Low-Cost Rapid Usability Testing for Health Information Systems: Is it Worth the Effort?

by

Tristin Baylis
B.Sc., University of Victoria, 2003

A Thesis Submitted in Partial Fulfillment of the
Requirement for the Degree of

Master of Science

In the School of Health Information Science

SUPERVISORY COMMITTEE

Low-Cost Rapid Usability Testing for Health Information Systems: Is it Worth the Effort?

by

Tristin Baylis
B.Sc., University of Victoria, 2003

Supervisory Committee

Dr. Andre Kushniruk, School of Health Information Science
Supervisor

Dr. Elizabeth Borycki, School of Health Information Science
Department Member

ABSTRACT

Supervisory Committee

Dr. Andre Kushniruk, School of Health Information Science
Supervisor

Dr. Elizabeth Borycki, School of Health Information Science
Department Member

Usability testing is a branch of usability engineering that focuses on analyzing and improving user interactions with computer systems. This testing technique has been used in different industries for years and has proven to be very useful in determining major issues with applications before they are released, however the use of this technique has been slow to gain widespread acceptance in testing health information systems. This study was designed to determine if a specific form of usability testing, Low-Cost Rapid Usability Testing, can be introduced as a standard part of the system development lifecycle (SDLC) for health information systems in a cost effective manner. To determine if this was possible a full cost-benefit analysis of Low-Cost Rapid Usability Testing was performed on a health information system, the BC Chronic Disease Management (CDM) Toolkit, tracking all of the costs involved in the testing process and comparing them against the possible costs that may have been incurred if this testing was not performed. It was found that by introducing this technique into the system development lifecycle to allow for earlier detection of errors in a health information system it is possible to achieve an estimated 36.5% to 78.5% cost saving compared to the impact of errors going undetected and causing a technology-induced error. Overall it was found that Low-Cost Rapid Usability Testing can be implemented in conjunction with other testing techniques in a cost effective manner to develop health information systems, and computer systems in general, which will have a lower incidence of technology-induced errors.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

Thanks to my employer CGI for allowing me to take the time from my work schedule to complete this project and for the funding assistance without which I would not have been able to be part of this program.

Special thanks to Rosemary Gray and Linda Low at British Columbia Ministry of Health Services for allowing me to perform and helping to recruit participants to be involved in this research.

Also thanks to Dr. Liz Keay who provided her expertise to help determine the possible cost of any associated medical error.

Above all, thanks Dr. Andre Kushniruk and Dr. Elizabeth Borycki at the School of Health Information Science at the University of Victoria. Without their guidance and advice I would not have been able to complete this project.

CHAPTER 1: INTRODUCTION

The introduction of information technology (IT) into any environment is meant to help resolve complex or repetitive tasks, make processes more efficient and improve workflow. The possible benefits of introducing IT into a healthcare or clinical setting have led to the wide spread adoption of health information systems across the healthcare industry with the goal of dramatically decreasing medical errors by streamlining workflow, providing features like automated alerts and reminders in patient monitoring systems, and automating error prone manual tasks such as packaging prescriptions (Bates et al., 1998; Leape & Berwick, 2005). However, the development of new health information systems is advancing faster than legislation and industry can keep up. Health information systems are being developed by companies with limited medical or clinical background and put into use without being thoroughly tested by the software provider or the end user of the application. This has led to the introduction of errors in the workflow of medical environments causing serious complications and even deaths in some cases (Leape & Berwick, 2005). Errors may result from the complex interaction between the computer system and the health professional. Errors of this type caused by the introduction of new systems are referred to as technology-induced errors (Kushniruk, Triola, Stein, Borycki & Kannry, 2004) or technology-facilitated errors (Koppel et al., 2005).

The term technology-induced error was introduced in a paper by Kushniruk et al. (2004). This study reviewed the use of a handheld prescription writing application characterizing two types of technology induced errors that were described as (1) slips (e.g. incorrect medication entry which at some point the user notices and corrects - typically resulting from unintentional events such as typos) and (2) mistakes (e.g. incorrect medication entry which is not corrected by the subject). The handheld application studied was designed to facilitate medication ordering. However, this new technology had not been introduced the

errors it caused would not have been an issue. In some cases the new errors caused by the introduction of an application could possibly outnumber and be more severe than the errors it is designed to resolve, the potential cost of these errors has been estimated to be considerable (Kushniruk et al., 2005).

A possible solution to deal with the growing issue of technology induced error comes from the field of usability engineering (Nielsen, 1993). Usability engineering refers to human-computer interaction and specifically with making human-computer interfaces that have high usability or user friendliness; this involves assessing the usability of an interface and recommending ways to improve it. Usability testing is a subset of usability engineering and has been used in many cases to identify usability problems, and to validate and improve the functionality of health information systems (Nielsen, 1993).

1.1 Usability Testing

Usability of a computer system, as defined by Preece et al. (1994), is the capacity of the system to allow users to carry out their tasks safely, effectively, efficiently and enjoyably (Preece, Roger, Sharp, Benyon, Holland & Carey, 1994). Usability testing is a branch of usability engineering that involves observing representative users of a system carrying out predefined tasks to determine usability issues with a system (Nielsen, 1993). This testing technique has been used in different industries, ranging from software development to aerospace design, and for decades has proven to be very useful in determining minor and major issues with tools or applications before they are released. In general the overall usability of a system is related to how effective, efficient and enjoyable a system is to use (Nielsen, 1993).

Traditional usability testing has often involved building very elaborate testing facilities with one way mirrors to simulate real world conditions. Although this type of testing has been found to be very effective it is also very expensive (Jeffries, Miller, Wharton & Uyeda, 1991). In a study by Jeffries et al. (1991), traditional usability testing was compared to three other methods (heuristic evaluation, application of guidelines and cognitive walkthrough). Although usability testing was found to be one of the most

effective techniques it was also found to be the most expensive costing almost seven times as much as the other techniques. Largely based on this type of finding a general impression has emerged that usability testing is too expensive, this has led to usability testing methods not becoming as widely practiced as might be expected. The cost of traditional usability testing has led researchers to look for more cost effective options. One such option proposed by Nielson (1994), called "Discount Usability Engineering", is intended to address this issue. The "Discount Usability Engineering" method (Nielsen, 1994) is based on the use of the following three techniques:

- **Use of Scenarios:** A set of scenarios are designed that test the key components of a program or interface.
- **Simplified thinking aloud:** A subject walks through an application scenario stating aloud the steps they are performing and what they are thinking while their actions and verbalizations are recorded for additional analysis.
- **Heuristic evaluation:** This involves a researcher stepping through a program or interface, and does not involve observing real subjects.

A cost-benefit analysis of usability testing and several other usability engineering methods showed a 2:1 dollar savings-to-cost ratio for relatively small development projects and a 100:1 savings-to-cost ratio for large development projects (Karat, 1997). However even with the introduction of more cost effective techniques and increasing acceptance in industry usability engineering still has not become common practice in the development of health information systems.

1.2 Usability Testing in Healthcare

Even though usability testing is not commonly used in the health sector, over the past decade it has been adapted for use in the testing of health information systems. In one study by Jaspers (2009), three usability engineering techniques (heuristic evaluation, cognitive walkthrough, and usability testing) were compared in evaluating health information systems. It was found that each of the three usability evaluation methods has its advantages and disadvantages. It was concluded a combination of different usability testing techniques can complement one another and can be used very effectively in the

testing of health information systems (Jaspers, 2009). Jaspers (2009) findings are consistent with the findings by Nielsen (1994) related to the effectiveness of usability testing in industry.

Usability testing has had multiple successes in validating health information systems ranging from the evaluation of mobile software applications to helping improve the workflow of existing systems (Kushniruk & Borycki, 2006). Innovations in implementing usability testing in health information systems have led to the development of a modified version of usability testing known as Low-Cost Rapid Usability Testing (Kushniruk & Borycki, 2006).

1.2.1 Low-Cost Rapid Usability Testing

Low-Cost Rapid Usability Testing as proposed by Kushniruk (2006) is a form of usability testing, similar to "Discount Usability Engineering" (Nielsen, 1994), where instead of setting up an expensive testing facility a portable laboratory is created. This laboratory may involve as little equipment as a laptop, a video camera and screen recording software taken into a real-world setting where testing is conducted. In addition, Low-Cost Rapid Usability Testing describes a simplified approach to analysis of video data obtained from testing sessions (Kushniruk & Patel, 2004). This overall objective of the approach is to allow usability testing to be done at a lower cost than traditional usability testing (since it does not involve an expensive laboratory set up) and allows for data to be collected in real world environments to more closely simulate how an application is used once implemented. This form of testing involves setting up a portable laboratory with a laptop with screen capture software installed and a video camera to record subject's actions as they perform a task. As the subject is performing a task they are typically asked to "think aloud" as they perform different actions, this allows the researcher to later correlate what the subject was thinking with any usability issues they may have been having. The approach also outlines a methodology for conducting analysis of the resultant video and audio recordings.

Designing a low-cost rapid usability test typically involves nine distinct phases (Kushniruk & Patel, 2004). In Phase 1 the evaluation objective is determined. In Phase 2 the study design is outlined. This involves characterizing the profile (i.e. characteristics and attributes) of the subjects and determining the number of subjects to involve. In Phase 3 representative tasks are selected that will be used to test the system being validated. Phase 4 involves creating any required questionnaires. This may involve creation of several different types of questionnaires including background, pretest or post-test questionnaires. In Phase 5 the evaluation environment is determined. This is the physical location where the subjects will carry out test scenarios while being observed. Phase 6 is the data collection phase when the selected subjects are video and audio recorded as they complete the predefined tasks. Subjects may be asked to "think aloud" or verbalize their thoughts while using the system under study. In Phase 7 qualitative and quantitative analyses of the data from the previous phase are performed involving the identification and classification of usability problems. In Phase 8 the analyzed data is interpreted. This step may involve quantifying the results. In the final phase, Phase 9, the results are fed back as specific recommendations for system improvement and re-design. Table 1.1 outlines the details of the phases involved in Low-Cost Rapid Usability Testing (Kushniruk & Patel, 2004).

Low-Cost Rapid Usability Testing Design Phases
Phase 1 – Identification of evaluation objectives
o Purpose is to address the "how", "when", "where", "who", "why" and "what" of the usability test o Information to be specified: • Purpose of testing • Problem statement • User profile • Method (test design) • Task list • Test environment and equipment

• Evaluation measures (data to be collected) • Report contents and presentation
Phase 2 – Sample selection and study design
o Must determine user profile of target population(s) – includes: • Describing the most critical skills, knowledge, demographic information and other relevant factors o Study may involve subjects (i.e. users) of differing levels of medical and computer experience and expertise: • e. g. nurses, attending physicians, residents, patients o Number of subjects: usability testing can be very informative with small number of subjects (e.g. 8-10 subjects)
Phase 3 – Selection of representative experimental tasks and contexts
o Task scenarios: • Simple written descriptions of usability tests • Scripts for simulated doctor-patient interactions
Phase 4 – Selection of background questionnaire
o Background questionnaire: • Given either before or after testing to obtain subject demographics o Pretest questionnaires: • Studies may involve tests of knowledge conducted before and after videotaping of subjects o Post-test questionnaires; • Can be used to assess learning resulting from experimental session
Phase 5 – Selection of evaluation environment
o Portable approach – can perform testing in subjects' real environment o Selection of evaluation environment depends on objective of study
Phase 6 – Data collection (recording of thought processes)
o Participants include: • The subject(s) • A test monitor, who interacts with the subject, presents the case scenarios and ensures testing proceeds properly

- Technical considerations during testing:
  - Video recording of the subject-computer interaction using one or more digital cameras
  - Capture of computer actions (screens) using video recording or screen capture

- "Thinking Aloud"
  - Usability tests may involve giving subjects tasks to complete including instructions such as:
    "Please think aloud, or verbalize your thoughts as you carry out the task"

Phase 7 - Analysis of process data

- Transforming data into recommendations may involve qualitative and quantitative analysis of the video-based data
- Coding of video and audio data to identify frequencies and categories of usability problems
- Advantages of using video recordings as data:
  - Provides a record of the "whole event"
  - Allows for data in a physical and social context to be integrated, including:
    - Audio transcriptions of Subjects' "Thinking Aloud"
    - A record of communication and patterns of Social Interaction

Phase 8 - Interpretation of findings

- Qualitative codes can be quantified;
  - Frequency of coding categories
  - Ranked by frequency, priority and ease of fixing

Phase 9 - Iterative input into design

- Results are fed back as specific recommendations for re-design
- Within the System Development Life Cycle (SDLC) several cycles of usability testing may take place
- The process of feedback must be streamlined to:

• Be efficient • Be informative • Provide useful feedback to designers

Table 1.1 Low-Cost Rapid Usability Testing Design Phase adapted from Kushniruk & Patel (2004)

It has been argued that this simplified form of usability testing allows for tests to be performed in a short time period allowing for quicker analysis of any issues (Kushniruk & Patel, 2004). However, even though usability testing has been gaining popularity in recent years it still has not gained mainstream acceptance in the health sector for evaluation and validating applications (Kushniruk & Borycki, 2006).

1.2.2 The Cost of Usability Engineering

One of the main arguments for the lack of widespread adoption of usability testing is that usability testing is considered by many project managers and stakeholders to be too expensive and time consuming (Jeffries et al., 1991; Karat et al., 1992). This is an argument that since the very early days of formal usability testing many researchers have endeavored to either prove or disprove (Jeffries et al., 1991; Karat et al., 1992).

An early study by Jeffries, Miller, Wharton & Uyeda (1991) performed a cost-benefit analysis of four different testing techniques: heuristic evaluation, use of software guidelines, cognitive walkthroughs, and traditional usability testing. Overall the study found that usability testing was effective in uncovering serious problems and it was second only to heuristic evaluation (i.e. a systematic inspection of a subject interface design for usability issues) in validating information systems. However in this study usability testing was found to be the most expensive of the four techniques that were examined. The study by Jeffries et al. (1991) led to a follow up study conducted by Karat, Campbell, & Fiegel (1992) entitled “Comparison of empirical testing and walkthrough methods in user interface evaluation”. This study found that usability testing was very effective at identifying issues in software applications, and contradicted the findings by Jeffries et al. (1991) in that it found that usability testing led to the identification of more

critical errors than heuristic evaluation. However this study did support the findings of Miller et al. (1991) that usability testing was the more expensive technique.

Another study by Karat (1993) entitled “Usability Engineering in Dollars and Cents” investigated the question of how to perform a cost-benefit analysis of usability engineering. A cost-benefit analysis is a method of analyzing projects for investment purposes and proceeds as follows (Karat, 1993):

1. Identify the financial value of expected project costs and benefits.
2. Analyze the relationship between expected costs and benefits.
3. Make the investment decision.

In this study Karat (1993) asserted that in order for usability engineering techniques to be adopted on the critical path of development in organizations software development managers had to be able to show that use of these techniques makes financial sense. To demonstrate this Karat performed a cost-benefit analysis of two usability engineering techniques, usability testing and rapid prototyping. It was found for the two sample applications that were reviewed the cost-benefit ratio of implementing usability testing was 1:2 (Cost: $20,700: Savings: $41,700) and rapid prototyping was 1:100 (Cost: $68,000: Savings: $6,800,000). Although the findings of the cost-benefits of usability engineering in this study are for two specific cases it gives a definite indication of the benefit to an organization when implementing usability engineering.

Additional studies by Karat (1994, 1997) have found that usability engineering techniques could very effectively be introduced into the system development life cycle (SDLC) allowing developers to identify application errors much earlier in the development process, leading to definite benefits. Up to 64% (Karat, 1994) of software development costs that occur during the maintenance phase of an application are due to unmet or unforeseen user requirements. It was therefore concluded that introducing usability engineering into the SDLC early on has the potential to significantly reduce the total cost of developing an application.

Despite a number of studies showing the benefits of usability engineering (Jeffries et al., 1991; Karat et al., 1992; Karat, 1993) a study by Dillon et al. (1993) entitled “A survey of usability engineering within the European IT industry” (Dillon, Sweeney & Maguire, 1993) found the adoption of usability engineering techniques was still very low. In a survey conducted as part of the study by Dillon et al. (1993) it was found that only 52% of the respondents reported that their organizations regarded usability engineering as 'very important' or 'essential' with almost 19% of organizations reporting that usability engineering techniques were either 'unimportant' or 'not essential'. This survey also found that 54% of organizations had no dedicated usability engineering staff and 74% had no testing facilities at all.

The lack of adoption of usability engineering techniques is an issue that was also identified by the pioneer of the discount usability engineering, Jacob Nielsen (Nielsen, 1993). Working to increase the adoption of usability testing in general, Nielson conducted several studies (Nielson, 1994; Nielson, 1995; Nielson, 2007) that investigated the overall cost-benefit of different testing techniques. In a study titled “Using discount usability engineering to penetrate the intimidation barrier” (Nielsen, 1994) Nielsen performed an analysis of the cost savings of implementing discount usability engineering. It was found that the main issues limiting the adoption of traditional usability engineering techniques is they are seen as being too time consuming and expensive. Nielson (1994) found one of the major costs of usability testing was the number of test subjects that have been required and their associated cost and time requirements. To help resolve this issue an investigation was completed which found that the maximum benefit for discount usability is achieved when using three to five participants (i.e. subjects) (Nielsen & Landauer, 1993) as demonstrated in Figure 1.1, which shows that the ratio of benefits to costs decreases at some point (as the number of subjects increases). In related work Virzi (1992) argued that with as few as 4 - 5 participants in a usability study up to 80% of usability problems can be determined. Although subsequent work has determined the number of participants (i.e. subjects) needed for usability testing of commercial applications often involves between 5 and 10 participants (Rybin & Chisnell, 2008).

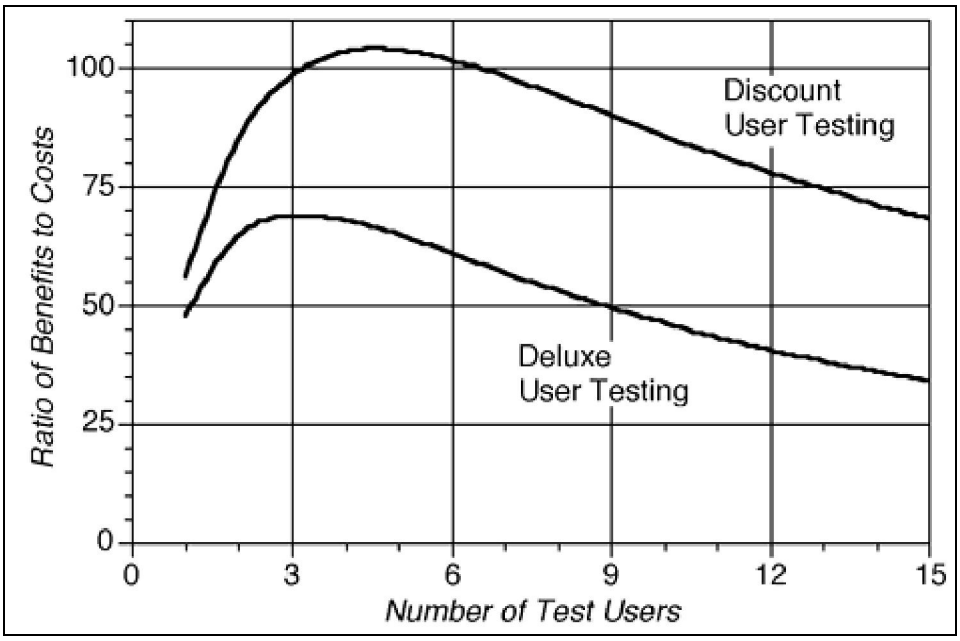


Figure 1.1 Usability Engineering Optimal Users (Nielsen & Landauer, 1993)

Reducing the overall number of subjects required in usability testing drastically reduces the time required for testing and the time required for analyzing the test results. Nielson showed that implementing discount usability engineering techniques instead of other usability testing techniques could reduce the total time required for testing by as much as 50% which on its own would lead to a significant cost savings. Nielson (1994) took this analysis one step further performing a full cost-benefit analysis of the different components of usability engineering using a sample application. A cost-benefit analysis of heuristic evaluation found a cost-benefit ratio of 1:48 (Cost: $10,500, Savings: $500,000), whereas the cost-benefit analysis of user testing found cost-benefit ratio of 1:56 (Cost: $4,650, Savings: $260,500). This indicates an average cost-benefit ratio for usability engineering of 1:52.

However, despite the findings of Nielson (1994) and Karat et al (1992,1993) many developers continued to have the impression that usability engineering activities lengthen projects, add expenses, and fail to prevent problems showing up when systems are released into production (Lund, 1997). In an attempt to overcome this perception Lund (1997) set out to indicate the benefit of usability engineering from a different perspective. Lund (1997) asserted that performing a cost-benefit analysis is too retrospective and fails

to show the value the usability engineering to stakeholders ahead of time. To overcome this obstacle Lund developed a set of key indicators for usability testing that linked all activities of usability testing to overall corporate earnings. To accomplish this Lund (1997) produced data linked to implementing usability testing that is often included in a business case and therefore allowed decision makers to make reasonable business decisions (e.g. deciding on whether to invest in usability testing or in other new product activities). This resulted in the development of a formula that directly showed the effects of implementing usability testing techniques reflected in the organization's overall stock price. Although somewhat unorthodox this approach found that the cost of usability engineering could be closely monitored, indicating an overall benefit to an organization without a significant cost impact.

In research similar to that performed by Lund (1997), Mayhew (1999) defined a set of steps to help determine the benefits of usability testing:

1. **Create a detailed usability plan:** Design a testing plan that will involve all steps of the usability testing lifecycle from start to finish. This should involve everything from interaction with subjects to analysis of the testing results.
2. **Establish analysis parameters:** Based on the characteristics of the application that is being tested, determine a specific set of parameters that should be evaluated during the usability testing process.
3. **Select relevant benefit categories:** Benefit categories are the benefits that are expected to be generated from the usability testing. Some examples are (Mayhew, 1999):
    - Decreased late design
    - Decreased user training
    - Increased user productivity
    - Decreased user errors
4. **Estimate benefits:** Determine the value of the benefits that have been received from the usability testing process. These benefits should relate to the benefit categories found in the previous step.

Following these steps using a sample application Mayhew (1999) indicated a cost-benefit ratio in the first year of a usability plan of 1:2 (Cost: $87,660, Savings: $175,104), and in the fifth year a cost-benefit ratio of 1:6 (Cost: $87,660, Savings: $558,320). These results are consistent with the earlier findings of Karat (1993) and Nielson (1994).

In a related study by Donahue (2001) it was found that in addition to the perception that usability testing is too expensive, many organizations typically see usability testing as an unneeded additional cost, if development doesn't include any formal usability activities (such as usability testing or prototyping) there won't be any usability costs. It was asserted that the assumption, that usability testing is an "additional" cost is wrong, as every software product has a user interface, whether it's usability-engineered or not. Donahue found that by correcting usability problems in a project's design phase, the development cost for fixing application errors can be reduced by 60% to 90%. This study also found that 80 percent of development costs occur during the maintenance phase, where many maintenance costs are associated with poorly defined user requirements and other problems that usability engineering can prevent. To support this view Donahue (2001) performed a cost-benefit analysis of improving the sign-on procedure of a single application. For a system used by more than 100,000 users and a usability testing cost of $68,000 a benefit of $6,800,000 was realized within the first year of the system's implementation. A cost-benefit analysis of such figures results in a cost-benefit ratio of 1:100 (Cost: $68,000, Savings: $6,800,000)

Taking a different approach to determining the benefits of usability testing, in a study by Bevan (2005) it was found that usability testing can help to reduce development time and cost by producing a product that has only relevant functionality. In many software applications only about 5% of features available to the customer are used 95% of the time. More importantly, some 70% of user-interface design features are never or rarely used. If usability tests are designed to identify the features that are used and eliminate those that are not, an immediate cost savings can be realized (Bevan, 2005).

In addition to the research that was conducted by Nielson (Nielson, 1994; Nielson, 1995; Nielson, 2007), an extensive review of the existing literature provides some evidence contrary to the perceived impression that usability testing is too expensive to be implemented in a cost effective manner (Nielson, 1994). It was found that since Nielson published his book simply entitled "Usability Engineering" (1993) there have been many studies showing that usability engineering (Nielsen, 1994; Karat, 1997; Lund, 1997; Donahue, 2001; Bevan, 2005) and specifically discount usability engineering methods involving usability inspection (involving trained analysts evaluating an interface or system, rather than studying real users by applying usability testing) can be done in a cost effective manner in main stream industry. It should be noted that many of the studies showing cost-effectiveness of usability methods have focused on usability inspection methods, which do not involve observation of users interacting with systems as is done in usability testing. However, the issue of proving the cost-effectiveness of usability testing involving video recording and detailed analysis of user interactions in complex domains remains unsettled, particularly in the area of healthcare IT.

Even though a significant amount of analysis has been completed in performing cost-benefit analyses of discount usability engineering, it appears that no similar research has been completed to validate the cost effectiveness of Low-Cost Rapid Usability Testing as proposed by Kushniruk et al. (2006) in complex domains such as health care. One publication by Kushniruk & Borycki (2006) entitled "Low-Cost Rapid Usability Engineering: Designing and Customizing Usable Healthcare Information Systems" outlined the procedure for conducting Low-Cost Rapid Usability Testing. In that paper a cost analysis was conducted for a sample application. It was found that the total usability testing cost, including the one-time cost of equipment, as well as the cost of data analysis, was $4,860. However even though this study alluded to the benefits of Low-Cost Rapid Usability Testing, a formal cost-benefit analysis has not yet been performed.

Based on the current literature it can be concluded that several cost-benefit analyses of discount usability engineering have been conducted in the IT industry. Studies have focused on methods such as heuristic evaluation and usability testing however, a

comprehensive cost-benefit analysis has not been conducted to determine if a method such as Low-Cost Rapid Usability Testing (Kushniruk & Borycki, 2006) can be carried out in a cost effective manner in the testing of complex information systems. In addition, the costs and potential benefits of applying usability testing techniques to identify and prevent technology-induced error in healthcare remains unexplored. Given recent research indicating that use of healthcare IT may inadvertently lead to a range of technology-induced errors (Koppel et al., 2005; Kushniruk et al., 2005), it will be important to determine not only if usability testing methods are cost effective in terms of lowering costs of system delivery, but also if such methods can be used to predict errors and reduce potential costs associated with occurrence of technology-induced errors.

CHAPTER 2: RESEARCH QUESTIONS

2.1 Problem

To date usability testing has not gained mainstream acceptance in the health sector for the testing of applications (Kushniruk & Borycki, 2006) even though a number of studies have demonstrated the usefulness of this technique (Karat, 1997; Kushniruk & Patel, 2004; Kushniruk & Borycki, 2006; Nielsen, 1994). Additionally even though a few of these studies have shown the cost effectiveness of usability testing in mainstream industry, no studies have been reported demonstrating this for health information systems. This leads to the following question: "Is Low-Cost Rapid Usability Testing a cost effective way of identifying both usability problems and potential safety issues in healthcare IT?"

2.2 Questions

There are several questions that this research addresses:

- Can Low-Cost Rapid Usability Testing effectively identify errors in health information systems?
- Is Low-Cost Rapid Usability Testing a cost effective way of determining usability problems and safety issues?
- Can usability testing be integrated into the system development lifecycle of a complex healthcare application in a cost effective manner?
- Are the results from Low-Cost Rapid Usability Testing useful in improving a chronic disease management application?

2.3 Rationale

Although usability testing has been used for decades in industry and to a limited degree in the development of health information systems, it has never gained widespread acceptance as the general perception in the industry is that setting up a usability

laboratory is extremely expensive (Kaner, James & Bret, 2001). However, by using a Low-Cost Rapid Usability Testing approach it may be possible to perform usability testing in a cost effective and productive manner (Kushniruk & Borycki, 2006). To determine this, in this thesis a cost-benefit analysis was performed to find if usability testing can be cost effectively integrated into the system development lifecycle of a complex healthcare application.

CHAPTER 3: RESEARCH METHODS

This chapter will outline the material, subjects, procedure and approach to the analysis applied in this thesis.

3.1. Materials

3.1.1Software Studied

In the study described in this thesis a Chronic Disease Management (CDM) Toolkit was analyzed for its usability. The CDM Toolkit is a secure web application that was developed in British Columbia in 2003 as an information management and technology decision support tool to be used to support decision making about chronic disease management. The purpose of this application was to provide an organized approach to improving the care of patients with chronic diseases. A screen shot of a primary screen from the application is shown below. The screen shot in Figure 3.1, which uses test data, shows the details for a specific patient with a chronic disease, congestive heart failure.

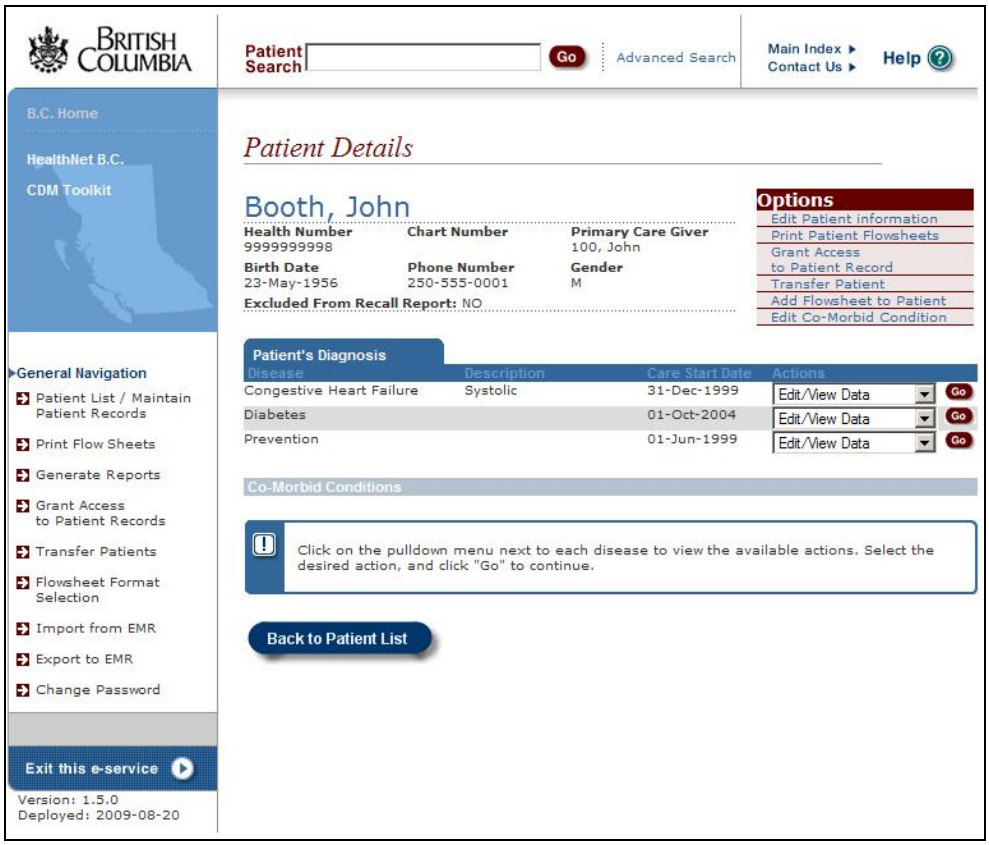


Figure 3.1 CDM Screen Shot

Some of the key benefits of this application are that it provides access to a range of helpful tools for maintaining patient records, using patient flowsheets, accessing clinical guidelines and generating clinical and administrative reports. It is designed to support physician decision making and ultimately lead to improvements in patient care.

The CDM Toolkit has a diverse group of users that all leverage the application for different purposes ranging from determining scheduling for patient tests to reporting on the state of chronic disease management for specific conditions within British Columbia. There are four main categories of users of this application. Each type of user has differing types of access (i.e. they can access different types of data in the application based on their role):

- **Physicians** use the application to aid in decision support and to track patient conditions. This involves recording patient specific data and creating reports as required (BC CDM Toolkit Documentation and Guides, 2011).
- **Nurses** use the application in much the same way as doctors but have more restricted rules on the data they can update and report on (BC CDM Toolkit Documentation and Guides, 2011), for example, nurses can only report on a limited subset of patients that have been assigned to them by a physician.
- **Medical Office Assistants** use the application to generate reports and follow up on patient care based on recommended treatment generated by the decision support component of the CDM Toolkit. This often involves using the system to determine when follow up appointments and tests may be required for a patient (BC CDM Toolkit Documentation and Guides, 2011).
- **Administrators** use the application to monitor usage and compliance as mandated by the Ministry of Health Services. Additionally with the ability to perform data mining they can generate advanced reports which allow them to provide data to analysts who work to determine if there are any trends in the data (e.g. diabetes is on the rise in a specific area) (BC CDM Toolkit Documentation and Guides, 2011).

A number of changes were made to the CDM Toolkit to enhance its functionality. The changes made to enhance the CDM Toolkit that were evaluated as part of this thesis involved an upgrade to the application database, screens, flowsheets, reports and tools, while also ensuring that no existing relevant functionality was lost from the current BC CDM Toolkit version. The Yukon Department of Health and Social Services version of the CDM Toolkit, which is functionally identical to the BC CDM Toolkit, was also upgraded during this initiative.

The CDM Toolkit is a very complex system with many different components and tools (as shown in Appendix A) that are accessed based on roles assigned to a specific user. The complete toolkit is made up of several different components that allow for data access from many different areas as shown in Figure 3.2.

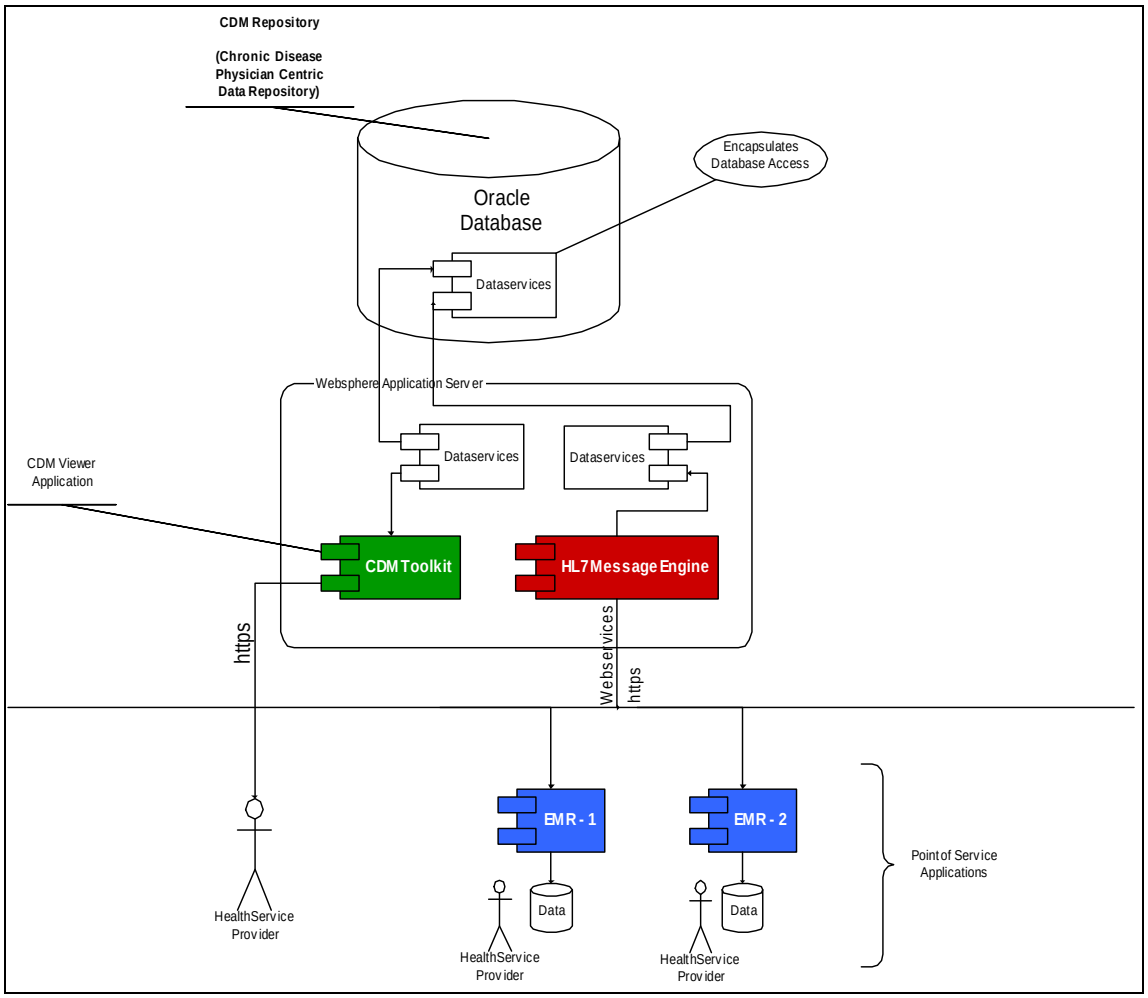


Figure 3.2 CDM Flow Chart

The bulk of the changes being made in the current release focused on the Health Service Providers interactions with the CDM toolkit via the Internet (HTTPS), therefore this is the area the usability testing performed in this study focused on.

3.1.2 Usability Testing Materials

Table 3.1 shows the list of the key materials that were required to perform the usability testing portion of this thesis.

Materials	
Material	**Description**
Laptop	The laptop was used to run the CDM Toolkit test scenarios.
Internet Access	Internet access was required as the CDM Toolkit application cannot be installed on the laptop due to its complexity. The application was accessed through the Ministry of Health Services test website.
Headset with microphone	This was used to record what the subject said as they performed the test scenarios.
Web Cam	The Web cam was linked to the laptop and used to record the subject's facial expressions.
Video Camera	The video camera was set up to record the subject's body language and physical activities as they carried out the test scenarios.
HyperCam© screen capture software	HyperCam© is a software application used to capture and record computer screens.
Room to conduct study	A private room was used to conduct the study. It contained a desk and a chair and had enough space to accommodate the researcher, equipment and the research subject.
Consent Form	The consent form was needed to obtain permission to record the subject for purposes of the usability testing (see Appendix C).
Usability Testing	This questionnaire was based on the standardized

Questionnaire	Questionnaire for User Interface Satisfaction (QUIS) designed by Shneiderman (1997) at the University of Maryland. It assesses the subject's perception of issues they encountered in the application based on carrying out the test scenarios performed (see Appendix D).
Usability Testing Evaluation Questionnaire	The purpose of this questionnaire was to determine the subject's perception of the Usability Testing process as a whole (see Appendix E).

Table 3.1 Usability Testing Materials

The testing station was set up with the web cam focused on the subject's face and the video camera filming the subject's physical actions. The general set up of the testing station is depicted in Figure 3.3 (Kushniruk & Borycki, 2006).

Figure 3.3 Usability Testing Station Setup (Kushniruk & Borycki, 2006)

3.2. Subjects

Usability studies in industry typically involve between eight to ten subjects (Nielsen, 1994). Furthermore it has been shown that the bulk of the surface level usability errors can be identified with as few as 3-5 subjects (Nielsen, 1994). A usability error is an error that occurred while designing the application that causes it to behave in a way that is not:

- **Effective:** which means the user cannot do what they want to do
- **Efficient:** which means the user cannot do what they want to do in a convenient or time saving way
- **Satisfactory:** which means the usage of the application is annoying or unacceptable to the user

To make the study in this thesis representative of real world commercial usability testing eight subjects were used (Kushniruk & Patel, 2004).

The primary subjects for this research were 8 current and representative CDM Toolkit users (i.e. 2 administrators, 2 physicians, 3 nurses and 1 medical office assistant). This allowed subjects to perform the test scenarios with minimal training and to more effectively evaluate the impact of the new application enhancements being evaluated (Kushniruk & Patel, 2004).

Different health centers across BC were approached and informed of the purpose of the study. Subjects were recruited from these centers. Subjects who were interested in participating in the study were requested to complete a written consent form for participation in the study.

The inclusion criteria used to determine if a subject could participate in the study (Nielsen, 1994; Kushniruk et al., 2006) are shown in Table 3.2.

Subject Criteria	
Criteria	**Description**
Subjects	All subjects were British Columbia Ministry of Health Services employees that currently use the CDM Toolkit. Subjects were to fall into one of the following categories of users: • physicians • nurses • medical office assistants • administrators (such as the current regional Practice Support Team)
Selection Criteria	
The following criteria were used to exclude subjects:	Familiarity with basic computer/internet use (i.e. searching info on the web, navigational tools such as a mouse, web interface) was required.
	Previous exposure to the application being studied was required.
	Must be proficient in the English language.
	Subjects could not be affiliated with the development and implementation of the application being studied.
	Subjects could not be application stakeholders.

Table 3.2 Usability testing Subject Criteria

For a subject to be included in the study it was required that he/she had experience using the CDM Toolkit and familiarity with basic computer/internet use as no training was supplied as part of this study. Subjects were also required to be proficient in the English language as the CDM Toolkit did not provide other language options. All stakeholders and individuals involved in the development of the CDM Toolkit were excluded from the study as they may have prior knowledge of changes to the application.

3.3. Procedure

There were several steps involved in each testing session. In order to obtain accurate results, the steps were performed consistently for each subject (Nielsen, 1993). The following sections outline the steps involved in completing each testing session.

3.3.1 Phase I – Usability Testing

It should be noted that all usability testing was performed using the test environment described above for the CDM Toolkit (BC MoHS CDM Toolkit Test, 2010). This environment contained no actual patient information and any changes made in this environment did in no way impact the production version of the application.

1. **Pre-Test Setup**

   Before each research subject participated in the predefined test scenarios the CDM Toolkit (BC MoHS CDM Toolkit Test, 2010) data was set to a consistent starting point. This was done by logging in as an application administrator and manually configuring the system before each session to have the user account and data at the starting point described in Table 3.3.

Pre-Test Setup	
Step	**Description**
User Configuration	Before each: • The username was set to **cdmtest107** and the password was set to **hello107.** • The user role was configured to allow the subject to run reports and create and edit flowsheets. • The user was configured to have 3 patients assigned to them, one of those patients being specifically “John Doe” (a fictitious test patient)
Data Setup	Before each test all data related to patient “John Doe” was set to a consistent starting point as outlined below. This was done by logging into the CDM Toolkit as an administrator and editing John Doe’s data through the Maintain Patients Records screen. • John Doe was assigned two chronic diseases, diabetes and asthma to allow for testing of multiple flowsheets. • Any required pre-populated data for John Doe was completed. This avoided unrelated system

	validation rules from being displayed when performing all test scenarios. • All pre-existing flowsheets for John Doe were removed.

Table 3.3 Pre-Test Setup

2. Overview

Before the start of a test each subject was informed of all the steps that would be involved in the testing process. Table 3.4 shows the items that were reviewed with the subject (see the table below and Appendix B for the complete checklist that was followed).

Overview Discussion Items	
Step	**Description**
Overview	A general overview of the study was given. The researcher first discussed the study at a high level (i.e. what usability testing is and then reviewed what the subject would be doing).
HyperCam©	It was then made clear to the subject that all of his/her actions and comments while using the CDM Toolkit would be recorded using the screen recorder HyperCam©.
Video/Web Camera	It was made clear to the subject that they would be recorded by both the web camera connected to the computer and a secondary video camera.
Use of Data	Subjects were informed that the data gathered as part of this research would be used for no other purpose.
Consent Form	Finally, subjects were informed that they would be required to sign a consent form indicating they were willing to participate in the study.

Table 3.4 Overview Discussion Items

If the subject was willing to continue with the test, they were asked to sign a consent form (the signed copy remained with the researcher, and a second copy was given to the subject). The consent form covered the following (see Appendix C for the complete consent form):

- Outlined who the research is being conducted by and the appropriate contact information.
- Detailed the purpose and importance of this research.
- Outlined how subject selection was completed.
- Reviewed all the risks and benefits associated with this research.
- Reviewed the use of the data collected for this research including anonymity, confidentiality, and dissemination of results and disposal of data.

3. Perform Test Scenarios

Each subject had received standardized training in the use of the CDM Toolkit prior to the study as all selected subjects were current users of the system. Subjects were asked to read a short clinical scenario and then to log into the CDM Toolkit to perform the task described in the scenario. The scenario consisted of the subject performing the actions described in Table 3.5 (see Appendix G for the full test scenario provided to subjects).

Test Scenario Tasks	
Step	**Description**
Login	Log into the application as "cdmtest107".
Select patient John Doe	Navigate to the "Patient List/Maintain Patient Records" screen and select the patient John Doe.
View the patient's Asthma Flowsheet	Navigate to the "Asthma Flowsheet" for patient John Doe.
Revise the Asthma Flowsheet	Revise the "Asthma Flowsheet" for the patient John Doe with data the subject feels is appropriate.
View the patient's Diabetes Flowsheet	Navigate to the "Diabetes Flowsheet" for patient John Doe.
Revise the Diabetes Flowsheet	Revise the "Diabetes Flowsheet" for patient John Doe with data the subject feels is appropriate.
Navigate to the Reports screen	Navigate to the screen for generating reports by selecting the "Generate Reports" link.
Generate Patient Education Report	Generate a Patient Education Report with the following specific criteria: • Patient: Doe, John • Condition/ Flowsheet: Asthma • Start Date: 01-Jan-2008 • End Date: 28-Feb-2010 • Disease Attributes: Spirometry - FEV1%, Spirometry - PEF%
Generate Run Charts Report	Generate a Run Charts Report with whatever criteria the subject determines is appropriate.
Exit the Application	Exit the application (Testing is complete).

Table 3.5 Test Scenario Tasks

Subjects were encouraged to "think aloud'' and verbalize their thoughts about the CDM Toolkit as they performed the tasks. Subject's verbalizations were audio and video recorded for subsequent protocol analysis (Kushniruk et al., 2005). In addition the subject's actions were recorded by HyperCam© (an application that records the subjects interactions with the system and their verbalizations and stores them in a movie file) as the subject completed the scenario using the CDM Toolkit (Kushniruk et al., 2005).

4. Post –Task Interview

Immediately after completing the scenario a brief semi-structured interview (see Appendix F) was conducted with the subject. During this time the recording equipment was left on to record the subject's responses. The interview involved reviewing the following questions, allowing the subject to elaborate on each question as they felt was appropriate.

- How did you find using the application?
- Did you have any problems using it? (if so explain each)
- Do you find that the application is usable?
- Do you find that the information provided would be useful for your practice?
- Do you have suggestions for improvement to the application?
- Do you have any other suggestions?

This interview allowed for a subsequent analysis of the usability testing component with the post-task interview to see if the subject's overall perception of the system aligned with issues they may have encountered during the test scenario (Kushniruk & Borycki, 2006).

5. Questionnaire

Once the subject had completed the scenario they were given a written questionnaire adopted from a questionnaire designed by Shneiderman (1997) to assess their

impressions of using the system (see Appendix D). The questionnaire designed by Shneiderman was selected as a basis for the questionnaire in this study as it has been tested in industry for validating the usability of a system as outline in the book "Designing the User Interface" (Shneiderman ,1997). The questionnaire involved the following closed ended questions each rated on a scale of 1 (indicating they disagree) to 5 (indicating they agree), except where otherwise specified.

- The task to be performed was clear
- The look and feel was pleasing
- The help files were useful
- Navigation was straight forward
- System information was accurate
- Instructions were understandable and accurate
- Error messages provided were understandable and accurate
- The application is effective
- Using the application was enjoyable
- Please circle the number which most appropriately reflects your impression of the system (On a scale of 1-5).
    - Terrible (1), Wonderful (5)
    - Frustrating (1), Satisfying (5)
    - Dull (1), Stimulating (5)
    - Difficult (1), Easy (5)
    - Rigid (1), Flexible (5)
- Screen layouts were helpful
- Sequence of screens
- Terminology of information on screens

In addition to the closed ended questions subjects were asked the following open ended questions which allowed them to elaborate more on their experience using the application.

- Please comment on any other application issues that you feel should be noted.
- Please make any other general comments.

3.3.2 Phase II – Usability Testing Review

As a final step once the initial round of usability testing was completed, a second questionnaire was given to the subject with the purpose of determining their opinion of the usability testing process in general. It was explained to the subject that the purpose of this questionnaire was to rate the usability testing process as a whole to help determine if it is perceived as a useful process that could be used at Ministry of Health Services in the future.

The follow up questionnaire (Appendix E) contains questions that focus on determining how a subject perceives the usability testing process and if they had suggestions about improving the testing process. The questionnaire involved the following closed ended questions, each rated on a scale of 1 (indicating they disagreed) to 5 (indicating they agreed):

- The purpose of the usability testing was clear.
- Usability testing was a useful process.
- This process helped to identify potential issues.
- Thinking aloud changed how I interacted with the system.
- Being videotaped changed how I interacted with the system.
- I would recommend usability testing be implemented as a standard step in the testing process.

In addition to the closed ended questions subjects were asked the following open ended questions which allowed them to elaborate more on their overall opinion of usability testing:

- Please comment on any improvements you would make to the usability testing process.
- Please make any other general comments.

3.4. Analysis

3.4.1 Phase I – Usability Testing

The first step in analyzing usability testing results involves synchronizing the two video recordings (web camera and video camera) with the HyperCam© screen recording of the application. Once this was completed, the researcher stepped through each subjects' testing session transcribing all audio and notable items from the video and HyperCam© recording into a usability test log for each testing session. A notable item from the recording was defined as an action or gesture a subject may have made but did not verbalize; for example, if the subject slammed the keyboard in frustration this would be noted by the researcher in the transcript (Kushniruk & Borycki, 2006).

The usability testing logs for each subject were then reviewed by two researchers to determine if any errors may have been encountered by subjects. To determine problem errors in the system as a whole the errors were coded based on a modified version of the scheme described by Kushniruk et al. (2005). This placed each noted error into one of the following categories (see Table 3.6).

Coding Categories	
Interface	
Data entry	Procedure
Display visibility	Speed
Navigation	Attention
Locating	
Content	
Data	Help
Defaults	

Table 3.6 Coding Categories

To help determine what errors must be fixed, the severity of the usability problems was also rated based on frequency, impact and persistence (Nielsen, 1995). These errors were then rated based on the rating scale outlined by Nielsen (1995) (see Table 3.7).

Severity Ratings	
Rating	**Description**
0	Not a problem
1	Cosmetic problem only: need not be fixed unless extra time is available on project
2	Minor usability problem: fixing this should be given low priority
3	Major usability problem: important to fix, so should be given high priority
4	Usability Catastrophe: imperative to fix this before product can be released

Table 3.7 Severity Ratings

In cases where a discrepancy in either the coding category or severity rating for an identified error occurred between the two researchers the discrepancy was discussed and a consensus was obtained in terms of assigning an appropriate coding category or severity rating.

During the usability testing many of the subjects encountered the same or very similar errors as each other or the same error multiple times while performing the test scenarios. The next step in analyzing the errors was to reduce the list of errors that were encountered to a distinct list of errors, eliminating duplicate errors. The distinct list of errors contained usability errors as well other types of errors (e.g. programming errors) that were discovered by subjects during testing.

Following this, the testing logs and the CDM Toolkit Evaluation questionnaire data (see Appendix D) were reviewed to determine if there were any errors that may not have

already been noted and to determine the subject's general opinion of the CDM Toolkit. The answers to all 13 closed ended questions were evaluated to determine the mean value and standard deviation for each question and the average mean value and standard deviation for all of the questions. The responses for the 2 open ended questions were reviewed to determine any common themes in the subject's responses.

In addition to the usability testing of the CDM Toolkit an analysis of data collected from administering the usability testing questionnaire (see Appendix E) was undertaken to determine how the subject's perceived the overall usefulness of the usability testing. The analysis involved tabulating the results of the closed ended rating questions in the questionnaire to determine the overall rating of the process as a whole. The answers to all 6 closed ended questions were evaluated to determine the mean value and standard deviation for each question and the mean value and standard deviation for all the questions. The responses to the 2 open ended questions were reviewed to determine if any common themes emerged from the subject's responses.

3.4.2 Phase II – Cost-Benefit Analysis

Although usability testing has been used for decades in industry and to limited degree in the development of health information systems, to the best of the researcher's knowledge a cost-benefit analysis of usability testing has never been undertaken (similar to the one performed by Karat (1997)) to determine the cost effectiveness of usability testing in health information systems. However, researchers have argued that Low-Cost Rapid Usability Testing should be possible to perform in a cost effective and highly productive manner (Kushniruk & Borycki, 2006). To test this assumption, a cost-benefit analysis was performed to determine the cost effectiveness of usability testing when integrated into the system development lifecycle (SDLC) of a complex healthcare information system. To determine if usability testing is cost effective all costs related to usability testing from the start to the finish of the research were tracked. This involved tracking everything from the cost of the required materials to the cost associated with each participant's time spent in performing usability testing.

The cost-benefit analysis of the usability testing was performed by comparing the cost of performing Low-Cost Rapid Usability Testing on the CDM Toolkit directly to the cost that would be estimated to be incurred if the errors that were found in the testing were found later in the SDLC. This analysis was completed calculating the possible costs in 3 different ways:

- Direct Measurable Savings
- Cost of Errors related to when resolved in the SDLC
- Cost of Medical Error

1) Direct Measurable Savings

The most basic calculation (ignoring evidence about when an error is introduced (Kaner, James & Bret, 2001)) for cost savings associated with catching application errors was first found. This was done by finding the number of application errors and the costs associated with having to do multiple migrations of the application to a production environment to resolve the errors (see Table 3.8).

Basic Usability Savings Calculation
usbCost = Usability Testing Cost = Material + Travel + Research Subject + Usability Analysis numErrs = Number of errors found in usability testing mgrCst = Cost of Migrating application to Production fixCost = $\frac{\text{Best Case Scenario + Worst Case Scenario}}{\text{Number of Scenarios}}$ = $\frac{\text{(Migrating All Fixes at Once) + (Migrating Fixes One at a time)}}{\text{Number of Scenarios}}$ = $\frac{\text{(mgrCst) + (mgrCst * numErrs)}}{2}$ **Total Savings** = fixCost - usbCost

Table 3.8 Basic Usability Savings Calculation

The cost of migrating an application was found based on the hourly rate of the resource required for performing the migration multiplied by the number of hours required to perform the application migration. For the purposes of this study the time and cost were based on the average hourly rate ($85/hour) of the company currently maintaining the CDM Toolkit and the time estimated for performing a migration of this application from Development to Production (38 hours, details in Table 3.9). Therefore it was assumed that the cost of migrating the application to production (mgrCst in the table above) was $3,230.00.

Migration Time	
Task	**Estimate (hours)**
Test Migration Preparation	4
Test Migration	6
Validate application in Test Environment	8
Production Migration Preparation	4
Production Migration	8
Validate application in Production Environment	8
Total	38

Table 3.9 Migration Time

2) Cost of Errors Related to When Resolved in the SDLC

The Direct Measurable Savings cost-benefit analysis does not take into account when an error was identified in the SDLC. However, when an error occurs in the SDLC can directly affect the cost of resolving that error (McConnell, 2004)). It has been found that the later in the development of an application an error is found the more expensive it was to fix (Kaner, James & Bret, 2001).

Table 3.10 [developed by McConnell (2004)] shows the related increase in cost if an error is not fixed until late in the development process. For example, if an error was not

caught during construction, and is not caught until Post-Release it can cost 10 to 25 more than it would have cost to fix if caught in construction.

Calculated Cost of Errors Based on when Introduced in the SDLC

Time Introduced	Time Detected				
	Requirements	**Architecture**	**Construction**	**System Test**	**Post-Release**
Requirements	1×	3×	5–10×	10×	10–100×
Architecture	-	1×	10×	15×	25–100×
Construction	-	-	1×	10×	10–25×

Table 3.10 Cost of Errors Based on when Introduced in the SDLC (McConnell, 2004)

Additional analysis were undertaken to determine the cost savings associated with usability testing based on when an error may have been introduced and the relative savings associated with identifying an error during usability testing (which could be done as early as the Requirements phase if prototypes are being generated). The final total cost savings was calculated using the formula as shown in Table 3.11 using the best case scenario for resolving errors when usability testing was not implemented to identify errors (i.e. when all errors that are introduced in the Requirements phase are resolved in the Architecture phase).

Cost of Error in the SDLC

usbCost = Usability Testing Cost
= Material + Travel + Research Subject + Usability Analysis

devCost = Estimated Development cost of fixing errors

costErr= Increased Cost of resolving errors (Best case Scenario)
= (Cost of Err in Requirements and Resolve in Architecture) - devCost

Total Savings = costErr - usbCost

Table 3.11 Cost of Error in the SDLC

3) Cost of Medical Error

Finally, an analysis was done to determine the cost of medical errors that may have been prevented by using Low-Cost Rapid Usability Testing (Kushniruk & Borycki, 2006).

To determine the possible medical errors that may have resulted from using the CDM Toolkit, a medical expert reviewed the usability testing logs of each subject to determine possible usability problems that could be associated with medical errors (the medical expert was a specialist physician with an extensive background in evaluating and auditing physician records). The medical expert noted each issue or error that could have possible negative medical impacts on the patient. Each negative medical impact was noted (e.g. lack of treatment could cause hospitalization). A cost-benefit analysis was then performed to determine the cost of the medical error from two different perspectives:

- Measurable Cost of Medical Error
- Complete Costs of Medical Errors

Measurable Cost of Medical Error

The measurable cost of a medical error was found by creating a list of the possible medical issues each medical error could cause (e.g. incorrect blood pressure could lead to a hospital visit). Then, using the BC Medical Services Commission Payment Schedule (2010) and the BC Ministry of Health Services Drug Data Files (2010), costs were found for each issue. These costs were then summed up to determine the total measurable cost of a medical error. Once the cost of the medical error was found, as shown in Table 3.12, the total cost savings was found by subtracting the total usability testing cost from the overall total measurable cost of the medical errors.

Measurable Cost of Medical Error
usbCost = Usability Testing Cost = Material + Travel + Research Subject + Usability Analysis costErr= Cost of each technology- induced error + Cost to Fix Errors **Total Savings** = costErr - usbCost

Table 3.12 Measurable Cost of Medical Errors

Complete Costs of Medical Errors

The cost of a potential medical error was determined by reviewing the existing literature to find the total cost of a medical error once all factors were taken into account. Total cost includes everything from the cost of treating the medical error to the cost of any litigation that may result from it. However depending on the exact factors considered in a study, the average cost of a single medical error can vary from $13,000 (Shreve et al., 2010) to as high as $600,000 (Thomas et al., 1999) per error. For the purpose of this study the conservative case was used assuming each medical error had an overall cost of $13,000 (Shreve et al., 2010). Once the complete cost of a medical error was found, as shown in the below calculation (see Table 3.13), the total cost savings was found by subtracting the total usability testing cost from the overall total complete cost of the medical error.

Complete Cost of Medical Error
usbCost = Usability Testing Cost = Material + Travel + Research Subject + Usability Analysis costErr= Complete cost of Medical Error **Total Savings** = costErr - usbCost

Table 3.13 Complete Cost of Medical Error

CHAPTER 4: RESULTS

The results of this study are discussed in terms of: (a) CDM Toolkit Usability Testing, (b) Usability Testing Review and (c) Cost-Benefit Analysis. The same subjects were involved in all areas of the study. The eight subjects were recruited from different health centers across British Columbia. All subjects had previous experience in using the CDM Toolkit. Table 4.1 provides information about the demographic characteristics of the subjects.

Study Demographics		
		Number (%)
Age(In years)	21-30	1(12.5%)
	31-40	3(37.5%)
	41-50	3(37.5%)
	51-60	1(12.5%)
	>61	0 (0%)
Location	Chilliwack	2 (25%)
	Greater Vancouver	4 (50%)
	Victoria	2 (25%)
Gender	Male	2 (25%)
	Female	6 (75%)
Occupation	2 Physicians	2 (25%)
	3 Nurses	3 (37.5%)
	1 Medical office assistants	1 (12.5%)
	2 Administrators	2 (25%)

Table 4.1 Study Demographics

Subjects were for the most part female (i.e. 75%) and 87.5% were under the age of 50. The eight subjects that were involved were from three main geographic areas; Chilliwack, Greater Vancouver and Victoria. As shown in Figure 4.1, subjects were representative of a diverse cross section of application user types. The majority of the

subjects were nurses, but the study also included application administrators, physicians and medical office assistants.

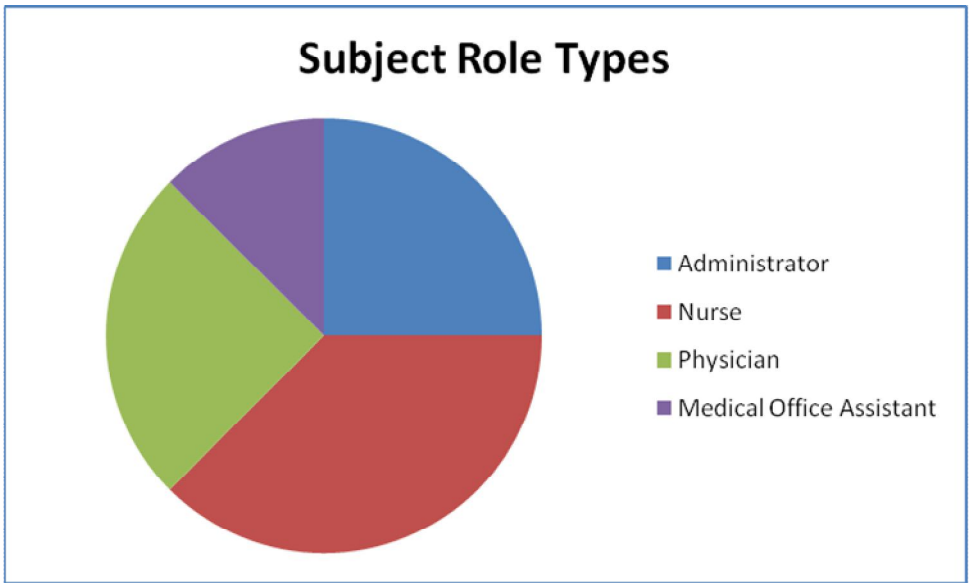


Figure 4.1 Subject Role Types

4.1 CDM Toolkit Usability Testing

As shown in Table 4.2, subjects had varying levels of experience in using the CDM Toolkit with 62.5% of subjects having 1 to 4 years experience in using the application.

Subject Experience with CDM		
Time Using CDM	Number of Subjects	% of Subjects
0	0	0%
< 1 year	2	25%
1 - 3 years	3	37.5%
3 - 4 years	2	25%
> 4 years	1	12.5%

Table 4.2 Subject Experience with CDM

The analysis of the usability testing involved 2 main components: (1) coding and classification of the errors encountered by the subjects and (2) analysis of the subject's survey responses.

4.1.1 Coding/Classification of Application Errors

The initial coding of each hour of video tape took 1 to 2 hours with an additional 4 hours required for coding the errors that were encountered during the testing. The coding of the errors was then reviewed by a second researcher. Any disagreements regarding the coding were resolved by additional discussion of the issue and review of the recordings of the testing session by two researchers. Table 4.3 is an example of a coded transcript of a subject "thinking aloud" while performing the test scenario (see Appendix G). The full transcript can be found in Appendix K. The "Time Stamp" is the actual time on the video. The "Subject Action" is what the subject is doing or saying. The "Observations" are the researcher's observations of what the subject is doing in different recordings. The "Coding Category/Severity Rating" is the coding category and severity rating for the individual error as outlined in the Methods section of this document.

Subject: 1			
Time Stamp	**Subject Action**	**Observations**	**Coding Category/Severity Rating**
2:43	"Waiting"	Subject is waiting for the screen to load	Speed/2
2:53	Clicks Add New Data		
3:09	"Hmmm … Enter observation date, is this the right page?"	The separate observation date page is confusing the subject. Enters Date and navigates to next page.	Navigation/3
3:26	"I think this is the right page"	Still not clear if in the right spot.	Navigation/3
3:27		Starts revising data.	
3:46	Clicks save		
3:50	"Waiting for page to save"	Performance Issue	Speed/2

Table 4.3 Sample Coded Transcript

The frequency of all coded usability problems for all 8 subjects is found in Table 4.4. On the left-hand side of the table, the differing categories of coded usability problems are listed. As indicated below the total number of usability problems related to the user interface was 60 while problems related to the content of information presented by the system accounted for 12 coded problems. Of the coded problems for the interface, the most frequently coded problem was related to speed: 41.7% of interface problems and 34.25% of all problems encountered were related to speed. Content problems were typically related to the application default values or the purpose of a field being unclear

(e.g. the subjects often did not know what to enter for observation date field on a flowsheet).

On the bottom row of Table 4.4, the total number of problems coded for each of the 8 subjects is given. As can be seen in the table, all of the subjects encountered an error. The number of errors per subject ranged from a minimum of 3 errors to a maximum of 16 errors. The average number of errors per subject was 10.9 errors.

Usability Problems and their relationship to Entry Errors										
Problem	Subjects								Total Usability Problems By Category	%Usability Problem Associated with Error
	P1	P2	P3	P4	P5	P6	P7	P8		
Interface										
Data entry	1				1	1	1	1	5	6.85%
Display Visibility			2		1	1			4	5.48%
Navigation	3	1	2	1	1	1	1	1	11	15.07%
Locating	1	2	1		1				5	6.85%
Procedure	1	1	2		2	1		1	8	10.96%
Speed	7	1	6	1		2	2	6	25	34.25%
Attention	1							1	2	2.74%
Content										
Data			1		1				2	2.74%
Defaults	2	1	1	1			2		7	9.59%
Help		1			1		2		4	5.48%
Grand Total:	16	7	15	3	8	6	8	10	73	

Table 4.4 Usability Problems and their relationship to Entry Error

The frequency of all the severity ratings for all 8 subjects is found in Table 4.5. On the left-hand side of the table, the severity rating categories for each of the coded problems are listed. The majority of the usability problems (i.e. 50.68%) were found to be category 2 errors or minor usability problems. 5.48% of the usability errors were category 4 errors or usability catastrophes that should be corrected before an application is moved to the production environment.

Usability Problems and their relationship to Severity Ratings										
Rating	Subjects								Total Usability Problems By Severity Rating	%Usability Problem Associated Usability Rating
	P1	P2	P3	P4	P5	P6	P7	P8		
0: Not a Problem	1	0	0	0	1	1	0	0	3	4.11%
1: Cosmetic problem	0	0	3	0	0	0	0	1	4	5.48%
2: Minor usability problem	9	4	7	2	2	2	4	7	37	50.68%
3: Major usability problem	5	3	4	1	3	3	3	2	24	32.9%
4: Usability Catastrophe	1	0	1	0	2	0	1	0	5	6.85%
Grand Total:	16	7	15	3	8	6	8	10	73	

Table 4.5 Usability Problems and their relationship to Severity Ratings

A total of 73 errors were found. The errors discovered were not just strictly usability errors, several errors were identified as programming errors that were not identified in the initial testing of the application (e.g. the Body Mass Indicator would not calculate). Errors varied from subject to subject.

The researcher further analyzed the data to determine a distinct list of the errors encountered as subjects often identified the same errors. Some of the errors identified were general issues with the entire application (e.g. Applications Speed Issues) where as other issue were very specific to one issue (e.g. Entering Height is Confusing). Table 4.6 shows the types of problems that were encountered by the subjects. The column on the left gives a brief description of the type of problem encountered, the column on the right outlines the details of the problems encountered while the other columns outline how often the problem was encountered and the related severity rating.

Distinct Errors Detected By Usability Testing				
Error Description	**# of Times Reported**	**% of Total Errors**	**Severity Rating**	**Details**
1. Applications Speed Issues	25	34.25%	2	It was noted at several different points of the test scenarios that the application was having performance issues (e.g. a screen was slow to load data).
2. BMI did not calculate	3	4.11%	4	The Body Mass Indicator (BMI) function on the diabetes Flowsheet would not calculate the BMI based on the entered height and weight.
3. Entering Height is Confusing	3	4.11%	3	The height could be specified in either imperial or metric (there were fields for both) but not both.
4. Entering Observation date when Navigating to Flowsheet	9	12.33%	3	It was unclear to the subject what value should be entered in the "Observation Date" field when creating a Flowsheet.
5. General Data Entry Issues	9	12.33%	2	Subjects were consistently having issues manually entering dates. It was not clear what the date format should be without using the calendar tool.
6. General Layout Issues	5	6.85%	1	The layout of the fields on the screen caused issues for subjects when trying to find specific data or links.
7. Invalid Application Certificate	8	10.96%	3	When starting the test scenarios subjects received a popup security warning stating that the application certificate was not valid.
8. Influenza Vaccine entered on Asthma did not show on Diabetes Flowsheet	1	1.37%	4	The influenza vaccine was entered and saved on one screen but did not carry over to the next screen.
9. Locating Flowsheet Links	3	4.11%	2	Subjects had trouble locating the links that were used to access the different flowsheets within the application.
10. No Error for Incorrect Blood Pressure	1	1.37%	4	When a blood pressure was entered that was outside of the clinically acceptable range no error message was generated.
11. Populating Other Meds Field	1	1.37%	3	It was unclear what data should be entered in this field.
12. Populating the Peak Flow Fields	2	2.74%	3	It was unclear what data should be entered in this field.
13. Popup window for Reports is Confusing	3	4.11%	3	When accessing the reporting portion of the application the reports were opened in a separate popup window. Subjects found this confusing.

Table 4.6 Distinct Errors Detected By Usability Testing

Of the total 73 problems encountered by subjects thirteen types of problems were found. Of the thirteen types of problems, the problem that occurred most often (i.e. 34.25% of the total) was application speed. Application speed has a severity rating of 2 (i.e. a minor usability problem). There were three other types of problems (i.e. 6.85% of the total). These problems had a severity rating of 4 (i.e. Usability Catastrophe). Given the magnitude of these problems they would require fixing.

4.1.2 Survey Results

Once the subject had completed the usability testing scenario they were asked to complete a survey about their experience in using the CDM Toolkit. The survey contained 13 close ended questions and 2 open ended questions. Figure 4.2 shows how the 8 subjects rated each question. On the left of Figure 4.2 is a brief description of the question being rated. The related bars show how many subjects selected each rating from 1 (Disagree) to 5 (Agree).

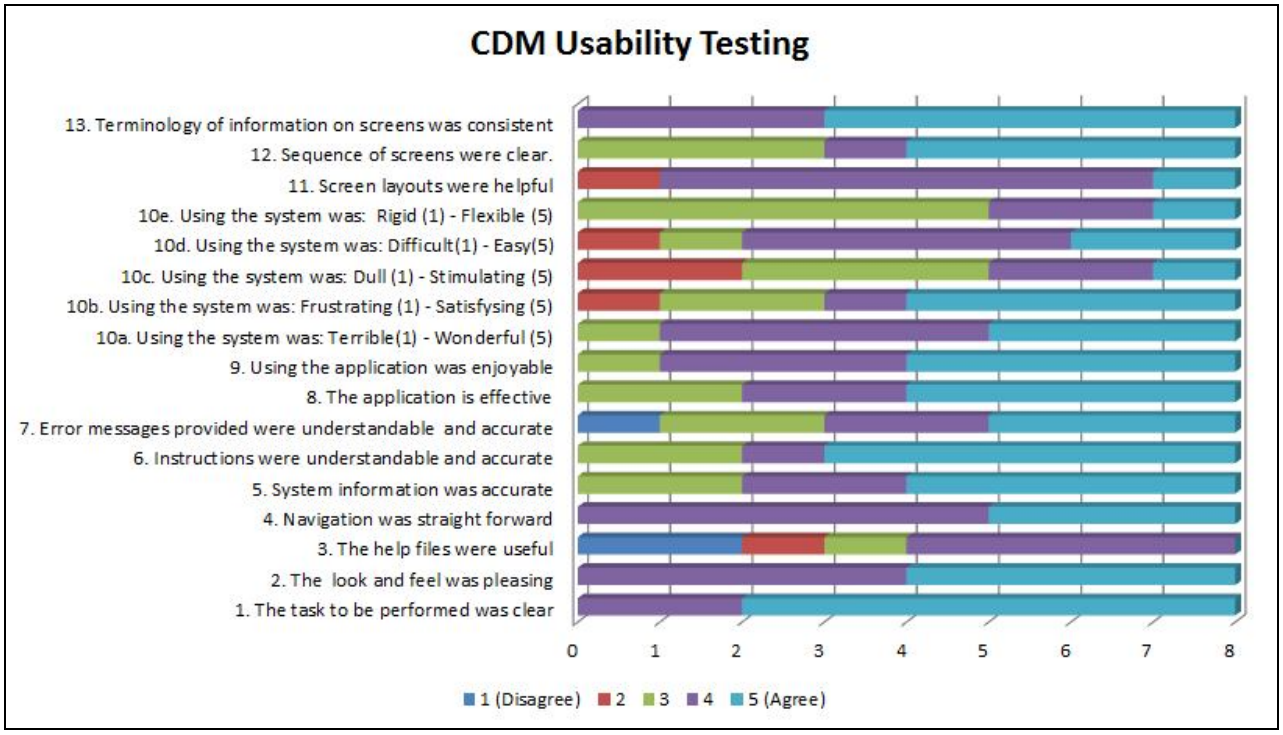


Figure 4.2 CDM Usability Testing Questionnaire Results

Further analysis of the particpant responses (as shown in Table 4.7) showed mostly positive responses with an average rating of 4.058 for all subjects. The left column of the table shows the question being evaluated. The middle column displays the mean value for each question and the right column shows the standard deviation for subject responses.

Survey Response Standard Deviation		
Question	**Mean Value**	**Standard Deviation**
1. The task to be performed was clear	4.75	0.463
2. The look and feel was pleasing	4.5	0.535
3. The help files were useful	2.875	1.356
4. Navigation was straight forward	4.375	0.518
5. System information was accurate	4.25	0.886
6. Instructions were understandable and accurate	4.375	0.916
7. Error messages provided were understandable and accurate	3.75	1.389
8. The application is effective	4.25	0.886
9. Using the application was enjoyable	4.375	0.744
10a. Using the system was: Terrible(1) - Wonderful (5)	4.25	0.707
10b. Using the system was: Frustrating (1) - Satisfying (5)	4	1.195
10c. Using the system was: Dull (1) - Stimulating (5)	3.25	1.035
10d. Using the system was: Difficult(1) - Easy(5)	3.875	0.991
10e. Using the system was: Rigid (1) - Flexible (5)	3.5	0.756
11. Screen layouts were helpful	3.875	0.835
12. Sequence of screens was clear.	4.125	0.991
13. Terminology of information on screens was consistent	4.625	0.518
Average	4.058	0.865

Table 4.7 Survey Response Standard Deviation

Additionally, subjects were also asked two open ended questions (the full results of which can be found in Appendix I). For the first question “Please comment on any other application issues that you feel should be noted:” 4 of the 8 subjects answered. The answers ranged from comments on the overall speed of the application to the BMI (Body Mass Index) function not calculating. For the second question “Please make any other

general comments:" 5 of the 8 subjects answered. Answers ranged from general comments about how the subjects felt about the changes, to specific comments on how the Flu Vaccine was not being treated correctly. Samples of some of the responses to the open ended questions are below:

- **Please comment on any other application issues that you feel should be noted:**
  - Reports were slow.
  - The new flowsheets are good.
  - The BMI was not calculating.

- **Please make any other general comments:**
  - The date of observation screen is annoying … every time I edit an observation date I need to click through it.
  - No carry forward on entry of Flu vaccines.

4.2 Usability Testing Review

Once the usability testing and surveys were completed, subjects were given an additional survey to obtain their opinion about usability testing in general. This survey consisted of 6 close ended questions and 2 open ended questions. Figure 4.3 shows how the 8 subjects rated each question. On the left of the figure is a brief description of the question being rated. The related bars show how many subjects selected each rating ranging from 1 (Disagree) to 2 (Agree).

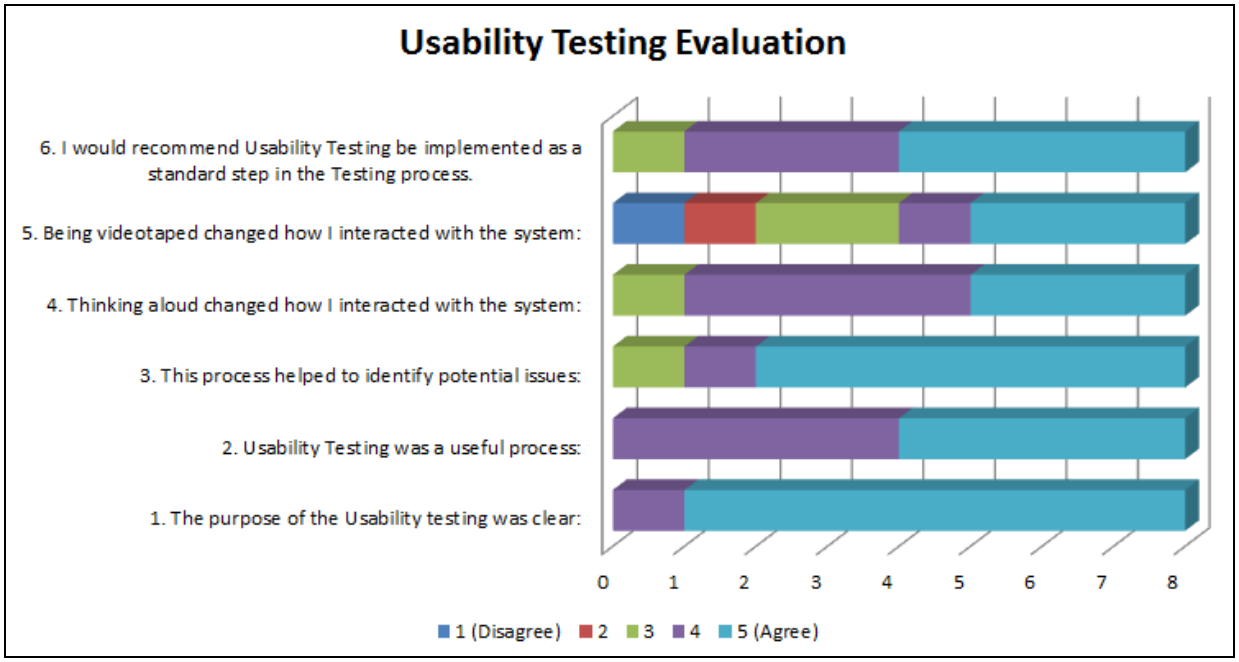


Figure 4.3 Usability Testing Evaluation Questionnaire Results

Further analysis of the subjects responses (as shown in the Table 4.8) revealled a positive average rating of 4.35. The left column of the table shows the question being evaluated. The middle column displays the mean value for each question and the right column displays the standard deviation for the subjects responses.

Survey Response Mean Value/Standard Deviation		
Question	**Mean Value**	**Standard Deviation**
1. The purpose of the Usability testing was clear:	4.875	0.354
2. Usability Testing was a useful process:	4.5	0.535
3. This process helped to identify potential issues:	4.625	0.744
4. Thinking aloud changed how I interacted with the system:	4.25	0.707
5. Being videotaped changed how I interacted with the system:	3.5	1.512
6. I would recommend Usability Testing be implemented as a standard step in the Testing process.	4.375	0.744
Average	4.354	0.766

Table 4.8 Survey Response Mean Value/Standard Deviation

Subjects were also asked two open ended questions (the full results for these can be found in Appendix J). For the first question "Please comment on any improvements you would make to the Usability Testing process:" 4 of the 8 subjects answered. The answers ranged from comments regarding the testing pool size, to ensuring detailed enough tests are included in the usability testing process to validate the entire application. For the second question "Please make any other general comments:" only 3 of the 8 subjects answered this question but with no significant amount of detail therefore this question was not considered as part of this study. Samples of some of the responses to the open ended questions are below:

- **Please comment on any improvements you would make to the Usability Testing process:**
  - This would have been useful in the first round of tests with the original application.
  - A more detailed testing brochure would be helpful.

- **Please make any other general comments:**
  - I thought this was a useful exercise, I hope to be using the toolkit soon with our EMR.

4.3 Cost-Benefit Analysis

One of the goals for this research has been to determine if implementing Low-Cost Rapid Usability Testing as part of the standard system development lifecycle is cost effective. To help determine this, all of the related costs associated with performing the usability tests were tracked. This involved tracking everything from the cost of office supplies to the cost of time spent analyzing the usability test results. As indicated in Table 4.9 the total cost of the usability testing was $8,362.91. This is the key value that was used for all cost-benefit analysis comparisons. For a detailed list of costs see Appendix L.

Usability Testing Cost Summary		
Cost Description	**Amount**	**% of Total Cost**
Material Total	$1,750.69	20.93%
Travel Total	$422.22	5.05%
Research Subject Total	$1,260.00	15.07%
Researcher Total	$4,930.00	58.95%
Total Usability Testing Cost:	**$8,362.91**	

Table 4.9 Usability Testing Cost Summary

The cost of all materials used in the study was $1,750.69. This cost included all the materials required for completing the usability study. This included the cost of the software (i.e. HyperCam©), hardware (i.e. laptop, video camera, headphones) and any office supplies that may have been required. The total cost of all travel required to complete this study was $422.22. As the study was completed at the individual subjects work sites, travel was required by one researcher to meet the subjects. The travel costs included the cost of gas, ferry travel, parking and meals.

All of the subjects were volunteers. However, for the purpose of this study a value has been calculated based on the time they provided and their relative hourly wage. The total cost of the 11.5 hours (in terms of work time) required by all 8 research subjects was estimated to be $1,260.00. The hourly rate for both of the researchers was based on the blended hourly rate of $85/hour of the company currently maintaining the CDM Toolkit. The time spent by the researchers (i.e. 54 hours for researcher 1 and 4 hours for researcher 2) in reviewing the results of the usability testing accounted for 58.95% of the usability testing costs with a total estimated cost of $4,930.00.

To determine if Low-Cost Rapid Usability Testing can be introduced as a cost effective, standard part of the system development lifecycle (SDLC) for health information systems, a cost-benefit analysis was completed. Three different perspectives were used:

- **Direct Measurable Savings:** This is the most basic case. Here a cost comparison is based on the cost of fixing the errors found in usability testing.
- **Cost of Errors related to when resolved in the SDLC:** This is a more complex comparison similar to the analysis performed by Karat (1997) where the costs were found based on when an error is found in the system development lifecycle (Shneiderman, 1997).
- **Cost of Medical Error:** A comparison was done where the cost of the medical error that could have been caused by system error was identified and tabulated.

As was stated above, all costs related to performing the usability testing (see Appendix L) were tracked (i.e. a total $8,362.91). This amount was used in the cost-benefit analysis. It must be noted, as shown in Figure 4.4, a large portion of the usability testing cost (i.e. 20.93%) were materials costs (i.e. the cost of materials used in testing).

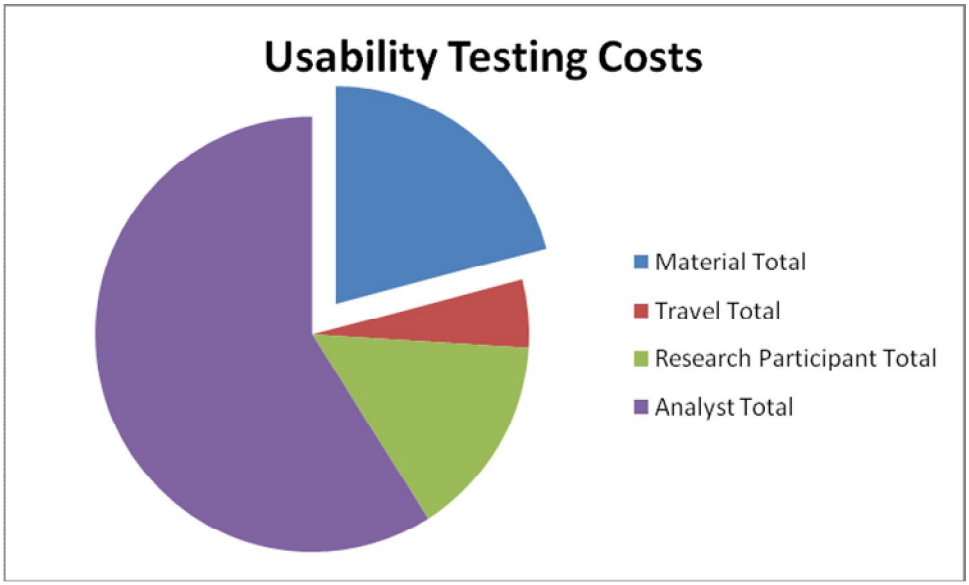


Figure 4.4 Usability Testing Costs

It should be noted that the materials used in usability testing (e.g. laptop, headphones, video camera) can be reused in future usability studies. Excluding the material costs from the usability testing cost would bring the total cost of the usability testing down to $6,612.22. However, to avoid skewing the results of this research in this cost-benefit

analysis the full amount (i.e. $8,362.91) will be used as the cost of performing the usability testing.

4.3.1 Direct Measurable Savings

The most basic and straightforward cost-benefit analysis case involves calculating the direct measurable cost. This is calculated by determining the cost of migrating the errors found during usability testing to production. This cost reflects the cost of the errors being found after the application has gone to production. It should be noted that this calculation does not account for the cost of fixing the errors as this cost would be incurred no matter when the error was found (ignoring the increased cost of when in the SDLC an error is found) in the SDLC.

Table 4.10 uses the formula that was outlined in **Chapter 3 - Research Materials and Methods** for calculating basic usability saving results. The average migration cost is assumed to be $3,230.00.

Basic Usability Savings Results

usbCost = Usability Testing Cost
= Material + Travel + Research Subject + Usability Analysis
= $8,362.91

numErrs = Number of errors found in usability testing
= 13

mgrCst = Cost of Migrating application to Production
= $3,230.00

$$\text{fixCost} = \frac{\text{Best Case Scenario} + \text{Worst Case Scenario}}{\text{Number of Scenarios}}$$

$$= \frac{(\text{Migrating All Fixes at Once}) + (\text{Migrating Fixes One at a time})}{\text{Number of Scenarios}}$$

$$= \frac{(\text{mgrCst}) + (\text{mgrCst} * \text{numErrs})}{2}$$

$$= \frac{\$3{,}230.00 + (13 * \$3{,}230.00)}{2}$$

= $22,610.00

Total Savings = fixCost - usbCost
= $22,610.00 - $8,362.91
= $14,247.09

Table 4.10 Basic Usability Savings Results

As indicated in Table 4.10 in the best case scenario the cost of fixing all errors would have been $3,323.00. The worst case would have been $41,990.00. For the purpose of this study, the average cost of resolving all of the errors encountered during usability testing was used (i.e. $22,610.00). This leaves a positive cost-benefit of $**14,247.09** after subtracting the cost of Low-Cost Rapid Usability Testing introduced before migrating the system.

4.3.2 Cost of Errors related to when resolved in the SDLC

As mentioned in the previous section, it is impossible to ignore when an error is identified in the system development life cycle (SDLC) in terms of relative cost of fixing an error. As outlined in a study by Kaner et al.(2001), the later in the development of an application an error is found the more expensive it will be to fix.

In order to determine the cost of an error over the SDLC, first a baseline cost for fixing the error must be found. Table 4.11 outlines the basic development cost of fixing each of the thirteen errors found during usability testing. Estimates were found by the researcher and then validated by a second researcher. Based on a blended hourly rate of $85/hour, it was found that the baseline development cost was **$10,880**.

Estimated Development Cost of Fixing Errors		
Error Description	**Time to Fix (hours)**	**Cost to Fix ($85/hour)**
1. Applications Speed Issues	40	$3,400.00
2. BMI did not calculate	6	$510.00
3. Entering Height is Confusing	8	$680.00
4. Entering Observation date when Navigating to Flowsheet	12	$1,020.00
5. General Data Entry Issues	2	$170.00
6. General Layout Issues	20	$1,700.00
7. Invalid Application Certificate	4	$340.00
8. Influenza Vaccine entered on Asthma did not show on Diabetes Flowsheet	8	$680.00
9. Locating Flowsheet Links	2	$170.00
10. No Error for Incorrect Blood Pressure	4	$340.00
11. Populating Other Meds Field	2	$170.00
12. Populating the Peak Flow Fields	4	$340.00
13. Popup window for Reports is Confusing	16	$1,360.00
Total	**128**	**$10,880.00**

Table 4.11 Estimated Development Cost of Fixing Errors

Based on a scale of increased cost for fixing an error relative to when the error was introduced and resolved in the SDLC (McConnell, 2004) (as outlined in the Materials and Methods Section above) Table 4.12 shows the cost of resolving the error.

Calculated Cost of Errors Based on when Introduced in the SDLC					
Time Introduced	**Time Detected**				
	Requirements	**Architecture**	**Construction**	**System Test**	**Post-Release**
Requirements	$10,880.00	$32,640.00	$108,800.00	$108,800.00	$1,088,000.00
Architecture	-	$10,880.00	$108,800.00	$163,200.00	$1,088,000.00
Construction	-	-	$10,880.00	$108,800.00	$272,000.00

Table 4.12 Calculated Cost of Errors Introduced in the SDLC (McConnell, 2004)

As shown in Table 4.12 the cost of resolving an error can vary from a baseline cost of $10,880.00 (if an error is introduced and resolved in the same phase of development) or the cost can significantly increase to a worst case scenario of $1,088,000.00 (if the error is introduced in the Requirements phase and not resolved until the Post-Release phase). If usability testing was introduced in the Requirements phase of the SDLC and carried through each phase of the SDLC (assuming all errors were caught) the cost of resolving the errors would never go above the baseline cost (i.e. errors introduced and resolved in the Architecture phase would cost $10,800.00 to fix). However, without usability testing, in the best case scenario, all errors are introduced in the Requirements phase and resolved in the Architecture phase. The increased cost of resolving errors in the Architecture phase is **$21,760.00.** In the worst case scenario, errors are introduced in the Requirements phase and not resolved until the Post-Release phase. In the worst case the increased cost of resolving the errors skyrockets to **$1,077,120.00**.

Whether you are considering the best or worst case scenario, as demonstrated in Figure 4.5, when usability testing is not undertaken the overall cost of resolving all errors rapidly increases if the error remains unresolved and the project moves through the different SDLC phases.

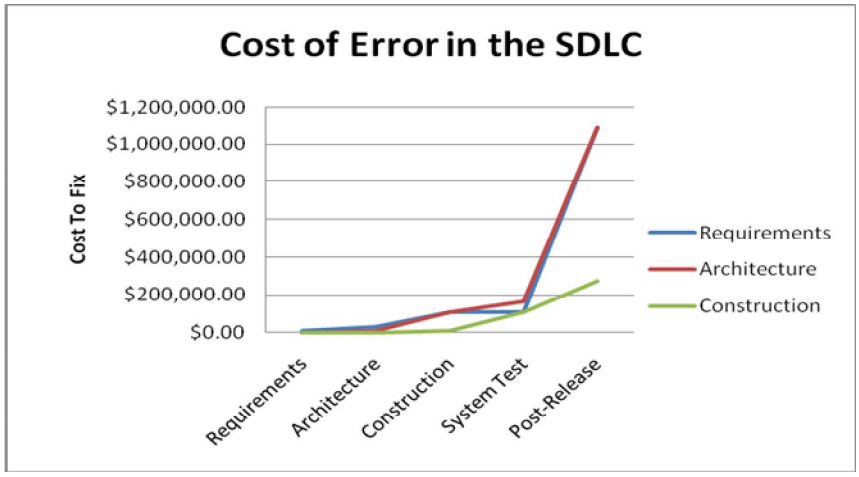


Figure 4.5 Cost of Errors in the SDLC

For the purpose of this study, only the best case scenario is considered (i.e. all errors are introduced in the Requirements phase and resolved in the Architecture phase). The best

case scenario is considered in calculating the overall cost savings incurred after introducing usability testing to the SDLC. This cost was found to be $21,760.00. As illustrated in Table 4.13, the overall savings was found to be **$13,397.09** by subtracting the total cost of usability testing (i.e. $8,362.91) from the best case scenario.

Calculation of Cost of Error in the SDLC
usbCost = Usability Testing Cost = Material + Travel + Research Subject + Usability Analysis = $8,362.91 devCost = Estimated Development cost of fixing errors = $10,880.00 costErr= Increased Cost of resolving errors (Best case Scenario) = (Cost of Err in Requirements and Resolve in Architecture) - devCost = $32,640.00 - $10,880.00 = $21,760.00 **Total Savings** = costErr - usbCost = $21,760.00 - $8,362.91 = $13,397.09

Table 4.13 Calculation of Cost of Errors in the SDLC

4.3.3 Cost of Medical Error

In the previous sections a cost-benefit analysis of usability testing was performed from a technical point of view. However this analysis did not take into account potential medical errors that may be caused by the system if not resolved before going live.

Medical errors contribute to adverse patient events, including death, that have been estimated to cost upwards of $19 Billion a year in the United States alone (Institute of Medicine, 2000). Preventing these errors (if possible) is not only cost effective it is ethically necessary. Although information technology has been touted to reduce such medical errors, the introduction of such technology has also been shown to lead to new types of errors arising from the technology's use in complex medical situations. These types of errors have been identified as technology-induced errors (Kushniruk et al., 2005). It is the potential cost of such errors that is one focus of interest in this thesis.

A review of the transcripts from the usability testing and analysis of the types of errors that arose during usability testing was undertaken by a physician to determine medical errors that could arise from the subjects' interactions with the system. The review involved 13 types of medical errors that were found during usability testing of the CDM toolkit. The following usability errors that could lead to a medical error (i.e. that could contribute to direct harm to the patient as the result of a technology-induced error) were the following:

- The Body Mass Indicator (BMI) did not calculate
- The Influenza Vaccine entered on the Asthma Flowsheet did not show up on the Diabetes Flowsheet
- There was No Error reported for Incorrect Blood Pressure

The possible medical errors that could have been caused by not resolving these errors can be measured in numerous ways. This study focused on two methods for determining the related costs: (1) direct measurable cost of medical error and (2) the total cost of a medical error after including all medical costs and related expenses.

4.3.3.1 Measurable Cost of Medical Error

The cost of medical errors that can be introduced through the implementation of a technology (i.e. a technology-induced error) is not only related to the cost of fixing the technology but the impact of the error on the treatment of the patient by a medical professional. In reviewing the three potential medical errors (i.e. BMI did not calculate, Influenza Vaccine Not Carried forward, No Error for Incorrect Blood Pressure) that were identified in the usability testing (in consultation with a medical professional) it was found that although there was a possibility of some very harmful effects to a patient, the most likely result of these three errors would be over treatment and possibly in some cases under treatment of these conditions. The case of over treatment will generally lead to additional visits to the patient's general practitioner (GP), unneeded prescriptions and in some cases visits to the emergency room. From a review of the three medical errors, a list of the most likely unneeded medical procedures that may have occurred was

developed. Using the BC Medical Services Commission Payment Schedule (2010) and the BC Ministry of Health Services Drug Data Files (2010), costs were found and associated with each of these unneeded medical procedures.

Measurable Cost of Medical Error	
Cost Descriptions	**Cost**
BMI did not calculate	
GP Consultation	$108.69
Glucose Testing	$164.15
GP Follow-up Consultation	$108.69
Subtotal	$381.53
Influenza Vaccine Not Carried forward	
GP Consultation	$108.69
Extra Dose of Influenza Vaccine	$62.15
Emergency Room Visit	$280.60
GP Hospital Visit	$73.71
GP Follow-up Consultation	$108.69
Subtotal	$633.84
No Error for Incorrect Blood Pressure	
GP Consultation	$108.69
Emergency Room Visit	$280.60
GP Hospital Visit	$73.71
Referral for Pulmonary function Testing	$225.14
Chest x-ray	$124.57
Renal Testing	$326.65
Thiazide Diuretics Prescribed (100 mg/day for 6 months)	$108.00
GP Follow-up Consultation	$108.69
Subtotal	$1,284.05
Total Cost	**$2,299.42**

Table 4.14 Measurable Cost of Medical Errors

As shown in Table 4.14 if each of the three possible technology-induced errors that were identified during usability testing occurred only once, the cost to the healthcare system would be $2,299.42. However, it is more likely that each of these errors could occur

several times before the errors are reported and the system is fixed. If each error occurred only four times, the cost to the healthcare system would be $9,197.68. This cost easily exceeds the cost of performing Low-Cost Rapid Usability Testing (i.e. $8,362.91).

However, for the purpose of this study a conservative approach was undertaken and it was assumed each technology-induced error only occurs once. The total cost of the measurable medical errors was calculated by adding the cost of fixing the application errors to the cost that the technology-induced errors for the healthcare system. As indicated in the Table 4.15 this cost was $13,179.42. Subtracting the total usability testing cost, $8,362.91, from this amount gives a total savings of $4,816.51.

Calculation of Measurable Cost of Medical Error
usbCost = Usability Testing Cost = Material + Travel + Research Subject + Usability Analysis = $8,362.91 costErr= Cost of each technology-induced error + Cost to Fix Errors = $2,299.42 + $10,880.00 = $13,179.42 **Total Savings** = costErr - usbCost = $13,179.42 - $8,362.91 = $4,816.51

Table 4.15 Calculation of Measurable Cost of Medical Errors

4.3.3.2 Complete Costs of Medical Error

The true cost of technology-induced errors may be much more than the direct cost to the healthcare system, as discussed in the previous section. To determine the actual cost of technology-induced errors many factors need to be considered, such as patient loss of life, permanent disability, loss of work and even ensuing legal action. In 2000 a ground breaking report by the Institute of Medicine entitled “To Err is Human: Building a Safer Health System” (Institute of Medicine, 2000), identified many of the factors related to medical errors in the United States. The Institute of Medicine found that the true cost medical error was in the range of $17-$19 Billion per year in the US. This report led to many healthcare institutions taking action. Quality initiatives were undertaken by

healthcare organizations to help improve patient safety. The electronic health record was identified as one means of reducing medical error.

In the years since the publication of "To Err is Human: Building a Safer Health System" (Institute of Medicine, 2000) many different studies (Leape & Berwick, 2005; Thomas, Studdert, Newhouse, Zbar, Howard, Williams & Brennan, 1999; Shneiderman, 1997; Shapiro, 2008; Shreve, Van Den Bos, Gray, Halford, Rustagi & Ziemkiewcz, 2010) have been conducted in order to determine if the number of medical errors has been reduced. Although the general consensus is that the number and severity of medical errors is being reduced, the number of medical errors that are still occurring remains high. Furthermore, it is not clear whether information technology is actually decreasing medical error rates overall, and what affect, if any, has technology had in terms of introducing new types of errors (i.e. technology-induced). The actual cost of medical errors varies from one study to the next but the cost continues to range between $8.8 Billion (Shapiro, 2008) to $19.5 Billion (Shreve et al., 2010). Depending on the study the average cost of a single medical error also varies greatly. One study, done using medical claims data from across the United States, found the average cost of a medical error to be $13,000 (Shreve et al., 2010) whereas another study conducted in Colorado and Utah reviewing adverse drug events, found the cost of a medical error could be as high as $600,000 (Thomas et al., 1999).

For the purpose of this study, as was done in previous sections of this thesis, a conservative cost will be used, assuming each medical error has an overall cost of $13,000 (Shreve et al., 2010). The total cost of medical errors will be calculated by multiplying the number of possible medical errors that were found during usability testing by the cost of a medical error. This assumes that each of these errors will only occur once. As indicated in Table 4.16 this cost was $39,000.00. Subtracting the total usability testing cost, $8,362.91, from this amount gives a total savings of **$30,637.09**.

Calculation of Complete Cost of Medical Error
usbCost = Usability Testing Cost = Material + Travel + Research Subject + Usability Analysis = $8,362.91 costErr= Complete cost of Medical Error = $13,000 x 3 = $39,000.00 **Total Savings** = costErr - usbCost = $39,000.00 - $8,362.91 = $30,637.09

Table 4.16 Calculation of Complete Cost of Medical Error

CHAPTER 5: DISCUSSION

The discussion in this chapter will focus on three main areas (1) the findings of the CDM Toolkit Usability Testing, (2) the subject's impressions of the usability testing process and (3) the cost-benefit analysis of the usability testing process.

5.1 CDM Toolkit Usability Testing

One of the main objectives of this study was to conduct a cost-benefit analysis of Low-Cost Rapid Usability Testing of a health application. As has been previously mentioned, the application that was chosen upon which to perform this analysis was the CDM Toolkit provided by Ministry of Health Services. A usability testing session was performed with each subject. The audio and video results from each session were transcribed and then analyzed for errors identified during the testing process. This analysis found that subjects encountered a total of 73 errors. These errors were coded and rated using a coding scheme developed by Kushniruk et al. (2005), and were also given a severity rating using a rating scheme developed by Nielsen (1995). The purpose of this analysis was to determine any common or major errors that were encountered by subjects and to correct them before the application was released to production. As shown in the Figure 5.1 the analysis of the 73 errors indicated that the majority of the errors encountered (50.68%) could be classified as "Minor Usability Problems" and would not necessarily have a major impact on the application. However, as indicated by the segregated portion of the pie chart in Figure 5.1, a small portion of the errors (i.e. 6.85%) were found to be "Usability Catastrophes". These errors require a resolution before the application goes to production.

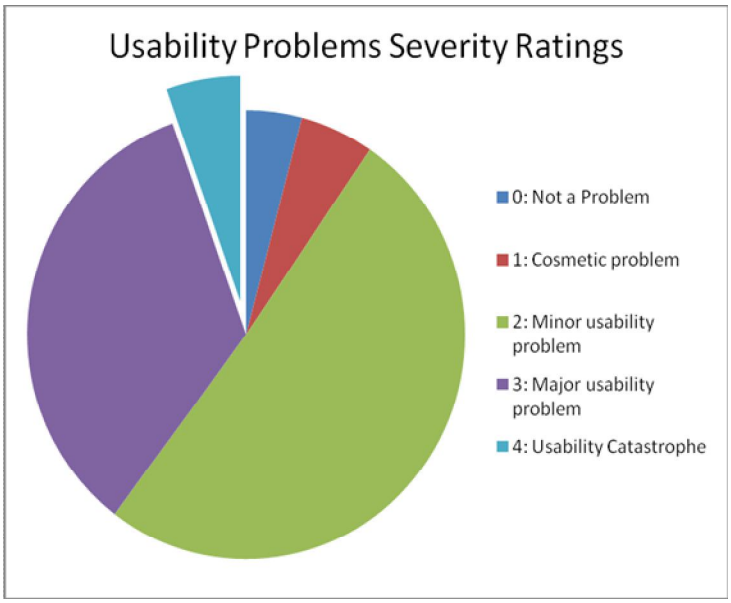


Figure 5.1 Usability Problem Severity Ratings

To further narrow down exactly what types of errors should be fixed from the 73 reported errors encountered by subjects, the errors were categorized according to type. As shown in the Results chapter this indicated that the subjects encountered 13 types of errors. It should be noted that several of the errors that were encountered were not usability errors but programming errors that were not identified during other types of testing (e.g. unit testing, black box testing).

The preferred result of usability testing is to fix all of the errors encountered by subjects. However, in some cases funding and time constraints make it necessary for project managers to prioritize what errors should be fixed. In the list of types of errors, 3 of the errors were rated severity of 4 “Usability Catastrophe”. These errors must be fixed. The 3 errors were:

- The Body Mass Indicator (BMI) did not calculate
- The Influenza Vaccine entered on the Asthma Flowsheet did not show up on the Diabetes Flowsheet
- There was No Error reported for Incorrect Blood Pressure

Additionally, it was found that the most common error was related to application speed (34.25% of total errors encountered). Even though this error was rated a Severity Rating of 2 or "Minor Usability Problem", due to the prevalence of this single error it was recommended that this error also be fixed before the application goes to production, as resolving this issue will help increase the overall usability of the system.

The prioritization of resolving the remaining errors should be based on their severity and number of occurrences. For example the next error to be resolved, if funding and time permitted, would be "Entering Observation date when navigating to Flowsheet" which was rated severity rating of 3 or "Major Usability Problem", this error occurred 9 times.

5.2 Usability Testing Review

The second objective of this study was to determine if subjects perceived Low-Cost Rapid Usability Testing as adding any value to the development process in general. To achieve this goal, each subject was given a survey (see Appendix E) to complete once they finished the usability testing. This survey contained 6 close ended and two open ended questions. In addition to this, once the usability testing was completed each subject was also given a post task interview (see Appendix F) containing 6 open ended questions. Here subjects were encouraged to express their opinions about usability testing while being audio recorded. Overall subjects found that usability testing was a useful tool. 87.5% subjects either agreed or strongly agreed that they would like to see usability testing implemented as a standard testing technique. However, the majority of subjects stated that "thinking aloud" (75% of subjects) and being videotaped (62.5% of subjects) could have an effect on how they interacted with the system. Additionally, after reviewing the comments subjects made in response to both the open ended questions in the survey and in the videotaped post task interview, two common responses were found to have been made by the subjects. The first was that subjects appreciated the opportunity to be involved in the development process of the application. Generally subjects felt they had systems forced on them without ever being asked their opinion; implementing Low-Cost Rapid Usability Testing could go a long way towards helping eliminate this impression. The second common response was that this form of testing would not be

robust enough to test entire systems. The impression was that the testing scenarios provided to the subjects were not detailed enough to test all possible paths through the system. This led to several discussions about the purpose of usability testing.

The purpose of usability testing as defined by Nielsen (1993) is not to have subjects test every possible scenario but rather to have subjects navigate common paths through the application and to identify errors or design flaws that may have been missed by other standard testing techniques. Usability testing should not be used as the sole testing technique for validating an application but should be used in conjunction with other standard testing techniques (e.g. unit testing, black box testing, white box testing, clinical simulations) to create applications that are as error free as possible.

5.3 Cost-Benefit Analysis

Usability testing has been in use for years in industry and many different cost-benefit analyses (Nielsen, 1994; Karat, 1997) have been performed to determine if it is an appropriate testing technique for validating applications. One notable study by Karat entitled “Cost-justifying usability engineering in the software life cycle” (Karat, 1997) found that usability engineering techniques could very effectively be introduced into the system development life cycle, allowing developers to identify application errors much earlier in the development process which led to a definite overall cost-benefit. However, after performing a review of the literature, it was found that no such cost-benefit analysis has been performed involving either discount usability engineering techniques (Nielsen, 1993) or Low-Cost Rapid Usability Testing (Kushniruk & Borycki, 2006) in the development of health information systems. This leads to the key question of this research, can Low-Cost Rapid Usability Testing be cost effectively introduced as a standard part of the system development lifecycle (SDLC) for health information systems.

As was stated in the Results chapter, depending on the factors that are considered in performing a cost-benefit analysis of Low-Cost Rapid Usability Testing, the results can vary. However, as indicated in Table 5.1, in all cases a positive benefit was found with

savings attributed to the implementation of this additional level of testing. The average cost savings was **$15,774.45,** which meant a total percent savings of **59.9%** compared to the impact of errors going undetected and potentially causing a technology-induced error.

Cost-Benefit Analysis Summary		
Cost-Benefit Analysis	**Savings**	**% Savings**
Direct Measurable Savings	$14,247.09	63.0%
Cost of Errors related to when resolved in the SDLC	$13,397.09	61.6%
Cost of Medical Error		
• Measurable Cost of Medical Error	$4.816.51	36.5%
• Complete Costs of Medical Errors	$30,637.09	78.5%
Average Savings	**$15,774.45**	**59.9%**

Table 5.1 Cost-Benefit Analysis Summary

Additionally, it should be considered that in performing the cost-benefit analysis for each scenario the most conservative case was used. If the very worst case scenario was used for each case the potential savings would be much higher. For example, as indicated in Table 5.2, the worst case scenario for all three medical errors that were found would be to assume each error had an associated cost of $600,000 (Thomas et al., 1999) as indicated in the Results chapter. This would mean the possible cost of a technology-induced error would be $1,800,000.00 which would lead to a massive total savings due to the implementation of Low-Cost Rapid Usability Testing of **$1,791,637.09**.

Calculation of Worst Case Scenario Cost of Medical Error
usbCost = Usability Testing Cost = Material + Travel + Research Subject + Usability Analysis = $8,362.91 costErr= Complete cost of Medical Error = $600,000 x 3 = $1,800,000.00 **Total Savings** = costErr - usbCost = $1,800,000.00 - $8,362.91 = $1,791,637.09

Table 5.2 Calculation of Worst Case Scenario Cost of Medical Error

5.4 Limitations

Although the findings of the study have been quite informative there were several limitations that should be noted and can hopefully be avoided in any future research related to this topic. One of the major limitations of this research was that shortly after the study began the Ministry of Health Services decided to move in a different strategic direction and the upgrade to the CDM Toolkit was cancelled. Although this did not stop the completion of the initial usability testing it did stop any of the recommended changes to the CDM Toolkit from being implemented and validated in a real world environment.

Additionally, due to a limited amount of time, the subjects volunteering for this study could participate in only one round of usability testing (involving a minimal set of test scenarios). It would have been beneficial to complete more test scenarios to evaluate additional areas of the application. Also performing more than one round of usability testing would have been useful to ensure that any changes made met the subject's expectations and that no new errors were introduced.

5.5 Possible Future Research

Although this study showed that Low-Cost Rapid Usability Testing could be cost effectively introduced into the system development lifecycle, if future research was performed in this area it would be beneficial to introduce Low-Cost Rapid Usability Testing into the system development lifecycle of an application from the initial Requirements phase and carry it through all the way to the application being released to production. The key benefit of extending the research this way is that it would allow the researcher to monitor the cost and possible application improvements of performing larger scale more detailed usability testing scenarios over multiple rounds of usability testing. It would also allow the researcher to investigate the benefits to the overall user friendliness and functionality of a system while allowing for end user input from the beginning of the system development lifecycle.

5.6 Study Implications

The goal of this study was to investigate if Low-Cost Rapid Usability Testing could be introduced into the SDLC of health information systems in a cost effective and productive manner. The results discussed in the previous sections indicate it can. This section will review the implications of this study's findings on existing literature and current system development practices.

5.6.1 Usability Literature and Theory

As has been discussed in previous sections of this thesis, adoption of usability testing is low in many cases due to the perception that usability testing is often expensive (as was shown in a study by Jeffries et al. (1991)). However, the results of this study refute those findings (supporting the findings of studies by Donahue (2001), Lund (1997), Karat et al.(1992), Karat (1993, 1994, 1997), Nielson (1994, 1995, 2007)), determining that implementing usability testing can lead to a cost savings (i.e. 61.6% - 63% savings) over the life of a project. Studies by Nielson (1994) and Donahue (2001) found that implementing usability testing could lead to a positive cost-benefit ratio ranging from 1:52 (Nielson, 1994) to 1:100 (Donahue, 2001) depending on the usability testing techniques that are utilized; the results of this study support those findings, determining that implementing Low-Cost Rapid Usability Testing could lead to a positive cost-benefit ratio ranging from 1:2.6 (cost: $8,362.91, savings: $21,760.00) to 1:2.7 (cost: $8,362.91, savings: $22,610.00).

Additionally, a study by Karat (1994) found that usability engineering techniques in general could effectively be introduced into the software development life cycle. The findings of this study support the findings of the Karat study (1994), it was found that Low-Cost Rapid Usability Testing could effectively be introduced into the SDLC.

5.6.2 Health Informatics Literature

Even though a number of cost-benefit analyses have been completed on usability testing in general industry (Donahue, 2001; Lund, 1997; Karat et al., 1992; Karat, 1994; Nielson, 1995; Nielson, 2007), a complete cost-benefit analysis of Low-Cost Rapid Usability Testing (as outlined by Kushniruk & Borycki, 2006) has (to the best of the researcher's knowledge) has never been published. A previous study by Kushniruk & Borycki (2006) entitled titled "Low-Cost Rapid Usability Engineering: Designing and Customizing Usable Healthcare Information Systems" calculated the cost of applying Low-Cost Rapid Usability Testing to the evaluation of a health information system and implied that it was very cost effective. However, the Kushniruk & Borycki (2006) study did not go as far as performing a full cost-benefit analysis. Demonstrating the effectiveness (in terms of cost-benefits) of applying such usability testing methods in healthcare is of importance given the problems encountered worldwide in the deployment and adoption of healthcare IT (Borycki & Kushniruk, 2008).

This study has taken the findings of Kushniruk & Borycki (2006) a step further by completing a full cost-benefit analysis of Low-Cost Rapid Usability Testing when implemented as part of the SDLC of a health information system. It was found that if Low-Cost Rapid Usability Testing is implemented as part of the SDLC of a health information system it is possible to achieve a 36.5% to 78.5% cost saving compared to the impact of errors going undetected and causing a medically induced error. Furthermore, to the best knowledge of the researcher, it represents the first time the potential medical cost of problems that are detected by usability testing have been estimated within a cycle of usability testing.

5.6.3 General System Development Practice

In a study by Dillon et al. (1993) entitled "A survey of usability engineering within the European IT industry" it was reported that 19% of organizations found usability testing to be 'unimportant' or 'not essential' and 54% of organizations had no dedicated usability

engineering staff. Contradicting the idea that usability testing is 'unimportant' or 'not essential' the results of this study show that usability testing leads to a definite cost savings (i.e. as high as 78.5%). It is the researchers hope that these findings would lead to higher adoption of usability testing practices in the development of health information system and in industry in general.

CHAPTER 6: CONCLUSION

It has been decades since usability testing was first introduced, since then it has been used in all forms of industry ranging from basic software development to the engineering of new airplane designs (Nielsen, 1993). However even though it has been shown that it could have potential in finding life threatening errors in health information systems it has still not gained main stream acceptance in healthcare (Kushniruk & Borycki, 2006), with one of the major barriers being the impression that usability testing is too expensive to implement on a regular basis. Previous studies (Karat, 1997) have shown that usability engineering techniques could very effectively be introduced into the system development lifecycle allowing developers the opportunity to identify application errors much earlier in the development process (which leads to a definite overall cost-benefit). The goal of this study was to show that this process could effectively be applied in the evluation of health information systems. To accomplish this, a cost-benefit analysis of Low-Cost Rapid Usability Testing was performed, considering the costs related to correcting technology-induced errors in a health information system from three main viewpoints.

- Direct Measurable Savings
- Cost of Errors related to when they are resolved in the SDLC
- Cost of Potential Medical Error

It was found that by introducing Low-Cost Rapid Usability Testing into the system development lifecycle errors could be detected prior to system release. Early detection of errors (i.e. prior to release) would allow organizations to achieve a 36.5% to 78.5% cost saving compared to the impact of errors going undetected and causing a medical error. Additionally, this study was designed to determine if users and stakeholders involved in the development of health information systems viewed Low-Cost Rapid Usability Testing as a useful process. In general, subjects involved in this testing process did see usability testing as a useful tool for improving health information systems. Subjects also appreciated the opportunity to have input into the development of a system before it was implemented as part of their everyday workflow.

Overall, Low-Cost Rapid Usability Testing was found to be an effective testing technique that can be implemented in conjunction with other testing techniques (e.g. unit testing, black box testing, white box testing, clinical simulations) in a cost effective manner to develop health information systems which will have a lower incidence of technology-induced errors. Additionally the results of this study also indicated that Low-Cost Rapid Usability Testing could lead to a cost savings of 61.6% - 63% (even when medical errors are not considered).

REFERENCES

Bates, D.W., Leape, L.L, Cullen, D.J., Laird, N., Petersen, L.A., Teich, J.M., Burdick, E., Kickey, M., Kleenfield, S., Shean, B., Vander Bliet, M., Sager, D.L. (1998). Effect of computerized physician order entry and a team intervention on prevention of serious medication errors. Journal of the American Medical Association, 280(15): 1311-16.

BC CDM Toolkit Documentation and Guides. Retrieved March 19, 2011 from, http://www.health.gov.bc.ca/access/guides/index.html

BC Medical Services Commission Payment Schedule. Retrieved August 25, 2010 from, http://www.health.gov.bc.ca/msp/infoprac/physbilling/payschedule/index.html

BC Ministry of Health Services Drug Data Files. Retrieved August 25, 2010 from, http://www.health.gov.bc.ca/pharmacare/outgoing/index.html

Bevan, N. (2005). Cost benefits evidence and case studies. Retrieved August 25, 2010 from, http://www.usabilitynet.org/papers/Cost_benefits_evidence.pdf

Bias, R.G., Mayhew, D.J. (2005): Cost-Justifying Usability: An Update for the Internet Age. Morgan Kaufmann; 2nd edition (April 4, 2005).

British Columbia Ministry of Health Chronic Disease Management Toolkit. Retrieved January, 01, 2010 from, http://www.health.gov.bc.ca/cdm/reqaccess.html

British Columbia Ministry of Health Chronic Disease Management Toolkit Test Application. Retrieved January, 01, 2010 from, https://cdmd3.moh.hnet.bc.

Borycki, E. M., Kushniruk, A. W. (2008). Where do technology-induced errors come from? Towards a model for conceptualizing and diagnosing errors caused by technology. In A. W. Kushniruk and E. M. Borycki (Eds.). Human, Social, and

Organizational Aspects of Health Information Systems. Hershey, Pennsylvania: IGI Press.

Dillon, A., Sweeney, M., Maguire, M. (1993). A survey of usability engineering within the European IT industry -- Current practice and needs. In J. L. Alty, D. Diaper, and S. Guest (eds.) People and Computers VIII. Cambridge: Cambridge University Press. pp. 81-94.

Donahue, G. (2001). Usability and the bottom line. IEE Software, 31–37. January/February 2001.

Ehrlich, K., Rohn, J. (1994). Cost-justification of usability engineering: A vendor's perspective. In Bias, R.G., and Mayhew, D.J. (Eds.), Cost-Justifying Usability. Academic Press, Boston, MA.

HyperCam. Retrieved January, 01, 2010 from, http://www.hyperionics.com/hc/

Institute of Medicine. (2000). To Err is Human: Building a Safer Health System. National Academy Press, DC.

Jaspers MW. (2009). A comparison of usability methods for testing interactive health technologies: methodological aspects and empirical evidence. Int J Med Inform. May; 78(5):340-53.

Jeffries, R., Miller, J. R., Wharton, C., Uyeda, K.M. (1991). User interface evaluation in the real world: A comparison of four techniques. Proc. ACM CHI'91 Conf. (New Orleans, LA, April 28 May 2), 119,124.

Kaner C, James B, Bret P. (2001). Lessons Learned in Software Testing: A Context-Driven Approach. Wiley.

Karat, C., Campbell, R., Fiegel, T. (1992). Comparison of empirical testing and walkthrough methods in user interface evaluation. In Proceedings of CHI'92 (Monterey, California, May 3- 7, 1992), ACM, New York, 397-404.

Karat, C. (1993).Usability Engineering in Dollars and Cents, IEEE Software, May 1993, pp. 88-89.

Karat, C. (1997). Cost-justifying usability engineering in the software life cycle. In Helander, M., Landauer, T., and Prabhu, P. (Eds), Handbook of Human-Computer Interaction. Elsevier Science, Amsterdam.

Karat, C. (1994). A business case approach to usability cost justification. Cost Justifying Usability. R.G. Bias and D.J. Mayhew, Eds. Academic Press, New York.

Koppel, R., Metlay, J. P., Cohen, A., Abaluck, B., Localio, A. R., Kimmel, S. E., Strom, B. L. (2005). Role of computerized physician order entry systems in facilitating medication errors. Journal of the American Medical Association, 293-1197-1203.

Kushniruk, A. W., Borycki, E. M.,(2006). Low-Cost Rapid Usability Engineering: Designing and Customizing Usable Healthcare Information Systems. Healthcare Quartely, 9(4), 98-100,102.

Kushniruk, A. W., & Patel, V. L. (2004). Cognitive and usability engineering methods for the evaluation of clinical information systems. Journal of Biomedical Informatics, 37(1), 56-76.

Kushniruk, A. W., Triola, M. M., Borycki, E. M., Stein, B., & Kannry, J. L. (2005). Technology induced error and usability: The relationship between usability problems and prescription errors when using a handheld application. International Journal of Medical Informatics, 74(7-8), 519-526

Kushniruk A, Triola M, Stein B, Borycki E, Kannry J. (2004). The relationship of usability to medical error: an evaluation of errors associated with usability problems in the use of a handheld application for prescribing medications. Medinfo. 2004;11(Pt 2):1073-1076.

Leape L, Berwick D. (2005). Five Years After To Err Is Human: What Have We Learned?. JAMA. Vol 293, No. 19.

Lund, A. (1997). Another approach to justifying the cost of usability. ACM Interactions, 4(3), 48-56.

Mantei, M., Teorey, T. (1988). Cost/benefit analysis for incorporating human factors in the software lifecycle, Communications of the ACM 31, 4 (April), 428-439.

Mayhew, D. (1999). The Usability Engineering Lifecycle: A Practitioner's Handbook for User Interface design, Morgan Kaufmann, San Francisco.

McConnell, S. (2004). Code Complete (2nd ed.). Microsoft Press. pp. 960.

Nielsen, J & Landauer, T (1993). A mathematical model of the finding of usability problems. Proc. ACM INTERCHI'93 Conf. (Amsterdam, the Netherlands, 24-29 April), 206-213.

Nielsen, J. (1993). Usability engineering. Boston: Academic Press, Inc.

Neilson J. (1994). Usability Inspection Methods. New York: Wiley.

Nielsen, J. (1994).Using discount usability engineering to penetrate the intimidation barrier, in R.G. Bias & D.J. Mayhew (eds), Cost-Justifying Usability, academic Press.

Nielsen, J. (1995). Severity Ratings for Usability Problems, http://useit.com/papers/heuristic/severityrating.html.

Nielsen, J. (2007). How to Conduct a Heuristic Evaluation. Retrieved August 25, 2010 from, ww.useit.com/papers/heuristic/heuristic_evaluation.html

Preece J, Roger Y, Sharp H, Benyon D, Holland S, Carey, T. (1994). Human-computer Interaction. Addison-Wesley Publishing Company.

Rigby M, Forsström J, Roberts R, Wyatt J.(2001).Verifying quality and safety in health informatics services. BMJ. 323(7312):552-6.

Rosson, M., Carroll, J. M. (2001). Usability engineering: scenario-based development of human computer interaction. Redwood City, CA: Morgan Kaufmann.

Ruben, R., Chisnell, D. (2008). Handbook of Usability Testing. Indianapolis, IN: Wiley Publishing Incorporated.

Shapiro S. (2008). Medical Errors Cost U.S. $8.8 Billion, Results in 238,337 Potentially Preventable Deaths. Health Grades.

Shneiderman B. (1997). Designing the User Interface. Addison Wesley. 3 edition 2 Jan

Shreve J, Van Den Bos J, Gray T, Halford M, Rustagi K, Ziemkiewcz E. (2010) The Economic Cost of Medical Errors. Society of Actuaries. Retrieved August 25, 2010 from, http://www.soa.org/files/pdf/research-econ-measurement.pdf

Thomas EJ, Studdert DM, Newhouse JP, Zbar BI, Howard KM, Williams EJ, Brennan TA. (1999).Costs of medical injuries in Utah and Colorado. Inquiry.Fall;36(3):255-64.

Virzi, R. (1992). Refining the test phase of usability evaluation: How many subjects is enough? Human Factors, 34(4), 457-468.

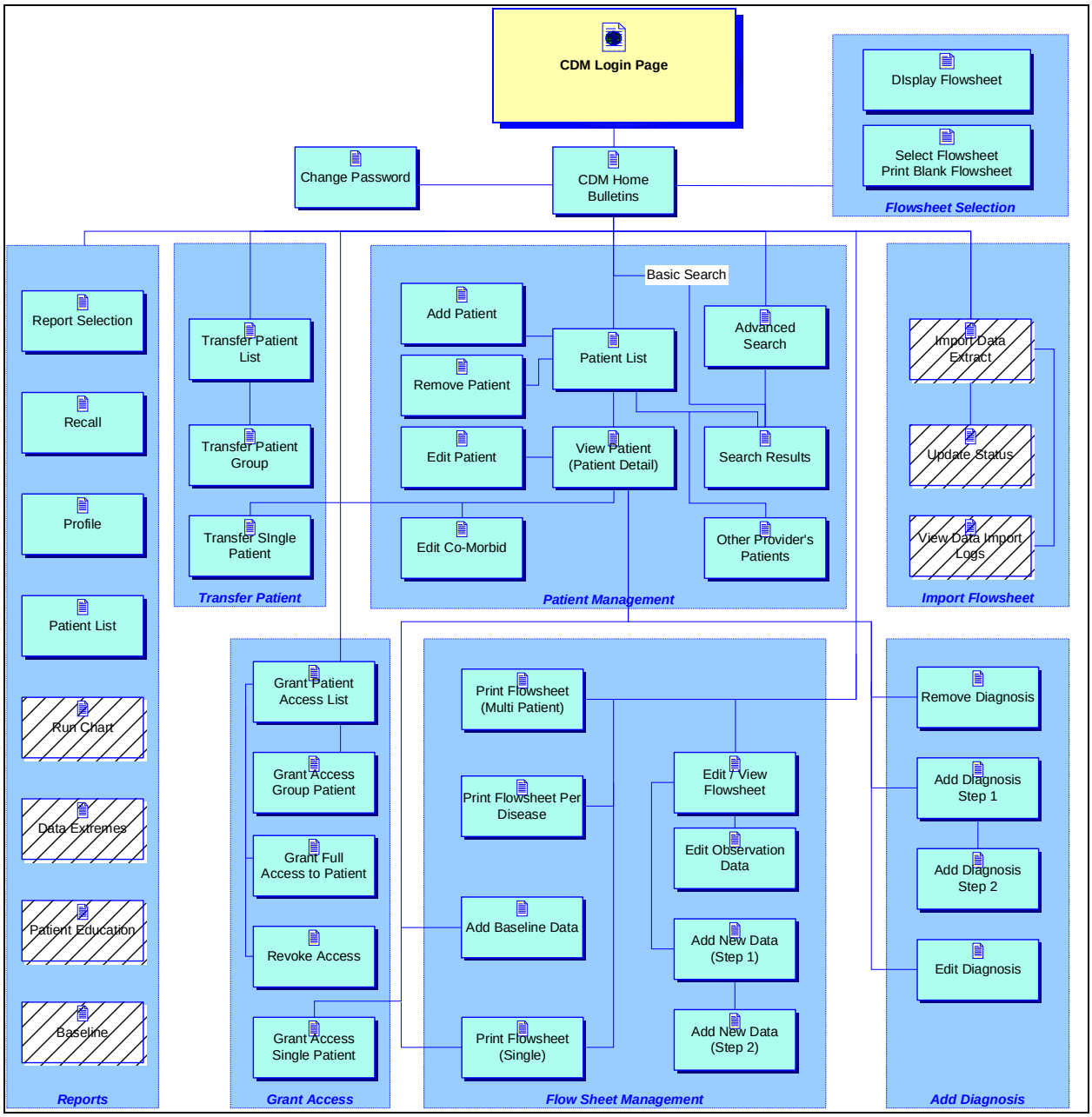
Virzi, R., Sorce, J., Herbert, L. (1993). A comparison of three usability evaluation methods: heuristic, think-aloud, and performance testing. Proceedings of the Human Factors and Ergonomics Society 37th Annual Meeting, (HFES, Santa Monica, CA) 309 - 313.

LIST OF ACRONYMS

Acronyms	
Acronym	**Definition**
BMI	Body Mass Index
CDM	Chronic Disease Management
EHR	Electronic Health Record
GP	General Practitioner
HCI	Human Computer Interface
HTTPS	Hypertext Transfer Protocol
IT	Information Technology
MOA	Medical Office Assistant
MoHS	British Columbia Ministry of Health Services
QUIS	Questionnaire for User Interface Satisfaction
SDLC	System Development Lifecycle
WHIC	Western Health Information Collaborative

LIST OF APPENDICES

Appendix A: *CDM Toolkit Site Map*



Appendix B: *Usability Testing Overview*

Study Overview / Checklist

Purpose : Review with each participant before beginning study.

1) Review what the test will involve.

- Overview
- HyperCam©
- Video Camera
- Use of Data
- Consent Form

2) Have user sign the consent Form

3) Review Test Scenario

4) Perform Usability Test

5) Perform Post testing interview with camera still on

6) Complete Usability Questionnaire

7) Complete Usability Evalution Questionnaire.

8) Ask the participant if they have any questions.

Appendix C: *Participant Consent Form*

Health Information Science
University of Victoria

Participant Consent Form

Low-Cost Rapid Usability Testing for Health Information Systems: Is it Worth the Effort?

You are invited to participate in a study entitled "Low-Cost Rapid Usability Testing for Health Information Systems: Is it Worth the Effort?" that is being conducted by Tristin Baylis.

Tristin Baylis is a Graduate Student in the department of Health Information Science at the University of Victoria and you may contact him if you have further questions by phone at (250)220-7942 or by email at tbaylis@uvic.ca.

As a Graduate student, I am required to conduct research as part of the requirements for a degree in Health Informatics. It is being conducted under the supervision of Andre Kushniruk. You may contact my supervisor at (250)472-5132.

Purpose and Objectives

The purpose of this research is to validate a previously unused type of health system software verification know as usability testing. Usability testing has been used with great success for decades in other industries but is still not consistently used in the validation of health systems.

Importance of this Research

Research of this type is important because this testing allows end users to validate a system before it is released into production with the goal of making the system more usable and reducing system errors.

Participants Selection

You are being asked to participate in this study because you are a users of the current version of the

Chronic Disease Management Toolkit.

What is involved

If you agree to voluntarily participate in this research, your participation will involve spending 45 - 60 minutes of your time to perform a predefined test scenario using the new version of the CDM Toolkit and then to complete a short questionnaire.

Risks

There are no known or anticipated risks to you by participating in this research.

Benefits

Being involved in this research will allow you to contribute your input to the overall usability of the new version of the Toolkit before it is put into production.

Voluntary Participation

Your participation in this research must be completely voluntary. If you do decide to participate, you may withdraw at any time without any consequences or any explanation. If you do withdraw from the study your data will not be used.

Anonymity

In terms of protecting your anonymity your actual name will not be used in the final results. Any video or voice recordings will only be used for analysis and also will not displayed in the final results.

Confidentiality

Your confidentiality and the confidentiality of the data will be protected by storing all digital data such as voice recordings, video and logging files in a password secure laptop with only access granted to the researcher. All paper records collected as part of this study will be stored in a locked filing cabinet.

Dissemination of Results

It is anticipated that the results of this study will be shared with others in the final thesis and dissertation.

Disposal of Data

Data from this study will be disposed of disposed of on completion of this study. All paper record will be shredded and all digital information deleted.

Contacts

Individuals that may be contacted regarding this study include Tristin Baylis (tbaylis@uvic.ca) or the supervisor for this study Andre Kushniruk (andrek@uvic.ca).

In addition, you may verify the ethical approval of this study, or raise any concerns you might have, by contacting the Human Research Ethics Office at the University of Victoria (250-472-4545 or ethics@uvic.ca).

Your signature below indicates that you understand the above conditions of participation in this study and that you have had the opportunity to have your questions answered by the researchers.

_____ _____ _____

Name of Participant Signature Date

A copy of this consent will be left with you, and a copy will be taken by the researcher.

Appendix D: *Usability Testing Questionnaire*

Health Information Science
University of Victoria

CDM Toolkit Evaluation

Please rate the following questions on a scale of 1 – 5:

The task to be performed was clear	Disagree 1	2	3	4	Agree 5
The look and feel was pleasing	Disagree 1	2	3	4	Agree 5
The help files were useful	Disagree 1	2	3	4	Agree 5
Navigation was straight forward	Disagree 1	2	3	4	Agree 5
System information was accurate	Disagree 1	2	3	4	Agree 5
Instructions were understandable and accurate	Disagree 1	2	3	4	Agree 5
Error messages provided were understandable and accurate	Disagree 1	2	3	4	Agree 5
The application is effective	Disagree 1	2	3	4	Agree 5
Using the application was enjoyable	Disagree 1	2	3	4	Agree 5
Please circle the number which most appropriately reflects your impression of the system.	Terrible 1	2	3	4	Wonderful 5
	Frustrating 1	2	3	4	Satisfying 5
	Dull 1	2	3	4	Stimulating 5
	Difficult 1	2	3	4	Easy 5
	Rigid 1	2	3	4	Flexible 5
Screen layouts were helpful	Confusing 1	2	3	4	Clear 5
Sequence of screens	Confusing 1	2	3	4	Clear 5
Terminology of information on screens	Inconsistent 1	2	3	4	Consistent 5

Health Information Science
University of Victoria

CDM Toolkit Evaluation

Please comment on any other application issues that you feel should be noted:

Please make any other general comments:

Thank you for your help.

Appendix E: *Usability Testing Evaluation Questionnaire*

Health Information Science
University of Victoria

Evaluation of Usability Testing

Please rate the following questions on a scale of 1 – 5:
(1) Strongly Disagree (2) Disagree (3) Neutral (4) Agree (5) Strongly Agree

The purpose of the Usability testing was clear:	1 2 3 4 5
Usability Testing was a useful process:	1 2 3 4 5
This process helped to identify potential issues:	1 2 3 4 5
Thinking aloud changed how I interacted with the system:	1 2 3 4 5
Being videotaped changed how I interacted with the system:	1 2 3 4 5
I would recommend Usability Testing be implemented as a standard step in the Testing process.	1 2 3 4 5

Please comment on any improvements you would make to the Usability Testing process:

Please make any other general comments:

Thank you again for your help.

Appendix F: *Post Task Interview*

Post Task Interview

1) How did you find using the application?

2) Did you have any problems using it? (if so explain each)

3) Do you find that the application is usable?

4) Do you find that the information provided would be useful for your practice?

5) Do you have suggestions for improvement to the application?

6) Do you have any other suggestions?

Appendix G: *Usability Testing Scenario*

Health Information Science
University of Victoria

Test Scenario

1) Log into the application with the following credentials:
   Username: cdmtest107 **Password:** hello107
2) View the Patient list by selecting the Patient List/Maintain Patient Records link from the left side menu.
3) View patient John Doe's record Under the My Patients tab.
4) View the patients Asthma Patient Diagnosis Condition/Flowsheet.
5) After accessing the screen to View the patients data select the option to Add New Data to the patients Asthma Flowsheet.
6) Revise the patients Asthma related data and select Save the changes.
7) Select from the Options to view the patients Diabetes Patient Diagnosis/Condition Flowsheet.
8) Select the option Add New Data to the patients Diabetes Flowsheet.
9) Revise the patients Diabetes related data and select Save the changes.
10) Next generate reports for John Doe by selecting the Generate Reports link from the left side menu.
11) Select to create a Patient Education Report.
12) Generate a Patient Education Report with the following criteria:
    Patient: Doe, John **Condition/ Flowsheet:** Asthma
    Start Date: 01-Jan-2008 **End Date:** 28-Feb-2010
    Disease Attributes: Spirometry - FEV1%, Spirometry - PEF%
13) Navigate back to the Generate Reports Screen and select to generate a Run Charts Report.
14) Enter any criteria you feel are appropriate for the Run Charts Report and select Go.
15) Exit this e-service.

Finished

Appendix H: *Post Task Interview Results*

Open Ended Questions	**Responses**
How did you find using the application?	Fairly straight forward Popups were confusing
	Great
	Liked the new features Easy to adapt to
	The application was straight forward The loop at the end when logging out was confusing
	Application was slow
	Different but acceptable
	I like the asthma flowsheets
	Great overall
Did you have any problems using it? (if so explain each)	No Comment
	No major problems
	Lag time in loading pages
	No Comment
	No real problems.
	Not familiar with asthma so the asthma flowsheets were confusing.
	Peak Flow and medications purpose was not clear
	Speed was an issue
Do you find that the application is usable?	The network was slow
	No Comment
	No Comment
	It satisfies the mandatory reporting requirements but could provide more functionality.
	No Comment
	No Comment

	No Comment
	Yes it worked very well
Do you find that the information provided would be useful for your practice?	Not Applicable
	Yes
	More details need to be provided
	This would be useful with more details. Currently the office is paper based, it's like a black hole.
	No Comment
	Yes, but currently not familiar with some of it.
	No Comment
	Yes
Do you have suggestions for improvement to the application?	Make the reports faster Making the editing observation paths more clear
	Better performance
	More detailed instructions Make the reports clearer
	Add a page with a patient health record summary
	No Comment
	No Comment
	No Comment
	No
Do you have any other suggestions?	No
	Improve overall navigation
	No Comment
	No Comment
	Currently importing data directly from an EMR does not work. It would be great if this could be fixed.
	Importing previous patients would be helpful
	Diabetes flowsheets need more details
	No Comment

Appendix I: *CDM Toolkit Evaluation Questionnaire Results*

Question	1 (Disagree)	2	3	4	5 (Agree)
The task to be performed was clear	0	0	0	2	6
The look and feel was pleasing	0	0	0	4	4
The help files were useful	2	1	1	4	0
Navigation was straight forward	0	0	0	5	3
System information was accurate	0	0	2	3	4
Instructions were understandable and accurate	0	0	2	1	5
Error messages provided were understandable and accurate	1	0	2	2	3
The application is effective	0	0	2	2	4
Using the application was enjoyable	0	0	1	3	4
Using the system was: Terrible(1) - Wonderful (5)	0	0	1	4	3
Using the system was: Frustrating (1) - Satisfying (5)	0	1	2	2	4
Using the system was: Dull (1) - Stimulating (5)	0	2	3	2	1
Using the system was: Difficult(1) - Easy(5)	0	1	1	4	2
Using the system was: Rigid (1) - Flexible (5)	0	0	5	2	1
Screen layouts were helpful	0	1	0	6	1
Sequence of screens were clear.	0	0	3	1	4

Terminology of information on screens was consistent	0	0	0	3	5

Open Ended Questions	Responses
Please comment on any other application issues that you feel should be noted:	Network latency was an issue Reports were slow
	No Comment
	Reports were clear and easy to read New Flowsheets are good
	Unable to print patient report on a graph
	No Comment
	BMI not calculating
	No Comment
	No Comment
Please make any other general comments:	The date of observation screen is annoying ... every time I edit an observation date I need to click through it
	No Comment
	No Comment
	Data entered under comments fields is lost on different dates
	Implementing these changes is "Preaching to the Choir"
	No carry forward on entry of Flu vaccines
	No Comment
	I am very pleased with the new changes to the CDM Toolkit

Appendix J: *Usability Testing Questionnaire Results*

Question	1 (Disagree)	2	3	4	5 (Agree)
The purpose of the Usability testing was clear:	0	0	0	1	7
Usability Testing was a useful process:	0	0	0	4	4
This process helped to identify potential issues:	0	0	1	1	6
Thinking aloud changed how I interacted with the system:	0	0	1	4	3
Being videotaped changed how I interacted with the system:	1	1	2	1	3
I would recommend Usability Testing be implemented as a standard step in the Testing process.	0	0	1	3	4

Open Ended Questions	Responses
Please comment on any improvements you would make to the Usability Testing process:	A large enough pool of tests should be use to find all critical system issues.
	No Comment
	Would have been useful in the first round of tests with the original application Good to have real users test
	Complete tests on our own computers
	A more detailed testing brochure would be helpful
	No Comment
	No Comment
	No Comment
Please make any other general comments:	No Comment
	No Comment

	No Comment
	Tests were very basic
	No Comment
	The testing made me nervous
	No Comment
	I thought this was a useful exercise, I hope to be using the toolkit soon with our EMR.

Appendix K: *Usability Testing Participant Transcript Example*

Below is a sample of the usability testing transcripts, for the complete data please contact the author.

Participant: 1			
Time Stamp	**User Action**	**Observations**	**Coding Category/Severity Rating**
00:00	User reviews directions	Testing Begins.	
1:03	User selects the login button		
1:21	"Login is slow"		
1:30	"Looks like recording is affecting the key strokes"	HyperCam may be having a slight effect on data entry	Speed/0
1:38	Selects the My Patient link		
2:06	"Looking for John Doe"	User is scanning the screen for the name John Doe	
2:29	Selects the Asthma flow sheet link		
2:43	"Waiting"	User is waiting for the screen to load	Speed/2
2:53	Clicks Add New Data		
3:09	"Hmmm … Enter observation date, is this the right page?"	The separate observation date page is confusing the user. Enters Date and navigates to next page.	Navigation/3
3:26	"I think this is the right page"	Still not clear if in the right spot.	Navigation/3
3:27		Starts revising data.	
3:46	Clicks save		
3:50	"Waiting for page to save"	Performance Issue	Speed/2

3:59		Returned to the View/edit data screen	
4:12	Clicks the Diabetes link		
4:25	"Waiting for the page to load"	Performance Issue	Speed/2
4:40	Clicks the Add Data button		
4:46	"Again I am clicking through"	By passes the observation date screen.	Navigation/3
4:50		Navigates to the view edit diabetes screen.	
5:13	Selects save	User changes no data.	
5:20		Returns to View Edit data screen	
5:27		Searches for Generate Report Link	
5:38	"I'm clicking generate reports" Selects the generate reports link		
5:49	Selects the Patient Education report		
6:01	Starts entering report criteria		
6:08	Selects Patient John Doe		
6:15	"Asthma is already selected"		
6:19	Sets dates	Types the dates without using the calendar tool.	Data Entry/2
6:49	Selects Disease options		
6:58	Selects Go		
7:02	"Popup window loads with no content"		Locating/3
7:23	"Still Waiting"		Speed/3
7:38	"Still waiting"		Speed/3
7:50	Clicks back and forward between browser windows	User is getting inpatient	Speed/3
8:22		Report Loads	
8:30	"No lines Just dots"	Reviewing content of graph	

		in the report.	
8:46	Navigates back to the original window leaving the popup window open.		
8:52	Clicks the Back button returning to the Generate Reports page		
9:00	Selects the Run Charts report and continue		
9:17	Enters dates	Dates are entered manually	
9:54	Selects Go		
10:05		New IE window opens with report already loaded	
10:10		Reviews report	
10:35	Selects the main page again.		
10:40	Selects exit this Eservice		
10:58		Receives an Invalid Certificate error	Procedure/3
11:02	"And I'm done."	Testing is complete.	

Appendix L: *Cost-Benefit Analysis Spreadsheet*

Item	Cost
Laptop	$1,024.86
Internet Access	$9.99
Headset with microphone	$34.20
Web Cam	$22.79
Video Camera	$568.86
HyperCam	$39.99
Office supplies	$50.00
Room Use Fee	$0.00
Material Total	$1,750.69

Item	Cost
BC Ferries	$159.72
Gas	$82.50
Parking	$60.00
Meals	$120.00
Travel Total	$422.22

Researcher	Researcher 1	
Hourly Rate	$85.00	
Task	**Hours**	**Cost**
Preparation Time	16	$1,360.00
Usability Testing Scheduling	4	$340.00
Usability Testing	16	$1,360.00
Analysis	12	$1,020.00
Documentation	6	$510.00
Total	54	$4,590.00
Researcher	Researcher 2	
Hourly Rate	$85.00	
Task	**Hours**	**Cost**
Analysis Assistance	4	$340.00
Total	4	$340.00
	Researcher Total	$4,930.00

Research Participant Type	Hourly Rate	Hours	Cost	Participant Type
Participant 1	$80.00	2	$160.00	Administrator
Participant 2	$80.00	1	$80.00	Administrator
Participant 3	$120.00	1	$120.00	Nurse
Participant 4	$120.00	1.5	$180.00	Physician
Participant 5	$120.00	1	$120.00	Nurse
Participant 6	$120.00	2	$240.00	Physician
Participant 7	$120.00	1	$120.00	Medical Office Assistant

Participant 8	$120.00	2	$240.00	Nurse
Total		11.5	$1,260.00	

Total Usability Testing Cost:	**$8,362.91**