions = new MapOptions();
	mapOptions.setControls(new JObjectArray(new JSObject[] {}));
	mapOptions.setNumZoomLevels(16);
	mapOptions.setProjection("EPSG:4326");
}

/**
 * This method creates the map widget
 */
private void createMapWidget() {
	mapWidget = new MapWidget("100%", "100%", mapOptions);
	map = mapWidget.getMap();
}
```

The code below adds the layers to the map and also adds different control options to the map.

```
/**
 * This method adds the layers to the map widget
 */
private void addLayersToMap() {
	map.addLayers(new Layer[] {wmsLayer});
}

/**
 * This method adds the controls to the map
 */
private void addControlsToMap() {
	map.addControl(new PanZoomBar());
	map.addControl(new MousePosition());
	map.addControl(new MouseToolbar());

// This sets the zoom level to 2 as a default and the position to center
	LonLat center = new LonLat(0, 0);
	map.setCenter(center, 3);
}
```

3.3.3.3 Connection service

In this section of the code we get a brief overview of how we make the connection to the database and how we retrieve the data.

Here one can see how we retrieve the reports from the database. These reports contain the data from the forms created in OpenXdata.

```
/**
  * This method establishes a connection and runs the SQL query for
  * getting the reports
  */
public ReportData[] getReportData() {
      // This array list stores the data retrieved from the SQL query
      ArrayList<ReportData> reportValues = new ArrayList<ReportData>();
      Properties oxdProperties = new Properties();

      try {
            // Loads the OpenXdata properties
            oxdProperties.load(new
                  FileInputStream("OPENXDATA_SETTINGS.properties"));

            // Gets user name, password and database name from OpenXdata
            properties
            username = oxdProperties.getProperty("hibernate.connection.username");
            password = oxdProperties.getProperty("hibernate.connection.password");

            // Creates the database connection
            Class.forName(driver);
            connection = DriverManager.getConnection(
                  databaseUrl + databaseName, username, password);
                  statement = connection.createStatement();

            // This is the report SQL query
            String sql = "SELECT report_id, form_definition_version_id, name,
                  description, query_sql FROM report";
            resultSet = statement.executeQuery(sql);

            // This is a while-loop which runs to the database and retrieves all
            // the rows of the given columns
            while (resultSet.next()) {
                  reportId = resultSet.getInt("report_id");
                  formId = resultSet.getInt("form_definition_version_id");
                  name = resultSet.getString("name");
                  description = resultSet.getString("description");
                  query = resultSet.getString("query_sql");

                  reportValues.add(new ReportData(reportId, formId, name,
                                    description, query));
            }

            resultSet.close();
            statement.close();
            connection.close();

            } catch (Exception e) {
                  MessageBox.alert("Error",
                              "Something went wrong! Error as follows: " + e,
                              null);
            }

            // Returns the values to the array
            return (ReportData[]) reportValues.toArray(new ReportData[reportValues
}
```

This next method in the connection service retrieves the marker data, that is, it gets the data from the report sequentially and adds it to different types of arrays depending on the data type.

```
/**
 * This method establishes the connection and runs the SQL query for getting
 * the data in the reports
 */
public MarkerData[] getMarkerData() {
    // This array list stores the data retrieved from the SQL query
    ArrayList<MarkerData> values = new ArrayList<MarkerData>();
    try {
        // Creates the database connection
        Class.forName(driver);
        connection = DriverManager.getConnection(
                databaseUrl + databaseName, username, password);
        statement = connection.createStatement();

        // This is the SQL query
        resultSet = statement.executeQuery(reportQuery);

        // Gets the meta data from the result set
        ResultSetMetaData rsMetaData = resultSet.getMetaData();
        // Gets the number of columns in the table
        int columnCount = rsMetaData.getColumnCount();

        while (resultSet.next()) {
            for (int i=1; i <= columnCount; i++) {
                // Gets the column names in the table
                String columnName = rsMetaData.getColumnName(i);
                // Gets the column types in the table
                String columnType = rsMetaData.getColumnTypeName(i);
                // Handles all of the values as objects
                Object value = resultSet.getObject(i);

                // Checks what type the object is and converts it accordingly
                // At the time being it supports INTEGER, VARCHAR, TEXT,
                DOUBLE and DATE.
                if (value != null) {
                    if (columnType.equals("INTEGER")) {
                        String valueData = value.toString();
                        setIntData(Integer.parseInt(valueData));
                    } else if (columnType.equals("VARCHAR")) {
                        String valueData = value.toString();
                        setStringData(valueData);
                    } else if (columnType.equals("TEXT")) {
                        String valueData = value.toString();
                        setStringData(valueData);
                    } else if (columnType.equals("DOUBLE")) {
                        String valueData = value.toString();
                        setDoubleData(Double.parseDouble(valueData));
                    } else if (columnType.equals("DATE")) {
                        String valueData = value.toString();
                        setDateData(sdf.parse(valueData));
                    }
                }
```

```java
                // Sets a 'null' value if the object doesn't contain any data
                if (value == null) {
                    valueData = "";
                } else {
                    valueData = value.toString();
                }

                values.add(new MarkerData(valueData, columnName, columnType,
                    columnCount));
            }
        }

        resultSet.close();
        statement.close();
        connection.close();

    } catch (Exception e) {
        MessageBox.alert("Error",
                    "Something went wrong! Error as follows: " + e, null);
    }

    return (MarkerData[]) values.toArray(new MarkerData[values.size()]);
}
```

3.3.3.4 Get marker data

In this class we generate the radio buttons from the reports retrieved from OpenXdata. By clicking one of these radio buttons we generate markers from the data in one of those reports. We will only be viewing fragments of the code since this is a rather comprehensive class.

Below one can see the code from inside a for-loop. In this for-loop we retrieve the report data from arrays and generate radio buttons which we add to the content panel.

```
reportId = reportSource.getIntCell(i,0);
formId = reportSource.getIntCell(i,1);
reportName = reportSource.getStringCell(i,2);
reportDesc = reportSource.getStringCell(i,3);
reportQuery = reportSource.getStringCell(i,4);

// Creates a radio button and adds the name and tool tip
Radio layer = new Radio();
layer.setBoxLabel(reportName);
layer.setToolTip("Report Id: " + reportId + " Form Id: " + formId +
                        " Report description: " + reportDesc);

// Adds the radio button to a hash map
radioButtonMap.put(reportName, layer);

// Adds the query to a hash map
queryMap.put(i, reportQuery);

// Declares a new radio button based on the values in the radio button hash
map
final Radio radioButton = radioButtonMap.get(reportName);

// Adds the radio button to a radio button group
radioGroup.add(radioButton);

// Declares a variable with the given query
final String query = queryMap.get(i);

// Adds the radio buttons to the marker panel in the west panel
if (MarkerPanelWest.getMarkerContentPanel().getItemCount() <=
    reportSource.getLength()+10000) {
            MarkerPanelWest.getMarkerContentPanel().add(radioButton);
}
```

The next step is to add a listener to each of the radio buttons as they get generated. This is a very comprehensive code snippet, so instead of adding it to this document, it is probably better to explain it rather than showing it. First off, when we click a radio button we use the query in the report to get the data. When the query is executed it retrieves the data from that given report and we get a dataset in return. We then add the different values to different arrays depending on the data value. The next step is to figure out which fields from the form that are designated for the coordinates. If these fields are found they are added to a longitude and latitude hash map. If the coordinate values are not found, the user gets prompted to define these field names depending on the form. After this done, we generate the markers for each registered patient. This is done by iterating through hash maps with the coordinate data. Lastly we create a listener to the marker, when the marker gets clicked it checks its position and adds the corresponding data depending on the position.

4. EVALUATION

This chapter is divided into four parts. The first part, goal, is the goal of the evaluation. The second part, evaluation design, relates to how the evaluation was designed, that is, how the questionnaire was set up. The third part, evaluation, copes with the purpose of the evaluation along with elements such as which subjects participated in the evaluation and the findings. The fourth and last part, discussion of findings, is the section where we look at the data from the evaluation and try to understand the findings and make it into qualitative information.

4.1 GOAL

The goal with this evaluation is to clarify whether this GIS application has the potential to function as a support system for the end user. This is done by adding sample data in OpenXdata to display in the GIS application. When the evaluators then evaluate the system, they are able to imagine how the system works in a real world context. Along with the possibilities to try out the existing functionality the evaluators are given the option to visualize other potential functionalities that may improve the usability and usefulness of the application. It is important to emphasize that the goal with this evaluation is not to determine whether this application is useful or not for the OpenXdata end-user, the goal is to determine whether the end-user can apply this application to support their work in collecting data by visualizing the collected data.

4.2 EVALUATION DESIGN

When evaluating this prototype, it was decided to perform two types of evaluations, one where the focus were on the usability experience and the other focused on the functionality of the application. Each of the evaluations had a set of task for the evaluator to complete before answering the questionnaire. These tasks were as follows:

1. Your first task is to get familiarized with the user interface:
    a. Look at the options in the top menu. Do you understand their functions?
    b. Can you imagine what functions are for the 'GeoServer' menu and the 'Settings' menu?

    c. Try changing the map to 'OpenStreetMap'. Then zoom in to a desired place on the map using the map controls.

2. Your second task is to view the data registered in OpenXdata as markers on the map:
    a. Try to find the section in the application where you can view the data from OpenXdata.
    b. Select the first data set retrieved from OpenXdata.
    c. Zoom in on a patient and view the information about that given patient.
    d. Zoom back out and then try to search for a patient by its ID (e.g. 1205782145).
3. You last task is to define the coordinate fields of the form from OpenXdata:
    a. When a user makes a form in OpenXdata, he or she has to define two fields in the form for the coordinates (longitude and latitude). The next task is to define the coordinate fields from OpenXdata if the GIS application can't understand which fields that are designated for longitude and latitude in the form. Do this by changing the data set retrieved from OpenXdata until you encounter a dataset that the application can't place on the map.
    b. After you have found such a dataset, try to define the values for the coordinate fields so that the application can understand what it has to look for in order to view the data on the map.

Both of the evaluations had the same tasks for the user.

The usability questionnaire consisted of two parts. The first part was the System Usability Scale (SUS) questionnaire, detailed information about SUS can be viewed in section *4.1.1 System Usability Scale (SUS)*. The second part was a semi-structured questionnaire with questions of their impressions and possible ideas for future work.

The usability questionnaire included the following questions:

Part I (SUS):

1. I think that I would like to use this system frequently
2. I found the system unnecessarily complex
3. I thought the system was easy to use
4. I think that I would need the support of a technical person to be able to use this system

5. I found the various functions in this system were well integrated
6. I thought there was too much inconsistency in this system
7. I would imagine that most people would learn to use this system very quickly
8. I found the system very cumbersome to use
9. I felt very confident using the system
10. I needed to learn a lot of things before I could get going with this system

Part II contained these questions:

1. How was your first impression of the user interface, did you find anything difficult and/or was there something you didn't understand?
2. Was there something you missed from the user interface that should be implemented?
3. What did you think of the marker functionality from a user interface perspective?
4. How do you think a system like this will work as a support system?
5. Other comments?

The functionality questionnaire consisted of only one part. This questionnaire only consisted of a few semi-structured questions mostly about their impressions of the application.

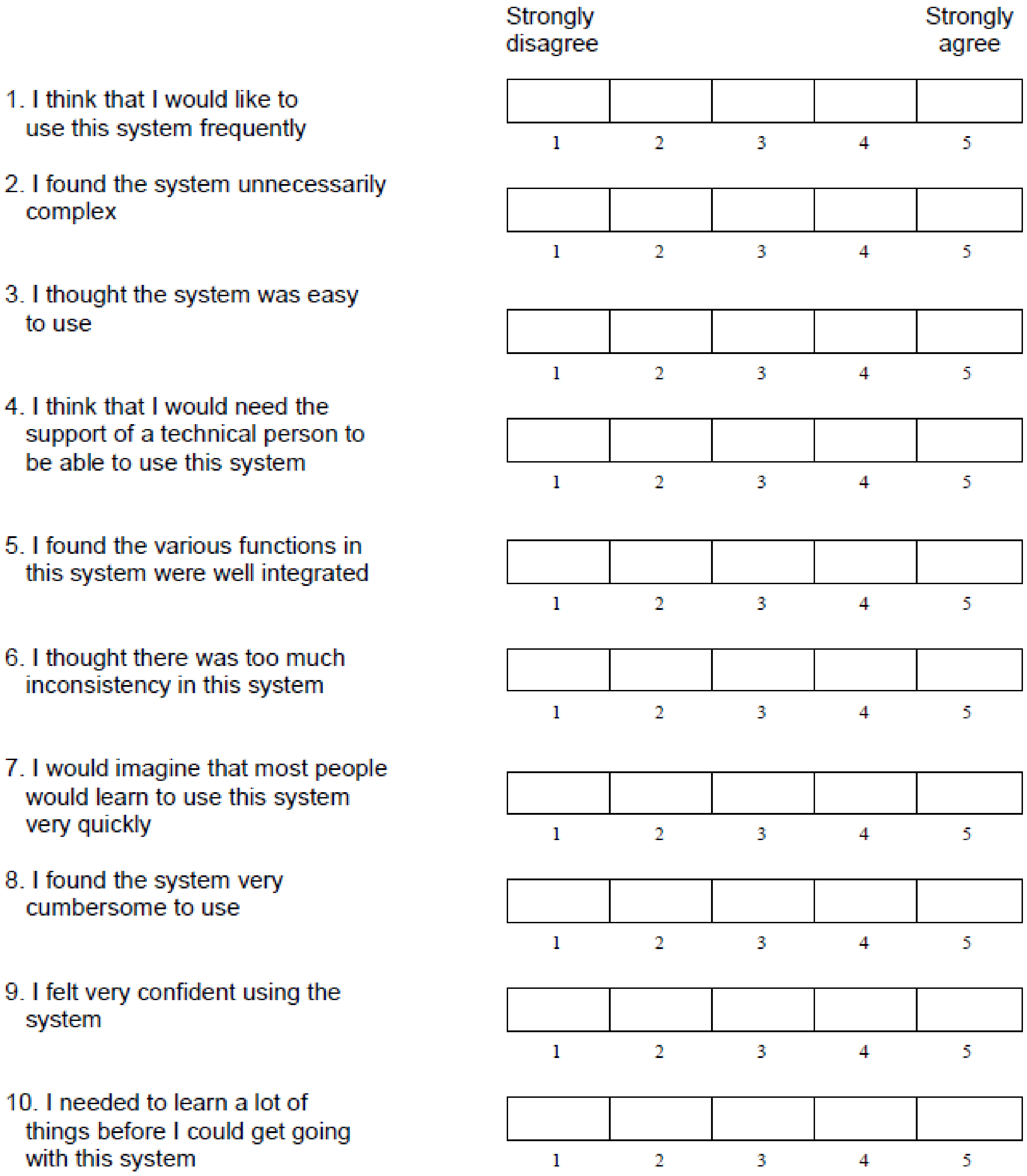
The functionality questionnaire included the following questions:

1. How was your first impression of the functionality, did you find anything difficult and/or was there something you didn't understand?
2. Was there any functionality you missed that should be implemented?
3. What did you think of the marker functionality?
4. Do you think you could use such a system in your work, please explain?
5. How do you think a system like this will work as a support system for this given purpose?
6. Other comments?

4.2.1 System Usability Scale

The System Usability Scale (SUS) is a result of an experiment trying to generalize the questionnaire for all types of usability tests. *The System Usability Scale (SUS) is a simple, ten-item scale giving a global view of subjective assessments of usability* (Brooke, 1986). SUS is a questionnaire based on scale from “Strongly disagree” to “Strongly agree”. The target evaluator then answer some questions based on this scale.

In Figure 10 one can see the SUS questionnaire:

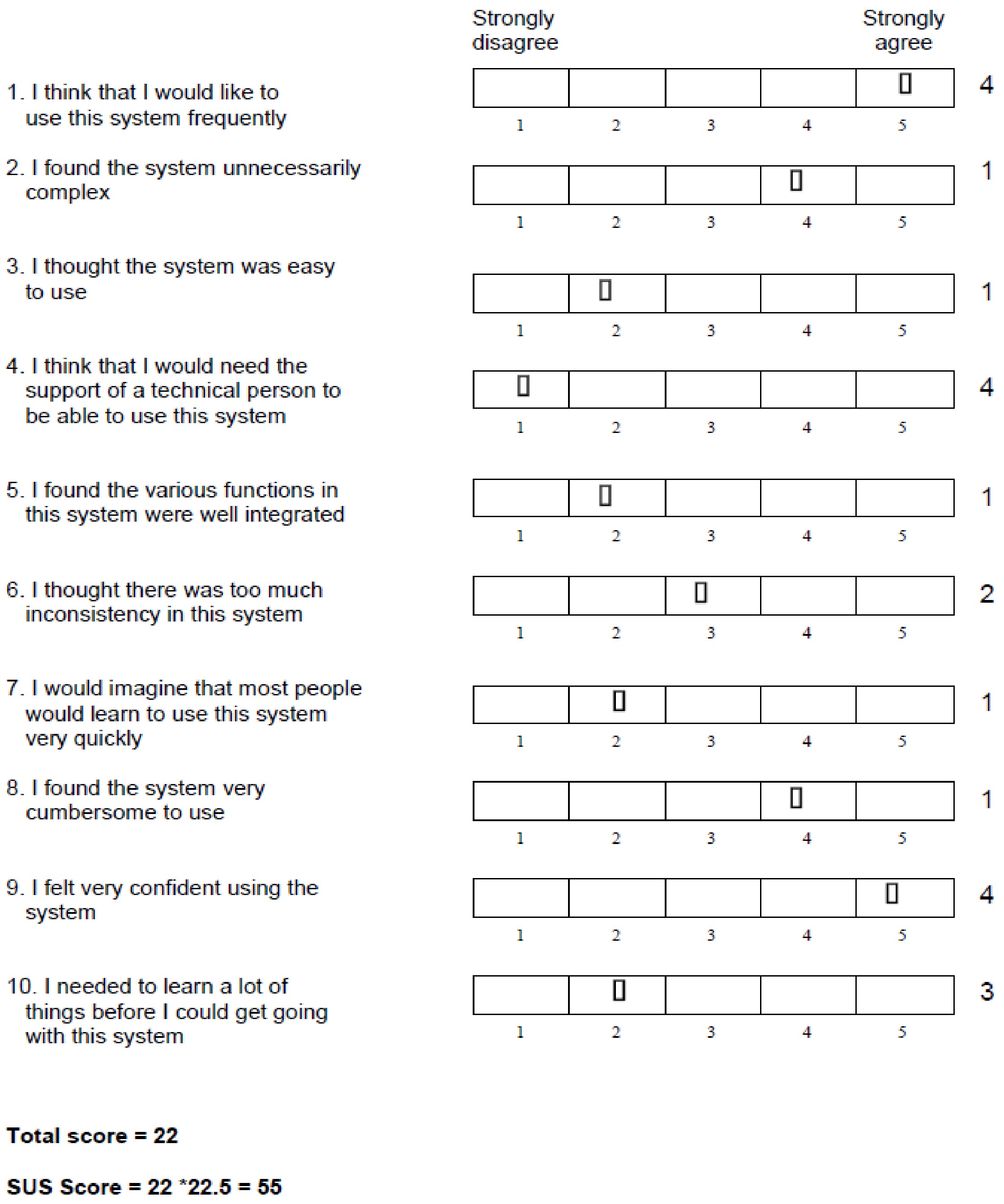


	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently	□ 1	□ 2	□ 3	□ 4	□ 5
2. I found the system unnecessarily complex	□ 1	□ 2	□ 3	□ 4	□ 5
3. I thought the system was easy to use	□ 1	□ 2	□ 3	□ 4	□ 5
4. I think that I would need the support of a technical person to be able to use this system	□ 1	□ 2	□ 3	□ 4	□ 5
5. I found the various functions in this system were well integrated	□ 1	□ 2	□ 3	□ 4	□ 5
6. I thought there was too much inconsistency in this system	□ 1	□ 2	□ 3	□ 4	□ 5
7. I would imagine that most people would learn to use this system very quickly	□ 1	□ 2	□ 3	□ 4	□ 5
8. I found the system very cumbersome to use	□ 1	□ 2	□ 3	□ 4	□ 5
9. I felt very confident using the system	□ 1	□ 2	□ 3	□ 4	□ 5
10. I needed to learn a lot of things before I could get going with this system	□ 1	□ 2	□ 3	□ 4	□ 5

FIGURE 10 THE SYSTEM USABILITY SCALE QUESTIONNAIRE

When interpreting the data we can score the data based on a range from 0 to 100, each question has points from 0 to 4. We will then get a score from each question telling us how the application performed according to the questions. *To calculate the SUS score, first sum the score contributions from each item. Each item's score contribution will range from 0 to 4. For items 1, 3, 5, 7, and 9 the score contribution is the scale position minus 1. For items 2, 4, 6, 8 and 10, the contribution is 5 minus the scale position. Multiply the sum of the scores by 2.5 to obtain the overall value of SU* (Brooke, 1986).

In Figure 11 one can see an example of how to sum up the score:



	Strongly disagree				Strongly agree	
	1	2	3	4	5	
1. I think that I would like to use this system frequently					▯	4
2. I found the system unnecessarily complex				▯		1
3. I thought the system was easy to use		▯				1
4. I think that I would need the support of a technical person to be able to use this system	▯					4
5. I found the various functions in this system were well integrated		▯				1
6. I thought there was too much inconsistency in this system			▯			2
7. I would imagine that most people would learn to use this system very quickly		▯				1
8. I found the system very cumbersome to use				▯		1
9. I felt very confident using the system					▯	4
10. I needed to learn a lot of things before I could get going with this system		▯				3

Total score = 22

SUS Score = 22 *22.5 = 55

FIGURE 11 SAMPLE OF HOW TO SCORE THE SYSTEM USABILITY SCALE

4.3 Evaluation

In this part of the chapter we look at what the evaluation consisted of. The first section gives a description of the subjects participating in the evaluation. Next we talk about the procedure for collecting the data and thereafter we do a data analysis of the data collected. Lastly we present the findings based on the data.

4.3.1 SUBJECT

When testing the GIS application two types of evaluators were included, five usability evaluators and five functionality evaluators. The first group, the usability evaluators, was a group of fellow students which did not have any or little experience with OpenXdata and had taken courses which deal with usability testing. The second group, the functionality evaluators, was a group of people working with the OpenXdata project or using it in their daily work.

The reason for including usability evaluators is to determine if the applications interface is sufficient enough based on an experts view. We may call these evaluators “experts” since they have experience with usability evaluations from previous courses they have attended when retrieving their degree in bachelor or master in Information Science at the University of Bergen.

The intention with including the functionality evaluators is to have a group that may represent the end-users of this application. The best scenario would off course be to actually test this application on real end-users; however, this was difficult considering that the OpenXdata application has yet to be fully taken advantage off by any organizations.

4.3.2 DATA COLLECTION PROCEDURE

When evaluating the GIS application the evaluators were given a brief introduction to what OpenXdata is and its functions. Along with this the evaluators were given an explanation of the functionality of the GIS application along with its intentions. They were told that the GIS application is intended to work as a data projector as well as a support system. Since the usability evaluators did not have any relations to OpenXdata, they were told to try to place themselves in a position where a system like this would be useful to them. The evaluators were then given a set of tasks to complete as earlier mentioned. While the evaluators tested the application, signed sat as an observer in the background to register their choices for the different tasks they were given. If the evaluators were not able to figure out how to complete some of the tasks, they were given tips on how to achieve this. After the testing of the system, the evaluators were set to answer some questions about their thoughts of the application.

4.3.3 DATA ANALYSIS

In this part of the chapter we will have a look at the data that we have collected and try to get an understanding of how the application performed. We will first off view the usability evaluation and then concentrate about the functionality perspective of the evaluation.

4.3.3.1 USABILITY ANALYSIS

The first pointer in getting an understanding of how well the user interface is developed is by comparing how well it scored on the SUS questionnaire. The application scored as follows; the first evaluator gave it 82.5, the second 37.5, the third 35, the fourth 87.5 and the fifth and last gave it 65.5. If we calculate these results to an average we get 61. The results from each question can be viewed in Table 2, "SD" refers to "Strongly disagree" and "SA" refers to "Strongly Agree".

TABLE 2 SYSTEM USABILITY SCALE SCORE

	System Usability Scale score	**SD**				**SA**
		1	**2**	**3**	**4**	**5**
1	I think that I would like to use this system frequently			2	3	
2	I found the system unnecessarily complex	1	1	2	1	
3	I thought the system was easy to use	1	1		2	1
4	I think that I would need the support of a technical person to be able to use this system	2	1		1	1
5	I found the various functions in this system were well integrated		1	2	1	1
6	I thought there was too much inconsistency in this system	3		2		
7	I would imagine that most people would learn to use this system very quickly		1	1	1	2
8	I found the system very cumbersome to use	2	1	2		
9	I felt very confident using the system	1	1	2	1	
10	I needed to learn a lot of things before I could get going with this system	1	1		2	1

As can be seen in Table 2, the evaluators were mostly neutral or agreed in some degree that this would be a useful application for its purpose based on question 1. Two of the evaluators placed them self as neutral and the remaining three agreed to some degree that they would use this application frequently.

Based on the questions about the complexity of the application, the answers were much more spanned. One of the evaluators strongly disagreed that the system was unnecessarily complex.

One disagreed to some degree that it was unnecessarily complex. Two placed them self as neutral and one of the evaluators agreed to some degree that it was unnecessarily complex.

One of the evaluators strongly agreed that the application was easy to use, two agreed to some degree. One of the evaluators disagreed to some degree and one strongly disagreed.

Based on the technicality of the application, one of the evaluators strongly agreed that they would need support from a technical person in order to use the application. One agreed to some degree, while one disagreed to some degree and the remaining two strongly disagreed.

Most of the evaluators found the functionality well or rather well integrated in the application. One of the evaluators strongly agreed on this, one to some degree, two placed them self as neutral and the last evaluator disagreed to some degree.

Three of the evaluators strongly disagreed on the question concerning the application to be too inconsistent while the remaining two place them self as neutral.

Two of the evaluators strongly agreed that people will quickly learn the application, one agreed to some degree, one was neutral and the last evaluator disagreed to some degree.

Most of the evaluators did not find the application to cumbersome to use. Two of the evaluators strongly disagreed on this, one disagreed to some degree, while the remaining two placed them self as neutral to this question.

However, none of the evaluators felt completely confident while using the application. One of the evaluators strongly disagreed on this, one disagreed to some degree, two were neutral and the last one agreed to some degree.

The last question about the need to learn stuff in order to use the application was rather spanned. One evaluator strongly disagreed on this, one agreed to some degree, two agreed to some degree, while the last one strongly agreed.

In the last part of the interview the evaluators were set to answer some questions where they were able to write down their thoughts about the application. The questions were as earlier mentioned in the thesis.

Based on question 1, all of the evaluators had some difficulties when using the application, one of the evaluators stated that:

"The user interface isn't completely intuitive but is fairly easy to use with an introduction." [Evaluator 1]

On the same question, another evaluator stated the following:

"It was hard to find the "View Data"-menu. I didn't understand the layers I could add from the GeoServer menu." [Evaluator 4]

The second question concerning features missed from the interface, three of the evaluators wrote the following:

"Some features and functions could be automated in the same way they are in comparable software. One example is auto zoom when selecting a patient." [Evaluator 1]

"Google Street View! Hard to keep track of all reports when it becomes more than 20? Search or categorize them maybe?" [Evaluator 4]

"Zoom to selected dataset." [Evaluator 3]

On question three four of the evaluators wrote the following:

"Following other map application standards." [Evaluator 2]

"Markers are good for visualizing data." [Evaluator 5]

"It's straight forward and represents what it is supposed to. The question is how it copes when the data set is scaled up." [Evaluator 1]

"Good. Easy to use. Similar to other map services." [Evaluator 4]

On maybe the most important question about how this application would work as a support system, the evaluators wrote the following:

"Given enough training it could work well." [Evaluator 3]

"Will support organizing of data, and is practical in terms of showing where the different locations are." [Evaluator 2]

"I don't know." [Evaluator 5]

"It's a good start, but could use more tools in the future to handle the data markers, in groups for instance." [Evaluator 1]

"Good." [Evaluator 4]

On the last question only two of the evaluators had comments extending the previous questions:

"Pop-up windows when choosing patients should close when another patient is selected." [Evaluator 2]

"Latitude and longitude should be standard." [Evaluator 5]

We will be having a closer look at these comments and what they might mean in section *4.2.5 Findings* and in *4.3 Discussion of findings*.

4.3.3.2 Functionality analysis

We will now view the results from the interviews with the subjects having a closer relation to OpenXdata. In this interview the subjects were set to complete the same tasks as in the usability interview. A difference however, is that the SUS questionnaire was not included in this interview. Rather than having the focus on the usability experience, this interview has its intentions to discover or indicate if this application has the potential to function as a support system for the OpenXdata users. We will now have a look at what the evaluators answered on the questions and later on discuss these answers in section *4.2.5 Findings* and *4.3 Discussion of findings*.

The questions are mostly the same as in the usability interview, however, there is one additional question which is: "Do you think you could use such a system in your work, please explain?"

On the first question, the evaluators stated the following:

“The instructions were fairly clean and the user interface is straightforward. This means exploring the functionality was easy. The functions themselves are impressive given that the software is still in an early stage of development.” [Evaluator 1]

“View data was hidden away, would expect to find it in the File-menu. “Save” and “Save as…” should have been beneath each other. The left menu should possibly been duplicated and function as a roll down menu. Should maybe the database data been viewable at all times?” [Evaluator 2]

“Some of the menus can be better named and placed. E.g. “View Data” above “Map providers” “Settings” could have the “GeoServer” as a sub item. “Database” could have been also sub to “Settings” [Evaluator 3]

“Most of the functionalities were kept simple and easy to understand. The application was pretty easy to work with.” [Evaluator 4]

Concerning missing functionality, the evaluators wrote the following:

“The options listed in the menu could all be functional. Doing this would be a lot for now.” [Evaluator 1]

“Click and choose on variables for longitude and latitude.” [Evaluator 2]

“After search, the found item should be marked and the info box showed. Google maps as a map provider. View at many markers close to each other is difficult. Could need some auto zoom or “explode”/or some grouped markers.” [Evaluator 3]

“There could be many functionalities that could be implemented e.g. choropleths, but the aim with which this application was created, I think the functionalities are sufficient.” [Evaluator 4]

On the third question, the evaluators stated:

“This is the most important feature of the application, it lets you see where patients are and what information is associated with each one.” [Evaluator 1]

“A menu for choosing which variables are shown would be nice.” [Evaluator 2]

“Works fine, gives correct info about items.” [Evaluator 3]

“This is a nice functionality. A user can easily identify with the marker that some information about some patient or a person is given. The click-and-view part is good and easy to work with.” [Evaluator 4]

On the fourth question about if they could use this application in their work, the evaluators wrote:

“Yes, since the organization I am associated with carries out several epidemiological studies. This system would quickly let us see where the patients are and what the latest data about them are.” [Evaluator 1]

“Yes, would be useful for monitoring, but would need a filter-function e.g. to show “today’s patients”. Follow-up of health personnel or data collectors would be easy.” [Evaluator 2]

“Yes, for monitoring origin of data or tracking work progress.” [Evaluator 3]

“We currently don’t have any such requirement, but this application could be very useful in a field setting, where it could be used for tracking field workers or getting information about patients in remote areas.” [Evaluator 4]

“Of course, it is very useful system, where we can get lot of information on the map.” [Evaluator 5]

On the fifth question the evaluators stated the following (note that one of the evaluators did not answer this question):

“I think that as the system currently stands, it will work well to display patient/health center locations along with basic data about them. This is quite helpful since the data are already being collected automatically and the data needs to go into a database/spreadsheet for analysis anyway. Having some or much of it automatically available for viewing in this manner is helpful.” [Evaluator 1]

“It will enable me to see e.g. if a fieldworker has collected all his/her information at one spot or where it was supposed to have been collected.” [Evaluator 3]

"It could be used for real time tracking of different patients in a remote area. So in the pop-ups, we could have a graph about some patient data e.g. sugar level. And doctor can continuously monitor his patients and see their sugar levels. As soon as it shows some aberrant behavior, doctor can call the patient and give some advice." [Evaluator 4]

On the last question, the evaluators wrote (note that three of the evaluators did not answer this question):

"Local storage of maps would be a plus." [Evaluator 2]

"Web + mapping is a very important tool in epidemiology, and it has been used to find sources of diseases and outbreaks for centuries after John Snow discovered the source of cholera in London long time ago." [Evaluator 3]

4.3.4 FINDINGS

This section presents the findings from the evaluation. We will first off present the findings from the usability evaluation and afterwards look at the findings from the functionality evaluation.

4.3.4.1 USABILITY FINDINGS

The initial plan when creating the user interface for this application was to make it as similar to existing map application solution (e.g. Google Earth) as possible. Along with this the plan was to make it as minimalistic and easy as possible in order to not mislead the users when using the application.

The usability evaluators gave an average score of 61 on a scale from 0 – 100. If we say that 50 is an application of mediocrity, we may say that this application place itself as "something good". Based on the SUS questionnaire, the results were extremely scattered ranging from 35 – 87.5.

According to the questionnaire and the observation of the evaluators, the most misleading element of the application seemed to be a misplaced menu for viewing the data from OpenXdata. Each of the evaluators looked for this option in the menu at the top of the application and none seemed to be able to locate it at the first glance (it is placed at the bottom in the left column). Along with this some of the menu names did not give any meaning to the evaluators, e.g. the “GeoServer”-menu and the “Configure Layer”-option in the “Settings”-menu.

When testing the different maps, some of the evaluators commented that they would have preferred to see a more detailed map in order to better locate where the patients were registered. One of the evaluators even mentioned that Google Street View[18] would be a great addition to this application.

Most of the evaluators also complained about the general feel of the usability elements of the application. E.g. zooming directly to a dataset when selected or a better search function where the searched patient get marked if found.

Lastly, some of the usability evaluators were concerned with how to cope with many markers located at the same point.

4.3.4.2 FUNCTIONALITY FINDINGS

When testing the application on the evaluators having a relation to OpenXdata, they were presented with the same questions concerning usability as the usability evaluators were. These evaluators also reacted on the misplaced option for viewing data from OpenXdata. Also, some of the menu options did not give immediate meaning to what its function was, that is, the same menu items that were confusing for the usability evaluators.

Two of the functionality evaluators suggested having a drop down list or a selectable list for defining the coordinates if they were not found initially. Along with this, most of the evaluators wanted a more functional search feature whereas the searched patient gets marked or the pop-up shown.

Concerning the map options, Google Maps was sought for along with the ability to locally store maps in order to not be dependent on having an internet connection.

One of the maybe most important feature requests was the ability to filter the data in order to retrieve only the wanted information about the patient along with the ability to cope with several markers in one location. One of the evaluators suggested to have a “explode”-function

[18] Google Street View is a service in Google Maps where the user is able to get a panoramic view of a street at ground level.

whereas the user click a cluster of patients and then all the patients gets branched outwards in order to get a better view of which patients are registered in that given area. Another suggestion was to create a list of the patients in that area where the user could select the desired patient.

When asked about what the evaluators thought of this application as a support system, all of the evaluators found reasons for using this in their work. However, additional features were requested in order to make this a fully functional and usable application as mentioned above. Some of the usage elements mentioned was to monitor the origin of data and work progress, where the patients have been registered and check if a fieldworker has collected his/her data in one spot or where it was supposed to be collected.

4.4 Discussion of findings

In this section we will look at the findings and discuss the issues surrounding them. By discussing these topics we will hopefully have an outcome resulting in a better understanding of which functions that needs improvement and if this application is useful as a support system for this given purpose.

We will first off look at the improvement possibilities of the user interface based on the feedback from the evaluators.

One of most criticized usability features of this application was the placement of the function for viewing the data from OpenXdata. Besides this, the functions for the GeoServer and the settings seemed to not give a clear indication of its purpose to the user. A third issue was the ability to cope with large quantum of data placed in the same location on the map.

As most of evaluators tried to locate the function for viewing the data in the top menu, it is probably a solution to move or add this option in the “File”-menu or add a new menu item for this. Along with this it is a possibility to expand the “View Data”-panel and by this hide the “Map provider”-panel on the application startup. This may lead to a better visualization of the actual options to the user.

Concerning the naming of the menu options, we need to give a better indication of what the different options do. A solution to this may be to rename the options to names which the user can relate to without the need of understanding technical terms. This could be achieved by renaming the “GeoServer”-menu to “Local map layers” or something in that category. Along with this we could add tooltips which give a brief description of the functionality. As a last resort we would be able to create a help menu where we could give the user the necessary introduction to understand the different features.

Touching the issue of coping with several markers at one location, several suggestions were made by the evaluators. One of the maybe most recognizable solutions to this may be to include the “explode”-function earlier mentioned. Mainly because the users who have previously used Google Earth would probably expect this since this functionality exists in Google Earth. However, to those users that are not familiar with this possibility may not be able to take full advantage of this or understand its functions when used. A solution to this might be to give the user some sort of feedback when using this feature. A feedback of this kind could be to have a descriptive text in a non-interactive window somewhere in the application which should not affect users who do not need it.

Looking at the functionality feedback, it is clearly many things that need improvement in order to make this a fully functional support application. The main issues based on the questionnaire seemed to be the ability to filter the retrieved data sets and better indication of various statues on the markers. Also the ability to use a map locally stored on a computer in order to not be dependent on an internet connection. In order to make all this happen, many new features need to be implemented in the application. A solution to the filter function might be to retrieve all the data from OpenXdata and add a function in the GIS application where the user can select which fields they want to show before the markers are generated. As to the status indications of each marker, this could be done by changing the icon color of e.g. the last registered patient and by this give a clearer indication. Another possibility is to use different types of icons for the different types of markers. Lastly, regarding the issue to locally store a map on the computer is something that is already being handled by developers which have been included recently to handle exactly this challenge. It is here that the “GeoServer”-menu comes into the picture. The idea is to include GeoServer alongside the GIS application so that the user can add whichever layer they want to in GeoServer and then import this layer into the

GIS application. In time it may also be a possibility to import layers into GeoServer from the GIS application which will ease the process for the user.

Based on the comments from the evaluators regarding if this applications has the potential to function as a support system, nearly all of the evaluators had a positive response. Sure, there are several improvement possibilities that may be needed for this application to fully live up to its potential. Even though, this application shows the possibilities and gives the users a clear indication of what they may achieve with such an application. The intentions as mentioned in the introduction were to create an application that could prove that it was needed and I think that have been achieved. An example of this can be based on one of the comments from the evaluators:

"Web + mapping is a very important tool in epidemiology, and it has been used to find sources of diseases and outbreaks for centuries after John Snow discovered the source of cholera in London long time ago." [Evaluator 3]

5. CONCLUSION AND FUTURE WORK

When starting the work on this thesis the research question was to some degree diffuse. As the thesis progressed, it was clear that this had to be more definite according to what we were trying to achieve. However, this did not affect the work of the thesis; the goal was still the same. The research question we ended up with is as follows: "*Can a Geographical Information System function as a support system for location based data collections?"*

The intention with this thesis is to clarify if a GIS can function as a support system for location based data collections. This was done by creating a GIS application that handled collected data from OpenXdata and placed them as markers on a map. The intention was not to create fully working software that could be directly put to use, but instead to function as a proof-of-concept prototype to the user. The idea was to give the user a pointer in how this could work and thereby give the user the opportunity to visualize other usage areas and functionalities. However, this initial plan was to incorporate more functionalities than this. E.g. a more specific functionality to track the field workers, but since time became an issue, this had to be discarded from the end result.

By testing this prototype on in total ten evaluators, the result became rather apparent. Five of these evaluators tested the user interface and the other five focused on the functionality. Concerning the user interface, some of the features confused the user. Besides this, the usability evaluators had some concerns about the detail of the map and the prototypes ability to cope with several markers on the map. Nevertheless, the prototype scored an average of 61 ranging from 0 – 100, which can be seen as a fairly good result considering this being an early prototype. The functionality evaluators were more or less happy with the prototype. The plan to let the evaluator visualize how this could be in the future seemed to work as planned. Most of the evaluators seemed to be able to imagine different functionalities that could be implemented in the future when testing the prototype. When asked about how they think this application can work as a support system, nearly all of the functionality evaluators had a positive response to this and saw the potential in incorporating this application with OpenXdata. However, comments were made that indicated that much work still remains on this prototype in order to fully live up to its potential. This was mostly concerning the functionalities of the prototype. A lot more features need to be implemented, that is, features such as local storage of maps, a filter function, better search function, more detailed maps, and so on. As the prototype stands at the moment, it is more or less general in the sense that it can be used not only to display patient data, but location based data in general.

We can by this summarize that using GIS to create a support application for e.g. clinical trials is absolutely helpful. There is clearly a need to investigate this further by actually testing it in a real environment with more features than the current one. Nevertheless, this thesis suggests that there is a need for a system like this in location based data collecting based on both the results from the evaluation and the literature review.

5.1 Reflections

We will now look retrospectively back on the work done in this thesis. We will focus on both the positive parts and the negative parts.

5.1.1 The design and development process

Looking back, there have been several wrong turns during the design and development phase of this thesis. The decisions made during the development phase lead to much futile time. Much of this could probably have been avoided if we in the beginning had decided to use the right tools and libraries. But this is easier said than done, many unforeseen issues surfaced as the thesis progressed which we had to allow for as they surfaced. As a result the time to write the theory part of the thesis was rather short. Besides this, if the development had started out using the current tools and libraries, we would probably have more and better implemented functionality than the current prototype has.

Even though these mistakes were made, the end result was satisfying considering the short time it was developed in. The intention of the prototype seemed to be fulfilled and the evaluators seemed to be able to understand its purpose bearing in mind the evaluation result.

5.1.2 The method for evaluation

The evaluation went rather well, as previously mentioned, the prototype seemed to live up to its intentions and the evaluators seemed to be able to imagine how this would work as a support system. But, the ideal situation would be to first do an evaluation and afterwards improve the application based on the feedback to then evaluate the application once more. In addition, the number of evaluators should ideally have been larger. Also, to test this application in a real life scenario would be a great addition to actually see if the system works as intended.

The evaluation itself was rather plain using simple methods for collecting the data. One might say that this should have been done more thoroughly, nevertheless, the results from the evaluation was satisfying and concluding to the research question.

5.2 FUTURE WORK

There is a lot of additional functionality needed in order to make this GIS application a fully working support system. As it is now, it only shows the possibilities to work as a support system for the users of OpenXdata. The current functionality for showing patients as a marker on the map with the ability to view the data in a pop-up is only one of many needed functionalities.

As mentioned by many of the evaluators, it is needed functionality such as filtering of data, local storage of maps, more detailed maps, and so on.

The ability filter what data one would like to view is probably an important feature. This could be done by creating a menu for the available data and the users have thereby the option to select which fields from the form one would like to view.

Local storage of maps was actually a considered feature when the development started. And this is something that other developers have started to work in lately. The plan is to use a local server for map storage which is called GeoServer. With this server one can store a local map and with a new functionality implementation in the GIS application the user will be able to apply these maps.

Another missing feature was the availability for detailed maps. This could be solved by importing such as Google Maps, but this have to be up to the user since it includes a fee if used for commercial purposes. In addition, one could implement a better version of OpenStreetMap than the current one which is not the official version.

The ability to view statistical data of the collected data would also be a great addition to this application. This could for instance be done by adding an option in the pop-up that opens a new window where one could see statistical information about the patient, e.g. weight gain or loss over time and so on.

At the time being, the GIS application only supports the ability to view information about patients. An addition to this could be to add support for other entities such as buildings, roads, areas and so on.

And last but not least, the user interface should be improved to give more sense to the user. That is, to change menu names that is hard to understand and move the “View Data” menu to a more obvious place in the application. Along with this, the application should be able to cope with several markers placed on the same spot. And the function for defining the coordinates could include a dropdown-menu for selecting the fields from the form that are defined to be longitude and latitude.

If all of these features were to be included, this GIS application would probably be a usable application that could be implemented into OpenXdata.

REFERENCES

Brooke, J. (1986). Retrieved May 04, 2010, from http://hell.meiert.org/core/pdf/sus.pdf

Das, A. (2003, April). Knowledge and Productivity in Technical Support Work. *Management Science* , p. 1.

Fazal, S. (2008). GIS Basics. *New Age International* , p. 6.

Geographic Information Systems. (2007, Febuary 22). Retrieved June 02, 2010, from http://egsc.usgs.gov/isb/pubs/gis_poster/

Kumar, R. (2005). *Research methodology: A step-by-step guide for beginners.* London: SAGE Publictations Ltd.

McLeod, K. S. (2000). Our sense of Snow: the myth of John Snow in medical geography. *Social Science & Medicine* .

Medeiros, C. B., & Pires, F. (1994, March). Databases for GIS. *SIGMOD RECORD* , p. 1.

Neteler, M., & Mitasova, H. (2008). *Open Source GIS.* New York: Springer.

Opensource.org. (2007). *Open Source Initiative*. Retrieved April 9, 2009, from http://opensource.org/

Preece, J., Rogers, Y., & Sharp, H. (2002). *Interaction design: beyond human-computer interaction.* John Wiley & Sons, Inc.

Snow, J. (1849). *On the Mode of Communication of Cholera.* London: Mr. Churchill's Publications.

Stockus, A., Bouju, A., Bertrand, F., & Boursier, P. (1999, November). Integrating GPS Data within Embedded Internet GIS. *ACM* , p. 1.

Studentportal.uib.no. (2009). Retrieved April 15, 2009, from https://studentportal.uib.no/dotlrn/classes/det-samfunnsvitenskapelige-fakultet/infomevi300/infomevi300-2009v/file-storage/download/54867961/InfoMedia_Master_Projects_2009.ppt

Wikipedia. (2009, May 21). *Wikipedia*. Retrieved May 26, 2010, from http://en.wikipedia.org/wiki/Vaccine_trial

APPENDIX A. USABILITY EVALUATOR RESULTS

Tasks

1. Your first task is to get familiarized with the user interface:

   a. Look at the options in the top menu. Do you understand their functions?

   b. Can you imagine what functions are for the ‘GeoServer’ menu and the ‘Settings’ menu?

   c. Try changing the map to ‘OpenStreetMap’. Then zoom in to a desired place on the map using the map controls.

2. Your second task is to view the data registered in OpenXdata as markers on the map:

   a. Try to find the section in the application where you can view the data from OpenXdata.

   b. Select the first data set retrieved from OpenXdata.

   c. Zoom in on a patient and view the information about that given patient.

   d. Zoom back out and then try to search for a patient by its ID (e.g. 1205782145).

3. Your last task is to define the coordinate fields of the form from OpenXdata:

   a. When a user makes a form in OpenXdata, he or she has to define two fields in the form for the coordinates (longitude and latitude). The next task is to define the coordinate fields from OpenXdata if the GIS application can’t understand which fields that are designated for longitude and latitude in the form. Do this by changing the data set retrieved from OpenXdata until you encounter a dataset that the application can’t place on the map.

   b. After you have found such a dataset, try to define the values for the coordinate fields so that the application can understand what it has to look for in order to view the data on the map.

Evaluator 1

Part 1

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently			X		
	1	2	3	4	5
2. I found the system unnecessarily complex	X				
	1	2	3	4	5
3. I thought the system was easy to use				X	
	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	X				
	1	2	3	4	5
5. I found the various functions in this system were well integrated					X
	1	2	3	4	5
6. I thought there was too much inconsistency in this system	X				
	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly					X
	1	2	3	4	5
8. I found the system very cumbersome to use	X				
	1	2	3	4	5
9. I felt very confident using the system			X		
	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system		X			
	1	2	3	4	5

Part 2

1. How was your first impression of the user interface, did you find anything difficult and/or was there something you didn't understand?

The user interface isn't completely intuitive but is fairly easy to use with an introduction.

2. Was there something you missed from the user interface that should be implemented?

Some features and functions could be automated in the same way they are in comparable software! One Example is auto zoom when selecting a patient.

3. What did you think of the marker functionality from a user interface perspective?

It's straight forward and represents what it is supposed to. The question is how it copes when the data set is scaled up.

4. How do you think a system like this will work as a support system?

It's a good start, but could use more tools in the future to handle the data markers in groups for instance.

5. Other comments?

Evaluator 2

Part 1

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently			X		
	1	2	3	4	5
2. I found the system unnecessarily complex				X	
	1	2	3	4	5
3. I thought the system was easy to use		X			
	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system					X
	1	2	3	4	5
5. I found the various functions in this system were well integrated		X			
	1	2	3	4	5
6. I thought there was too much inconsistency in this system	X				
	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly			X		
	1	2	3	4	5
8. I found the system very cumbersome to use			X		
	1	2	3	4	5
9. I felt very confident using the system		X			
	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system				X	
	1	2	3	4	5

Part 2

1. How was your first impression of the user interface, did you find anything difficult and/or was there something you didn’t understand?

Search ID field should be place more centrally towards the other functions. Hard to imagine what GeoServer options do for an outsider.

2. Was there something you missed from the user interface that should be implemented?

3. What did you think of the marker functionality from a user interface perspective?

Following other map application standards.

4. How do you think a system like this will work as a support system?

Will support organizing of data, and is practical in terms of showing where the different locations are.

5. Other comments?

Pop-up windows when choosing patients should close when another patient is selected

Evaluator 3

Part 1

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently				X	
	1	2	3	4	5
2. I found the system unnecessarily complex			X		
	1	2	3	4	5
3. I thought the system was easy to use	X				
	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system				X	
	1	2	3	4	5
5. I found the various functions in this system were well integrated			X		
	1	2	3	4	5
6. I thought there was too much inconsistency in this system			X		
	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly		X			
	1	2	3	4	5
8. I found the system very cumbersome to use			X		
	1	2	3	4	5
9. I felt very confident using the system	X				
	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system				X	
	1	2	3	4	5

Part 2

1. How was your first impression of the user interface, did you find anything difficult and/or was there something you didn’t understand?

Not always a intuitive user interface, hard to find the “View Data”-tab.

2. Was there something you missed from the user interface that should be implemented?

Zoom to selected dataset.

3. What did you think of the marker functionality from a user interface perspective?

See number 2.

4. How do you think a system like this will work as a support system?

Given enough training it could work well.

5. Other comments?

Evaluator 4

Part 1

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently				X	
	1	2	3	4	5
2. I found the system unnecessarily complex	X				
	1	2	3	4	5
3. I thought the system was easy to use					X
	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	X				
	1	2	3	4	5
5. I found the various functions in this system were well integrated				X	
	1	2	3	4	5
6. I thought there was too much inconsistency in this system			X		
	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly					X
	1	2	3	4	5
8. I found the system very cumbersome to use	X				
	1	2	3	4	5
9. I felt very confident using the system				X	
	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system	X				
	1	2	3	4	5

Part 2

1. How was your first impression of the user interface, did you find anything difficult and/or was there something you didn’t understand?

It was hard to find the View Data-menu. I didn’t understand the Layers I could add from the GeoServer menu.

2. Was there something you missed from the user interface that should be implemented?

Google Street View! Hard to keep track of all reports when it becomes more than 20? Search or categorize them maybe?

3. What did you think of the marker functionality from a user interface perspective?

Good. Easy to use. Similar to other map services.

4. How do you think a system like this will work as a support system?

Good.

5. Other comments?

No.

Evaluator 5

Part 1

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently				X	
	1	2	3	4	5
2. I found the system unnecessarily complex			X		
	1	2	3	4	5
3. I thought the system was easy to use				X	
	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system		X			
	1	2	3	4	5
5. I found the various functions in this system were well integrated			X		
	1	2	3	4	5
6. I thought there was too much inconsistency in this system	X				
	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly				X	
	1	2	3	4	5
8. I found the system very cumbersome to use		X			
	1	2	3	4	5
9. I felt very confident using the system			X		
	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system					X
	1	2	3	4	5

Part 2

1. How was your first impression of the user interface, did you find anything difficult and/or was there something you didn't understand?

It was hard to understand all the different features. I don't understand the layers…

2. Was there something you missed from the user interface that should be implemented?

Don't know. I don't need this program. It is hard to see the future features without a personal need.

3. What did you think of the marker functionality from a user interface perspective?

Markers are good for visualizing data, I like.

4. How do you think a system like this will work as a support system?

I don't know.

5. Other comments?

Latitude and longitude should be standard!

APPENDIX B. FUNCTIONALITY EVALUATOR RESULTS

Tasks

1. Your first task is to get familiarized with the user interface:

a. Look at the options in the top menu. Do you understand their functions?

b. Can you imagine what functions are for the 'GeoServer' menu and the 'Settings' menu?

c. Try changing the map to 'OpenStreetMap'. Then zoom in to a desired place on the map using the map controls.

2. Your second task is to view the data registered in OpenXdata as markers on the map:

a. Try to find the section in the application where you can view the data from OpenXdata.

b. Select the first data set retrieved from OpenXdata.

c. Zoom in on a patient and view the information about that given patient.

d. Zoom back out and then try to search for a patient by its ID (e.g. 1205782145).

3. Your last task is to define the coordinate fields of the form from OpenXdata:

a. When a user makes a form in OpenXdata, he or she has to define two fields in the form for the coordinates (longitude and latitude). The next task is to define the coordinate fields from OpenXdata if the GIS application can't understand which fields that are designated for longitude and latitude in the form. Do this by changing the data set retrieved from OpenXdata until you encounter a dataset that the application can't place on the map.

b. After you have found such a dataset, try to define the values for the coordinate fields so that the application can understand what it has to look for in order to view the data on the map.

Evaluator 1

1. How was your first impression of the functionality, did you find anything difficult and/or was there something you didn’t understand?

The instructions were fairly clean and the user interface is straight-forward. This means
exploring the functionality was easy. The functions themselves are impressive given that
the software is still in an early stage of development.

2. Was there any functionality you missed that should be implemented?

The options listed in the menu could all be functional, doing this would be a lot for now.

3. What did you think of the marker functionality?

This is the most important feature of the application. It lets you see where patients are and
what information is associated with each one.

4. Do you think you could use such a system in your work, please explain?

Yes, since the organization I am associated with carries out several epidemiological
Studies. This system would quickly let us see where the patients are and what the latest
data about them are.

5. How do you think a system like this will work as a support system for this given purpose?

I think that as the system currently stands, it will work well to display patient/health center
Locations along with basic data about them. This is quite helpful since the data are already
Being collected automatically and the data needs to go into a database/spreadsheet for
analysis anyway. Having some or much of it automatically available for viewing in this

manner is helpful

6. Other comments?

Evaluator 2

1. How was your first impression of the functionality, did you find anything difficult and/or was there something you didn’t understand?

View data was hidden away, would expect to find it in the File-menu. “Save” and “Save as” should have been next to each other.

2. Was there any functionality you missed that should be implemented?

Click and choose on variables for longitude and latitude.

3. What did you think of the marker functionality?

A menu for choosing which variables that are shown would have been nice.

4. Do you think you could use such a system in your work, please explain?

Yes, would have been useful for monitoring, but would soon needed a filter function. E.g. to show “today’s patients”. Follow up of health personnel or data collectors would be easy.

5. How do you think a system like this will work as a support system for this given purpose?

Yes, look above.

6. Other comments?

Local storage of maps would soon be a plus.

Evaluator 3

1. How was your first impression of the functionality, did you find anything difficult and/or was there something you didn’t understand?

Some of the menus can be better named and placed. E.g. View Data above Map Providers. Settings could have the GeoServer as a sub-item. Database could have been also sub to settings.

2. Was there any functionality you missed that should be implemented?

After search, the found item should be marked and the info box showed.

Google maps as a map provider.

View of many markers close to each other is difficult. Could need some auto-zoom or “explode”/or some grouped markers.

3. What did you think of the marker functionality?

Works fine, gives correct info about items.

4. Do you think you could use such a system in your work, please explain?

Yes, for monitoring origin of data or tracking work progress.

5. How do you think a system like this will work as a support system for this given purpose?

It will enable me to see e.g. if a fieldworker has collected all his/her information at one spot or where it was supposed to have been collected.

6. Other comments?

Web + mapping is a very important tool in epidemiology, and it has been used to find
sources of diseases and outbreaks for centuries after John Snow discovered the sources of
cholera in London long time ago.

Evaluator 4

1. How was your first impression of the functionality, did you find anything difficult and/or was there something you didn’t understand?

Most of the functionalities have were kept simple and easy to understand. The application was pretty easy to work with.

2. Was there any functionality you missed that should be implemented?

There could be many functionalities that could be implemented e.g. choropleths, but the aim with which this application was created, I think the functionalities are sufficient.

3. What did you think of the marker functionality?

This is a nice functionality. A user can easily identify with the marker that some information about some patient or a person is given. The click-and-view part is good and easy to work with.

4. Do you think you could use such a system in your work, please explain?

We currently don’t have any such requirement, but this application could be very useful in a field setting, where it could be used for tracking field workers or getting information about patients in remote areas.

5. How do you think a system like this will work as a support system for this given purpose?

It could be used for real time tracking of different patients in a remote area. So in the pop-ups, we could have a graph about some patient data e.g. sugar level. And doctor can continuously monitor his patients and see their sugar levels. As soon as it shows some

aberrant behavior, doctor can call the patient and give some advice.

6. Other comments?

Evaluator 5

1. How was your first impression of the functionality, did you find anything difficult and/or was there something you didn’t understand?

The functionality is really good and impressive, for me I think everything is clear while performing the tasks, only task 3, which I couldn't understand well. Otherwise all other functionalities are well presented.

Task 3 is difficult to understand. Try to make it easy to understand.

2. Was there any functionality you missed that should be implemented?

Metacarta is not displayed on the screen, should be fixed.

When we click on the marker to see the information, it should be closed automatically, when we click on the other marker.

3. What did you think of the marker functionality?

It is very good functionality. Especially when we have lot of data

4. Do you think you could use such a system in your work, please explain?

Of course, it is very useful system, where we can get lot of information on the map.

5. How do you think a system like this will work as a support system for this given purpose?

This type of system can be used for other type of activities and environment. For example in other health activities.

6. Other comments?
