USITRES

-Usability Issues TRacking Electronic System-

-An Ontology Web-based tool for Tracking Usability Problems-

GABRIEL SCHOR

A THESIS
IN
THE DEPARTMENT
OF
COMPUTER SCIENCE
AND SOFTWARE ENGINEERING

PRESENTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF COMPUTER SCIENCE AT
CONCORDIA UNIVERSITY
MONTREAL, QUEBEC, CANADA

APRIL 2007

Your file Votre référence
ISBN: 978-0-494-34774-4
Our file Notre référence
ISBN: 978-0-494-34774-4

Abstract

USITRES – An Ontology Web-based tool for Tracking Usability Problems

Gabriel Schor

The results of usability tests including user knowledge, qualitative and quantitative data are generally fragmented and stored in various formats and tools. This limits our capacity to extract meaningful patterns and recommendations from the huge set of data. The interaction between the testers and developers in charge of making changes in the design is limited and filtered by usability professionals which results in losses in terms of the user message. During usability studies, the representative users are designated by those conducting the study and this is often wrong as users evolve or change with time. The software users, developers and designers need a common and shared referential to discuss in a cohesive manner the results of tests from different perspectives.

We propose a novel ontology-based platform for collecting, managing and disseminating test results as well as for mediating the communication line between all the people involved in tests including users, usability experts, developers and quality managers. The ontology clearly describes the issues and applications based on a well defined set of characteristics.

The set of characteristics, that are made available to usability professionals, clarify and improve usability related knowledge exchanges and therefore improve the effectiveness, objectivity and accuracy rate of usability studies.

The proposed ontology has been implemented as part a tool, called USITRES, a central usability repository. Several functionalities allow USITRES users to test the usability of applications and web sites during their use, verify and validate the assumptions made during design and have a voice during the redesign. USITRES allows all users to log usability bugs and trace them. It allows the design team to interact much more effectively with the users, acknowledge and address their concerns much faster. It also supports the developers in the decision making process regarding what the users need in terms of interaction and behaviour patterns based on a more accurate tests results.

ACKNOWLEDGEMENTS

This thesis was made possible by the following people and I would like to thank them all:

Dr. Ahmed Seffah for his guidance and dedication throughout my Master's degree. He gave me the chance to pursue my dream when few others would. His dedication to the field of usability and to his students is exemplary.

Ms. Halina Monkiewicz, Graduate Program Advisor, for her patience in always answering my questions and helping navigate through the process of acquiring my diploma.

Mr. Charles Adkins, my manager, and my colleagues from CGI for understanding the importance of this project for me and for investing in my development.

I am grateful to my family. I thank my parents and my grandmother for continuously believing in me. You helped me believe in myself even more.

I am thankful to my wife of 10 years, Alexandra, who agreed to set many projects that were very important for her aside so that she can help me during my Master's period. I would have never finished this endeavour without you.

Table of Contents

List of Figures

List of Tables

1 Chapter 1 – Introduction

1.1 Usability

Usability is today recognized as an important quality factor among the others factors such reusability, security, etc. Various international standards provide definitions for the different aspects of usability as well as for the field as a whole. Usability is a **quality attribute** that assesses how easy user interfaces are to use. The word "usability" also refers to methods for improving ease-of-use during the design process.

The ISO 9241-11: Guidance on Usability (1998) standard (which is part of the ISO 9241 series) provides the definition of usability that is used in subsequent related ergonomic standards:

> **The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.**

In the software engineering community the term usability has been more narrowly associated with user interface design. ISO/IEC 9126, developed separately as software engineering standard, defined usability as one relatively independent contribution to software quality associated with the design and evaluation of the user interface and interaction:

> **Usability: a set of attributes that bear on the effort needed for use, and on the individual assessment of such use, by a stated or implied set of users.**

ISO/IEC 9126 (1991) has recently been replaced by a new four part standard that has reconciled the two approaches to usability. ISO/IEC 9126-1 describes the same six

categories of software quality that are relevant during product development: functionality, reliability, usability, efficiency, maintainability and portability

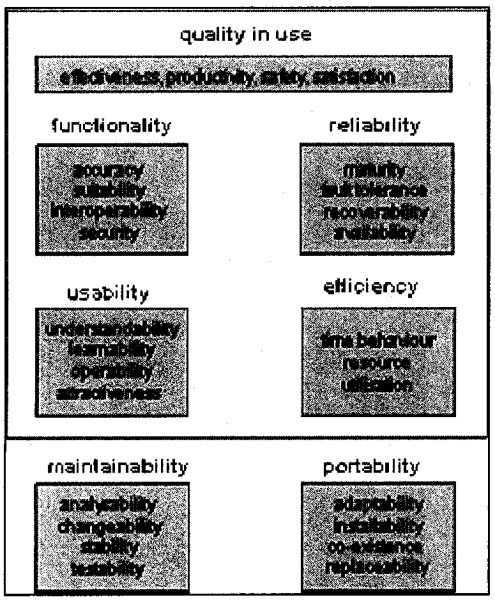

Figure 1-1 Quality in use

The definition of usability is similar to the previous standard:

> **The capability of the software product to be understood, learned, used and attractive to the user, when used under specified conditions**

The term "when used under specified conditions" is equivalent to "context of use" in ISO 9241-11. It was added to make it clear that a product has no intrinsic usability, only a capability to be used in a particular context. Other standards focused on different aspects of usability while providing complementary definitions such as:

- The use of the product (effectiveness, efficiency and satisfaction in a particular context of use) or product quality.
- The user interface and interaction, quality in use, user performance and satisfaction or products use in contexts.

- The process used to develop the product, development process, user centered process.
- The capability of an organization to apply user centered design, life cycle processes and usability capability.

A more complete list of usability related standards is provided below.

	Principles and recommendations	Specifications
Use in context	ISO/IEC 9126-1: Software Engineering - Product quality - Part 1: Quality model	ISO 20282: Usability of everyday products
	ISO/IEC TR 9126-4: Software Engineering - Product quality - Part 4: Quality in use metrics	
	ISO 9241-11: Guidance on Usability	
Interface and interaction	ISO/IEC TR 9126-2: Software Engineering - Product quality - Part 2 External metrics	ISO 9241: Ergonomic requirements for office work with visual display terminals. Parts 3-9
	ISO/IEC TR 9126-3: Software Engineering - Product quality - Part 3 Internal metrics	ISO/IEC 10741-1: Dialogue interaction - Cursor control for text editing
	ISO 9241: Ergonomic requirements for office work with visual display terminals. Parts 10-17	ISO/IEC 11581: Icon symbols and functions
	ISO 11064: Ergonomic design of control centres	ISO 13406: Ergonomic requirements for work with visual displays based on flat

		panels
	ISO 14915: Software ergonomics for multimedia user interfaces	ISO/IEC 14754: Pen-based interfaces - Common Gestures for text editing with pen-based systems
	IEC TR 61997: Guidelines for the user interfaces in multimedia equipment for general purpose use	ISO/IEC 18021: Information Technology - User interface for mobile tools
		ISO 18789: Ergonomic requirements and measurement techniques for electronic visual displays
Documentation	ISO/IEC 18019: Guidelines for the design and preparation of software user documentation	ISO/IEC 15910: Software user documentation process
Development process	ISO 13407: Human-centred design processes for interactive systems	ISO/IEC 14598: Information Technology - Evaluation of Software Products
	ISO TR 16982: Usability methods supporting human centred design	
Capability	ISO TR 18529: Ergonomics of human-system interaction - Human-centred lifecycle process descriptions	
Other	ISO 9241-1: Part 1: General Introduction	
	ISO 9241-2: Part 2:Guidance on task requirements	
	ISO 10075-1: Ergonomic principles related to mental	

	workload - General terms and definitions	
	ISO DTS 16071: Guidance on accessibility for human-computer interfaces	

Table 1-1 International standards for HCI and usability

Generally, usability is measured indirectly via the five components, also named criteria or measures:

- **Learnability**: How easy is it for users to accomplish basic tasks the first time they encounter the design?
- **Efficiency**: Once users have learned the design, how quickly can they perform tasks?
- **Memorability**: When users return to the design after a period of not using it, how easily can they reestablish proficiency?
- **Errors**: How many errors do users make, how severe are these errors, and how easily can they recover from the errors?
- **Satisfaction**: How pleasant is it to use the design?

There are many other important quality attributes. A key one is **utility**, which refers to the design's functionality: Does it do what users need? Usability and utility are equally important: It matters little that something is easy if it's not what you want. It's also no good if the system can hypothetically do what you want, but you can't make it happen because the user interface is too difficult. To study a design's utility, you can use the same user research methods that improve usability (Nielsen 2007).

1.2 Usability testing

To asses usability, several methods have been proposed by the HCI research community and practitioners. Some methods used to test for and implement usability are listed in

(Hom 2007). The proposed methods can be classified under the three following categories:

- Inquiry
- Testing
- Evaluation

1.2.1 Inquiry Studies

Usability evaluators obtain information about users' likes, dislikes, and understanding of the system by talking to them, observing them using the system in real work, or letting them answer questions verbally or in written form:

- Contextual Inquiry is mainly a discovery method as opposed to a testing method. During contextual inquiry the interviewer questions users in their environment to discover three key information types: Contextual inquiry is based on three core principles: what is the *context* of use, who is the user as a *partner* in the design process, and what should be the *focus* of the final product.(Beyer 1998: 416)
- Ethnographic Study / Field Observation The interviewer observes system use in user's environment. Notes are taken on the *artefacts* present such as notebooks, computers, forms and reports used in task resolution and the *outcroppings* present, size of cubicles, amount of light present, uniforms and dress code that conveys status in the enterprise. (Buxton 1995:191)
- Interviews and Focus Groups are a quick and cheap way to obtain subjective feedback from users. The questions can be based on a pre-specified list of items (structured) or can encourage the participant to freely provide their views (unstructured) or they can be a combination of both (semi-structured). Focus groups are facilitated and informal discussion groups. More than one can be conducted to focus on various aspects of the final product. Focus groups are used to identify the typical users requirements and to validate their implementation at later time. Some of the limitations of user groups are the peer pressure generated

in such environments that can lead to inaccurate results. Focus groups can also produce a lot of information that is hard to quantify and assimilate. It is also a known fact that humans act differently in groups than when alone. (Greenbaum 2000: 3)

- Surveys - Interviewers ask test participants specific questions. SUMI (the Software Usability Measurement Inventory) is an example of the use of questionnaires to collect subjective feedback. Test participants fill in a standardized 50-statement psychometric questionnaire after using a product and their answers are analyzed with the aid of a computer program. SUMI data provide information on: perceived efficiency, affect (likeability), control, learnability, helpfulness. The Questionnaire for User Interface Satisfaction (QUIS) is another subjective assessment questionnaire. This questionnaire focuses directly on the user interfaces and was developed by Shneiderman and colleagues in the late eighties. It consists of sections that assess the users overall reactions to the software, screen, terminology and system information, learning, and system capabilities. A different type of surveys are cognitive workload surveys that are used in assessing how much mental effort a user expends when using a prototype or deployed system. For example this can be obtained with questionnaires such as the Subjective Mental Effort Questionnaire (SMEQ) and the Task Load Index (TLX). It is also possible to collect objective data from heart and respiration rate. Questionnaires differ from surveys in that they are written lists, not ad hoc interviews, and as such require more effort on the part of your users to fill out the questionnaire and return it to you. (Fink 2005: 1)
- Journaled Sessions are conducted when a disk with the software prototype is distributed to the test participants. The disk also contains software that records the user actions. Upon completion of the tasks the disk that now contains a journal of the steps performed by the participant in their task is returned to the usability team for the inspection of the recordings. (Lindroth 2001: 7)
- Self-reporting Logs where test participants record themselves UI operations. A high degree of subjectivity can be useful in finding the exact feelings produced by the use of the product.

- Screen Snapshots is a very simple method where a user takes screen snapshots at various relevant times during his or hers interaction with the interface. The method will be rendered obsolete by increased adoption of video capturing tools that can capture the whole interactive session and not just snapshots.
- Inspection. In usability inspection, usability specialists - and sometimes software developers, users and other professionals – examine usability-related aspects of a user interface. (Khosrowpour 2005)
- Heuristic evaluation is used to identify usability problems in paper-based screen designs, draft training plans, draft documentation etc. Users, developers and usability specialists review a set of designs individually, and then meet to discuss each element of the design in turn. The method is useful early on in the design cycle when a quick and cheap assessment of a system or product is needed, along with an assessment of user reactions. The method provides a collaborative forum for users and developers to evaluate different designs and so eliminate defective design elements prior to implementation. As with expert evaluations, a list of problems and their severity ratings is generated and contained within a report. Analysts evaluate the system with reference to established guidelines or principles, noting down their observations and often ranking them in order of severity. The analysts are usually experts in human factors or HCI, but others, less experienced have also been shown to report valid problems. (Nielsen 1994: p154)
- Cognitive Walkthroughs. Users, developers and usability specialists review a set of designs individually, and then meet to discuss each element of the design in a walkthrough meeting. Problems and their severity are identified and noted. Up to 5 people may participate in this method, which includes 1 usability specialist with knowledge of HCI issues and 2 people from both the user and designer communities. All participants must be provided with the paper designs that are to be reviewed and a room will be required to hold the walkthrough meeting in. The facilitator then starts the walkthrough meeting by making sure that everyone is introduced to each other and that everyone understands how the walkthrough will be conducted. Each design element is examined in turn, letting users have their

say first in order not to have the specialists dominate the discussion. A list of problems is drawn up by consensus and corresponding severity ratings are defined as they arise. When all the design elements have been looked at, the problem list and severity ratings should be reviewed and any changes that may be required should be made. (Redish 1999: p68)

- Formal Usability Inspections were designed from the early code inspections used at IBM to discover code defects and from the later documentation inspections used to discover documentation defects. Organization members with various backgrounds inspect the interface and express their opinions directly to the interface owner (the person who developed the particular interface feature or features being inspected). A recorder or scribe takes note of all the comments and observations and the document will be used by the owner to fix all the defects found. (Lazar 2001: p230)
- Pluralistic Walkthroughs. Participants from most stakeholder groups are present in this technique. Users, developers, usability professionals and even project and product managers sit together to discuss and evaluate each element of interaction in the interface. The method provides the advantage of using a wide range of skills and also of establishing a more personal connection between most of the stakeholders in the project. (Buxton 1995: p84)
- Feature Inspection focuses on the feature set of a product. The inspectors are usually given use cases with the end result to be obtained from the use of the product. Each feature is analyzed for its availability, understandability, and other aspects of usability. For example, a common user scenario for the use of a word processor is to produce a letter. The features that would be used include entering text, formatting text, spell checking, saving the text to a file, and printing the letter. Each set of features used to produce the required output (a letter) is inspected. (Feher 2006: p60)
- Consistency Inspection has for its purpose to ensure that similar functionality is implemented and presented in a similar way throughout all the modules of a particular software. The software inspection is usually undertaken by a usability professional. The inconsistencies will be presented to a team of decision makers

from the various development teams and a decision about which particular way of implementing the feature throughout the whole package is taken. This new implementation design can be one of the designs already implemented or a new design that includes the best features of all the implemented designs. (Lazar 2001: p230)

- Standards Inspection ensures that software design complies with industry standards. Usability expert with extensive knowledge of the standard inspects the interface in question and produces a report that is presented to the design team. Changes will be implemented based on this report. (Pradeep 1998)
- Guideline checklists start by deciding on a set of usability guidelines that will be implemented in the product. These guidelines are then communicated to the developers and during one of the iterative design cycles they are verified in connection with heuristic evaluations or consistency inspections. (Riva 2001 :p116)

1.2.2 Usability Testing

In usability tests, a representative set of users work on typical tasks using the system (or the prototype) and the evaluators use the results to see how the user interface supports the users to do their tasks. Representative users work on typical tasks using the system (or the prototype) and the evaluators use the results to see how the user interface supports the users to do their tasks. A small number of participants work with the system while an observer makes notes. The technique can be used to identify the most significant user-interface problems. The real world working environment and the product under development is simulated as closely as possible. Users undertake realistic tasks while observers make notes, timings are taken and video and/or audio recordings made. The observations are subsequently analyzed to derive metrics. Design problems are also identified. It is very important that the test has a clear purpose and that the specific goals of the tests are measurable.

- Thinking Aloud protocol encourages the test participant to voice his or her thoughts during the test. Their comments as well as the qualifiers being used can be used to determine if the interface design is clear or not. Another benefit is that their vocabulary used during the test can be used to identify certain features or can be at least included in the documentation. (Winters 2002 :p4)
- Co-discovery method Users employ a prototype as they work through task scenarios. They explain what they are doing by talking or 'thinking-aloud' and this is recorded on tape and/or captured by an observer. The observer also prompts users when they are quiet and actively questions the users with respect to their intentions and expectations. Supportive evaluation is a participatory form of evaluation. Users and developers meet together and the user representatives try to use the system to accomplish set tasks. The designers who observe can later explore the issues identified through a facilitated discussion. (Riva 2001 :p119)
- Question asking protocol has the tester ask the participant direct questions in addition to recording his comments. This technique is meant to fill in the gaps created by the inherent conscious or unconscious self-filtering of participant comments. It can determine if the participant ignored a functionally because its purpose was unclear or because it didn't even see it. The technique can also be used to direct the participants actions. (Campbell 2003 :p261)
- Performance measurement collects and studies performance data such as mouse, trackball movement and keystroke speed and relating them with the interface design. The measurements can be used as design objectives as well and various designs options can be tested until the desired result is achieved. (Karwowski 2006 :p2022)
    - Eye-tracking allows testers to identify what the participants are looking at during the test. It involves the use of very expensive equipments such as skin electrodes, marked contact lenses, image processing cameras, and reflector trackers. The reflector trackers measure the light reflected by the eye in several fixed, well identify positions before the tests begins, and then uses this data together with measurements of the light reflected by

the eye during the test to track the iris position and inherently what the participants are looking at.

1.2.3 Usability Evaluation

In his paper, 'How to Conduct a Heuristic Evaluation', Jakob Nielsen talked about an inspection method where one or more usability experts will inspect and evaluate a finished application or web site. They will compare the interface characteristics with a set of well defined usability guidelines and standards and evaluate the quantity and quality of the differences between the particular implementation and the standards.

Often the evaluations are conducted by more than one expert. His studies show that different evaluators will find different types of problems given their different experiences and backgrounds. The same studies showed that although the most obvious problems are spotted by everybody involved, the more subtle problems are only spotted by very experienced evaluators. These experienced evaluators will each find a relatively low number of advanced issues, therefore there is a need for several of these evaluators to find all the advanced issues.

Based on these studies he made the famous statement that 3 to 5 evaluators will be sufficient to find the vast majority of usability bugs in an interface. These evaluators were to work alone with an observer. The observer is present to record the observations that the expert makes. The observer's presence is needed so that the expert can concentrate on examining the interface and not on recording his findings. The usability experts are rarely also domain experts in the domain that the particular software that they are inspecting but they will be able to ask the observer domain specific questions.

The output of heuristic evaluation is a list of inconsistencies between the implementation and a list of general usability heuristics. The output of heuristic evaluation can sometimes be a list of inconsistencies between the implementation and a more domain specific set of usability heuristics. It is important, that while this method doesn't provide any fixes for

the problems found, that it identifies all the facets of each problem. For example if a certain dialog violates some of the heuristics, all the rules that are breached must be described such that the future solution incorporates them all.

1.3 Usability Metrics and Measurement

Some of the major metrics that have been proposed in the literature include(ISO 9126_4,2000):

“

- Effectiveness

  The accuracy and completeness with which users achieve specified goals.
- Efficiency

  The accuracy and completeness of goals achieved in relation to resources.
- Satisfaction

  Freedom from discomfort, and positive attitudes towards the use of the system.

Effectiveness measures of usability

Effectiveness refers to the accuracy and completeness with which users can achieve their goals. Typical measures include:

- Number of power tasks performed;
- Percentage of relevant functions used;
- Percentage of tasks completed successfully on first attempt;
- Number of persistent errors;
- Number of errors per unit of time;
- Per cent of users able to successfully complete the task;
- Number of errors made performing specific tasks;
- Number of requests for assistance accomplishing task;
- Objective measure of quality of output;
- Objective measure of quantity of output;
- Per cent of users who can carry out key tasks without reading the manual.

Efficiency measures of usability

Efficiency refers to the amount of effort users need to put in to achieve their goals. Typical measures include:

- Time to execute a particular set of instructions;
- Time taken on first attempt;
- Time to perform a particular task;
- Time to perform a particular task after a specified period of time away from the product;
- Time to perform task compared to an expert;
- Time to learn to criterion;
- Time to achieve expert performance;
- Number of key presses taken to achieve task;
- Time spent on correcting errors;
- Number of icons remembered after task completion;
- Time to install a product;
- Per cent of time spent using the manual;
- Time spent relearning functions.

Emotional measures of usability

Emotion refers to how users feel about the system. Typical measures include:

- Ratio of positive to negative adjectives used to describe the product;
- Per cent of customers that rate the product as "more satisfying" than a previous product;
- Rate of voluntary use.
- Per cent of customers who feel "in control" of the product;
- Customer rating on a 7-point scale anchored with "makes me more/less productive";
- Per cent of customers who would recommend it to a friend after two hours' use;

- Per cent of customers that rate the product as "easier to use" than a key competitor. " (Discount Usability)

Other, more complicated metrics are implemented in QUIM (A framework for quantifying Usability Metrics in Software Quality Models):

Metric	Task Effectiveness
Definition	TE=Quantity*Quality/100
Explanation	
Terms	Quantity=proportion of the task completed Quality=proportion of the goal achieved
Source	Bevan and Macleod (1994)
Metric	Essential Efficiency
Definition	EE=100*S_essential/S_enacted
Explanation	Estimates how closely a given user interface design approximates the ideal expressed in the use case model.
Terms	S_essential = The number of user steps in the essential use case narative S_enacted = The number of steps needed to perform the use case with the user interface design (rules for counting the number of enacted steps has come in the reference).
Source	Constantine & Lockwood, 1999
Metric	Layout Appropriateness
Definition	LA=100*C_optimal/C_designed
Explanation	Favors arrangements where visual components that are most frequently used in succession are closer together. Higher Layout Appropriateness means better usability.
Terms	$C= \Sigma P_{i,j} \times D_{i,j}$ where $i \neq j$ $P_{i,j}$ = Frequency of transition between visual components i and j

	$D_{i,j}$ = Distance between visual component i and j.
Source	Sears 1995
Metric	Task Concordance
Definition	TC=100 * D/P P=N(N-1)/2
Explanation	Measures how well the expected frequencies of tasks match their difficulty, favors a design where more frequent tasks are made easier (e.g. fewer steps).
Terms	N=The number of tasks being ranked D=Discordance score, i.e. The number of pairs of tasks whose difficulties are in right order minus those pairs whose difficulties are not in right order.
Source	Constantine & Lockwood, 1999
Metric	Task Visibility
Definition	TV=100 * (1/S_total * Σ Vi)
Explanation	The proportion of interface objects or elements necessary to complete a task that are visible to the user.
Terms	S_total=Total number of enacted steps to complete the use case Vi=Feature visibility (0 or 1) of enacted step I
Source	Constantine & Lockwood 1999
Metric	Horizontal or Vertical Balance
Definition	Balance=200 * W1/(W1 + W2)
Explanation	These metrics evaluate how well balanced the screen is both vertically and horizontally (a score of 100 indicate perfect balance).
Terms	W1=Weight of side one W2=Weight of side two Weight of a side = Number of pixels used * side's distance from the center Center = Halfway between the left edge of the left-most visual element and the right edge of the right-most element.

Source	Sears 1995

Table 1-2 Examples of calculable metrics in QUIM (Seffah A)

Although these metrics might produce correct results, they fail to identify the exact usability issues and their resolution. They rather leave this to the interpretation of usability experts. Our system will eliminate this interpretative, possibly subjective and definitely speculative step and allow users to rate usability issue directly, tell developers directly which usability feature is more important to them, increasing the reliability of the process.

Beside metrics or data for calculating metrics, the results of usability studies also include usability problems. In the following section we will discuss usability problems while introducing a taxonomy based on severity rating.

1.4 Usability Problems

In his usability studies Nielsen found that while major usability problems are easy to find, minor usability issues are much more numerous. Surprisingly he also found out that the expert evaluators don't take less time to discover the minor usability bugs. To focus the efforts of the developing team on fixing the major issues, the one that will impact more heavily a larger number of users, a rating system was needed.

However, each evaluator inspects the system individually and the severity rating that he will assign to each problem is relevant to the incomplete set of usability problem that he found. Additionally, during the inspection, evaluators are focused on finding usability bugs and at any point they are only aware of the bugs that they already found. Nielsen proposes that the usability issue rating should be performed after the inspection to eliminate this distraction during the actual process and to allow each evaluator to look at descriptions of usability issues found by other evaluators as well. Even in this case its appreciation of the severity of usability problems found by others will be skewed in one way or another as he has a very different experience from the bugs that he found. To workaround this problem Nielsen proposes the use of an average severity rating which

will be provided by the mean of at least three severity ratings given for the same usability issue.

In his view the severity of a usability issue is dictated by a combination of 3, possibly 4 factors. These factors are the frequency, the persistence and the impact or the time it takes the users to overcome the problem. The forth factor that Nielsen presents without necessarily putting it on the same level with the previous three is the market impact. This is the impact in sales of a software product or service caused by usability problems. One reason why this factor is presented separately might be that while the first three are experienced by the user, this forth factor is experienced by the software producer.
Nielsen proposes the following severity scale to rate usability problems:

"

0= I don't agree that this is a usability problem at all.
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.

"

Although the scale is valid in itself there is no connection between this scale and the frequency, persistence and the impact of an usability issue, at least no apparent connection. This scale can give developers and project managers an idea about which problem to tackle firs but it doesn't help the user in assigning a particular severity rating to a problem that he is facing.

We developed an improved problem taxonomy as part of our usability ontology. The improved problem categorization takes into account the number of impacted users, the type of impacted users and the domain of the impacted users (health, financial, industry). While the Nielsen severity scale is focused on issue metrics such as frequency,

persistence and duration, our priority model is based on the study of the environment within which the issue occurs. This approach is more appropriate because the issues that we study are not important in themselves but due to the impact they have in their environment.

1.5 Research Statement and Specific Methodology

The following are the problems related to usability testing and evaluation addressed in this thesis:

- Usability testing is based on experts' assumptions about who the user is. These assumptions can be wrong and often are by various margins.
- Traditional usability testing is performed by a reduced number of users, therefore excluding a significant part of the user community.
- Usability testing is executed only periodically based on development schedule, in the best case once at each development phase and in the worst case scenario only just before releasing a software product on the market.
- Usability testing is executed with the involvement of usability experts which places a limit on the testing activity due to the limits of the expert resource.
- Usability testing, even when executed remotely, is expensive.
- Communication between the users and the developers is inexistent, restricted or filtered.

We propose the creation of a central repository of usability testing and usability knowledge that would be so easy to use that all usability stakeholders would benefit from its existence and the IT companies will be able to use and reap its benefits.

Within this repository we will implement an usability ontology to act as a common language for all usability stakeholders and improve usability related communication. The ontology will also be used to better identify the application users, removing the need to assume or predict who these users are and what their characteristics are.

The web based characteristic of the repository will allow any application user with a web connection and internet browser to take an active part in the usability improvement process.

The move from detail usability to mass usability will create a revolution similar to the introduction of mass production by Henry Ford. This is what the IT industry needs now as most software design paradigms have been in use for a long time and still fail to make a huge difference.

USITRES will allow users to test the usability of applications and web sites during their use, verify and validate the assumptions made during design and have a voice during the redesign. It will allow users to log usability bugs and trace them; rate the usability of applications and web sites, etc. It will allow the design team to interact much more effectively with the users and acknowledge and address their concerns much faster.

The objectives of my research and thesis are to:

- Document the usability and usability testing objectives, methods and best practices.
- Document the IT environment surrounding usability, identify challenges brought into the usability field by this continuously changing environment and propose solutions to this challenges
- Develop a usability ontology that will incorporate users as well as usability issues.
- Design, implement and test USITRES, a web-based usability issues logging and repository application that will guarantee the future adaptability of usability testing to further changes in the surrounding IT environment. USITRES will help the focus of the usability testing efforts by becoming the central repository of all usability issues. All issues and users tracked by USITRES will be classified based on the above mentioned ontology. USITRES will broaden the scope of

usability testing by being completely open to the user community and not an expert only tool.

- Validate USITRES by logging a usability issues already discovered within our research group. Simulate their resolution using the USITRES work-flow and produce project and system wide reports for all the interested parties.

My research is a combination of literature review, technology review and selection for the implementation of USITRES and various usability issues review.

1.6 Structure of the thesis

The thesis is structured in 6 chapters.

Chapter 1 - Introduction. We will present various usability definitions, usability testing definitions and motivations as well as testing techniques. We will describe the current problem as we see it and present the solution. In the same chapter we will present the objectives and methodology of the research and the structure of the thesis.

Chapter 2 – Usability Literature and Tools. In chapter 2 we will describe the current existing tools for tracking and managing usability problems. We will also review the literature to highlight some of the concepts that inspired our work.

Chapter 3 – Ontology. In chapter 3 we will explain the reasons that drove us to the creation of an ontology. We will also describe in detail the problems, criteria, users, context, task, solution categories of our ontology.

Chapter 4 – Proposed tool. In this chapter we will describe USITRES. We will start by presenting the platform of the tool. We will explain the architecture the application date model and user interface. We will demonstrate its intended use through use cases from the point of view of each type of users as well as by following a usability issue through its whole life cycle in a scenario of use.

Chapter 5 – USITRES Versions, Evaluation and Limitations. In chapter 5 we will evaluate USITRES through the same criteria that will be used for the other applications tracked in USITRES. We will present which are the limitations of our application that we discovered during its validation. We will also present the modifications that will be made to the tool in future versions.

Chapter 6 – Conclusion, Contribution and Future Work. In the final chapter we will summarize our work and research and give a conclusion to the thesis.

2 Chapter 2 – Usability Literature and Tools

In this chapter we will present the influences and findings that shaped the usability tracking process that we will be implementing in USITRES.

Through our literature review we will show the importance of knowing who the users are, what are their characteristics and capabilities. We acknowledged this and made our process *user centric*. We will also show the importance of being aware of the user's perceptions. We also made our process *perception centric*. The users' culture influences their expectations and our process will be *culture aware*. *Web usability* studies showed the importance of perceived usability over task success rate and our process was designed based on this finding as well. Furthermore literature showed the importance of conducting usability studies in the user environment through remote usability and not in the lab, which is why USITRES will be focused on *remote usability*.

The tool review showed that the tools that use *text data format* are very efficient as opposed to those that use the *video format* which needs a more complex setup, but that both their results are opened to interpretation. Also the *auto-collection tools*, that automatically capture usage data, are un-intrusive but their findings lack feedback from the user and the observer and are very impersonal. The *all inclusive usability suites* are very time consuming and necessitate a very complex setup, making them tools that are exclusively used in organizations with extensive resources. Based on these findings about the existing tools characteristics we decided that USITRES will use text to store data, will be manipulated by the users itself to capture their emotions as well as other characteristics and be web based for the outmost simplicity of installation and use.

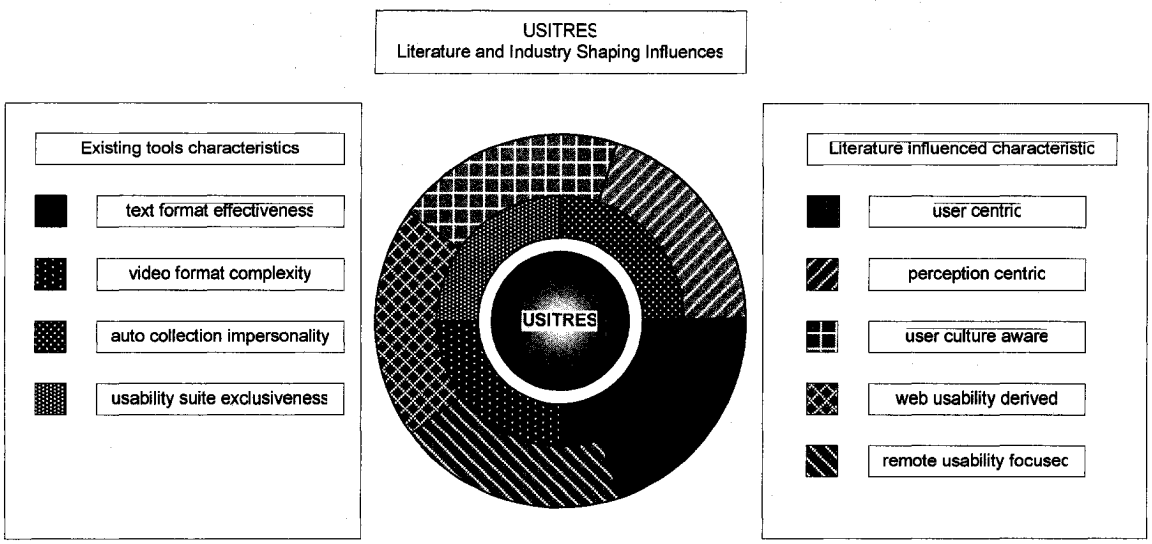

Figure 2-1 USITRES Shaping Influences

From the above illustration of the shaping influences we can see a summary of all the literature and industry shaping influences. All 9 influencing factors are equally important to moulding USITRES in its final form.

The findings from our literature review, together with the lessons learned of what not to do from the tool review shaped the USITRES process and tool. The process is designed to obtain a clear picture of the users and their usability issues derived from their perceptions. The tool is designed to be easy to use, accessible, personal and portable.

2.1 Background Work

Through our literature review we will demonstrate the importance of the *user centric approach* in software design. When designing for the user we will always have to keep their perceptions in mind, focus to deliver on their expectations and give the development process a characteristic of *perception centricity*. To understand the user and extract the meaningful nuances of their perception it is very important to account for the *user culture* influence on their profile and opinions. Although it is very hard to quantify the influence of culture on user actions all application stakeholders must be aware of this influence.

We will present some of the current views on *web usability*. Web usability had some new approaches to the usability issues and some of the implications of these new approaches will be presented such as the fact that perceived usability is more important than task success and that user focused measurements are more valuable then application focused measurements. The current development effort of USITRES is focused on taking usability out of the lab and we will review literature on other similar tendencies in usability, more specifically, *remote usability*.

2.1.1 User centric approach

There are two aspects of user centricity. First we have to learn who the user of the various applications are, what are their various characteristics and their capabilities. Second, we also have to develop a tool that is designed with the user in mind and capable of changing at the same time with the user.

The user centric approach attempts to have each step in the development effort, as well as the final product, performed and respectively produced with the final user in mind as opposed to the traditional approach that had the written specifications as the base for all decisions.

2.1.1.1 The Essence of Human – Computer Interaction

The author of 'The Essence of Human – Computer Interaction', Faulkner, describes in this book the multitude of the building blocks of human-computer interaction and illustrates them in Figure 2-1.

Computer Science	Artificial Intelligence	Anthropology
Ergonomics	Linguistics	Philosophy
Art	Sociology	Design
Psychology	Engineering	Physiology.

Table 2-1 Relationship between Computer Science, Artificial Intelligence and Anthropology

Christine Faulkner advocates the centrality of the user and the user task to the development and success of software. The ability to classify different aspects of the problem in "packets" may look like a success on paper and even the final product can look flawless but neither guarantees that the users' job just became easier. HCI is not static (therefore its concepts must change with time, more accurately with the user).

For her, the interface is the cushioning between the user and the task and its implementation is greatly influenced by the environment in which the system will operate. To better identify the user she classifies them into novice, knowledgeable /intermittent and expert/frequent user. At the same time she recognizes that other classifications might be appropriate for different systems. As far as task analysis goes she identifies that for this step in the development process all that is needed is a list of inputs, outputs, transformations and task composition (how often, dependencies on other tasks). The system is defined primarily through interviews with the users (that we will implement), observation, activity sampling and activity logging (in our opinion the last three are unreliable because they necessitate interpretation of the results by experts which is influenced by the experts experience rather than by the user experience). As strategies for representing design she presents storyboards, state transition diagrams and rapid prototyping. For the USITRES we are going to use user feedback and any of the above as the feedback will can be applied at any stage in the development cycle.

Evaluation (usability testing) is also pictured as a base of HCI. She identifies several problems that can appear during the evaluation phase such as mismatching between the tester characteristics and the user characteristics. This would be completely eliminated in USITRES as the testing will be done by the user.

She introduces the term of Usability Engineering and defines it as the process of developing the system that the user needs by evaluating and designing out any deficiencies. In other words, recursive usability testing should be performed until the user is satisfied with the end results.

2.1.1.2 Web User Interface Development

The solutions that Oracle implements, as described in 'Web User Interface Development at Oracle: How the Evolution of the Platform Influenced Our Products' Human-Factors Efforts', are developed based on simplicity and completeness focused design. They are first tested and used in-house before being deployed to the customer. Oracle is aiming to develop an all-inclusive platform that will satisfy all of their customer needs. Their customers will then be able to leverage the interface specific knowledge acquired using some of Oracle's products in all of their IT operations. The customers' costs are also reduced by the fact that the tools come ready to use. Oracle can also host the application which also reduces the maintenance costs for the client company. Part of this approach is facilitated by the fact that Oracle is the second largest software company in the world and has the resources to develop an all-inclusive solution. Even if the approach is valid it is hardly a silver bullet as it can only be applied by the largest players in the industry.

What Oracle also tries to establish is the sense of the usability experts being part of the overall design team. The atmosphere of collaboration as opposed to just criticising actually ensures a much higher percentage of usability recommendations being implemented. Similarly, we will try to involve the user in the development process by providing a direct link between him and the developer across USITRES.

Their preference to web based tool is motivated by the fact that there will only be one version of the software in use. Gone will be the days of countless coexisting applications versions that drive the costs of client support way up. Gone will be the days of countless functional and security patches that are sometimes overlapping and therefore the maintenance costs for the users will be much lower and the delivery times will be much shorter. The customer will also appreciate the elimination of the installation process of the full application or just of thin clients on all the computers in the company.

There are however several drawbacks to web tools such as download times, that even if they are short, because the web pages are kept simple, they are definitely longer that moving from screen to screen on a local application. Also HTML design doesn't always facilitate the implementation of all the functionality artefacts that can be implemented on traditional GUIs.

2.1.1.3 User-Centered Web Design

The most important benefits of a web site design geared towards usability described in 'User-Centered Web Design' by John Cato are:

- *Increased usefulness*: the more appropriate to the user tasks the product is the greater the acceptance rate of the product is by the user community and the greater is the user's desire to use it.
- *Increased efficiency:* matches the way in which the user naturally performs tasks therefore increasing the user's efficiency in using the web site.
- *Increased productivity:* the user will be able to concentrate its energy in resolving the tasks rather than coping with a cumbersome interface.
- *Fewer errors:* the user will spend less time debugging interface errors and looking for workarounds or applying patches.
- *Reduced training time:* a consistent and natural interface would be naturally simple to adopt by the users therefore decreasing the training time.
- *Improved acceptance:* the more efficient the user is the more he will enjoy his work and the more he will use the newly designed web site.

It is critical to involve users in the early stages of design. But it is also important to realize that design is a continuous process and that is often repeated 6 to 10 times as a product reaches maturity. In between these stages the user base changes and the context of use changes. It is important to realize this and to allow the newest users to have an impact on the overall site design. It is also important to realize that the mass user and the

early adopter have different backgrounds and they have to be allowed to express themselves in their own vocabulary.

The initial design concept will be modified based on the technological limitation and on the context of use. The designer hopes to produce a product that will satisfy a need of a large or influential population of users. The users have their own expectations and the two dimensions will help to define the last incarnation of the product. Up to now the expectations of users have been estimated based on the experience of people involved in creation of new products, although developing new products based on previous experience limits the innovation component of the new products.

2.1.1.4 The Co-Construction of Users and Technology

It is important to realize that while technology has a definite impact on the lives of the users, the users are equally powerful in the human-computer interaction relationship and they can have an equal impact on the development and success of various technologies and applications as shown in 'How Users Matter—The Co-Construction of Users and Technology' by Nelly Oudshoorn and Trevor Pinch.

When the camera was first developed, it was an expensive device that was used by highly skilled professionals for the needs of a very restricted group of rich individuals. When George Eastman redesigned the camera to be an inexpensive device easy to use, he was faced with another problem: there were no users for this type of device as the public opinion was still unchanged about the new device.

USITRES will not suffer this type of problems as it will always be in touch with its user base by allowing them to log their usability issues and concerns related to the system.

The SCOT approach defined by Pinch and Bijker (1984) identifies the user community as a social group with a role in the construction of technology. The approach identifies the co-construction process as a highly dynamic process where initially exists more than one

type of user groups, each with a slightly different interpretation of technology and with a slight different approach in using the technology. One study that supported the development of this theory showed that the older users of the high-wheeled bicycle defined it as the "dangerous bicycle". This in turn helped design the bicycle as we know it. The constant interaction between the users, the technology and its designers eventually drive the creation of a common vocabulary, a common interpretation of the use of the technology and a main type of usage. In this study the users are identified as agents of change.

USITRES allows the users to explicitly act their role by actively listening to their usability concerns.

Traditional studies of technology and discovery are focused on the designers of technology. Ruth Schwartz Cowan argues that these studies should focus on the final users of technology instead. In contrast with the co-construction theories which argue that users and technology have relatively equal powers in shaping each other, Cowan argues that we should focus on the users and their informal networks as the main driving force in the development of technology.
A blatant example of ignoring the users is the typical portraying of the women users as passive users of technology or even victims of technology. This attitude has changed in the recent decades and this important group of users that constitutes on average half the numbers of users for any technology is currently getting a much more prominent role in usability studies. Where as the earliest studies ignored the gender of the user, the most recent studies track this characteristic and so will USITRES.

". As Cowan (1987) suggested, users come in many different shapes and sizes. Medical technologies, for example, have a wide variety of users, including patients, health professionals, hospital administrators, nurses, and patients' families. "Who is the user?" is far from a trivial question. The very act of identifying specific individuals or groups as users may facilitate or constrain the actual roles of specific groups of users in shaping the development and use of technologies"

USITRES will take the guess work out of identifying who the user is by allowing the user to identify himself. Any one user of technology can be a tester in USITRES by simply creating a user in the system and then logging his or hers usability issue. A tester is then assigned to a manager but only for tracking purposes. USITRES will not only create a bridge between managers and developers who can be found anywhere in the world but also between users and managers closing the circle opened by managers who drove the creation of websites or application that touch users.

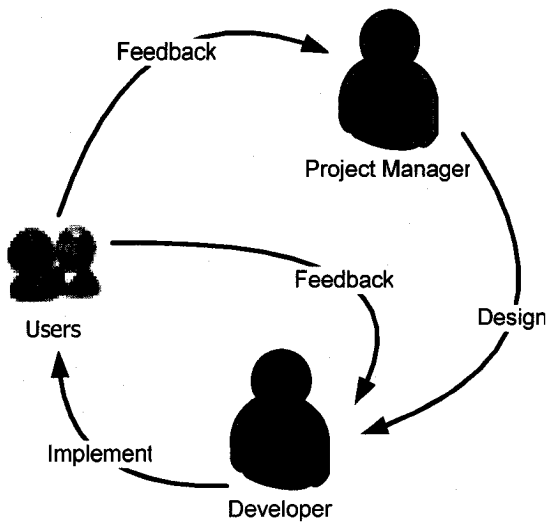

Figure 2-2 USITRES Communication Bridges

"To capture the diversity of users and the power relations between users and other actors in technological development, feminist sociologists have differentiated "end users," "lay end users," and "implicated actors." End users are "those individuals and groups who are affected downstream by products of technological innovation" (Casper and Clarke 1998). The term "lay end users" was introduced to highlight some end users' relative exclusion from expert discourse (Saetnan et al. 2000: 16). Implicated actors are "those silent or not present but affected by the action" (Clarke 1998: 267). And there are two categories of implicated actors: "those not physically present but who are discursively constructed and targeted by others" and "those who are physically present but who are generally silenced/ignored/made invisible by those in power" (Clarke, forthcoming)."

USITRES will give all these users a voice through feature requests and even product request. These requests will be implemented as regular usability issues, with the only difference that they will not be attached to an existing product but to a forthcoming one. These requests will then be auctioned off to USITRES participating companies for implementation of products or features. The companies' interests in these requests are guaranteed as they have already a guaranteed user base in the users that requested the particular application or feature. The funds obtained in this way will support further development of USITRES. Our system can act as a direct line of communication between the development companies and the technology users but in a direction opposed to the traditional one. Where traditionally companies educated users through documentation of the intended use of technology, this time around users will educate development companies on their needs and eventually on the modalities of satisfying them. **This will be the essence of our system as the main driver of mass usability where large user communities drive the development of technology as opposed to mass production where an abundance of instances of a particular technology shapes the users' lives even if only by the fact that the technology cannot be ignored, or through its use.**

Together with the theories on consumption and domestication supported by Bourdieu's (1984) this shift in the traditional relationship between designers and users can become a powerful shaping force of future software products. Bourdieu's theories converge on the fact that in today's society consumption is increasingly important in human relationships. In the most commercially powerful countries the human relationship and identities as well as one's place in the informal hierarchy of society are increasingly defined by consumption as opposed to production which was the traditional identity definer. This is apparent as people spend more and more time discussing their consumption habitudes and start to see their jobs as a mere tool in getting them to the desired consumer segment. USITRES users can be a driving force in this societal change by explicitly using their consumption patterns to dictate production. While the traditional efficient consumer would be an informed consumer, one that is always in the know about the latest available product and how this can make his life easier by satisfying some of his needs and wants, the new model of an efficient consumer will be an activist consumer that has an active

and explicit role in the design of products. This role can be fulfilled through the use of USITRES which allows the users to communicate their needs and wants and even how they can be fulfilled to the product designers. More specifically, these needs and wants will be formalized as usability issues of existing software products and requests for new products.

2.1.1.5 The Innovation Link between Organisation Knowledge and Customer Knowledge

Mehdi Khosrowpour postulates in 'Chapter 25: The Innovation Link between Organisation Knowledge and Customer Knowledge' of 'Challenges of Information Technology Management in the 21st Century: 2000 Information Resources Management Association International Conference' that the knowledge economy is the future and that organizations need to gather and leverage their knowledge if they are to survive in the future. While some of the knowledge is already stored under traditional document formats being on paper or stored in information systems, a very important quantity of information resides with the employees and the users. The research that the author conducted was carried to identify if indeed the organization recognize the fact that some of the knowledge is not yet structured and stored but indeed resides with their customers. The author argues that bringing together all the available repositories of knowledge can trigger meaningful innovation which will in turn guarantee the survival of the organization in the future.

Therefore organizations should focus on even more than just usability testing but overall knowledge collection or knowledge collection in the form of usability testing. Knowledge collection as advocated by the author is quite different than the traditional usability lab testing which is closer to knowledge validation as it is mostly performed to validate assumptions made by experts. The best proof of this fact is that the researchers develop typical user personas based on their intended user and not the actual user and then invite

for tests some users that represent these intended users. Testing with five users might be enough for validation but unlikely to be enough for knowledge collection.

"While knowledge is accumulated in organisations it is not harnessed in a way that will provide benefit. Each employee has an individual knowledge bank acquired as a result of education, skills and competencies developed over time, along with life experiences. The longer they work in an organisation the greater the knowledge they acquire about the organisation, its products, and its customers. The organisation also has a repository of knowledge through its documents, processes and routines Marshall, Prusak, Shpilberg, 1996; Demarest, 1997; Jordan and Jones, 1997; Fahey and Prusak, 1998). Marshall et al (1996) point to the volumes of information sitting in organisations, and Evans and Wurster (1997) identify every business as being an information business. Questions arise as to why this information is not put to use, and it may simply be there is a lack of knowledge to interpret it to provide possible opportunities.(p1)"

One of the obstacles in face of centralizing and consolidating the knowledge and also in face of extracting additional intelligence from it is a common knowledge vocabulary. We will provide this common vocabulary through the usability knowledge ontology that we will develop and implement in USITRES. We will provide various knowledge consolidations at the project, organization and industry levels.

"Knowledge is important for developing new products or services, and for adding value. While much of that knowledge can be obtained primarily through employees, databases, and business documentation, it will also come from meetings with customers (Byrne, 1993; Kanter, 1996; Amidon, 1997; Evans and Wurster, 1997; Jordan and Jones, 1997; Fahey and Prusak, 1998; Teece, 1998). Customer knowledge is extremely valuable.
"

"The literature indicates the value of working alongside customers. It is from good organisation/customer relationships and working collaboratively together that innovative approaches to problems or the development of new products emerge. Responses indicate that working with their customers is indeed something that many organisations do.

Further enquiry is, however, needed to identify how they do this and how each side benefits. They realise the value of developing good relationships and the benefits that are likely to accrue to both the customer and the organisation."

From the above quote it is important to retain the high value that the author puts collaborative work between the user and the developer. The collaborative component will be a step up from the actual type of usability testing where the users play a rather passive role when they are being observed by the usability professional. In USITRES, the users will be allowed to take initiative by having them log the usability issues themselves and even propose solutions that they might see fit.

2.1.2 Perception Centric

The final product usability is directly influenced by how close the product is to users perception of how well the product performs the needed tasks. The users' perception of whether a software is fast or usable is a much better measuring stick than measures such as time per task or clicks per task. Although the latest measures are much easier to collect, their results are opened to interpretation which can distort their real meaning. This is why we will be better off if we record the users' perception.

2.1.2.1 The Role of Expectations in Defining Stakeholders' Evaluations

Mehdi Khosrowpour equates information systems effectiveness with their success in Chapter 13 - Exploring the Role of Expectations in Defining Stakeholder's Evaluations of IS Quality' of 'Challenges of Information Technology Management in the 21st Century: 2000 Information Resources Management Association International Conference'. Of course success is a measure that is most dependent on the users' attitude. The author goes on to show the evolution of the meaning of the term "quality" from the close fit between product and specifications or production and standard process as shown by (Levitt 1972; Crosby 1979) to appropriateness for use (Juran et al 1974) to overall value delivered to the client (Cronin and Taylor 1992; Garvin 1988) and the final variation on the value

theme, meeting and exceeding customers' expectations (Gronroos 1983, 1990; Parasuraman et al 1984; Zeithaml et al 1990; Buzzell and Gale 1987). He notes interestingly that this progress was triggered by the economic shift from a production focused industries to a service focused economy, and more importantly a focus on information technology services.

His research is based on the equation that quality is equal to perceptions less expectations. Therefore to measure quality we have to find a way to measure the perception that the users have of a certain product that they use. The inclusion of the expectations in the formula is meant to account for the developer side of the equation while the inclusion of the perceptions is meant to account for the user side of the equation. The expectations are established through a series of contacts between the user and the developer that materialize in a variety of documents but more often than not in a specification document. To materialize the perception we have to track the users' opinions during their use of the product.

Although the study results are not conclusive as to when to measure perceptions and expectations it is clear that they are both important in determining final product quality.

2.1.3 User Culture

In this section we expose the importance of the internet in the user culture. In our tool we will explore the user exposure to the internet to understand the user culture. We review and we analyze the user experiences in using the internet. The goal for us it to highlight some of the cultural attributes of the web and the best practices in accessing and using the internet as a medium for communication and work.

The user culture is shaped by numerous factors, some traditional like particular history, language and national apartenance which dictates a different type of behaviour in the face of external stimuli such as applications designed by a different culture, some modern.

One such modern factor that is definitely shaping culture around the world, albeit in different measure, is the internet and its development and growth.

2.1.3.1 Human Factors and Web Development

Culture can have an impact on the success of a web site. Masaaki Kurosu identifies in 'Human Factors and Web Development Edited by Julie Ratner published by Lawrence Erlbaum Associates, London 2003 – A Cultural Comparison of Web Site Design from a Usability Engineering' two dimensions of culture its variety and its depth.

The cultural impact has a much larger role in ethnically diverse countries like Canada. It will have an increasing role globally as the economic globalization forces people from different countries to interact, often through web sites, to keep their competitive advantage by leveraging local know-how.

Usability engineering literature developed in the United States is much more readily available through out the world than lets say Japanese literature. The consequence of that is that most web sites that pay attention to usability follow U.S. guidelines which even though are valid can be improved upon if convergence with literature from other countries could be achieved.

2.1.3.2 The Culture Influence in Users' Attitudes

"The attitude towards computer related tasks, computer anxiety and spatial visualization ability (SVA) of a group of first year computer science students were measured just before their study commenced. The results presented in 'Challenges of Information Technology Management in the 21st Century: 2000 Information Resources Management Association International Conference Chapter 99: The Influence of Experience, Culture and Spatial Visualization Ability on Users' Attitudes and Anxiety Towards Computer Use' **by** Mehdi Khosrowpour were analyzed empirically based on two independent

variables, i.e. culture and computer experience. It was found that African and European users generally have the same attitude towards computer use. Users' attitudes improved after experience with computer related tasks. African students experienced significantly higher levels of computer anxiety than their European counterparts with the same amount of experience. It was also found that African users generally have a lower SVA than European users. Users with higher SVA generally have a better attitude towards working with computers and experience a lower level of computer anxiety."

It is obvious that culture influence users, the way they use software product and just as importantly the way they evaluate it. An organization that is not aware of the cultural makeup and diversity of their customer base is doomed to disappoint at least one segment of their target market. In the example above the cultural dimension that was studied was geographical but in today's globalized multi-cultural environment any other number of cultural factors can be the differentiator. We are all very aware that today's 20 years olds and the 50 years olds are exposed to very different cultures and we can infer that their perceptions as well as their expectations can be influenced by this difference.

"Users' attitude towards computer systems, the computer anxiety they experience and their self-confidence are deterministic of their performance with a system. System developers should understand users and their unique attributes in order to develop systems that will motivate users to use it and that will lead to reduced computer anxiety. The cultural diversity of users in South Africa should be taken into account when a system is to be developed that will be used by a mixed population. It was found that European users and African users show more or less the same attitude towards computer use. However, African users experience more computer anxiety than European users.(p445)"

There are global cultural factors, cultural forces that shape the perceptions of software users around the globe. Most of these cultural factors are rooted in North America and also interestingly to note most usability books are published in English another cultural component that can influence usability workers around the globe. Users come in contact

with these global shapers or with other relatively smaller cultural currents through the internet. Their internet accessing characteristics directly influence their cultural persona, such that if the user leaves in a country that censures the internet use he will only be aware of partial information. Similarly in a poor region with reduced bandwidth infrastructure, users will only be influenced by a certain cultural trend only later than some of their richer neighbours. The web is therefore a facet of culture and is important to be aware the users' exposure to it.

2.1.3.3 Cultural Diversity

Claude Ghaoui states in 'Cultural Diversity and Aspects of Human Machine Systems in Mainland China Encyclopedia of Human Computer Interaction ' that culture has a strong influence on the way we use software and therefore cultural considerations should be taken into account when designing software. Indeed, in the same way some of us prefer chopsticks to forks and yet some other people eat with their hands based on various culturally directed behaviour it is only predictable that we would use software differently based on our culture.

Culture shapes not only the way we use software but also the way we communicate our needs during design and our satisfaction with the final product. Software that will be used globally has to take into account these differences and implement an all encompassing approach.

Quoting Bourges-Waldegg (2000) the author says:

"...Design changes culture and at the same time is shaped by it. In the same way, globalization is a social phenomenon both influencing and influenced by design, and therefore by culture..., both globalization and technology have an effect on culture, and play a role in shaping them."

He shows here that users and technology shape each other, which is a concept advocated by many other authors but what is new here is that he recognizes the shift from a local user community to a global user community. He also recognizes more specifically than other authors that the mutual shaping is not constrained to one product, that technology in general shapes the user which in turn shapes each individual product. The technology that is the most far reaching these days and that has the potential to influence the most users is the internet, that is why it is important to study the role it plays in the life of the various user populations across the globe.

In the conclusion, the author states again:

"For researchers and developers working on advancing user-oriented design, one must realize that, in time of globalization, the culture-orientation is an essential component for successful usability and user friendliness. Culture is an influence factor on user interface design, and it is also an element of user-experiences. Engineers of products for the global market have to address this issue (see Röse, 2001)."

As the internet is now part of the global culture it is essential for us to understand the choices and limitations some users make and have when using the internet.

The study of the internet use as part of the user culture will provide us yet another important characteristic of our user, another step in our effort to develop a clear picture of who the user is.

2.1.4 Web usability

Web usability studies showed that perceived usability is more important than task success rate and our process was designed based on this finding, to capture the users' perceived vision of the system.

Web site usability has been treated differently than software usability. It had higher visibility due to the recent high profile in computer science of internet related research and higher visibility in financial circles due to the rise of e-commerce. Working practices devised for web usability, for increasing the efficiency of the user interaction with web sites, can and should at least partially be implemented in other applications.

2.1.4.1 Web Site Usability

'Web Site Usability – A designer's Guide' by Jared M. Spool is a study about the validity of the usability principles that are currently applied to web design. It measures the relevance of usability principles to the actual usability of a web site based on real measurements of user performance.

The study used 9 major corporate web sites for testing, web sites that are designed by entire design teams that follow usability and design guidelines as most of these companies bottom line depends on the success of their web site. The web sites tested are as presented in Table 2-2 in the book:

Site Name	Content
C|net www.cnet.com	Technology resources and information, product reviews, listings of where to buy products.
Disney www.disney.com	Games, videos, merchandise, Disneyland and Walt Disney World theme part information and reservation.
Edmund's www.edmunds.com	Car and truck prices, specifications, reviews, and other resources for vehicle buyers.
Fidelity www.fidelity.com	Fidelity mutual funds, personal and corporate investing opportunities.
Hewlett Packard www.hp.com	Product information, financial information, job opportunities.

Inc. www.inc.com	Small business resources, book reviews, articles, conferences, contact information for organizations.
Olympics (no longer available)	Schedules and results from the 1996 Olympic summer games in Atlanta, merchandise, tickets.
Travelocity www.travelocity.com	Airline tickets, car and hotel reservations, guide to recreational activities wordwide.
WebSaver www.websaver.com	Annuity information.

Table 2-2 Web Site Usability Study Web sites and Content

After numerous tests the results were unexpected with some of the sites that are geared toward mass audiences like c.net and Disneyland getting the worst results and all of the web sites coming in at under 50% usability.

The results of the study had several major implications as described and explained below:

1. Graphic design neither Helps nor Hurts: the graphic elements do not influence the user task success.
2. Text Links are Vital: due to downloading speeds, text links are viewed by the user before image links and are sometimes more explicit giving the user more confidence in predicting the type of information found if following a text link.
3. Navigation and Content are Inseparable: web sites that have some designers create the navigational elements separately than other designer that create the content guarantee that at one moment the navigation will become less and less relevant to the displayed content.
4. Information Retrieval is Different than Surfing: informational web sites need to be different than surfing web sites even though they are aimed at the same user.
5. Web Sites aren't Like Software: while in software usability testing there was a correlation between user success and usability there is no such correlation in web site usability testing.

One of the major pitfalls of web site design was the use of domain knowledge in the web site navigation. The use of business specific terms as navigational links, such as "Trip Segment" on travel web sites and "Daily NAV's" on investing web sites, only succeeded in getting the user stuck. As opposed to software users, web site users don't form mental models of web site structure, therefore different design models like "star" or "sequential" didn't make a difference in the success rate.

Frames, the web design tool, which is a "No No" in most usability books doesn't actually have an impact on user success. Navigational bars improve user success with top and bottom horizontal bars having more impact than vertical bars.

Links, on the other hand, are a good indicator of user success. Link characteristics such as predictability and differentiability make a difference between a user getting lost and actually finding the information they are looking for. Using the user's language and sometimes providing further explanations actually increased the user's confidence in the web site. Links embedded in sentences or wrapped links are negatively correlated to user success.

Sites searches proved to be for the most part ineffective as they didn't present explicitly the scope of the search actually being performed. Also, the search results were often inappropriate for the task the user was trying to perform and the result presentation order was random which doesn't bring any benefit to resolving a particular task.

Most sites proved rather inadequate to perform product comparisons which is rather surprising as all of the web sites studied are commercial web sites trying to sell products and services. When asked to compare two products or services, the users turned to inventive methods such as opening several browser windows, writing information on paper or printing the information. However, when they had to compare more than two products they gave up in frustration. These behaviours occurred if the web site had product comparison tools or not. These tools were largely ignored.

Interestingly the usability of a web site was influenced by the opinion that the user had about the company. When the WebSaver search returned a list of 5 annuities that were supposed to be the most appropriate for the user based on the search criteria provided, the user dismissed the results as unreliable explaining that the results probably display the 5 annuities the guarantee the bigger profit for the company. There is little that a web designer can do to change the opinion that the user has about the company that the web site is designed for.

Traditional web design wisdom says that more readable a web site is more usable it is. There are usability indices that have readability as an index of usability. In fact the opposite occurs. As the readability indices for the studied web sites decreased the following user's perception about a web site increased: authority, clarity, completeness, satisfaction and usefulness. The explanation comes from the fact that based on early web design principles, a user will use a web site the same way it will use the print materials when in fact this is wrong (another usability principle that nobody took the time to double check). The users are not reading web pages, they are rather scanning them for information. During this process, web sites with lower readability ratings, longer sentences, less connecting words, presented the user with more key words, meaningful words that helped him decide whether he was in the right place or not. Additionally the white spaces that were supposed to increase the usability of a web site created the need for the user to do more scrolling, to visit more pages before getting to the information that he was looking for, therefore decreasing his perception of the usability of the web site.

Other design principles derived from related fields such as newspaper layout or even web design experiences didn't fare much better. The principle that most content on a page must be above the fold (the place at which a user has to scroll) proved to be inconsequential as same interest was manifested by the user for information above or below the fold. Also the principle of proximity whereby buttons are presented to the user as close as possible to where he might be at the time he makes the decision to click the

button proved ineffective as users invariably scrolled to the bottom of the page to look for the submit, cancel or continue button. (button gravity).

The amount or type of graphics didn't have an impact on the usability of the web site either. Based on the current study all the talk about how graphics affect the page download time and consequently the usability of the web site appear to be false. The users inspected the text and the text links as it was downloaded first and then they skimmed the images once they were completely downloaded. Additionally if they saw an interesting text link they would jump to it before the images were loaded. Animation and movement proved to be more of an annoyance than a visual help with instances where users actually blocked an animation with their hands to be able to concentrate on the content.

In software usability testing users prefer the application that they are most successful with. This is not true with web sites. Below are the final results of the usability study as depicted in the book:

	User Success	Users Like Most	User Dislike Least
1	Edmund's	Travelocity	Tie: Edmund's, Olympic
2	Olympic	Olympic	
3	Hewlett Packard	Hewlett Packard	Travelocity
4	WebSaver	Tie: Edmund's, Disney (new)	Hewlett Packard
5	Travelocity		WebSaver
6	Inc.(new)	Disney(old)	Inc.(old)
7	Disney (new)	Inc.(new)	Disney (new)
8	Inc.(old)	C|net	Fidelity
9	C|net	WebSaver	Tie: C|net, Disney(old)
10	Fidelity	Tie: Fidelity, Inc.(old)	
11	Disney(old)		Inc.(new)

Table 2-3 Web Site Usability Study Results

As it can be seen there is no strong correlation between user success and usability. This finding can have a huge impact on usability research because it would mean that the way we've been measuring usability through user success is fundamentally flawed. But the question would be how then do we measure usability? It seems that the only way to find out what is the perceived usability of a web site by a user would be just that: ask the user and that is what USITRES will do.

2.1.4.2 E-commerce Usability

'E-commerce Usability – tools and techniques to perfect the on-line experience' by David Travis focuses on customers of e-commerce web sites. In the introduction it notices that "the only other industry that refers to its customers as "users" (and treats customers with equal contempt) is the drug industry."

To better define the usability of web sites a new concept is introduced together with screen design and consistency which in the author's opinion only count for 15% and 25% of the web site's usability. This new concept that accounts for the other 60% is task focus. The book concentrates on the description of this usability component.

"You know a system has task focus when you get a warm feeling that the person who designed this system knew what they were doing. You find you are able to use the system to do exactly what you want.[...] The crucial first step is to drive your design with a set of realistic tasks that real users are likely to carry out."

While the potential customers of a particular web site can be difficult to identify precisely the intended functionality of the web site can be easily identified by writing the specifications of the project. Consequently the design team most often concentrates on the technical specifications rather than on the client. Between the stakeholders of the project, the author identifies 15 major types as listed below:

- Clients or sponsors
- Customers or users (57 varieties)
- Shareholders or investors
- Testers
- Business analysts
- Technical support
- Legal experts
- System designers
- Documentation experts
- Marketing experts
- Competitors
- Technology experts
- Domain experts
- Regulatory bodies in the industry
- Representatives of trade associations

Their opinions will be weighted based on the number of stakeholders in each group and on their importance to the success of the project. For example the CEO, while alone, will have a higher importance as described in the importance matrix below (with the most important being 1 and the least important being 5) We will be using this concept to rank usability issues in USITRES, with the user importance being identified through the user profile:

Size				
	H	3	2	1
	M	4	3	2
	L	5	4	3
		L	M	H
		Importance		

Table 2-4 Opinion Importance Based on Group Size and Rank

Same matrix can be used to classify the relevance of user tasks based on importance and frequency.

The author also identifies the fact that in the beginning the web designer and the web user were "from the same social and intellectual block". The web was designed most of the time for the person sitting in the next desk since the term "next-desk design". The same mentality today can be lethal to the success of a web site.

This brings into focus the importance of market segmentation. While this activity obviously is meant at improving the bottom line, the economical success of a web site, it will also improve the reliability of the usability testing results. Deciding what type of users will use the web site and testing their use of the web site would provide much more meaningful results. It is also important to recognize that as the web site moves through its life cycle (from early market to main stream and to end of life) the target customer segment also changes its characteristics. If the testing of new web site can provide meaningful feedback from a group of technology enthusiasts, a redesign of web site with 1.000.000 registered users will benefit much more if the testing is conducted with more main stream type of users with no particular interest in the underlying technology but focused on the task they are trying to accomplish.

Another example of cross-domain confusion is the fact that the marketing experts (another group of our stake holders) identify the customer with the person making the purchase. This would only make sense in the traditional retail model but on the web, the 6 year old, above the average computer literate kid who surfs the Walt Disney and decides that he absolutely needs the latest super-hero figurine doesn't yet own a credit card. And the very influential teenager so preoccupied by his image when he is dropped off at school or at soccer practice, browsing the latest models from the most hip car manufacturer is far from having the means to buy by himself the next family car. They are not making the purchase but they can heavily influence it, therefore designing for the user and not for the customer makes sense in this case.

As described by the author one of the best methods of getting to know the customer is the interview. Interviews use a much smaller sample than surveys and have an almost 100% rate of response while mail surveys they typically only get under 10% and in that case there is a good probability that those 10% are professional survey takers. The interview also allows for extended comments from the users and follow-up questions from the interviewer. (both these features will be implemented in USITRES; we will implement an example of context specific question when one response will change the next questions; this gives the user the impression that his opinion will be taken into account, that this is not just another robot that he is talking to.)

An effort that is equal in importance to getting the to know the customers is presenting the customers to the design team in a manner that is clear and with all the properties needed to be leveraged in the final design clearly differentiated. For this purpose, in usability testing, personas have been traditionally used as stereotypical users. As described in the book personas usually have a picture for easier identification, a quotation that captures the customer's key objective, a short narrative describing the customer, a list of the customer's key goals and a status to describe this particular user importance to the overall project.

A quote that the author is particularly fond of is "Something for everybody equals everything for nobody." Additionally to user profiles, the author place equal importance on task profiles. One of the most effective examples in the book is the search feature that most web sites have. Consequently designers of new web sites will be tempted to use one because the code needed for implementation is readily available in the developer community and the space used on the page is very small and unobtrusive. This natural impulse is based on the false believe that the users want to search when in fact the users want to find and the search box is just another way of finding. If provided with more paths to the required information the user will have to spend time deciding which one to use. Additionally some paths might be longer than others and yet some others paths that seem to be leading to the desired result are in fact dead-ends. Because usually there is no relationship between the manner in which a search facility is presented to a user and the

actual path length to the needed information, the probability that the users will chose a path other than the optimal one increases with the number of search paths offered.

"Usability is a property that emerges when we design a product that sits in the triangle made by the customers, their tasks and the environment in which they use it".

Similarly to the way users are represented using personas, different tasks that will be performed on the site will be represented using scenarios.

The definition of usability as detailed in ISO 9241-11 is as follows:
"Extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.

Note the three components in this definition:

- Effectiveness: the accuracy and completeness with which the customers achieve specified goals;
- Efficiency: the accuracy and completeness of goals achieved in relation to resources;
- Satisfaction: freedom from discomfort and positive attitudes towards the use of the web site."

The author then goes on to list a number of criteria and metrics for the measurement of effectiveness and efficiency. He agrees that satisfaction is a rather soft criterion and that the only measurement that really counts is the client perceived level of satisfaction. We will argue that the same is true with effectiveness and efficiency especially on web sites that are not necessarily geared toward selling a product or a service but mostly toward brand recognition and brand promotion (sites that people surf instead of searching).

Another cross-domain inference that in my opinion and experience has the greatest negative impact is the innate obsession of project mangers with the time-to-market. This measurement is usually relevant to the personal performance of the project manager on

how he was able to estimate the correct development time and cost and then follow-up and push the designers and developers to deliver by that date. Of course in this mad dash to the finish line one of the first project activities to be axed is usability testing. From an organization point of view what is more important than time-to-market is the time-to-profit, the time it takes for a new or improved web site to deliver on its promises and user acceptance. In this case a web site that was released three months after the initially decided time-to-market might achieve the intended traffic of lets say 100.000 visits per day much sooner than a web site that was released without bothering to test the users opinion about it.

It is important that during usability testing we try to find errors and not fix them, otherwise usability testing sessions will change into design session. Also it is important to follow-up with the usability bugs discovered until they are eliminated, similarly to what is done with functionality bugs.

There are two major types of usability tests, with or without users. The methods that don't include users have usability experts performing heuristic evaluations and cognitive walkthroughs. Although these techniques are valuable when working on tight deadlines or when looking to find gross design errors they are in not way a valid replacement for the users. This is true as the experience and the knowledge to the experts tend to vary from one expert to another and they are forced by the nature of their investigation to make a lot of assumptions about the users that might be false in the end. As far as the usability testing with users is concerned it can be done against benchmark values or not. When is not done against benchmark values it is called customer talkthrough and the customer is simply asked to think out loud as he navigates through the site. In our opinion this is the most valuable of all the usability testing techniques as it is only the perception of the user about a web site that counts. In other words if the user finds the path to the needed information too long, even if it was only a click away, the usability designer must find a way to include the needed information on the home page where the user will have access to it immediately. Measures such as number of click, number of wrong choices and time to complete a task are not that valuable in this context. In USITRES we will

concentrate on the user focused measurements and not on the software focused measurements.

2.1.5 Remote Usability

In the literature reviewed bellow it is apparent that most effective usability studies are conducted in the user environment with the users having an active role in the study. This is the type of study that will be conducted with USITRES.

As explained in 'Remote Usability Evaluation At A Glance' by José C. Castillo "Much traditional user interface evaluation is conducted in usability laboratories, where a small number of selected users are directly observed by trained evaluators. However, as the network itself and the remote work setting have become intrinsic parts of usage patterns, evaluators often have limited access to representative users for usability evaluation in the laboratory and the users' work context is difficult or impossible to reproduce in a laboratory setting. These barriers to usability evaluation led to extending the concept of usability evaluation beyond the laboratory, typically using the network itself as a bridge to take interface evaluation to a broad range of users in their natural work settings."

In an age when even the usability evaluation, activity typically restrained to the laboratory, is performed over the network, directly within the user context, even becoming a part of the user context, studying the user context becomes even more important as before. The general IT user context is dominated and defined by web interaction and paradigms.

The paper presents most types of remote usability testing. It starts by presenting the various types of remote usability testing in 6 main categories as commercial, remote questionnaires and reviews, collaborative, video-supported, automated and user reported.

The commercial usability services are usability services offered by third party organizations through laboratory testing or software inspections. The remote

questionnaires are triggered by specific events in the application and have the advantage of storing the immediate reaction caused by software behaviour. Collaborative usability evaluation implies the concurrent implication of the user and developer in the evaluation process. This can be achieved through real-time application sharing, audio-conferencing, shared white-board applications and file transfer. The video-supported methods encompass all the setups where the users and the developers are connected through a video link being in a room across the hallway in a usability lab or across the continent through a web link. Instrumented or automated data collection involves transparent logging and sending of usage data by the software being used or by another software that is meant specifically for tracking, consolidating, evaluating and sending the usage data to the developers. Altogether, the author classifies all the remote usability techniques in the following types:

- User-reported critical incident method (UR)
- Remote questionnaire or survey (RQ)
- Instrumented or automated data collection (AC)
- Collaborative remote evaluation (CR)
- Video-conferencing-supported evaluation (VC)
- Third-party laboratory evaluation (3TE)
- Third-party usability inspection (3TI)
- Traditional lab-based usability evaluation (LB)

The author proceeds to classify all the testing methods. Based on the types of users involved the author distinguishes 4 categories: real users (UR, RQ, AC), representative users (none), both (CR, VC, LB and 3TE) and no users (3TI).

Based on time and user location the author categorizes the types on tests in 4 categories. When the user is tested in their own environment the methods can de classified based on time as asynchronous (when the user and the developer are logging and inspecting at different times)(UR, AC, RQ, CR), synchronous (CR, VC). The user can be tested in a controlled environment (3TE and LB) or the time or location of the user can be irrelevant

(3TI). Out of the 4 categories the one that can have the most impact on usability testing was the one in which the users were tested in their own environment the results were inspected asynchronously, because this category gives the possibility of consolidating the results and of the largest costs savings possible.

Based on the person who identifies the critical incidents the testing methods are split into 3 categories. The user might be the one who identifies the incident (UR, RQ), the evaluator can identify it (3TI, AC), or they can both identify the incident at the same time (3TE, CR, VS, LB).
Based on the level of interaction between the users and evaluators on one side and the type of tasks being performed during the tests as real or representative the author classifies the types of testing in 3 categories. One category involves methods where the tasks are real and there is no interaction between the user and evaluators (UR, AC, RQ), another category where the tasks are representative and there is significant interaction between the user and the evaluator (3TE, CR, VS, LB) while the last category is the one where the type of tasks or the level of interaction is not of interest (3TI).

Based on the types of data collected, the winner is collaborative remote evaluation (CR) with 6 out of 8 types of data being collected. However no information is provided as to how relevant this types of data are to finding and fixing usability problems. A weighted graph might have produced different results.

Based on the quantity of data collected versus the amount of equipment needed the clear winner is the automated collection with the less equipment required to collect the most data while video-conferencing-supported evaluation collects a lot of data but it also requires a lot of equipment. Remote questionnaires use the least equipment but they also traditionally support a reduced number of questions in order to reduce the impact on the main user tasks.

Based on the relative cost to collect versus the relative cost to analyze the data the remote questionnaires are the definite winners while at the other side of the spectrum we find grouped together the traditional collection methods (VS, CR, 3TE, LB).

Based on the quality of the data relative to the total cost of data (to collect and analyze), the best ratio is produced again by remote surveys even though it is limited to relative low quality of data. The second best ratio is provided by user-reported critical incident method that also produces the second best quality of data.

Throughout these classifications the third-party inspections appear to be the most out of touch with the users, their environment and the types of tasks they perform.
USITRES would be part of the user-reported critical incident method and remote questionnaire or survey types. Especially based on the analysis of the quality of the collected data, the combination of the remote questionnaire with the user-reported critical incident method is the most efficient way to perform usability testing. In USITRES the interaction is always initiated by the user that reports an incident and proceeds with the user filling out the questionnaire that would help the evaluator and the developer classify and fix the problem.

2.2 A Survey of Existing Tools and their Limitations

Most enterprise strength usability tools cover most aspects of usability testing, but some are stronger in some area of the domain. We will start by presenting tools that are used by usability professionals and are focused on usability *text format data collection* from the observer or directly form the user, followed by tools that are primarily used to collect usage data in *video format* and then by some that *automatically capture logs* of user interaction. Some other tools don't have particular strength but try to *cover all aspects* equally efficiently.

2.2.1 Text Format Data Collection

These tools collect data in text format directly from the user or from the observer based on user actions. They are easy to set up and effective

- Noldus Observer: The Observer is a modular software that can be used for the data collection, analysis, presentation and management. You can buy and use only the module that is needed for some particular research. You can also synchronize the recorded data with physiological data collected through other means using their external data module. http://www.noldus.com/products/index.html?observer/index
- Usability test data logger V4: uses modified and customizable excel spreadsheets to log and analyze usability data. It is a free tool that can work on any PC platform that support Microsoft Excel. Upon completing the data entry which is a manual process, independent of the actual test, it can even generate usability reports. http://www.userfocus.co.uk/resources/datalogger.html

2.2.2 Video Capture

These tool record the actual user – computer interaction in a video format so that it can be replayed at a later time for the observer to complete his recordings or by another stakeholder interested in seeing an example of a user interacting with the software product, without having this stakeholder being present at the testing site at the time of the test. They need additional video recording equipment and their results are opened to the interpretation of the experts watching the video recording.

- OCS from Triangle Research Collaborative: Observational Coding System or (OCS) aids in the collection of visual data including digital video data

and transforms visual data into information through statistical analysis and reporting. http://www.trctech.com/ocs.htm

- Camtasia Studio: Camtasia Studio is a video screen-recording program that records any action on your Windows desktop. The tool can be used to remotely observe user actions as well as to deliver intended method of use presentations and other types of demonstrations. http://www.techsmith.com/

2.2.3 Automatic Data Collection

These tools are often installed on the users' computers, as opposed to other products that are installed only at test sites or on the observers computer. They operate in a stealth mode, that is transparent to the user, and they record, often in proprietary formats, the user keystrokes, mouse movement and other types of interaction with an application. They are the less disruptive type of tool but also the most impersonal with absolute no type of feedback from either the user or observer.

- WebVIP & FLUD: Web Variable Instrumenter Program (WebVIP) - instruments a website to capture a log of user interaction. http://zing.ncsl.nist.gov/WebTools/WebVIP/overview.html_Framework for Logging Usability Data (FLUD) - a file format and parser for representation of user interaction logs. The only tools that record and read FLUD format are however until now the tools developed by NIST. http://zing.ncsl.nist.gov/WebTools/FLUD/overview.html
- Keystroke recording: Keylogger Pro - Stealth Keystroke Recording Software http://www.1spysoftware.com/keylogger-pro.shtml_SpyBuddy Stealth Keystroke Recording Software which includes web based logs that can be viewed from another computer. http://www.exploreanywhere.com/

2.2.4 Overall Process Tracking

These are product suits that include various components that are used at various points in the usability testing process from planning, to user selection, to the actual test and finally for reporting the results. They are very time consuming, even if it only to learn all their features, and necessitate a very complex setup, making them a tool that is exclusively used in organization with extensive resources.

- UsabilityWare 4.8: UsabilityWare™ 4.8 is a tool that covers most of the usability testing needs: define, recruit and track test respondents, record testing data (high definition (HD) hard disk recording, real-time recording of up to 3 signals, immediate retrieval media & text assets, live streaming for remote viewing & testing, instant playback and editing of media files, improved step-by-step usability testing process), analyze, present and report the results. http://www.usabilitysystems.com/prod_usability_software.html

These usability tools track speed of hand movements, record facial expressions track eye movements and other external indicators of user emotions and satisfaction with the application that they are using. Although the measurements can be very exact their interpretation is highly subjective and I would say error prone. USITRES eliminates all the guess work by asking the users directly of how they feel about the tool that they are using because anyways this is the only important measure of the usability of a web site, the PERCEIVED degree of usability.

Additionally, instead of having several usability professionals guiding and measuring the responses of one user in the lab we replace this situation with one where the user himself will conduct the usability study. Maybe each user will not test each advanced functionality individually but he/she will test the functionality that he/she will be most interested in. At the same time we will know the relevance of different functionalities to

different groups of users allowing the programmers to devote more of their time to the implementation of the user recommended changes to the most popular features. The reduced cost and increased focus of this approach will greatly influence the perceived productivity of the programmers.

All the presented tools have a limited user base because they have to be purchased and installed. The user base is somewhat artificially determined by the users' financial and technological capabilities. An all inclusive usability tool is needed to reach the entire user base without any additional barriers. USITRES can be used by all users with access to a web browser enabling mass usability and with it a usability revolution. The multitude of existing tools stores usability data in a multitude of repositories and formats. USITRES will be the centralized repository where the currently fragmented data can be gathered and used.

2.3 USITRES requirements and overview

We will present bellow the interactions between system users, technical components and different processes implemented according to rules that were developed during the creation of this document. Together they represent the USITRES system architecture:

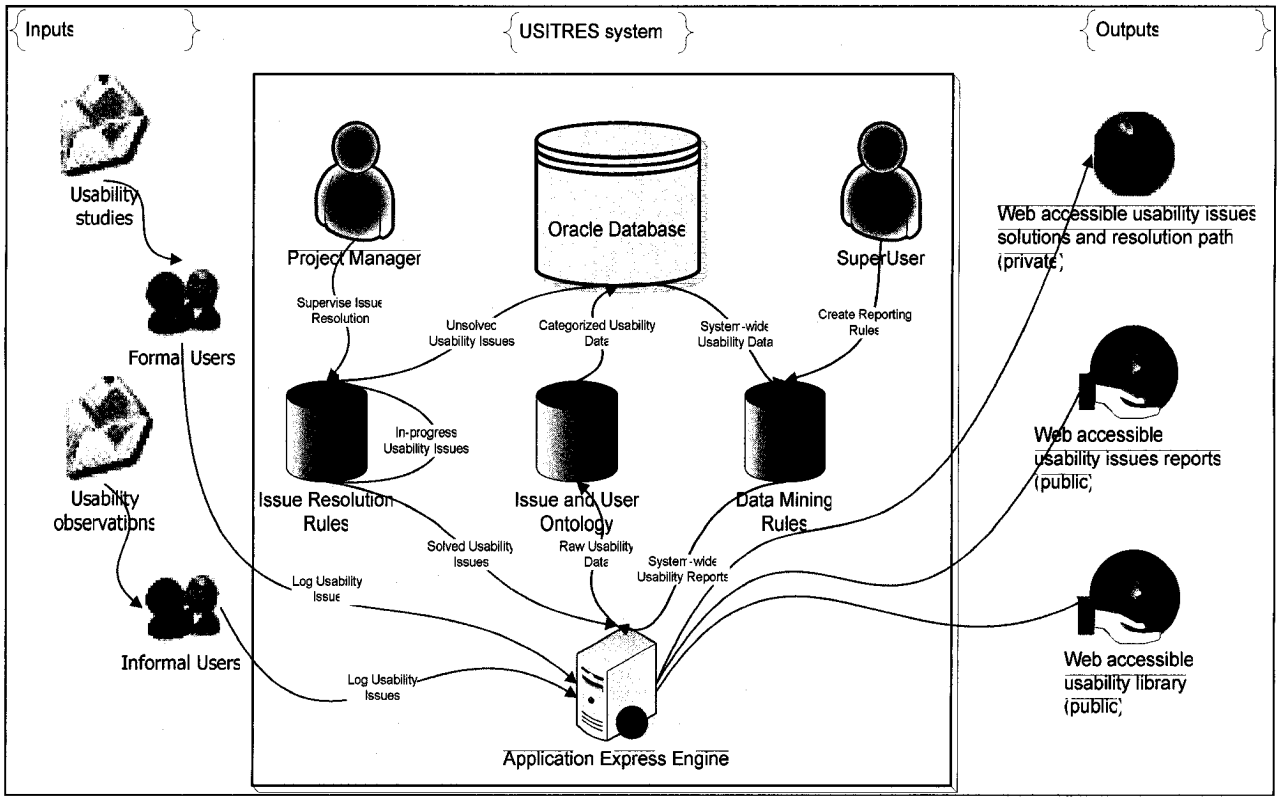

Figure 2-3 USITRES Requirements Overview

USITRES' inputs are formal usability studies conducted by different entities and logged by testers, formal users of the system. Other inputs are usability observations logged by casual users. These inputs are logged into the system through web pages constructed and served through the Application Express Engine (HTMLdb). The options offered to the user while logging the issues expose the relevant characteristics of the issue and of the user, reveal the importance of the issue to the user and, in the process, apply the USITRES ontology rules on the issue and the user. The categorized usability data is stored in the Oracle database. The usability issue then proceed to be resolved by going through the issue resolution rules under the supervision of a project manager. Through out the iterations of the issue resolution process the issue is made available to the tester and the developer through web pages served by the Application Express Engine. The entire set of issue data and user data will be corroborated in usability reports based on data mining rules that will be created by the system super user. These reports will be made available to the entire usability community in an effort to improve the quality of the

products developed by the industry. Additionally a library of usability white papers and other types of documentation are stored in USITRES and made available on the web to the usability community. To summarize, the outputs are solved usability issues that are only be accessible to the stakeholders of the particular system being studied, usability reports based on all data collected in USITRES that are accessible to the whole community as well as public usability documents that are stored in this usability database. All these outputs are meant to improve the quality of the products delivered by the usability industry.

All the usability rules proposed by the experts although they make sense in the lab are largely ineffective when tested by a large audience as showed by the usability study.

The customer segment market is both hard to identify and hard to follow as their preferences and needs evolve through time. Additionally, web sites and applications must change to appeal to new customers segments as they move through their life cycle. These are two very important entities in continuous evolution and to find their convergence point is only possible through mass usability testing with the various results being used differently based on customer profile the immediate task at hand.

The instances where principles from traditional retail are being applied to e-commerce are numerous. What the project managers implementing these principles fail to understand is that e-commerce and traditional retail are fundamentally different. Not only are they different but the users of e-commerce, our target customer, has its whole outlook about commerce influenced by the huge opportunities offered by the web and in turn will have its opinion about traditional commerce changed. Therefore, quite the opposite should happen, with successful web sites being analyzed and with the keys of their success implemented in traditional commerce.

Although the scientific community by enlarge and the IT industry (in a smaller measure) recognize the importance of the user in the design process they both claim in one way or another that they know better than the user. The proof of this claim is that even though

they take note of user opinion they take the decisions in which to structure and apply this opinion.

Similarly to the Oracle approach described by Luke Kowalski, USITRES is a one stop usability solution. This will entice more and more companies buried under the clutter of usability applications built in-house, freeware open-code solutions that are not supported, out-of-the box un-customizable bug tracking applications that don't properly address usability problems to adopt proper usability testing and tracking techniques. The fact that it is web based greatly increases its reach and it eliminates completely set-up and maintenance costs. The fact that it is itself subject to continuous usability testing by the user community reinforces the user confidence in the design principles that are be used on our web site. The success of USITRES could be connected to an increase rate in the adoption of usability principles.

Additionally, the increased involvement of the end users in the final product will increase their sentiment of apartenance to the development team, their pride in contributing to the development of a successful product, their rate of adoption and use of more of the implemented features, and therefore the overall success of the web site.

USITRES allows an improved application of traditional usability testing techniques through its improved reach but also through its customization (the fact that it is continuously under user testing) allows the user to design and use new usability design and testing procedures.

The users are allowed to influence only the usability testing procedures but because usability testing is a continuous process that has to be repeated at each step through the development process they control the development process. Although the IT companies might see this as a loss of control over their development process, the current state of the IT industry with most of its technology in stagnating pace, the never ending series of patches that follow software releases by even the largest and most prestigious software companies, this might be a good thing. The democratization of usability might bring over

anarchy in the development process but we can be sure that the order that will emerge from the initial anarchy can be an improvement over the actual state (we don't see how it can be worse).

3 Chapter 3 - Ontology

In USITRES we will be tracking a variety of usability issues from a variety of application types and various types of users. Our immediate goal is to facilitate the solving of these issues by providing a communication platform for all the interested parties in the form of the logging system. Our ultimate goal is to be able at the project planning stage to infer the types of possible issues that might arise with a particular type of application that targets a particular type of user. We will achieve this by studying all the previously logged issues for the same type of application by the same type of user. To increase the accuracy of these predictions we have to be able classify very precisely our users, issues with all their components (users, context, task and applications) as well as our solutions.

3.1 The need for an ontology

Our effort is meant to produce an effective communication platform for the field of usability testing. The application provides the usability testing professionals and the application users the common forum to communicate. The database backend supports the aggregation of data for reporting purposes. We developed an ontology for usability issues to create a common language that can be used for communicating, grouping and reporting. The main justifications for developing an ontology are:

Firstly, below you'll find an illustration of the current state of information fragmentation in the usability field. If different labs use different tools we can see the information fragmentation that results. One of the obstacles in face of centralizing & consolidating the knowledge and also in face of extracting additional intelligence from it is a lack of a common knowledge vocabulary. We provide this common vocabulary through the usability knowledge ontology that we developed and implemented. We will take the existing fragmented information and organize it and connect it to USITRES information through our ontology.

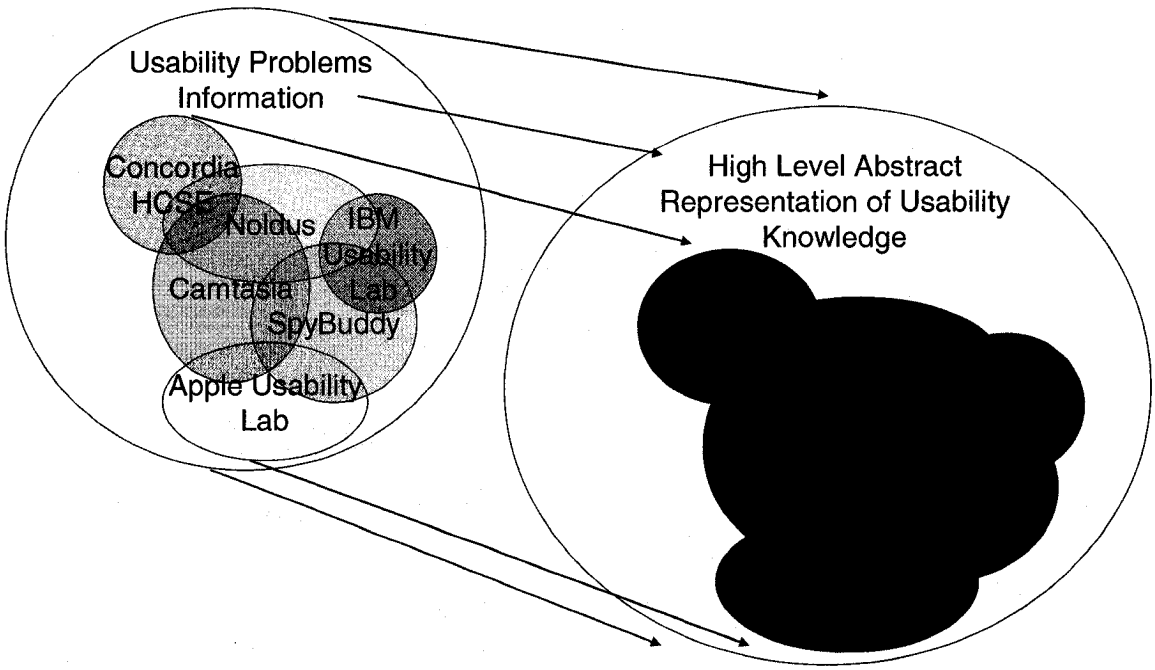

Figure 3-1 Usability Information Consolidation

Secondly, as illustrated bellow, there is a significant difference between the current expert centric situation and the user centric flow of managing usability problems. The illustration presents the following situation: a usability study is conducted by an usability express that selects a supposedly representative set of users. The user message of the fact that they don't like the current green user interface is conveyed to the project manager as a requirement to change the user interface colour. In turn this is translated to the developer as a specification to make a blue user interface. As we can see this would displease another set of users. If the users would have been in direct contact with the developer, he would have been able to take the correct decision to implement a black and white user interface. Our ontology allows for the proper identification of the problems to the developer and at the same time allow for the bypassing of the experts' interpretation of user problems.

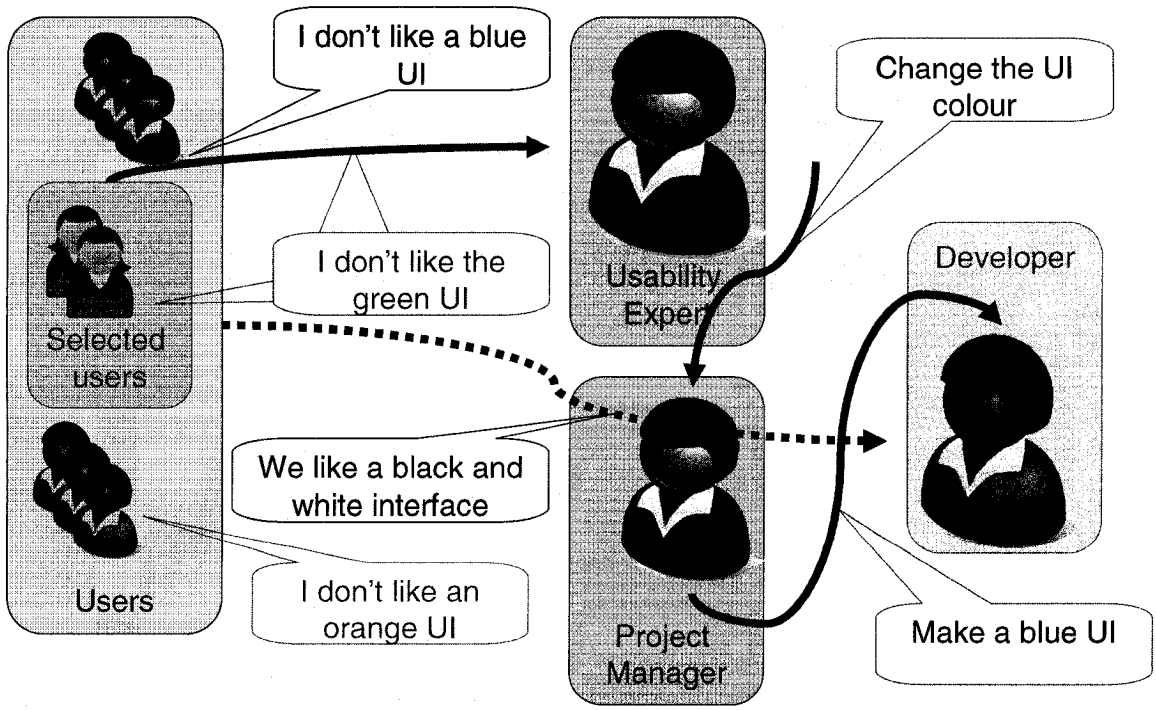

Figure 3-2 Expert Driven and User Driven Information Flow

Thirdly, our methodology allows for continuous usability by removing the need for the expert involvement, and therefore the limitations created by the limits of the expert resource.

3.1.1 Ontology classification

There are several ways to classify ontologies and some of these classifications are presented by Dieter E. Jenz (2003). He divides an all encompassing ontology in three parts, the core business ontology, the industry specific ontology and the organization specific ontology. In traditional sciences such as medicine or in highly regulated industries such as banking and insurance the core and industry ontology are meant to cover up to 95% percent of the whole domain. In a much more recently established domain such as usability and usability testing, that is only loosely regulated through high level standards, and whose development is driven by the user community as well as by the domain experts, the responsibility for developing ontologies is taken upon by the organization implementing usability testing and developing usability testing applications

such as USITRES. We are presenting an organization specific ontology in the remaining part of this chapter.

3.1.2 Ontology definitions

Another way, other than ontology classification, to get a better understanding of what an ontology is and how is it important to a domain and organization is to look at some of the definitions that are given such as the one presented by David G. Schwartz (2006).

"While never pretending to duplicate exactly the workings of the human imagination or experience, ontologies attempt to capture conceptually the rational building blocks of the mind by modeling our knowledge of reality. The whole purpose of this is to give the computer humanlike, albeit modest, thinking ability, by providing an explicit vocabulary for things, ideas, actions, relations, and approved behaviours.[...]An ontology comprises the explicitly articulated and shared concepts of a knowledge community or domain. These concepts are arranged formally in a taxonomy and are governed by specifically defined rules and axioms."

David Schwartz classifies the ontologies as formal and informal. The formal ontologies represent complete but simplified versions of a domain world and they attempt to create domain rationalization models. These models will support various decision systems, mostly computerized, be developing a rigid classification of the domain entities as well as relationships between the classes such as "is-a" and "part-of". Together the classification and the relationship form the vocabulary and the grammar used in that particular domain to formalize the communication and the decision taking process.

The informal ontology is a classification of entities in a particular domain and the USITRES ontology is an informal ontology. Based on the presence of hierarchy they are called taxonomies, directories, subject heading lists, or thesauri. If they include definitions they are called lexicons or glossaries. In the case they are just lists that include a definite list of terms that might be used they are called "controlled vocabularies" or

"synonym rings". Our ontology will be a taxonomy because each question belongs to one of the "Users", "Context", "Tasks" or "Applications" classification categories which all together represent all the facets of an issue. The measuring dimensions will be used to grade the other groupings.

The current release of USITRES implements a knowledge repository. In the third release we will implement a more formal ontology together with containment rules when implementing the usability issues forecasting feature.

The ontology solves two problems. It creates a common language between the users and the usability professionals. It alse provides the structure needed by the IT system supporting the communication to store all the communication instances and to create the final reports that will detail and visualize the overall project or system wide usability issues trends.

There is no one generally accepted definition of an ontology. Yet another definition is presented by Laura C. Rivero (2006):

"According to its modern computer-science technical meaning (Gruber, 1993), a consensus definition says that, "Ontologies represent a formal and explicit specification of a shared conceptualization," (p. 199) where:

Conceptualization refers to an abstract model of some phenomenon/situation in the world, where the model results by the identification of the relevant concepts that characterize this particular phenomenon/situation. To avoid any hype, "concepts" can be simply understood here as the discrete, important notions that must be necessarily utilized to describe the phenomenon/situation under consideration.

Explicit means that the type of concepts used and the constraints on their use are explicitly defined.

Formal refers to the fact that the ontology should be machine-usable.

Shared reflects the notion that an ontology captures consensual knowledge, that is, this knowledge is not private to some individual but must be accepted by a group."

All these criteria are met by the ontology used by USITRES. By identifying all the concepts that characterize an usability issue through the questions that we ask the user about the issue, himself, the context of use we achieve the conceptualization requirement. By asking the system user these questions without inferring any of the results we satisfy the explicit requirement. The direct mapping between the questions asked and the results stored in the database demonstrate that the ontology is formal enough to be reliably used in a computerized system. The presentation of these criteria to the users, together with the system help files and documentation and with the thesis describing the USITRES ontology satisfies the shared requirement, therefore satisfying all the requirements to be designated as an ontology.

3.2 An overview of the proposed ontology

In this section we will present the ontology that will be implemented in USITRES. Bellow you will find the ontology map that shows the classifications and the relationship between them.

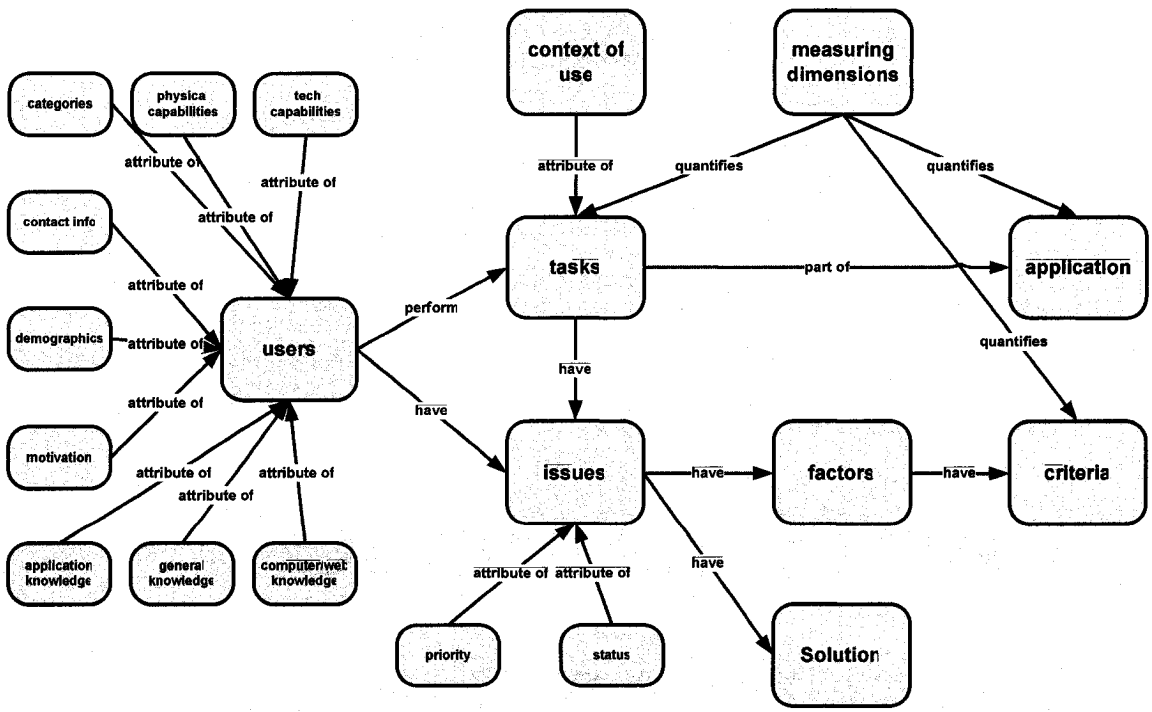

Figure 3-3 Ontology Map

Our ontology was developed by extracting the most important features from various usability forms and tools as well as general issue tracking tools.
In our ontology, users perform tasks which are part of an application, as each task is a materialization of the multitude of possible tasks that an application can perform. The tasks are performed in a context of use and together with the applications are measured in their relationship to the user.

Users have usability issues with the tasks performed. The issues have usability factors which in their turn have usability criteria. Issues also have solutions and their attributes are a priority rating and a status.

USITRES is user centric and the vast majority of attributes tracked are relative to the user. The types of user attributes tracked are: user categories, physical capabilities, technological capabilities, contact information, demographics, motivation, application knowledge, general knowledge and computer and web knowledge.

3.2.1 Users

In describing the user we will start by acquiring characteristics about his on the job persona in **user category identifiers**. We will then proceed to learn about his off the job persona by asking a series of **demographic questions**. We will explore his **attitude and motivation,** his **general knowledge level** as well as the **application and domain specific experience and knowledge**. We will then ask a series of questions that will help us understand some external pressures on his applications usage. By asking **internet usage questions** we will determine his or hers capability of finding additional help information as well as an important facet of his cultural make-up. We will then be asking **technological capability questions** to learn if certain application responses might be influenced by the hardware that the he or she is using. Various **physical characteristics** can also alter his interaction with the application. Finally we will be asking the usual

contact questions to be able to further communicate with the user. This complete user portrait will allow us to understand differences in response for a particular application from various user groups. It will also tell us who the application user really is, validating or contradicting the initial assumptions of the designers about who will be using the application or feature being studied. The questions presented bellow are the actual questions that will be asked in the application and the available answers will also be the ones that the user can choose from within USITRES:

3.2.1.1 User Category Identifiers

With this series of questions we will determine the user's job role within the company.

- Q: What is the job title that best describes your current job:
  - Clerical.
  - Operational.
  - Supervisor.
  - Manager.
  - Other (please describe).
- Q: In which geographic area is your main office located?
  - Specify country and province.
- A new feature is likely to equally impact all the employees with a similar job description. At the same time we need to estimate the knowledge base available through these similar employees that the user can tap into. Q: Please estimate how many people in your job title are working in your geographic area :
  - Number of people with the same job title.
- We need to estimate what percentage of the job tasks are accomplished manually or with a computer. Users with a higher exposure to automatic tasks will have an easier time in adopting another automatic task. Q: Describe the current level of automation of your job title in your office by checking one choice below:

- None (No users in my job title have or use a computer workstation).
- Low (All users in my job title who use the computer share a workstation with other users).
- Medium (Some users in my job title who use the computer share a workstation with other users, but some have their own workstations).
- High (All users in my job title have their own workstations).

- Businesses that have special characteristics should be treated apart from all the others. Q: Special business characteristics:
  - Minority-owned business.
  - 8A business(certified for delivering to the US government).
  - Woman-owned business.
  - Equal Employment Opportunity Company (EEOC).

3.2.1.2 Demographic Questions

- A higher income might account to a lower sensitivity to optional feature prices as well as an increased sensitivity to quality as these higher earners are accustomed to excellent quality in all products and services that they acquire. This is especially true of McIntosh product users who are willing to pay a higher price for better quality. Q: Which of the following categories includes your household's annual income?
  - Drop-down: $20,000, $20,001–$29,999, $30,000–$39,999, $40,000–$49,999, $50,000–$59,999, $60,000–$69,999, $70,000–$79,999, $80,000–$99,999, $100,000–$119,999, $120,000–$149,999, $150,000
- A full-time employee is more likely to experience a higher commitment to learn and use a new application then a part-time employee who is unsure about how long he or she will have to use this new application. Self-employed users might experience even higher commitment levels then their full-time counter parts. Q: What is your current employment status?
  - Employed full-time
  - Employed part-time

- Not employed
- Self-employed

- Q: What is your company or organization's primary product or service?
  - Describe product or service

3.2.1.3 Attitude and Motivation

- Overall attitude towards computers can have a deciding effect on the success of a new application. A person that intrinsically dislikes working with computer will not change his attitude towards a new applications unless this application is truly revolutionary in the way it interacts with the users. Q: In general, how do you feel about working with computers?
  - I don't like working with computers.
  - I have no strong like or dislike for working with computers.
  - I am interested in computers but only as a means to help me do my job better and faster.
  - I like working with computers.
- Users that have been positively impacted by the introduction of computers or more computerized tasks in their jobs will be more likely to adopt another new application or to better interact with existing applications. Q: How have computers affected your job?
  - Computers have made my job easier.
  - Computers have not affected my job in any particular way.
  - Computers have made my job more difficult.
- IT training can be perceived in very different ways based on its perceived payback value. Some employees might regard IT training as a mere interruption in their daily tasks, while others might see it as essential to their on the job performance. Q: Is the amount of time it takes to learn new software applications usually worth it?
  - Yes, it pays off because computer systems usually help me do my job better or faster.

- Sometimes it pays off, and sometimes it doesn't.
- No, computer systems are usually not useful enough to justify the training time.

3.2.1.4 General Knowledge

- Typing skill can influence the user proficiency differently in command line based applications and in visual interface based applications. Q: What is your level of typing skill (Using the 0-3 experience scale) ?
  - None (0).
  - "Hunt and peck" typist (less than 15 words per minute) (1).
  - Moderately skilled touch typist (between 15 and 50 words per minute) (2).
  - Highly skilled touch typist (greater than 50 words per minute) (3).
- A higher academic degree might denote a higher tolerance for learning activities. Q: What is your highest academic degree?
  - No degrees.
  - High school degree.
  - Trade or vocational school degree (beyond the high school level).
  - College degree (for example, B.A., B.S., Associate College degree).
  - Graduate degree (for example, M.A., M.S., Ph.D., Ed.D., M.D., R.N.).
- Novice and intermediate users are usually more likely to quickly adopt new applications as they are generally expected to spend a considerable amount of their time learning. On the other hand experienced users are expected to spend a considerable amount of their time teaching others, therefore having less time to explore and learn new applications. Q: How would you describe your experience level in your current job title (Using the 0-3 experience scale) ?
  - Novice (less than 1 year) (0).
  - Intermediate (1–3 years) (1).
  - Experienced (3-5 years) (2).
  - Expert (more than 5 years) (3).

- Most documentation is written in English and only then translated in other languages. Overall English proficiency directly influences the degree to which the users can understand the original product documentation. Q: What is your native language?
    - English
    - Spanish
    - French
    - Other (please name)
- Although major software providers have local support centers throughout the world, smaller companies only have one support center where service is usually provided in the language of the country where the company is headquartered and in English. Spoken English proficiency will definitely impact a user's capability of taking advantage of the support offered. Q: If your native language is not English, how well do you speak English (leave blank if English is your native language) (Using the 0-3 experience scale)?
    - Poorly (I have trouble communicating with English speakers.) (1).
    - Adequately (I speak well enough to get around.) (2).
    - Fluently (I speak almost as well as a native speaker.) (3).
    - Other (please describe) (0).
- Q: If your native language is not English, how well do you read English (leave blank if English is your native language) (Using the 0-3 experience scale)?
    - Poorly (I have trouble reading documents in English.) (1).
    - Adequately (I read well enough to get around.) (2).
    - Fluently (I read almost as well as a native speaker.) (3).
    - Other (please describe) (0).
- Q: How would you describe your general level of computer experience (Using the 0-3 experience scale)?
    - None (I have never used any software applications.) (0)
    - Low (I have used only one or two software applications.) (1)
    - Moderately low (I have learned and used multiple software applications but have no programming skills.) (2)

- High (I have used many different software applications and have some programming skills.) (3)

3.2.1.5 Application and Domain Specific Experience and Knowledge

- Trained users are expected to be able to adopt new software much quicker then others, therefore if this is contradicted by usability testing results the training program might have to change or the application interface might have to be changed. Q: Have you participated in any application training programs?
  - No.
  - Yes.
- Experienced users should already be familiar with application artefacts and more importantly with the help system. They are expected to absorb new information at a different rate than novice users. (Traditional forms present the user with drop-downs with various intervals to allow then to input their experience level. This type of input is reminiscent of oversimplified paper questionnaires and will be replaced with text boxes where the users will write the approximate number of months or years of elapse and effort. This will greatly increase the exactitude of our data as we will go from answers of the type: "Between 1 and 3 years" to "Approximately 1 year and 6 months of elapse and 6 months of effort." This new type of control will be used throughout USITRES wherever appropriate.) Q: How long have you been working with the application?
  - Months/Years.
- Experience acquired while working with other similar applications can still count as experience. Q: Before beginning to use this application, how much experience did you have working with any other similar applications?
  - Months/Years.
- Q: If any prior experience, on which application(s)?
  - Application name.

- An increased exposure to programming languages can increase the likelihood that users might understand underlining technology problems that cause interface usability problems. This can lead to various behaviours such as ignoring misleading error messages due to an understanding of the root cause of the problem. Some of the more experienced users might also show increased levels of feedback to the developers due to their personal interest in programming. Q: How much experience with and exposure to technical language and methodologies have you had (e.g., math, science, engineering, programming) (Using the 0-3 experience scale)?
    - None above high school level (0)
    - Less than four classes at college level (1)
    - Four or more classes at college level (2)
    - Graduate training or equivalent work experience (3)

3.2.1.6 Computer/Internet Use Questions

Computer usage experience is one of the major factors influencing the adoption rate of new applications. Experienced users will have far less issues adopting a new application while novice users will have to spend time adjusting to all computer use paradigms in addition to the particular application being studied.

- Q: How long have you been using computers?
    - Months/Years
- Q: How often do you use a computer?
    - Once a month
    - Once a week
    - Every day
    - Several times a day or most of the day
- Q: Where is the computer located that you use most frequently?
    - Home
    - Work
    - School

- Public computer lab
- Wherever you are (laptop/PDA)

- We have to evaluate the user's ability with email to timely respond or take note of information sent by email from USITRES. Q: How long have you been using email?
  - Months/Years
- One of the most important facets of the user profile is web proficiency. The internet contains a lot of resources, and more time he spent on the web, more likely that he is aware of this wealth of resources, more likely that he or she will look and find for answers there instead of by contacting the support center. Therefore web experienced users will report on average less bugs then other users as they will find solutions and workarounds by themselves rather than from customer support or by logging an USITRES issue. Q: How long have you been using the World Wide Web?
  - Months/Years
- On average, how long do you spend on the Web in a given session?
  - Minutes/Hours

3.2.1.7 Technological Capability Questions

These details are important for both determining the type of user we are dealing with as well as determining if the hardware might have any influence in creating a usability issue.

- Q: What operating system do you use most frequently?
  - Windows
  - Macintosh
  - Linux
  - Other
  - Don't know
- Q: What is the CPU speed of your primary computer?
  - Less than 1000MHz

    - 1000–2000MHz
    - More than 2000MHz
    - Other
    - Don't know
- Q: What is the resolution of your primary monitor?
    - 800 by 600 pixels or less
    - 1024 by 768 pixels
    - 1600 by 1200 pixels or more
    - Other
    - Don't know
- Q: What is the speed of your connection to the Internet?
    - 56K or less
    - DSL or cable modem (144K–1.5M)
    - T1 or T3
    - Other
    - Don't know
- Some application features and their associated usability issues might only be apparent on a certain type of computer due to hardware particularities. Q: Who is the manufacturer of the computer (Dell, HP, Compaq, IBM)?
    - Manufacturer name
- Q: What is the model name of your computer (Armada, Latitude etc.)?
- Q: What is the CPU brand (AMD, Intel)?
- Q: What is the CPU model name (Sempron, Pentium, Athlon)?
- Q: What is the memory size (in Gigabytes)?
- Q: What is the memory brand?
- Q: What is the hard disk size (in Gigabytes)?
- Q: What is the hard disk maker (Maxtor, etc.)?
- With the new technologies, PC with more than 1 CPU will become more and more frequent and we have to account for that. Q: How many CPU/cores does your computer have?

3.2.1.8 Physical Characteristics

- It is important to measure the differences in response between our male and female users. Even in the best cases where the product was designed with both genders in mind, previous versions of the product or competing versions of the product might differentiate between male and female users. This might have an influence on how the different users receive and use the product. Q: Are you :
    - Male
    - Female
- For graphically intensive applications it is important to determine and account for user dexterity. Based on this information different interfaces might be developed for different users to minimize mouse hand movements. Q: Are you:
    - Right-handed
    - Left-handed
    - Ambidextrous (equally coordinated with both hands)
- Many applications use coloured coded buttons and error and warning messages and signs. Knowing which percentage of their users is incapable of decoding these messages might determine the designers to change their approach. Q: Are you color blind in any way?
    - No.
    - Yes (please describe).
- Age can definitely influence the way users interact with various applications. Solitaire software and a music editing software will use significantly different interactions patterns. Q: How old are you?
    - Birth year.
- If a large percentage of the users have visual impairments, the designer might consider interface changes such as using larger icons and fonts. Q: Do you wear glasses or contact lenses?
    - No

- Yes (Please describe your vision problem and correction method, for example, nearsighted, farsighted; bifocals, contact lenses).

- Special customization can be designed for other disabled users, such as super sensitive mouse movement to decrease the need for movement from arthritis patients or increased sound levels for users with reduced audio sensitivity. Q: Do you have any physical handicaps other than vision deficiencies that computer technology would need to accommodate or support (for example, hard of hearing, arthritis in hands, wheelchair)?
  - No
  - Yes (Please describe)

3.2.1.9 Contact coordinates

- These will be used to contact developers as well as users as they are both defined in the system as actors.

  - Fax Number.
  - Phone Number.
  - Building Address: The building where traditional mail can be sent to the user.
  - Floor: The floor on which the actor is working.
  - Cubicle: The cubicle number.
  - Work Hours: Hours when the actor can be contacted.
  - Cell phone Number.
  - IM Network: Instant messaging network (MSN Messenger, Yahoo Messenger, IRC, Gtalk etc).
  - IM User Name: Instant messenger network user name that can be used to contact the actor.
  - Pager Number: Actor's pager number.
  - Pager Type: Numeric or Alphanumeric.

- Email Address: Actor's email address where testing or development tasks details can be sent.
- Preferred Notification Method: Phone, Cellphone, Email, Fax, IM Message etc.
- Nickname: In today's remote IT business environment is important to try to personalize the relationships between actors and using nicknames can achieve this purpose. Also for informal actors that don't feel comfortable providing their real name, this can be a method of addressing them.
- Web Page: Internet or intranet web page with additional actor information.

All this apparently technical information does not only show us what are the users' technical capabilities but also reveal the type of user that we are dealing with. A user with the latest versions of software, browsers, wide bandwidth and fast computer is likely to have a long experience using and adapting to new software and overall be more likely to quickly learn and use another software.

3.2.2 Task characteristics

Each action is performed by the user to accomplish a specific goal that brings them closer to even more important goals. The combination of the actions and the specific goal they accomplish is called a task. While the actions are accomplished using a product, the goal is user specific. It is therefore important to understand that for a task to be successful the goal must be understood and that the actions must be made available to the user with that goal in mind. To better understand the user goal we will have to learn the task frequency and criticality.

The following table will show the type of issues that the user might encounter while executing a task, their satisfaction with how the task addresses the issue and the importance of the task issue for themselves.

Issue Type	Satisfaction	Importance
Issue to address	**1-7**	**1-7**
Instructional and technical		
Accuracy of information and instructions.		
Fonts		
Consistency of style.		
Legibility of text.		
Visual clutter resulting from use of multiple fonts in a single document; question of availability of fonts on the targeted platforms.		
Colors		
Suitability of background colors.		
Suitability of foreground colors.		
Suitability of font colors.		
Haphazard use of color can be negative and confusing.		
Subtle, complementary color choices are generally more pleasing than saturated, contrasting colors.		
Borders		
Three-dimensional effects on command buttons can be effective visual cues for users.		
Use of three-dimensional effects on non-interactive elements can be confusing.		
Which one of these guideline does the user feel the particular issue violates (Nielsen (1994)) :		
Use simple and natural dialog.		
Speak the users' language.		
Minimize memory load.		
Be consistent.		
Provide feedback.		

Provide clearly marked exits.		
Provide shortcuts.		
Provide good error messages.		
Prevent errors.		
Include good help and documentation.		
Other.		
Which error type is more typical of this particular issue with operational features:		
The default state of UI control is incorrect.		
A poor choice of default state was made.		
The updated state of UI control is incorrect.		
The default input value is incorrect.		
A poor choice of default value was made.		
The updated input value is incorrect.		
The initial input focus is not assigned to the most commonly used control.		
The most commonly used action button is not the default one.		
The form or dialog box is too wide or long under minimum support display resolution (e.g., 800 x 600).		
Invalid inputs are not detected and handled.		
Other.		
Which error type is more typical of this particular issue with error handling:		
Displaying incorrect error message for the condition.		
Missing error messages.		
Poorly worded, grammatically incorrect, and misspelled errors.		
Messages were not written for the user and, therefore, are not useful to the user. For example, "Driver error 80004005."		
Error message is not specific nor does it offer a plausible solution.		
Similar errors are handled by different error messages.		
Unnecessary messages distract users.		

Inadequate feedback or error communicating to users.		
Handling methods used for similar errors are not consistent.		
Other.		
Images (web only)		
Visual cues and design details should blend with background not compete with it.		
Legibility of image labels.		
Legibility of button images.		
Suitability of size of images.		
Frames (web only)		
Display settings and browser types affect how frames are displayed.		
Use of back buttons have unexpected results.		
Tables (web only)		
Nested tables (tables within tables) slow down HTML load time.		
Presentation may vary depending on display settings and browser type (improper scaling or wrapping may result).		

Table 3-1 Element Type

While we may want the user to answer to our question with very specific answers for which we provide pick lists and other scales we also want him to be able to express other opinions about the software being tested in his own words. This will allow the user to answer in ways the product designer did not imagine and at the same time discover the way the user talks and refers to specific features or functionalities of the product. We might discover more intuitive ways of describing the way the product should be used by this type of user.

What do you call that (to discover user jargon and terminology)?

What are the main *bottlenecks* in this task?

What *work-arounds* have you found to get around the problems and bottlenecks?

What things would you most like changed?

Do you have any specific ideas for improvement?

3.2.3 Application characteristics

The application as a whole will also be rated by the user. The users' general impression of an application influences their approach to specific tasks. The general impression will be quantified based on the overall user satisfaction with particular application characteristics and the importance of such characteristics for the particular user.

How well does this application		
Perform		
Load time.		
Uptime.		
Tasks accomplished per use.		
Self-Descriptiveness		
Explain what is required in each fill-in field on the screen, so that you don't need to reference a manual to decide what to enter?		
Prompt you, so that you always know what the application expects you to do next and what your options are?		
Help you keep track of where you are in the application so you know how to return to other menus?		
Lay out prompts and fields on each screen in a clear, easy to read and understand format?		
Give error messages that clearly indicate what you did wrong and what to do next?		
Allows for User Control		
Allow you to cancel commands you have initiated but don't wish to execute, without undesirable side effects?		

Allow you to bypass irrelevant steps and get efficiently to the menu, field, or function you want?		
Allow abbreviated input of commonly used sequences of commands?		
Provide a command language that is consistent?		
Facilitates Learning		
Facilitate learning with a minimal amount of training and reference to manuals?		
Facilitate learning for users with no prior computer experience?		
Facilitate learning about it and using it with minimal ongoing human assistance?		
Facilitate remembering how to use it from one session to another?		
Facilitation Task Completion		
Reduce the effort required to perform routine or repetitive tasks?		
Protect the user from catastrophic results of normal human error?		
Help you to perform your job more efficiently and effectively?		
Provide all the functions that you require in your routine use of it?		
Help you to improve the quality of your work?		
Consistency and Predictability		
Behave similarly and predictably in similar situations?		
Require similar and predictable operations for similar functions?		
Provide clear feedback on the results of executed operations?		
Provide consistent response times across usage of the same functions or commands?		
Provide equal response times for similar activities?		
Provide reduced required input and output according to user training level?		
Flexibility in Task Handling		
Allow the experienced user to define his or her own set of functions?		
Provide shortcuts for the experienced user to perform tasks?		
Provide alternative ways for doing the same thing in different situations?		
Provide application information at different levels of detail on request?		

Table 3-2 Applications

3.2.4 Context of use

Web use is important in terms of the context in which the users are going to use the product and their expectations for it. An intensive web user also expects some of the paradigms found on the web to be present in applications and if they are found, they will be easier learned. This dimension is quantified in the user profile.

The company size is also relevant to the context of use as users are able to draw on more communal knowledge in a larger company but in a smaller company the information exchange might be swifter and more effective as it is more informal. The user will be asked to indicate the estimated number of employees in the company that he works for.
In today's IT environment, tools are rarely used in isolation, they are rather used as part of a toolkit that each IT worker develops based on his needs and experiences or is directed to used by the employer. Workers use different tools to accomplish different tasks and sometimes used different tools to accomplish the same task as directed by external factors such as security constraints and network bandwidth. A graphical tool might be easier to use but it might be using automatic sign-on which might be considered unsecured or it might place a too heavy burden on the network. The users will be asked to express in their own words what other applications they are using and how important they are to them.

For a user to be satisfied with a particular application or while accomplishing a particular task he must be able to accomplish a certain task but he must be able to do so in a manner that is appropriate to his values and attitudes. Some users might be looking for different things in a tool besides being able to accomplish a task and such as thrift, speed, ease, fun, comfort. It might also be important to discover through which media the user found out about the software being studied, how does he keep up with the latest versions and features is it specialty magazines, newsletters, web sites etc. The same medium should

be used for communicating other important knowledge. The user will be asked to identify the preferred medium of communication.

3.2.5 Measuring dimensions

We will be using a variety of scales and measurements to qualify and quantify the other issue characteristics. The most commonly used are illustrated below.

Frequency of Use		**Ease of Use**		**Ease of Learning and Memory**		**Location of Keys**		**Importance of use**	
0	Never	**1**	Very hard	**1**	Very hard	**1**	Very awkward	**1**	It helps me in one or more *minor* tasks in my job.
1	Almost never	**2**	Hard	**2**	Hard	**2**	Awkward	**2**	It helps in one or more of the *major* tasks in my job.
2	Seldom	**3**	Moderate	**3**	Moderate	**3**	Adequate	**3**	It *is* my job.
3	Sometimes	**4**	Easy	**4**	Easy	**4**	Good		
4	Often	**5**	Very easy	**5**	Very easy	**5**	Excellent		
5	Very often								

Table 3-3 Individual Function Satisfaction Ratings

The above individual function will be used to rate the individual application functionalities.

Each scale is relevant in a specific way to the overall quality of the product. The users will try to rate each feature on the four available scales, except in the case where the

frequency of use is rated at 0 when no other ratings need to be picked from the other scales. If not completely mutually exclusive the scales are meant to be used separately with no correlation being expected to be observed between the ratings of a specific feature.

We will be using a 4 levels experience scale. The scale will be used to rate the domain experience and application experience.

Experience Grades	
0	None
1	Low
2	Medium
3	High

Table 3-4 Experience Grades

The tasks' and applications' importance for the user as well as the user's satisfaction with a particular task or application will be quantified. We will be using a 1-7 scale where 1 means trivial/dissatisfied and 7 means crucial/completely satisfied.

3.2.6 Issues

A usability issue will arise only as a combination of specific conditions between the person using the application, the context in which he or she is using it, the particular task that they are trying to accomplish and the overall application characteristics.

"Usability is a property that emerges when we design a product that sits in the triangle made by the customers, their tasks and the environment in which they use"
Travis (2002, p38).

In addition to the classification of each of these components separately, the issues logged will be defined through their usability factors and criteria, and this combined set of characteristics will become the usability issue that will be logged in USITRES.

3.2.7 Issue Factors and Criteria

The criteria that the usability factors will be designated as are: Time Behaviour, Resource Utilization, Attractiveness, Likeability, Flexibility, Minimal Action, Minimal Memory Load, Operability, User Guidance, Consistency, Self-Descriptiveness, Feedback, Accuracy, Fault-Tolerance, Resource, Safety, Readability, Controllability, Navigability, Simplicity, Privacy, Security, Insurance, Familiarity, Loading Time and Appropriateness. (Seffah, QUIM)

Some of the usability factors that will be tracked are: Efficiency, Effectiveness, Satisfaction, Productivity, Learnability, Safety, Trustfulness, Accessibility, Universality and Usefulness. (Seffah, QUIM)

The mapping of criteria to factors is the following (Seffah, QUIM):

	Factors									
Criteria	Efficiency	Effectiveness	Satisfaction	Productivity	Learnability	Safety	Trustfulness	Accessibility	Universality	Usefulness
Time Behaviour	+			+						+
Resource Utilization	+			+					+	
Attractiveness			+							
Likeability			+							
Flexibility		+	+					+	+	+
Minimal Action	+		+		+			+		
Minimal Memory Load	+		+		+			+	+	+
Operability	+		+				+	+		+
User Guidance			+		+			+	+	
Consistency		+			+	+		+	+	

Self-Descriptiveness					+		+	+	+	
Feedback	+	+							+	+
Accuracy		+				+				+
Completeness		+				+				
Fault-Tolerance						+	+			+
Resource Safety						+				
Readability								+	+	
Controllability							+	+	+	+
Navigability	+	+					+	+	+	
Simplicity					+			+	+	
Privacy							+		+	+
Security						+	+			+
Insurance						+	+			
Familiarity					+		+			
Loading Time	+			+					+	+
Appropriateness							+	+	+	+

Table 3-5 Usability Factor Criteria

The user will rate each available criteria for each factor on a scale from 1 to 7 where 1 means trivial/dissatisfied and 7 means crucial/completely satisfied.

3.2.8 Solution categories

The solutions to the usability issues will be classified through log closure codes which can be any of the following:

Closure code	Details
Issue Solved	Problem fixed through software modifications.
Issue Unsolvable	Problem cannot be fixed due to various limitations.
Specification Changed	The technical specifications are changed to reflect the various implementation limitations.
Requirements Changed	The usage specifications changed to reflect the differences between the intended usage and the experienced usage.

Recommendation Issued	In case the usability developer is not mandated to make actual changes.
User system configuration	Issue solved by changing the user software configuration.
Application system configuration	Issue solved by changing the application server software configuration.
User hardware related	Issue source identified as being the user hardware.
Application hardware related	Issue source identified as being the application server hardware.
Third party software related	Issue source being identified as being a third party software issue that cannot be fixed through configuration settings.
Client education	Issue solved by explaining to the client what are the various software, hardware and environmental limitation.
User education	Issue solved by explaining the users what is the intended use of the feature.
Cancelled	The issue was declared as cancelled.
Normal condition	After further investigation the behaviour was designated as normal and conform to user specifications.

Table 3-6 Closure Codes

3.3 Priority System

The priority system will be developed as a rating system that takes in consideration the individual problem criticality and the overall system criticality.

The overall system criticality will be implemented as follows:

Criticality Index	Application Type	Application Details
1	Life Support	Health related applications used in operation rooms and elsewhere in hospitals
2	Critical Industry Support	Applications used in critical industries such as nuclear energy plants, military etc.
3	Financial Industry Support	Applications used by banks, governmental financial departments and other financial institutions.
4	Industry Support	Other industry applications
5	Entertainment applications	Multimedia applications used for entertainment purposes such as music, video players, games etc.

Table 3-7 Criticality Index

The users will also be able to define what their requirements for issue resolution are by defining their priorities as in the example bellow.

Priorities:
1: To be solved within 2 hours or less from the time of discovery
2: To be solved within 2 hours or less during working hours
3: To be solved within 24 hours or less from the time of discovery

The priorities will be used to determine which issues should be solved first but are only going to be inspected in detail for detail studies on user priority type and behaviour. For example a developer will only inspect the priority index (1, 2 or 3) to determine which issue to solve first, but other reports might try to uncover what first priority means for different types of users and use that knowledge in designing other systems.

The overall priority will be decided by the lowest priority index obtained by multiplying the criticality index of the application by the priority index of the individual issue. Using this decision system a critical industry support application (criticality index 2) second priority issue (priority index 2) will have an overall priority of 4 and will be solved before

a financial industry support (criticality index 3) second priority issue (priority index 2) that will have an overall priority of 6. Issues that have equal overall priority such as in the case of a critical industry support application (criticality index 2) first priority issue (priority index 1) with an overall priority of 2 and a life support (criticality index 1) second priority issue (priority index 2) with an overall priority of 2 as well will be solved in the order in which they were logged.

The top three criticality rated issues are rated as critical and will be solved based on the severity rating as determined. The other issue will be rated based on a severity scale first discovered during our literature review.

In the book E-commerce Usability David Travis introduces a severity scale that is based on the size of the stakeholder group and its importance.

Size				
	H	3	2	1
	M	4	3	2
	L	5	4	3
		L	M	H
		Importance		

Table 3-8 Severity Scale

Based on this scale, issues are rated based on the group size and group importance. The group scale can be determined by the number of users that identify with the same group or as declared by group representative as some users can log issues on behalf of a user group. Group importance will have commonalities between various companies such as the fact that the CEO group is the most important group and that the developer group is the least important. However other groups will be assigned various importance ratings between different companies. As an example, regulatory bodies will have higher importance ratings in heavily regulated industries such as medical equipment and software then in the entertaining industry.

3.4 Methodology

Our methodology solves three specific problems:

- To manage the problems, it defines a workflow between the tester, developer, manager and super user.
- To track problems to resolution, it defines issue stages and rules to move from one stage to the next.
- To connect user types to solution types, it defines closure codes.

A problem will typically be identified by a stakeholder. He will log the issue in USITRES. A project manager that is assigned to that particular project will assign the resolution to a developer. The developer will implement a solution or make a recommendation. The manager will validate the developer solution and if appropriate will inform the stakeholder. If not appropriate it will be reassigned to the developer for further refinement. Through its resolution the issue will have one of the following statuses (we will also be tracking the times when the status changes from one state to another).

- **New**: when first logged the issue will have its status at "New".
- **Assigned**: after a project manager inspects the details of the new issue it will assign it to a developer and the status will change to "Assigned" to show that work is being performed to fix the problem.
- **Work in progress**: once the developer actually finds time in his schedule to start working on the problem he or she will change the status to "Work in progress".
- **Pending**: during the whole process, progress might be stalled by a variety of reasons. During these periods the status of the issue will be set to "Pending". The action that the issue is pending for can be client solution approval, manager implementation approval, developer action, further desting.
- **Resolved**: once the user will inspect the newly implemented solution or any other development that fixed the usability issue and is satisfied with it he will set the status of the issue to "Resolved".
- **Closed**: after the user declared himself satisfied with the solution, the manager will inspect all of the issue details, fill in any missing details and then set the

status to "Closed". Additionally a closure code will be assigned to the issue to better identify the type of solution found as detailed above in 3.2.2 Solutions:

- Issue Solved.
- Issue Unsolvable.
- Specification Changed.
- Requirements Changed.
- Recommendation Issued.
- User system configuration.
- Application system configuration.
- User hardware related.
- Application hardware related.
- Third party software related.
- Client education.
- User education.
- Cancelled.
- Normal condition.

The depth of our ontology will serve the two particular purposes that we set out to attain in the begging. Specific issue characteristics will be clearly defined and in the process will help the developer understand and solve the problem. The user base characteristics will also be fully discovered and made available to help for future application design. The designers of new features for existing applications won't have to rely on the initial assumptions about who will use their application and will simply have to consult USITRES to find out who the real application users are. The designers of new applications will be able to use the collected data to discover which type of users use various type of applications, similar to the one that they developing, and use this hard data to designate their intended target, instead of having to assume or hope that a particular group of users will use their application.

4 Chapter 4 – Proposed Tool

In this chapter we will present the platform that will support our application and the application architecture and its design.

4.1 Platform

The platform was selected with the final application characteristics in mind. The final application is designed around a data repository therefore our platform had to be database centric. The final application is designed for remote usability and therefore we chose a platform that is web centric.

4.1.1 Platform Architecture

The development of USITRES will be completed on the Oracle Application Express development platform (formerly known as HTMLdb) and the data will be stored in an Oracle Database 10g Express Edition.

The platform is comprised of three components. At the backend we have the database that stores data and procedures. At the middleware level we have the Application Express application server with its Apache server. The application and the development front end that are available to users and developers are web based.

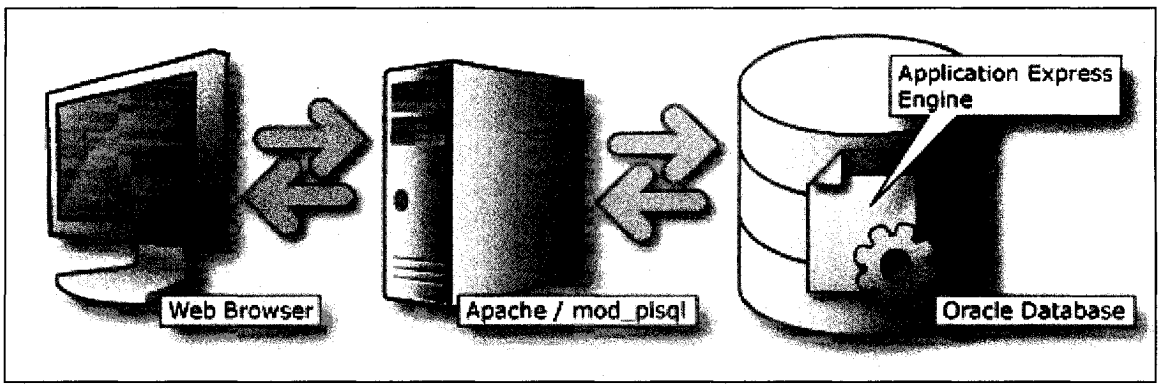

Figure 4-1 Platform Architecture (Oracle)

The Oracle database stores both the USITRES application data and the data and procedures that are used to create the Oracle Application Express development pages and the USITRES application pages. The two types of data are separated in schemas, with the USITRES application data stored in the USITRES schema and the various application data and procedures stored in system schemas called SYS, XDB, WMSYS, FLOWS_FILES, FLOWS_1600 etc.

The Apache server serves both the USITRES application web pages and the Oracle Application Express web pages used for the development environment. The Oracle Application Express server and the Oracle database run on the same machine. The Oracle application server itself uses information that is stored in the database.

The development pages, that are used to create the application pages, are created and displayed using procedures and metadata that come packaged with Application express. The USITRES application web pages are created in the development environment. They are displayed using procedures that come packaged with Oracle Application Express and metadata that is generated during the development process.

The platform excels in simplicity, portability, and accessibility. Its reduced number of components makes it easy to install and maintained. The platform is available for almost any platform from commercial UNIX flavours such as SUN, HP and AIX, to Windows to

major Linux distributions like Red Hat and others. It can be accessed for maintenance and development from any computer that has a web browser installed.

Examples of the various platform administration screens are included in Appendix A.

4.1.2 Platform Characteristics

To be able to store a lot of usability data we chose a platform that is **database centric**. To be able to reach a wide user base we chose a platform that is **Web centric**. To be able to quickly implement changes as suggested by the USITRES users we chose a platform that offers a **rapid development environment**.

4.1.2.1 Database Centric

The platform will be **database centric**. This will allow the application to store practically infinite amounts of data (32 Petabytes) and most importantly give the supporting staff the tools through SQL to mine the data, discover patterns and trends in the data etc. It is very important for a data centric application to be able to provide the needed tools to mine through the collected data. A database centric application is ideal for an application that will store large amounts of data, but which at the same time will have to accommodate numerous sessions of concurrent access to the data. The transactional model implemented by Oracle in its database server is ideal for accommodating at the same time the online transactional processes (OLTP) that require fast response time and the decision support system (DSS) processes that access tremendously large amounts of data in their periodical reports.

If new types of usability data are deemed necessary to usability studies as requested by any of the USITRES stakeholders, the database will be modified to include the new data type to be stored and consequently the application will be modified to allow for the input and reporting on this additional data. In the later stages of life of USITRES, its success will depend on mining large amounts of data. The ultimate purpose of USITRES is to

predict possible issues given a certain type of application and a certain type of intended users. It will do that by extrapolating from all the stored data about past issues logged for other applications. The predictions will be more accurate if more data is available to base the predictions on. To store such large amounts of data and have the available data mining tools we had to select a **database centric platform** that guarantees the capability of storing such large amounts of data and various data mining procedures.

4.1.2.2 Web Centric

To achieve our goal of mass usability testing we selected a **web based platform**. This will allow us to reach a much larger number of active users as opposed to using a traditional local application that has to be downloaded and installed on every user workstation. Another class of users are the passive users which will only access the USITRES usability library and system wide reports. The web platform will allow us to reach these users anywhere they are located. This type of users will be typically academic users and it is important for the success of USITRES to obtain an increased visibility with this class of users by having a web presence similarly to other research tools such as CiteSeer and Portal.org.

This will also eliminate the need for client side application installation and will drastically reduce the time to connect the testers and the project managers. The maintenance efforts will only have to focus on one unified platform as opposed to different components which is the case with classical client-server issue tracking applications.

Some of the drawbacks of desktop tools over a web based application are pointed out by the HTMLDB developers(Oracle):

- Desktop tools are fragmented storing data in many places.
- Desktop tools are vulnerable and the stored data is not secure.

- Desktop tools are platform dependent usually only developed for MS Windows while web based tools can be accessed from any platform, such as Linux, MacOS and even PalmOS and any other number of wireless device operating system.
- Desktop tools are web unfriendly and need the installation of client software.
- Desktop tools are costly especially for the user.

Users will be able to log usability issues from the computer where it was first discovered or from any other device with internet access, being that a laptop in a WI-FI hot spot, a computer in an internet café, wireless enabled personal assistant device such as a Palm or smart phone.

HTMLDB has translation facilities allowing its web pages to be displayed in different languages based on user browser preferences or major internet provider settings. This will help bridge some of the gaps that exist between the different user communities as well as between the designers, programmers and users, allowing them both to take full advantage of the proximity created by a web environment.

4.1.2.3 Rapid Development Environment

The development environment is designed for rapid development with most of its controls having a predominant visual component. Only a limited knowledge of SQL or PL/SQL is required for initial development. Expertise with these tools will be required for advanced feature.

While the development of the application requires minimum skill investments, the maintenance of the backend requires expert Oracle DBA skills for backup and recovery architecture design and implementation as well as for all the performance tuning activities. This separation of skill sets in turn can contribute to an increase in the development speed of USITRES.

4.1.2.4 Other Characteristics

In the early stages of life of USITRES, its success will be highly dependent on the number of users adopting this tool as their usability tool of choice, as we will be looking to gather enough user and issue information to be able to extract a meaningful correlation between the two. Two of the major characteristics that users look for especially in web based applications are trustfulness and reliability. The fact that we will be using a solution based on a infrastructure developed by a **major technology supplier** will give those two characteristics to USITRES. A powerful database engine and the very high level security available will give us the trustfulness characteristic, as the users will be confident that their data cannot be lost or hacked. Various high availability solutions based on very effective backup and recovery practices will allow us to attain a very high availability percentage and reliability.

Most users hesitate before using free open source tools as the open source technology providers are, with a few exceptions, prone to disappearing from the IT scene, most of the technologies that they propose being more like fads. Building on a solid base provided by a **major technology supplier** guarantees to the user that the tool will be around for a while. Therefore the user will be more inclined to invest some time in learning how to use the tool, knowing that the skills acquired will be useful for a long period of time.

The **free development** platform will allow third parties or even freelancer developers to write code and components that can be used by USITRES. It is easily foreseeable that users that will start by using the system for their testing purposes will also become interested in the underlying platform, start experimenting with the development platform and end up by developing functionality that can then be plugged into USITRES.
Oracle summarizes the main advantages offered by its platform as follows:

- Qualities of a Personal Database: productivity, ease of use, flexibility.
- Qualities of an Enterprise Database: integrity, availability, portability, security, reliability, scalability, manageability.

- Free Development Environments: the totality of the software components of our application are free to use for development purposes through the Oracle Licence Agreement.
- Large user community: the user community for Oracle products is one of the largest software user communities on the web with a variety of expertise domains and a wealth of knowledge that will also support the development process. This is unlike most other open-source user communities which are rather ephemeral, they emerge with the initial hype of a new open-source language or tool but they also disappear as soon as the initial hype passes.
- Designed for Consolidation: usability testing teams are presently using numerous forms and other type of resources to log and track usability test results and issues. Most importantly various usability labs have each their own user database, in either proprietary or vastly varying formats from one lab to another. The HTML DB development platform is designed for consolidation purposes supporting a variety of import types and providing easy to use wizards for importing from all these various sources. This represents the major advantage of creating the possibility of merging or at least sharing the information in all the user databases, presenting to the project managers the opportunity of finding an increased number of testers and better matching these testers to the particular product being tested. We will also be able to store in one place all the forms used during testing data gathering, user logging and tracing.

4.2 USITRES Architecture

Due to the data centric nature of our application, USITRES was designed from the database up. The interface was designed to provide the most intuitive access to the database while being focused on the user's tasks at the same time.

4.2.1 Data Model

The main purpose of the USITRES system is to track usability issues and the characteristics of the users logging them. Upon gathering a lot of this type of data we will be able to forecast the type of users that might have a specific usability issue. Therefore the data model is focused on the user and issue identities. The illustrated data model will store the Online Transaction Process (OLTP) data, data that is inserted live by the users in the day to day operation of USITRES. The data warehouse component of the system that will store historical data, data older than 1 year, will be developed at a later time.

The tables are split in three major areas, one that focuses on usability issues, one on all the actors in the USITRES system, users, developers, managers, clients and a variety of other type of users and one that focuses on reference data. Figure 4-2 shows the set of tables that will store the actor or user details.

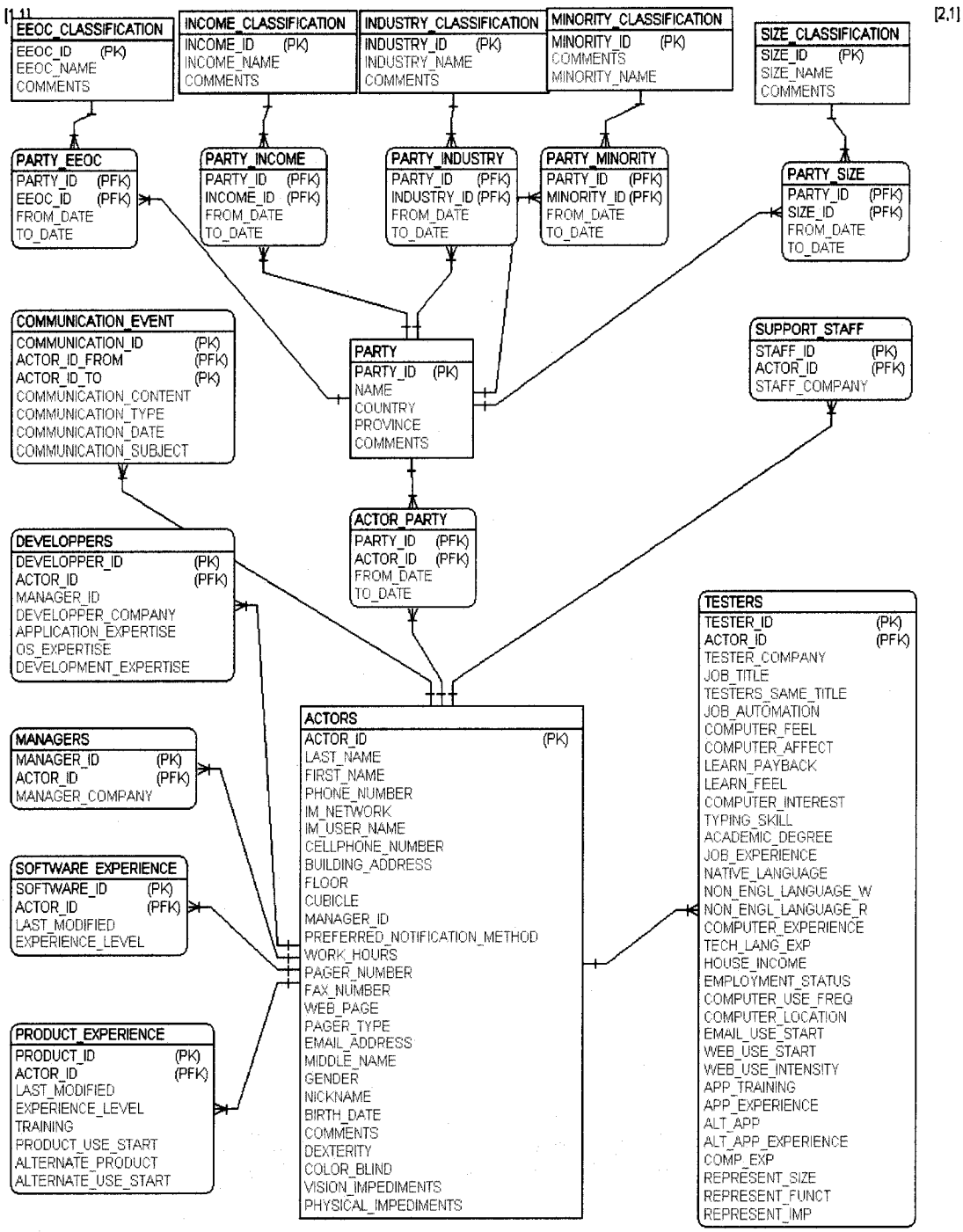

Figure 4-2. USITRES Actors Subject Area

Figure 4-3 shows the set of tables that will store the usability issues details.

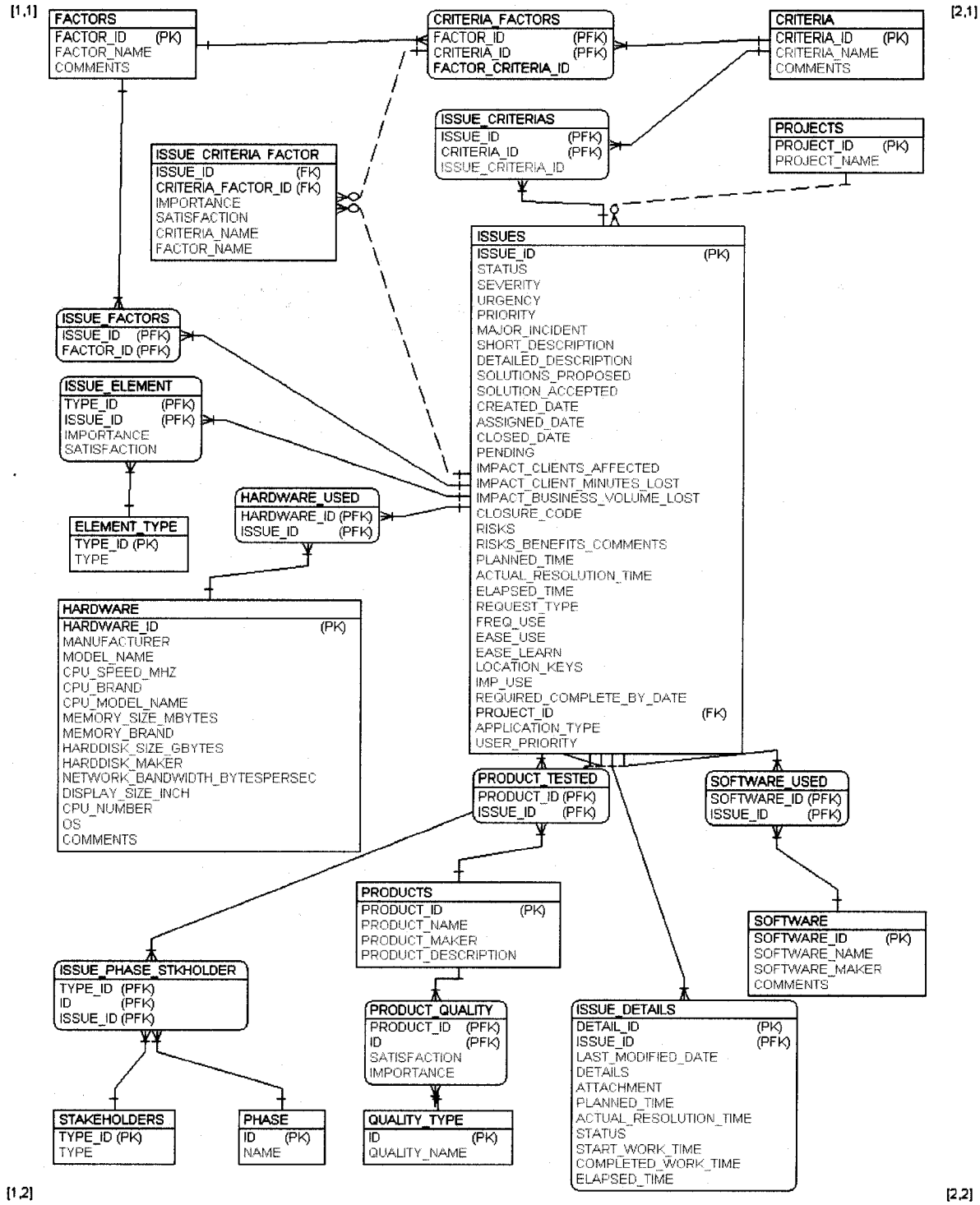

Figure 4-3. USITRES Issues Subject Area

Figure 4-4 shows the tables that will store reference data.

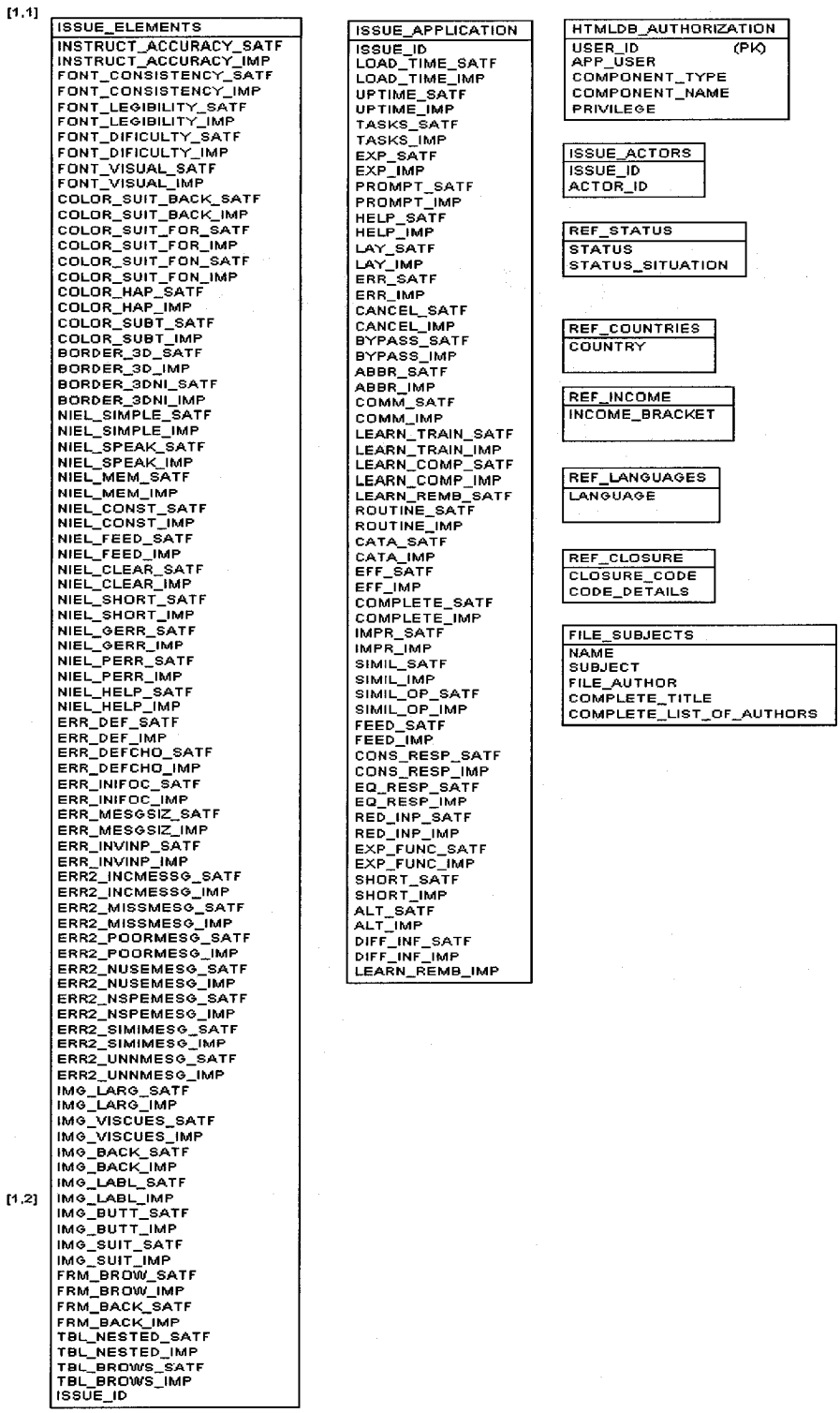

Figure 4-4 Reference Subject Area

Although the "party" tables are usually meant to model the traditional aspects of the developer, manager and client appurtenance to a corporate identity, in USITRES they have the added function of modelling representative users and testers.

Users can be split in 3 main categories: those that actively use a product to accomplish a task, those that are in the immediate proximity of active users and although not using the product, they are impacted by it, and those that don't use a product due to a prohibiting product characteristic such as extreme complexity, incompatibility with other products and price. While the first category of users was the traditional target market and main focus of all product designers the second and third were largely ignored. Through the use of the party identity we will model a representing user, a user that will log usability issues on behalf of non-active users (although these non-active users can log the issues by themselves we will be offering this alternative as the experience shows that users that are not directly involved with a product are unlikely to get involved with the customer service departments of the product maker). This will inform the designer of some of the barriers to use that unintentionally slipped into the product design.

While the active users were treated in one on one relationship by the traditional customer service and help desk department, this approach ignores the largest part of the customer base, the customers that chose to not call. Through the use of the party model identity we will model the representative user, a user that logs usability issues on behalf of lots of users of the same type. These representative users can be user group advocates or representatives, forum moderators, paper or web editors. These users are informed by user opinions in a variety of ways. The consumer group advocates are in charge of formal consumer groups and it is their foremost responsibility to represent the consumers. The internet forum moderators are in touch with the most active and interested of the consumer community. The messages that are exchanged in the forums are therefore very important to a product designer. The newspaper, magazine and web editors also hear from the consumer through a variety of feedback mechanisms. These representative users are clearly more important to the designers than any one user taken in isolation. Their issues will be given a heavier weight through various party properties in USITRES,

feature that is largely absent from other issue tracking software products. This way a much larger number of consumer opinions will be accounted for and looked at when designing or redesigning a product to fix usability issues. At the same time the involvement of these representative users with USITRES guarantees that USITRES will gain increased exposure to a larger consumer base.

The representing user, that logs issues on behalf of different type of users, and the representative user, that logs issues on behalf of a multitude of users of the same type are given a voice in USITRES through the use of the party identity.

Figure 4-5. USITRES Actor Data Model.

4.2.2 User Interface

The user interface design was built around the tasks that the various users will have to fulfill with the application.

Upon login a tester will be presented with a system description and direct navigation to the tester home, the usability library and warehouse and the custom bug tracking component.

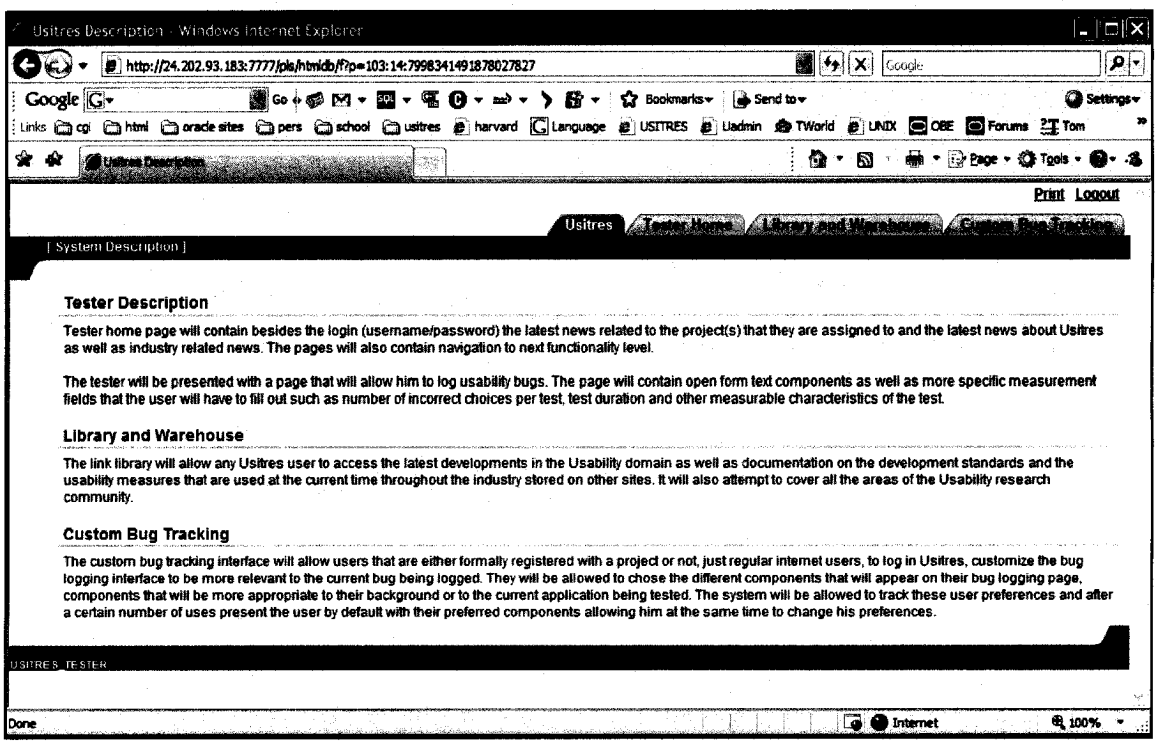

Figure 4-6 USITRES System Description

On the profile page the tester will be able to enter demographic details and contact information. They will also have access to third level navigation in the form of a right hand list with links to the default page "User Details" and other pages such as "Tester Details", "Context", "Tasks" and application.

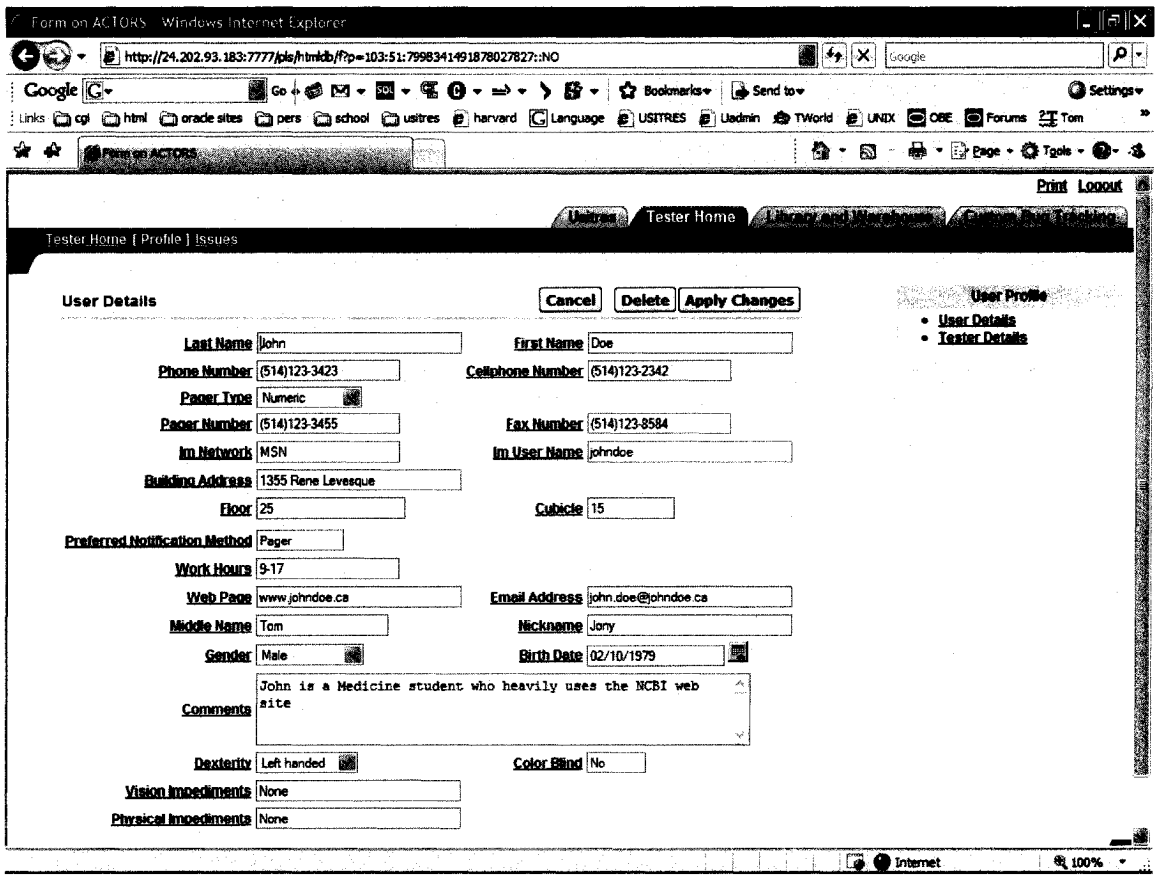

Figure 4-7 USITRES User Details

On the issues page, the tester can chose from one of the projects that he is involved in and then from the project issues. Upon doing that, the issue details are displayed. In the screen below the project doesn't yet have any issues and the tester will proceed by creating the first one.

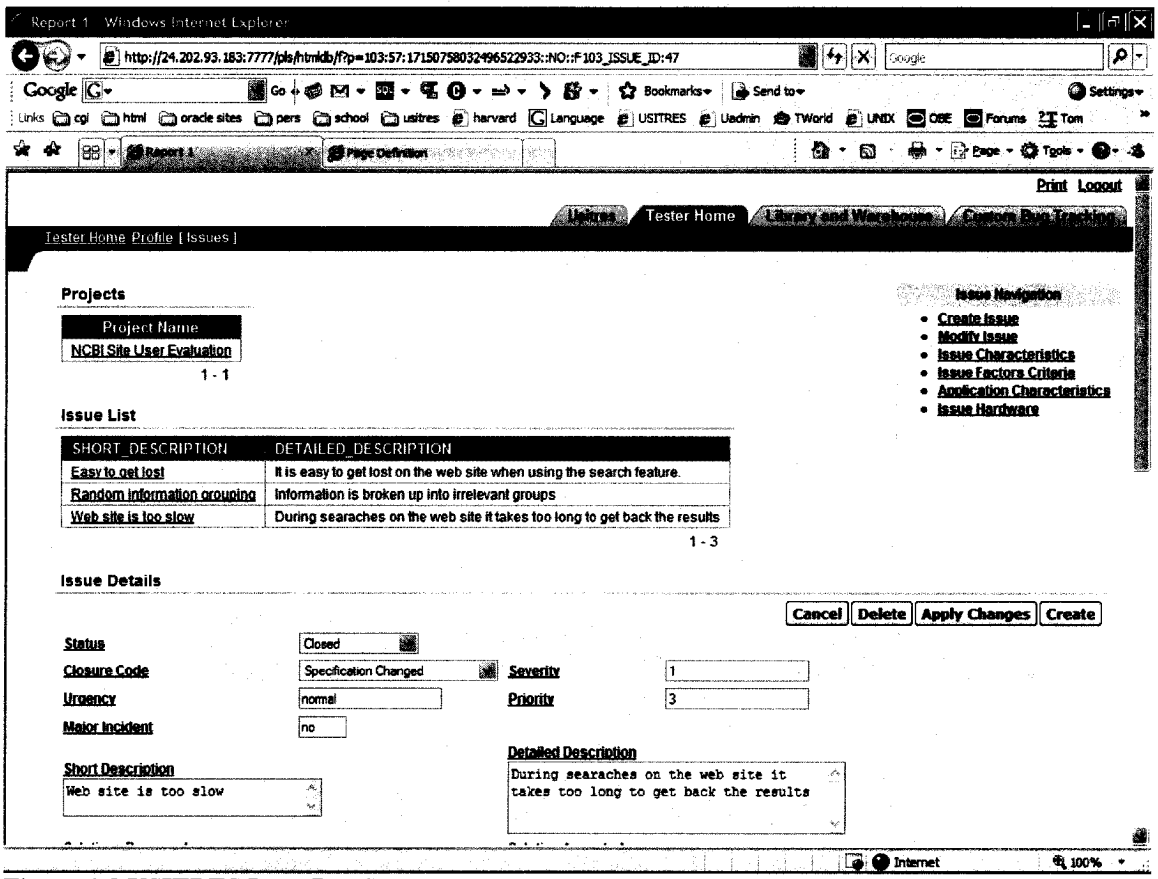

Figure 4-8 USITRES Issue Details

The tester can change the status of the issue by choosing one of the available status codes.

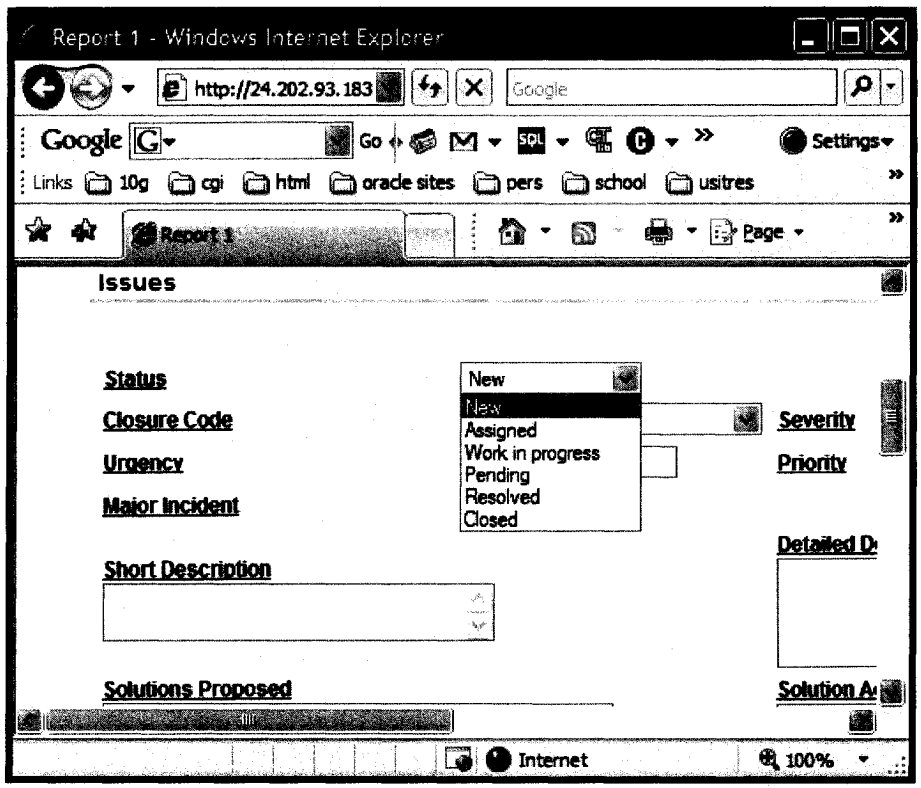

Figure 4-9 USITRES Issue Possible Statuses

On the issue elements page the tester can fill in the details of the feature elements importance and satisfaction to him/her.

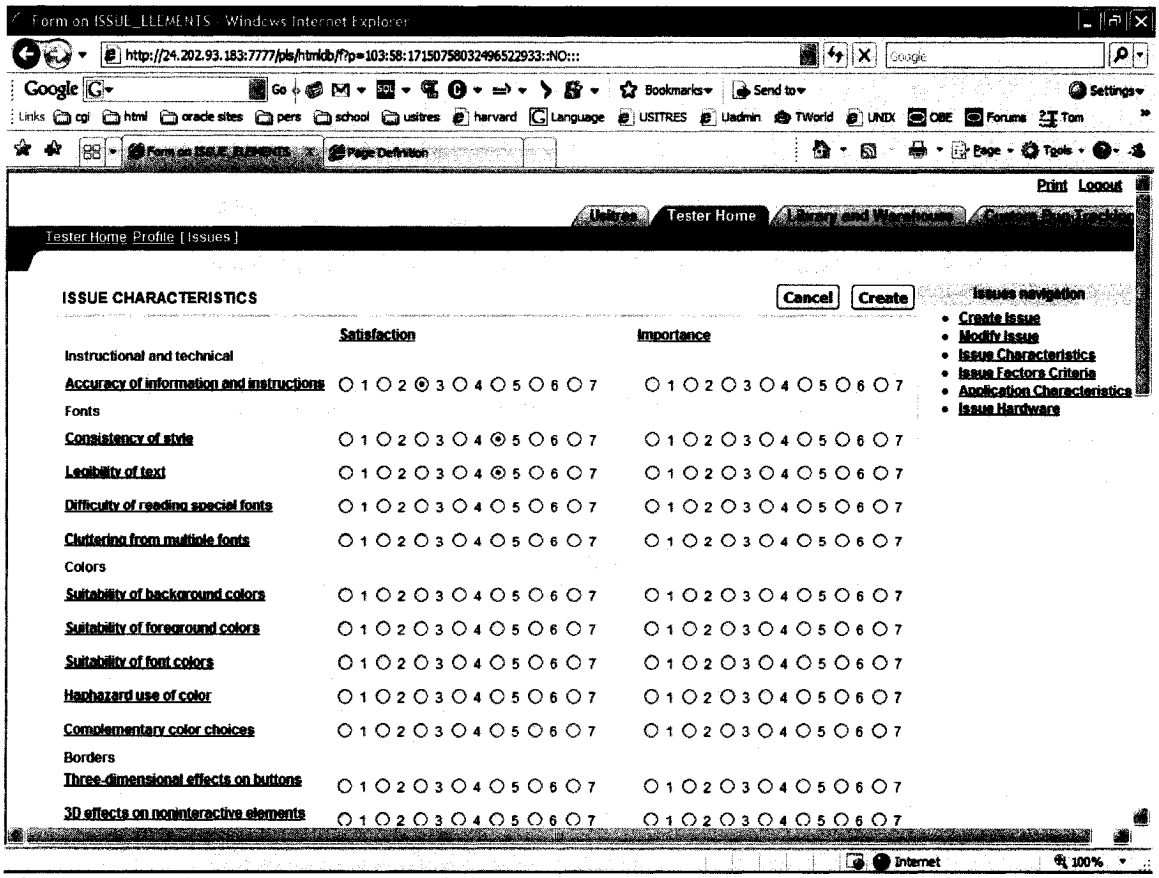

Figure 4-10 USITRES Issue Characteristics

On the "USITRES Document Library" page the user will be able to upload and download usability documents. The document library now holds most of the reference materials

used in this thesis. It currently holds 115 documents in various formats such as PDF, Word and Power Point.

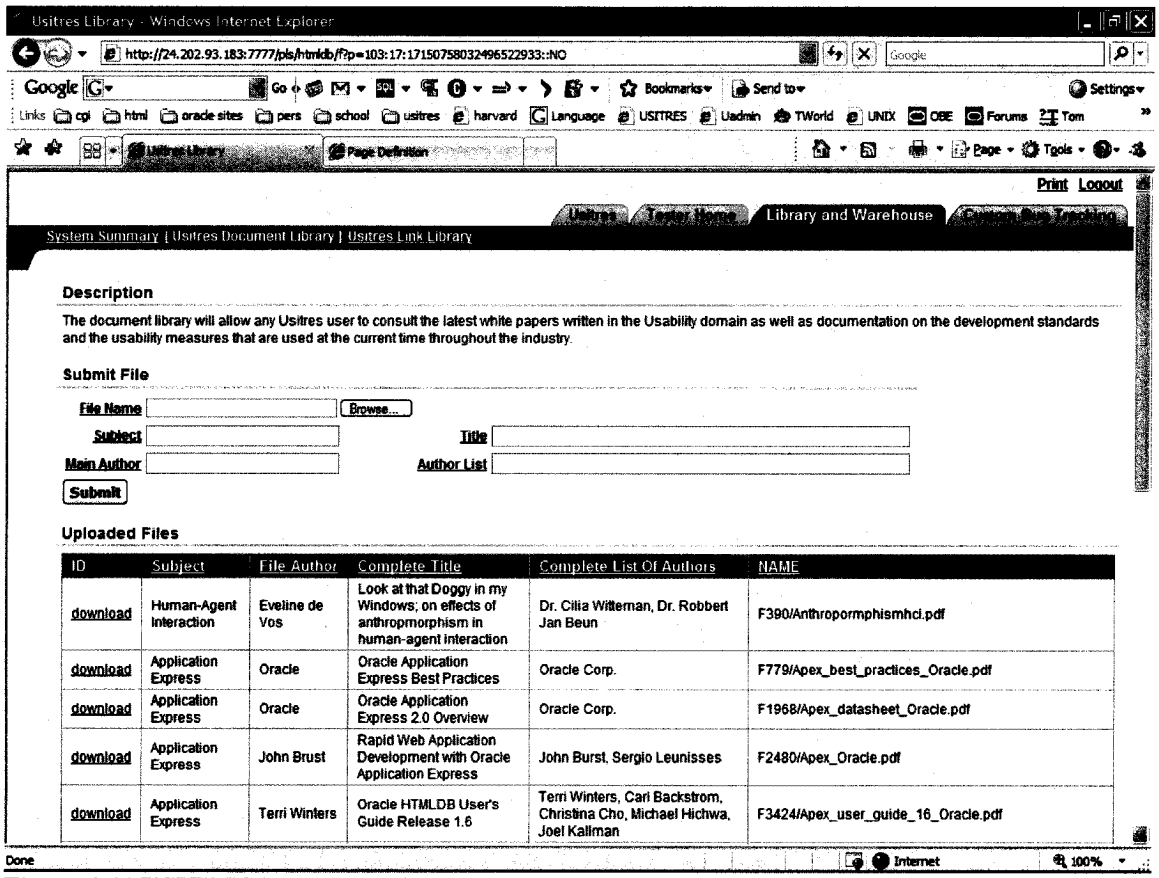

Figure 4-11 USITRES Document Library

4.3 USITRES Design

In this chapter we will described the proposed tool. The system design will be described by enumerating the system actors and illustrating their interaction with the system through use cases. We will present the backbone of the system through the data model and the much more visible user interface through a series of annotated tool screen shots.

The functionality of USITRES will be released gradually in 3 major releases. The system design was developed around this strategy. The first release will include an out of the box tester and project manager interfaces for usability bug tracking and the supporting

database. The second release will include a customizable user interface for logging usability bugs in an open end question format. The third release will expand the functionality to include a data warehouse, support data mining in this warehouse and a new complimentary database and web component to support a library of documentation and white papers.

4.3.1 Process and Uses of this tool

To illustrate the process steps that will be implemented in USITRES we will be presenting the system actors and several processes from the point of view of these actors. We will be using use cases to depict these processes.

4.3.2 System actors

The system actors are(rephrase):

- **Your customers:** the people or organizations that fund and task USITRES. They don't interact directly with USITRES.
- **The client's customers:** the people or organizations that buy things from your client. They don't interact directly with USITRES.
- **Users** (they interact directly with USITRES)**:**
    - Testers (formal and informal): the testers are using the various client applications and web applications being tracked by USITRES. Formal testers are those that were selected by the system customers or customers' clients. Informal testers are those that use some of the applications being tracked by USITRES and decide on their own to log a usability issue. Informal testers select themselves to be part of USITRES testers.
    - Managers: USITRES users that are in charge of managing applications being tracked by our system. They review and sometimes modify tester and developer profiles, and assign issues to testers to be tested and to developers

to be fixed. They execute project wide reports and send the results to the customers.

- Developer: users that log into USITRES to receive their work tasks as assigned by an USITRES manager. They performed these tasks and then they log in updates to issue resolution history and status.
- Super-user: users that assign managers to projects and create manager type users. They execute system wide reports and post the reports on the web.
- Library user: users that log in to browse through the usability documents or through the links stored in the library section. They will also be able to browse system wide reports.

4.3.3 USITRES use from the tester view

From the tester view we will be presenting how a tester creates a usability issue, which is similar to how a manager will create a usability issue, how a tester will update an issue after it was created and how a tester updates his/hers own profile.

4.3.3.1 Use case name: formal user (or manager) creates usability issue.

Use case short description: upon testing an application and finding a usability issue, the user logs in to USITRES and logs a usability issue.

Pre – conditions:

- The tester tested an application and identified a usability issue.
- The tester had its profile created in USITRES.
- A project was created for this application.
- The tester was assigned to a project.

Actor's goal: to track the resolution of an usability bug that he or she as an application user discovered while using the application. In the case of the manager, the client brings to the manager's attention an usability bug that the client wants to track.

Narration: this use case starts when the actor accesses the main USITRES login page. The system prompts for username and password. The tester (or manager) enters the username and password. The system validates them and upon successful validation the system presents the actor with the USITRES home page. The actor navigates to the testers' (or managers') home page. The actor navigates to the projects page. The system retrieves and presents the actor with selected details of assigned projects on the projects page. The actor selects the project that this particular issue is relevant to. The system retrieves and presents the actor with a list of previously logged usability issues and the options of modifying them or creating a new usability issue. The actor selects the option of creating a new usability issue. The system present the issue characteristics page where the actor can fill out details such as issue description, solution proposed, risks, number of clients affected. After the actor fills all the relevant characteristics he or she confirms that the characteristics are complete by indicating to the system to apply the changes. He or she then selects the issue elements link. The system presents to the actor a page where the actor will rate based on importance and satisfaction various issue elements such as "Accuracy of information and instructions", "Consistency of style", "Legibility of text" etc. After the actor fills all the relevant elements he or she confirms that the elements are complete by indicating to the system to apply the changes.

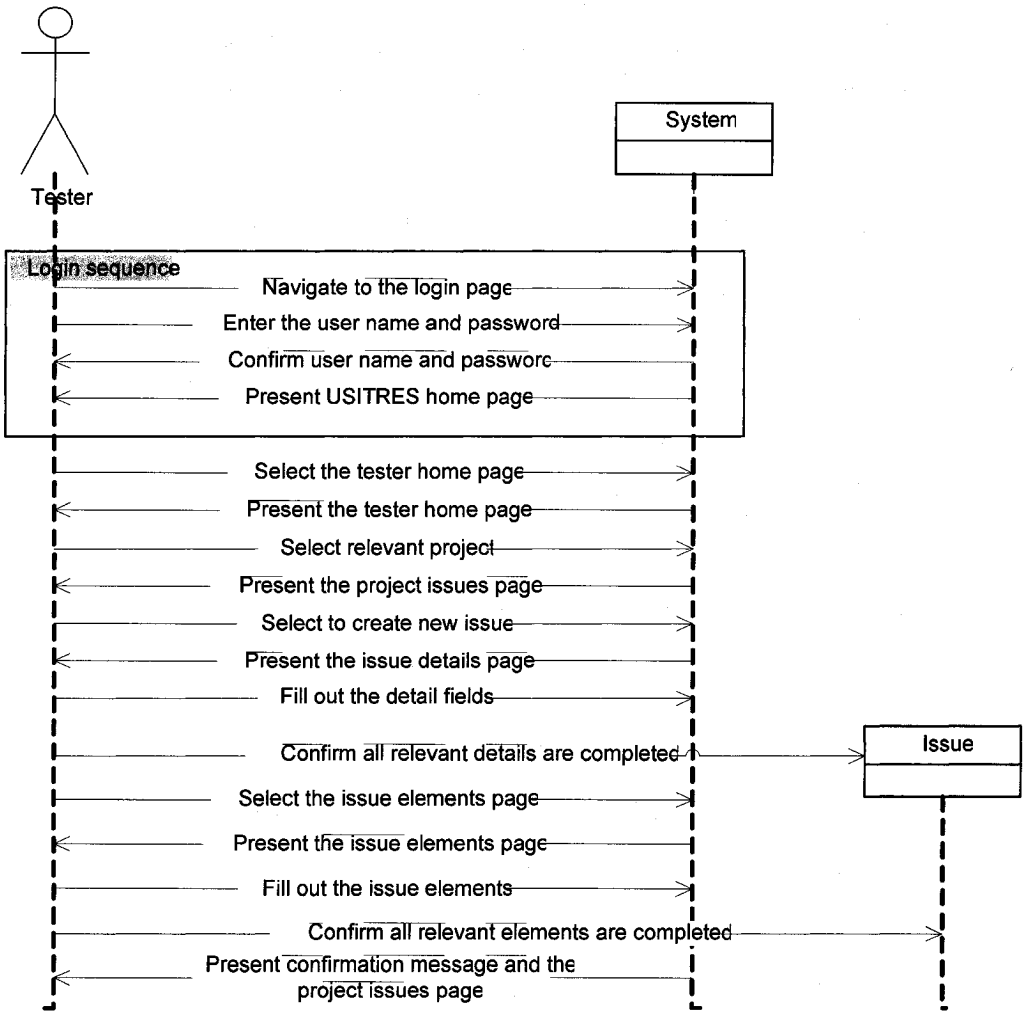

Figure 4-12 Use case: formal user (or manager) creates usability issue.

4.3.3.2 Use case name: formal user updates usability issue.

Use case short description: the user logs in to USITRES and modifies certain characteristics of a usability issue previously logged.

Pre – conditions:

- The usability issue was previously logged.
- The tester had its profile created in USITRES.
- A project was created for this application.

- The tester was assigned to a project.

Actor's goal: to append or modify information related to an usability issue. Once new facts become apparent relevant to a logged usability issue, the user wants to make them part of the tracked issue.

Narration: this use case starts when the user accesses the main USITRES login page. The system prompts for username and password. The tester enters the username and password. The system validates them and upon successful validation the system presents the user with the USITRES home page. The user navigates to the testers' home page. The user navigates to the projects page. The system retrieves and presents the user with selected details of assigned projects on the projects page. The user selects the project that this particular issue is relevant to. The system retrieves and presents the user with a list of previously logged usability issues and the options of modifying them or creating a new usability issue. The user selects one of the listed issues. The system fills out the characteristics form where the user can modify details such as issue description, solution proposed, risks, and number of clients affected. After the user modifies the relevant characteristics he or she confirms that the modifications are complete by indicating to the system to apply the changes. He or she then selects the issue elements link. The system presents to the user a page where the user will rate based on importance and satisfaction various issue elements such as "Accuracy of information and instructions", "Consistency of style", "Legibility of text" etc. After the user modifies the relevant elements he or she confirms that the modifications are complete by indicating to the system to apply the changes.

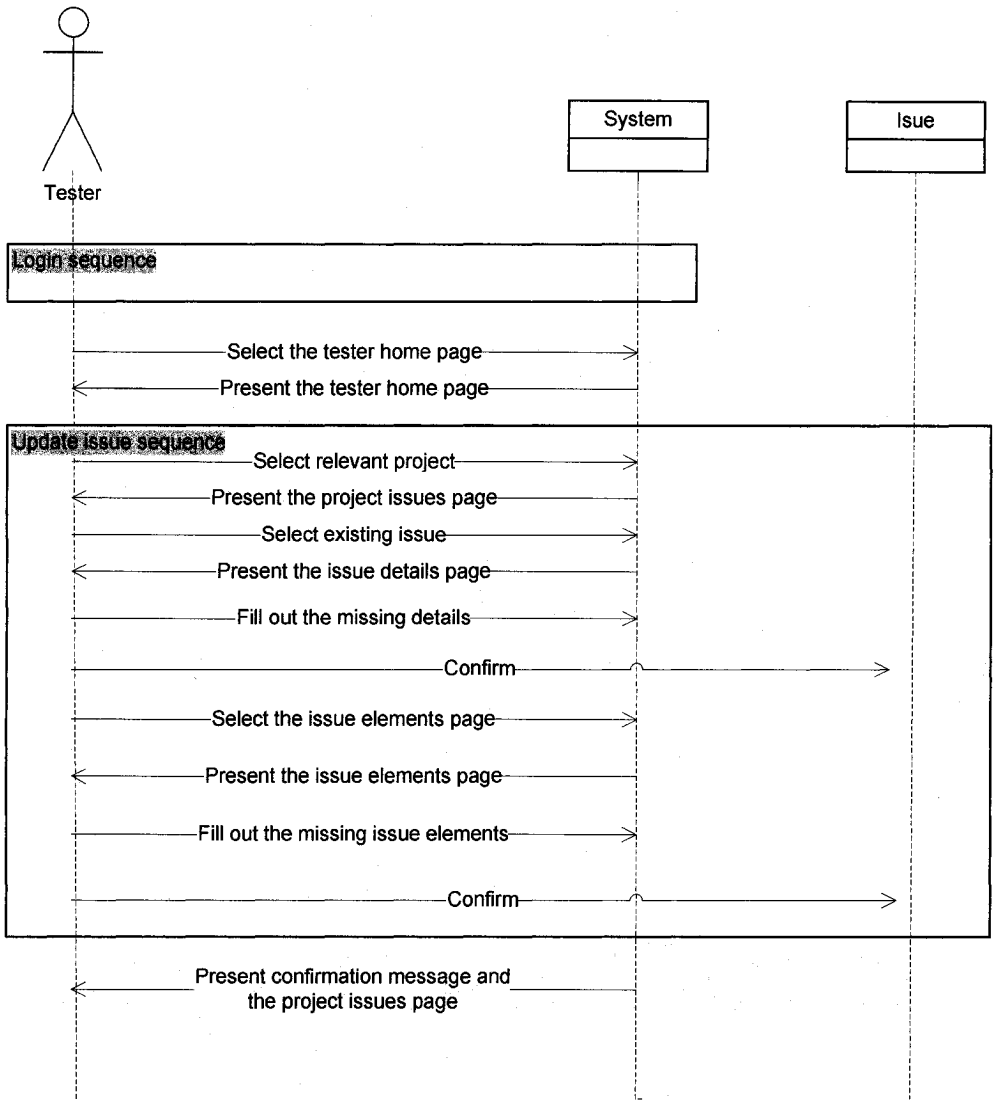

Figure 4-13 Use case: formal user updates usability issue.

4.3.3.3 Use case name: formal user updates profile.

Use case short description: the user logs in to USITRES and modifies certain characteristics of his profile.

Pre – conditions:

- The tester had its profile created in USITRES.

Actor's goal: the user appends or modifies information related to his or hers own profile. A manager first creates the user profile with limited information, mostly contact

information. The user wants to make available information about his attitude towards computers, a particular application and other details that the manager was not aware of when the profile was originally created. Additionally some user characteristics might change over time such as marital status, household income, experience in a job position or with a particular application and these changes must be made known to the tracking system.

Narration: this use case starts when the user accesses the main USITRES login page. The system prompts for username and password. The tester enters the username and password. The system validates them and upon successful validation the system presents the user with the USITRES home page. The user navigates to the testers' home page. The user navigates to the profile page. The system presents the tester with a user details page. The details on this page are common to most actors in the system and include name, general demographic information and detailed contact information. The tester enters the information details that the manager was not aware of and signals the system that the user information is complete by telling the system to apply the changes. The tester navigates to the tester details page that allows the tester to enter tester specific information that will be used to define this his or hers persona by presenting him with multiple choice questions like "How do computers make you feel?", How have computers affected your job?", "Is the amount of time it takes to learn new software applications usually worth it?".

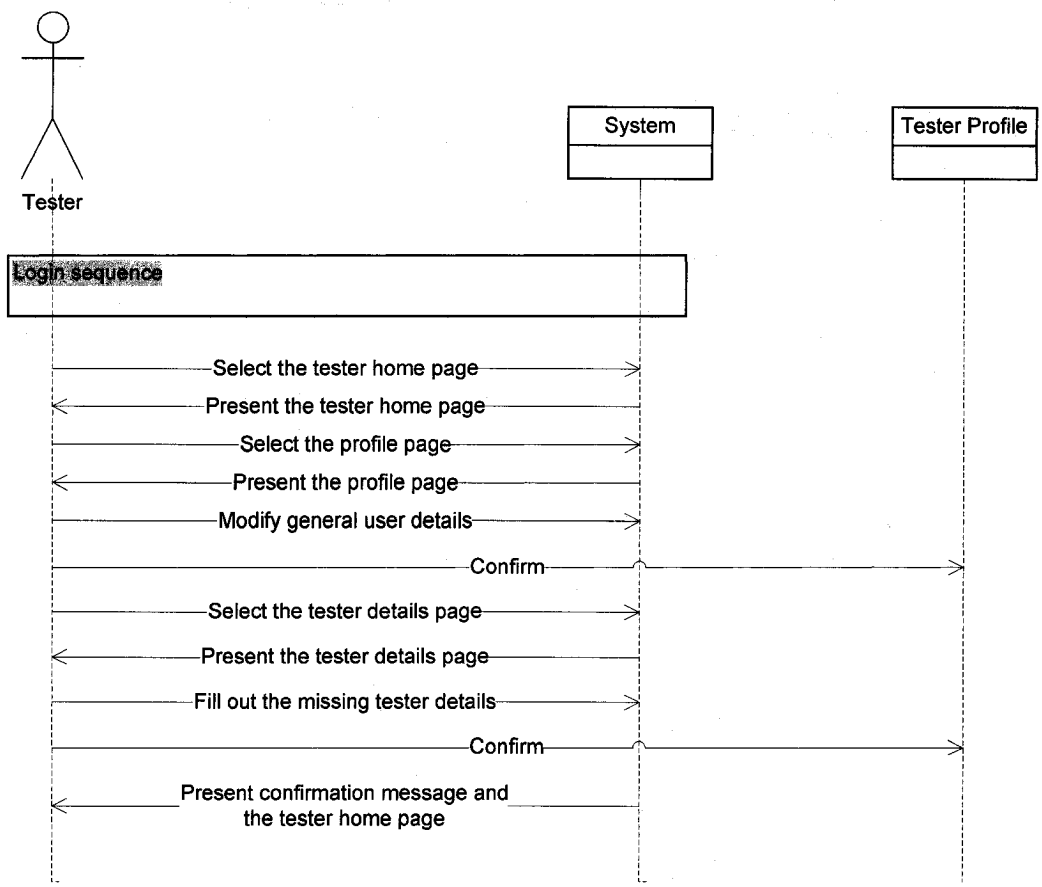

Figure 4-14 Use case: formal user updates profile.

The tester can belong to two groups of testers, the formal testers and the informal testers. The formal tester is a user that was formally identified by the project manager to be a tester and was assigned to one or more testing projects. The tester is a representative user of the product.

The tester will receive an automated email from USITRES with the tasks testing tasks assigned. The tasks can specify the use of the product in several ways:

- Test a specific feature of a product.
- Use a specific product in the day to day tasks.
- Conduct a specific task with the new product.

Upon identifying a usability issue the tester logs in to USITRES. Upon login he will be presented with a list of projects that he is assigned to work on. He will click on the link identifying the project that he identified the issue for and he will be taken to a page that lists all the usability issues associated with the current project. If the usability issue was just identified he will use the create button to navigate to the page that will allow him to create a new usability issue and associate it with the current project. He will fill out all the other text boxes that pertain to the details of the issue.

After the product goes through a new development iteration, the tester will receive another email announcing him that the product sustained changes. The user will attempt to reproduce the usability issue. He will connect to USITRES, select the correct project and on the subsequent page select the issue that he created on the last visit. He will then update the issue history by updating the history text field. He will describe the latest actions he performed in the tool and the changes that affected the usability issue, if it was solved or if it was just changed and it is still considered a issue.

The informal user, upon identifying a usability issue will navigate to the USITRES web page. On the first visit he will create a user profile that will identify the user characteristics, including a username and password. On the subsequent visits he will log in with the previously created username and password. Due to the fact that the informal user is not assigned to any specific project and to the fact that usability issues cannot exist outside a virtual, all inclusive project will be created for each informal user. The user will click on the project link and will be taken to the issue page. On this page he will be able to create usability issues or modify the existing ones similarly to the formal user.

4.3.4 From the manager view

From the manager view we will be presenting how a manager creates a tester or developer profile and how the manager assigns an issue to a tester or developer.

4.3.4.1 Use case name: create tester (or developer) profile.

Use case short description: the manager logs in to USITRES and creates a tester (or developer) profile.

Pre – conditions:

- The manager profile was created.

Actor's goal: to create an USITRES user of the tester or developer type. A pre-condition for a tester or developer to log in and interact with USITRES is for this profile (USITRES user) to be created by a manager. By creating a profile, the manager fills this pre-condition.

Narration: The system prompts for username and password. The manager enters the username and password. The system validates them and upon successful validation the system presents the user with the USITRES home page. The user navigates to the manager home page. The user navigates to the tester (or developer) page. The system presents the manager with a list of system testers (or developers) as well as the options to modify, remove or create testers (or developers). The manager select the create option. The system presents the manager with a user details page. The details on this page are common to most actors in the system and include name, general demographic information and detailed contact information. The manager enters the information details that he is aware of and signals the system that the user information is complete by telling the system to create the user.

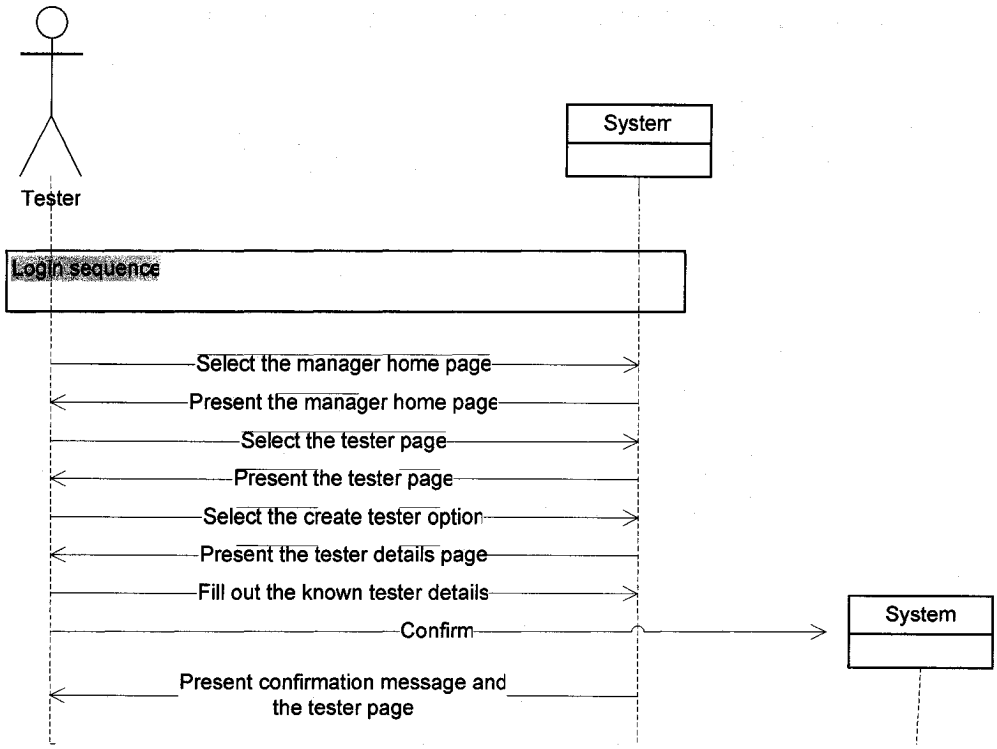

Figure 4-15 Use case: create tester (or developer) profile.

4.3.4.2 Use case name: assign usability issue to a tester (or to a developer).

Use case short description: the manager logs in to USITRES and assign a usability issue to a tester.

Pre – conditions:

- The manager, tester and developer profiles were created.
- A project for this particular application was created.
- An issue in this project was created.

Actor's goal: designate a tester to test a usability issue or a developer to find and implement a resolution.

Narration: The system prompts for username and password. The manager enters the username and password. The system validates them and upon successful validation the

system presents the user with the USITRES home page. The manager navigates to the projects page. The system presents the manager with a list of projects that he is in charge of. The manager selects the project that the particular issue belongs to. The system presents the manager with a list of all the issues logged for this project. The manager selects one of the issues from the list. The system presents the user with the details of the issue. From the details page the manager navigates to the tester (or developer) assignment page. The system presents the manager with a list of eligible testers (or developers). Upon confirmation of his or her selection the manager is redirected to the issue details page. The system sends notification emails to the tester (or developer).

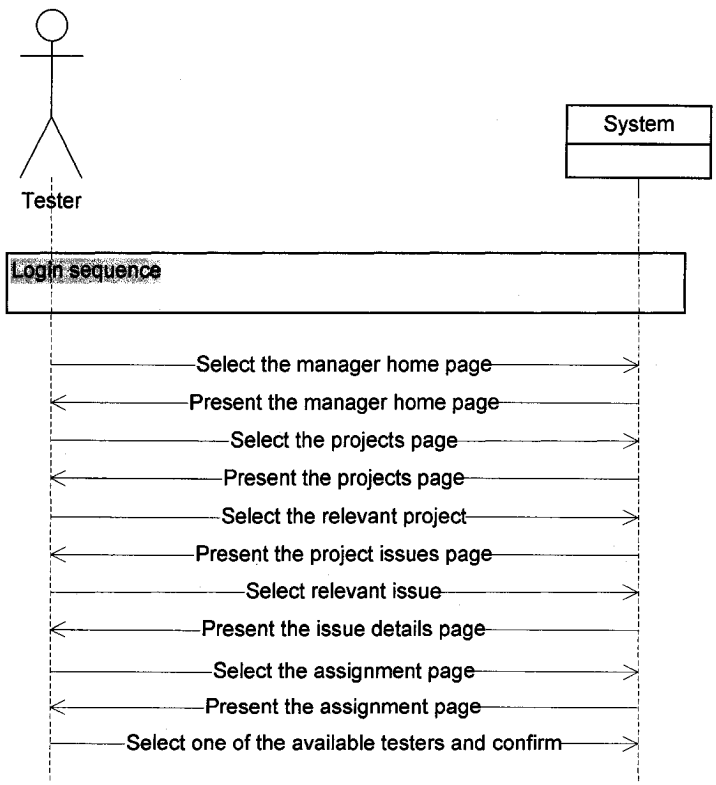

Figure 4-16 Use case: assign usability issue to a tester (or to a developer).

The manager can accomplish a variety of tasks in USITRES:

Create or modify formal user.

Create or modify a developer.

Modify a usability project.

Create or modify a usability issue: the manager can create a usability issue. In the case a usability was identified by somebody else other than a tester, such as a client or any other stake holder, it will be up to the manager to log the usability issue in the system.
Assign a usability issue to a tester.
Assign a usability issue to a developer.
Track the progress of solving usability issues and approve solutions as well as estimates from developers.
Execute reports and identify global issues and solutions and convey them to the client.

4.3.5 From the developer view

From the developer view we will be presenting how a developer updates an usability issue.

4.3.5.1 Use case name: update usability issue.

Use case short description: the user logs in to USITRES and modifies certain characteristics of a usability issue previously logged.

Pre – conditions:

- The usability issue was previously logged.
- The tester had its profile created in USITRES.
- A project was created for this application.
- The tester was assigned to a project.

Actor's goal: indicate to the system that progress is being made in issue resolution. Upon completing a development cycle the developer lets the other actors involved with this particular issue, the tester and the manager, that they can proceed with the next testing cycle or other necessary action.

Narration: this use case starts when the user accesses the main USITRES login page. The system prompts for username and password. The developer enters the username and password. The system validates them and upon successful validation the system presents the user with the USITRES home page. The user navigates to the developers' home page. The user navigates to the projects page. The system retrieves and presents the user with selected details of assigned projects on the projects page. The user selects the project that this particular issue is relevant to. The system retrieves and presents the user with a list of previously logged usability issues and the options of modifying them or creating a new usability issue. The user selects one of the listed issues. The system fills out the characteristics form where the user can modify details such as issue description, solution proposed, risks, number of clients affected. After the user modifies the relevant characteristics he or she confirms that the modifications are complete by indicating to the system to apply the changes. As a developer, the user will modify most of the time the resolution history, the issue status, the solution proposed filled and other similar details related to the resolution of the issue.

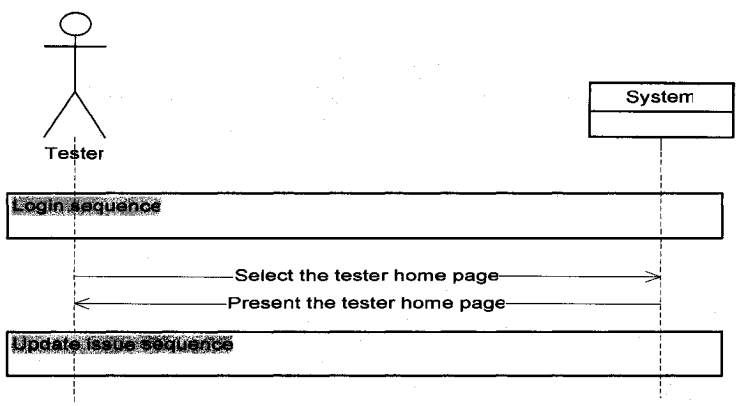

Figure 4-17 Use case: update usability issue.

4.3.6 From the super-user view

From the super-user view we will be presenting how he/she will create an USITRES project, a manager profile, how to assign a project to a manager and how to review and remove if needed some of the recently uploaded documents in the library.

4.3.6.1 Use case name: create a project.

Use case short description: the super-user logs in to USITRES and creates a project.

Pre-conditions: none.

Actor's goal: create an USITRES artefact that will group usability issues, testers, developers and managers relevant to a particular application.

Narration: The system prompts for username and password. The super-user enters the username and password. The system validates them and upon successful validation the system presents the user with the USITRES home page. The super-user will navigate to his home page. The super-user will navigate to the projects page. The system will present him with a list of already created projects as well as the option to create a new project. The super-user selects the option to create a new project. The system will redirect the user to the project details page where he will be able to enter project details.

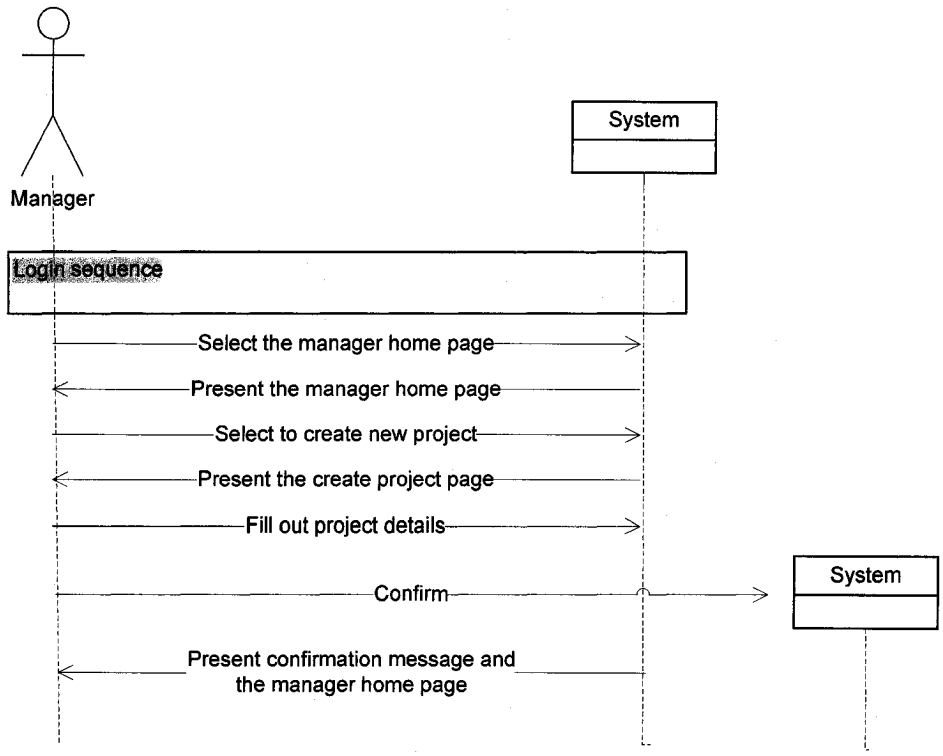

Figure 4-18 Use case: create a project.

4.3.6.2 Use case name: create manager.

Use case short description: the super-user logs in to USITRES and creates a manager.

Pre-conditions: none.

Actor's goal: create a manager type user. While a manger user can create tester and developer type users, only a super-user can create a manager type user.

Narration: The system prompts for username and password. The super-user enters the username and password. The system validates them and upon successful validation the

system presents the user with the USITRES home page. The super-user will navigate to the managers page. The system will present him with a list of already created manager profiles as well as the option to create a new manager profile. The super-user selects the option to create a new manager profile. The system will redirect the user to the project details page where he will be able to enter manager details, such as name, contact information and general demographics.

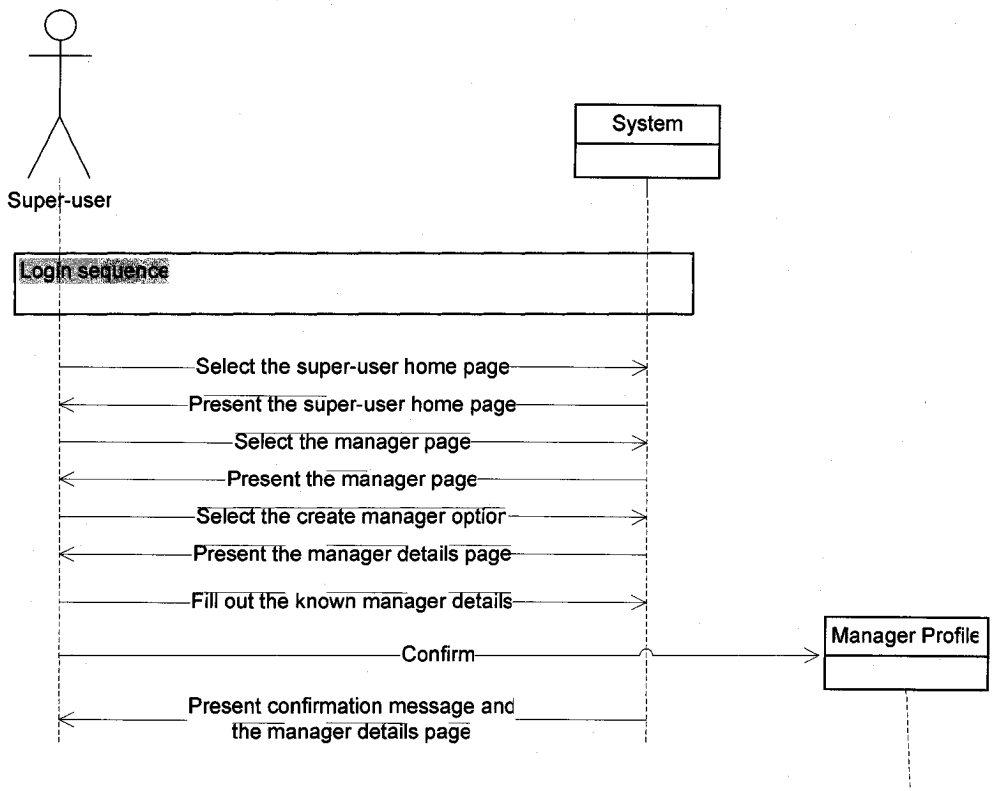

Figure 4-19 Use case: create manager.

4.3.6.3 Use case name: assign project to manager.

Use case short description: the super-user assigns an existing project to an existing manager.

Pre-conditions:

- The tester profile was created.

- The project was created.

Actor's goal: designate a manager to manage a usability project

Narration: The system prompts for username and password. The super-user enters the username and password. The system validates them and upon successful validation the system presents the user with the USITRES home page. The super-user will navigate to his home page. The super-user will navigate to the projects page. The system will present him with a list of already created projects. The user selects the project that will be assigned to a manager. The system presents the user with the details of the project. From the details page the user navigates to the manager assignment page. The system presents the user with a list of eligible managers. Upon confirmation of his or her selection the super –user is redirected to the project details page. The system sends notification emails to the manager.

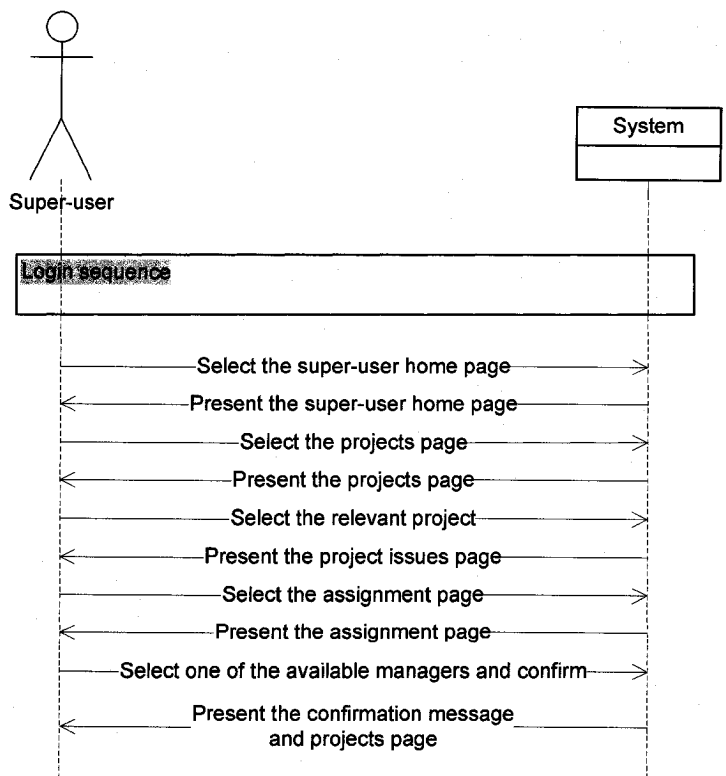

Figure 4-20 Use case: assign project to manager.

4.3.6.4 Use case name: review latest uploaded documents and delete if needed.

Use case short description: open an uploaded document and delete it from the database.

Pre-conditions:

- a document was uploaded to the USITRES web site.

Actor's goal: validate that the uploaded document are relevant to the usability field.

Narration: The super-user logs in the USITRES database. He uses sql to query and sort the documents by uploaded date. He then logs in to the USITRES web page. The system prompts for username and password. The super-user enters the username and password. The system validates them and upon successful validation the system presents the user with the USITRES home page. The user navigates to the Library and Warehouse home page. The user navigates to the library web page. He inspects the latest uploaded documents as detailed by the results of the query executed in the database by downloading them and opening them on his client computer. In the case they are deemed not relevant to the field of usability the super – user logs in the database and deletes those documents using SQL delete statements.

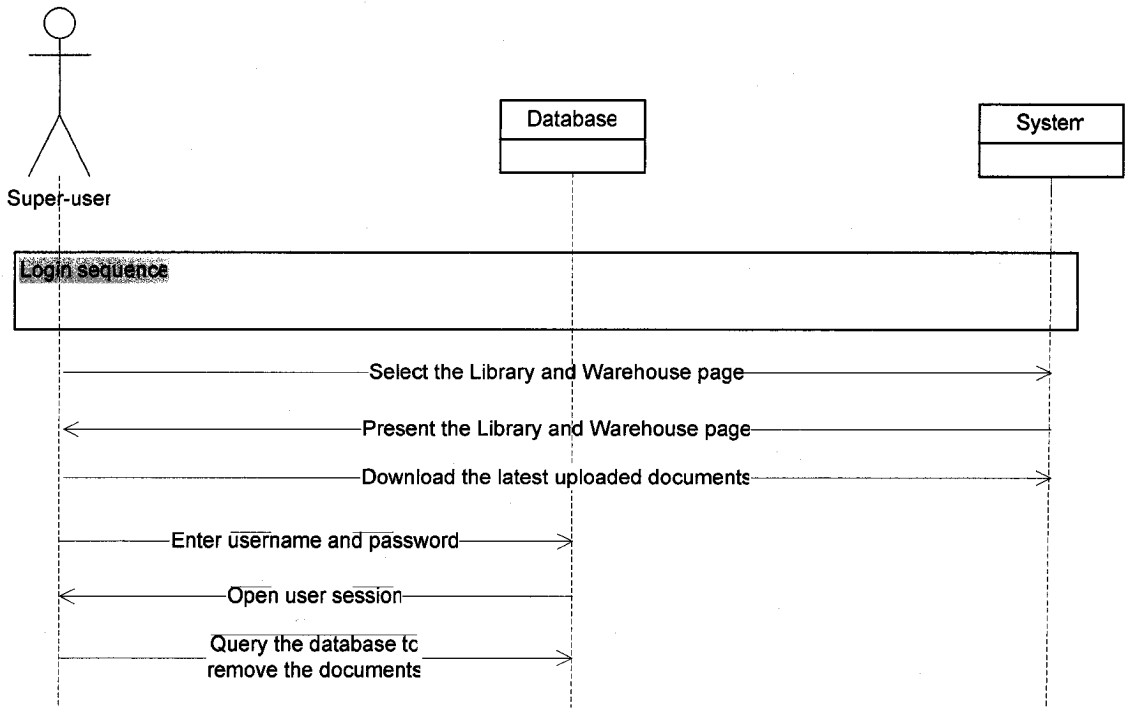

Figure 4-21 Use case: review latest uploaded documents and delete if needed.

All the other users will only have access to a subset of all the issues logged into USITRES, namely the ones that are assigned to them, created by them or managed by them. The super user will have access to all the issues. He will also have access to reports on system wide data. These reports can be usability reports on all the issues that are logged in the system. Upon reviewing them the super user will post some of them on the reports page. These reports are executed or scheduled by the super user at times of low system usage as they can place a performance burden on the whole system. There will also be reports on the web site traffic as well as a wide array of database use and performance. This is another advantage of database centered solution as we can track the usage of the system down to the relevant system data.

4.3.7 From the library user view

From the library user view we will be showing how he/she can upload and download an usability document from USITRES.

4.3.7.1 Use case name: upload document.

Use case short description: upload a document from the user computer to the USITRES web site.

Pre – conditions: none.

Actor's goal: make a usability document available to the usability community.

Narration: The system prompts for username and password. The library user enters the generic username "guest" and identical password. The system validates them and upon successful validation the system presents the user with the USITRES home page. The user navigates to the Library and Warehouse home page. The user navigates to the library web page. The user will be allowed to browse his client computer and select the file to be uploaded. The user will then fill out information on the subject, author, and document name. Upon confirmation of the information the system will upload the document and store it in its database.

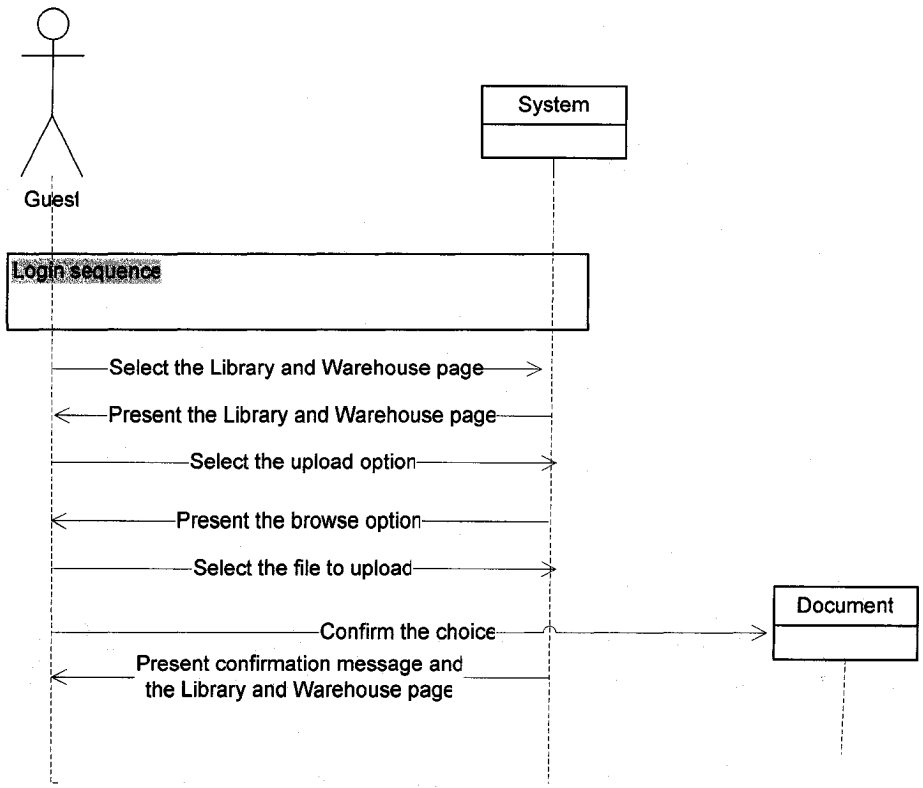

Figure 4-22 Use case: upload document.

4.3.7.2 Use case name: download document.

Use case short description: browse the document library and download a document.

Pre – conditions: none.

Actor's goal: save a usability document on the client computer for viewing.

Narration: The system prompts for username and password. The library user enters the generic username "guest" and identical password. The system validates them and upon successful validation the system presents the user with the USITRES home page. The user navigates to the Library and Warehouse home page. The user navigates to the library web page. The system presents a list of all available documents. The user can sort the

documents by subject, author, title or name. The user chooses to download a particular document. The system presents him with a choice for a download location on the user computer. Upon user's confirmation of the download location, the system will start the download operation.

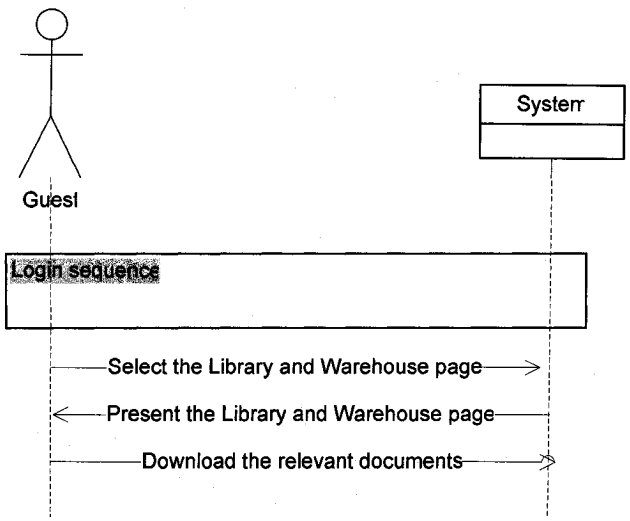

Figure 4-23 Use case: download document.

4.3.8 Scenario of usage

Through the following scenario of usage we will summarize the above use cases by following an issue from creation to resolution together with the creation of all relevant actors and entities.

The super user will be contacted by an external stakeholder with the purpose of tracking usability issues relevant to a new software product.

The super user will create a project manager system user. The project manager system user will create an empty usability issue grouping in the form of an usability project. The project manager will create a tester system user and a developer system user by filling in available demographic information.

The tester will log in and complete its profile by adding more demographic characteristics and tester specific details such as attitude towards computers. They will then log in and create a usability issue by filling in short description and a detailed description, number of clients affected, business minute and volume lost due to this particular usability issue etc. . The tester will then complete the issue by filling in the details of the importance of the various issue elements and his/hers satisfaction with the current implementation of the feature. At this point the issue has a status of "new".

The project manager will log in and assign the issue to a developer. At this point the issue will have a status of "assigned".

The developer will log in and complete its profile by adding more demographic characteristics and developer specific details such tool and industry experience. It will then inspect the issue details and propose a solution or several solutions together with their risks and benefits. At this point the issue will have a status of "pending".

The project manager will log in and select a proposed solution and reassign the issue to the developer.

The developer will log in and after taking note of the selected solution will begin to implement it. At this point the issue will have a status of Work in progress. After completing the work the developer will log in and change the status of the issue to "resolved".

In the final step the tester will log in, acknowledge the implemented solution and test it out. Upon satisfactory results the user will change the status of the issue to "closed".

The data stored in the oracle database will become inherently part of the system wide usability reports that will be made available to the entire usability community.

5 Chapter 5 – USITRES Versions, Validation and Limitations

The planned USITRES functionality will be implemented in several versions. We will present the validation of the current version through expert evaluation, its limitations and what it is planned for the future of USITRES.

5.1 USITRES Versions

USITRES will be completed in three versions. The first version was developed during this thesis creation and the subsequent versions will follow the directions described bellow but will also be based on expert evaluations of the previous versions and user feedback.

- Version one: Select the technologies to be used and create the implementation platform. Implement USITRES which will contain the standard user logging interface and other supporting modules such as the project manger and developer interface. Using the standard interface, the user will be able to fill in pre-defined fields that will allow him to define and log a particular usability issue. This same phase will also a library repository for usability documents that the whole usability community can see and use.
- Version two: implement a wizard driven logging interface where the user will be able to, at the same time, design the form that he will be filling and fill it with the details of the usability issue. This version will include a report interface that will display both project specific and system general reports.
- Version three: implement data mining and business intelligence reports that will allow high level USITRES users to forecast some of the usability issues that might arise with new applications, based on their own and historical USITRES data, and then use one of the solutions that was declared as acceptable in USITRES for similar previous issues. With an increased data volume and report

validation, these report will become increasingly effective, helping software designers to release usability issues free first version applications.

5.2 USITRES Validation

The system will be validated by being loaded with several usability issues logged during other usability studies. Their resolution will be simulated by logging in with the different system actors and going through the various status states until the issue is closed with one of the closure codes.

The process was evaluated based on the same ontology that was implemented in USITRES.

Most labels through out the application have links to pop-up help pagelets. This feature is very helpful as the relevant help pages are displayed without any need for searching through user manuals. Through this feature many usability notions are explained to the user such that the user becomes increasingly familiar with the field as a whole. Some options that are available to the user, such as the issue factor criteria ratings, are a bit too advanced for novice users and they might be slightly intimidating for novice users. These features were implemented to be accessible for advanced users and for developers and managers but they should be better identified as optional for novice. In general more features should be clearly identified as mandatory or optional. Mapping from other surveys is effort intensive and open to interpretation when access is only available to the input data. When inserting data collected during other surveys like we did with the NCIB web site usability study, some of the data collected doesn't automatically fit in the ontology that we developed. At that point, interpretation of the results is needed to be able to mould them into our ontology. This interpretation process is open to the same type of errors as the assumption based user identification process that is currently common to various lab conducted study. The application top two level navigation is much more explicit than the side list navigation by better identifying the area that the user is currently accessing. The sharing of layout between all actors, managers, and testers allows us to fix

common problems quickly but removes the possibility of customization. Therefore some advanced options are presented to users and irrelevant options are available to managers such as tester gender which is only relevant in reports.

The application was also evaluated through expert heuristic evaluation on the same criteria that are available to the tester in USITRES. In the following tables the criteria is showed in normal font and the evaluation is presented in italics. We will present the most relevant findings.

In the first table we will present findings that became apparent during completion of several tasks.

Issue Type
Instructional and technical
Accuracy of information and instructions.
The task information displayed is limited because of lack of space, the user might have to scroll up and down the page.
Fonts
Consistency of style.
The style is 100% consistent due to the use of templates.
Visual clutter resulting from use of multiple fonts in a single document; question of availability of fonts on the targeted platforms.
Some pages contain too many elements; they might interfere with task focus.
Colors
Suitability of font colors.
The red colour theme can prove offensive in some cultures. A more subtle gray theme will be used in subsequent versions.
Haphazard use of color can be negative and confusing.
The interface is very consistent and that will reduce learning time as similar features have similar implementation.

Which one of these guideline does the user feel the particular issue violates (Nielsen (1994)) :
Use simple and natural dialog.
The application doesn't always use simple and natural dialog but all terms are explained through help pages.
Speak the users' language.
Doesn't speak the users language but tries to educate the user.
Provide feedback.
Messages are clear but generalized.
Provide shortcuts.
It provides good access to all pages; however the on the page navigation is less than optimal.
Provide good error messages.
The error messages are Oracle error messages, irrelevant for the user. An interpretation mechanism is needed.
Include good help and documentation.
The application has very good help features.
Which error type is more typical of this particular issue with operational features:
The form or dialog box is too wide or long under minimum support display resolution (e.g., 800 x 600).
The pages resize very well when the browser is resized.
Invalid inputs are not detected and handled.
Invalid inputs are not allowed through the use of radio buttons and check box groups which restrict user choices.
Which error type is more typical of this particular issue with error handling:
Messages were not written for the user and, therefore, are not useful to the user. For example, "Driver error 80004005."
Error messages are Oracle generated and only relevant for the DBA.
Error message is not specific nor does it offer a plausible solution.

Error messages don't offer solution.
Similar errors are handled by different error messages.
Because errors are Oracle generated they are similar for similar problems
Images (web only)
Legibility of button images.
Most buttons are just linked text therefore very legible.
Suitability of size of images.
The application doesn't use any images to help the user focus on his task.

Table 5-1 Task Related Issues

In this second table we will present findings that transpire through the use of the whole application.

How well does this application
Perform.
Load time.
The application loads very fast
Uptime.
The application or database server doesn't crash by itself but access is sometimes interrupted because of complex network configuration.
Allows for User Control
Allow you to cancel commands you have initiated but don't wish to execute, without undesirable side effects?
The application has no undo feature, if details are modified there is no way to get back to previous version of the record.
Allow you to bypass irrelevant steps and get efficiently to the menu, field, or function you want?
The testers have 3 click access to any page from the home page.
Allow abbreviated input of commonly used sequences of commands?
No command line language is available but not necessary because users are mostly

novice.
Facilitates Learning
Facilitate learning with a minimal amount of training and reference to manuals?
Facilitates learning through quick links to pop-up help pages that explain all the terms.
Facilitate learning for users with no prior computer experience?
Some web experience is needed for the use of the various artefacts for navigation and user input.
Facilitate remembering how to use it from one session to another?
All preferences are session based therefore no preferences are remembered from one session to another.
Facilitation Task Completion
Help you to perform your job more efficiently and effectively?
Yes, if you are a tester, because the tool is efficient and accessible.
Help you to improve the quality of your work?
Yes, because it is consistent.
Consistency and Predictability
Behave similarly and predictably in similar situations
The application is very consistent.
Provide equal response times for similar activities?
Very consistent response time as most operations are accomplished by record insert and the time is similar.
Flexibility in Task Handling
Allow the experienced user to define his or her own set of functions?
Not in this release.
Provide shortcuts for the experienced user to perform tasks?
The application's consistency was implemented at the expense of flexibility, therefore no customization was implemented for various user types.

Table 5-2 Application Related Issues

Other findings were made during the development process and refer to the development platform. Although they are not specific to the current application they are still relevant to future versions of USITRES. When developing the so called tabular forms, forms that allow users to modify all columns in a table, radio groups not available for displaying the available choices. In this case custom forms must be developed and they are much more effort intensive than tabular forms. This is one reason to upgrade the application to APEX version 3. Only one table can be modified per application page, therefore some items that are logically related will have to be modified on separate pages. The creation of records is much more complicated then modifying them because all the join with child tables must be populated. Some of the procedures used through out the application were implemented through application features and some other ones have been implemented in the database through triggers and PL/SQL procedures. In the second release we will attempt to implement all features through database components for uniformity and functionality. The development process should be documented as well as the interface for support and repeatability.

Overall the application is well suited for the issue that is was set to solve. The interface is seamless, the tasks are clear and the resolution path is obvious. The help features are relevant and readily available. The performance was satisfactory with fast web page loads and no fatal failures. The navigation is clear and the user can easily see where he is on the web page and how to navigate to other areas of the web page.

6 Chapter 6 – Conclusion, Contribution and Future Work

6.1 Conclusion

The current usability tools have several limiting characteristics such as the fact that they are either designed for usability professionals or the fact that they are prohibitively expensive for most users. They also have to be installed on the users' machine which introduces more technological constraints on the users.

The current usability methods call for heavy involvement of usability experts either in user and test selection and design or in the actual usability testing being performed exclusively by experts such as in expert or heuristic evaluation. Most importantly the experts are called upon to interpret the results of usability measurements.

The literature recognizes the fact that currently the actual users are left out from the usability testing process and recommends a greater involvement of the users in the process and a greater recognition of their various cultural and technological backgrounds.

In this thesis we showed through our tool review that existing tools are fragmenting the usability information by storing it in various formats and they are not very portable due to the fact that they must be installed on the client computer. Through our literature review we showed the need for the usability field to be more user centered, focused on finding who the users are instead of assuming who they are. We also showed, based on the literature review, that usability should be driven by user perceptions and not by abstract measurements and that it should be evaluated in the user environment. We then developed a common usability vocabulary to be used by tester, developers and usability professionals in the form of an usability ontology.

6.2 Main Contribution

The main contributions of this thesis to the field of usability are the ontology, the web based tool and the comprehensive methodology for collecting and managing usability problems.

The ontology will provide the common language that will be used by usability professionals and regular users to communicate their usability issues. The ontology will remove the need for professionals to have to interpret abstract usability measurements and will directly record the users opinions about the software or web site tested. By mapping the existing data from usability studies into our ontology we homogenize the existing usability information.

The web-based tool will provide the communication platform between all usability stakeholders. The tool will eliminate the existing barriers, geographical and financial, to mass usability testing by implementing a remotely accessible usability testing platform. More importantly the tool will allow the actual users to identify themselves doing away with the artificial and potentially erroneous tester identification process presently used in lab conducted usability tests.

The issue resolution methodology will clarify and accelerate the process of collecting and managing usability problems by providing an explicit and repeatable path from the stage of issue discovery to the stage of issue closing.

The major contribution of this thesis is to show that it is possible to deliver a unified solution by implementing a user centric ontology on a web enabled platform in an application that provides a very accessible tool to all usability stakeholders. The USITRES application delivers in the users' environment a solution that tracks and stores usability issues as well as a general usability knowledge repository.

6.3 Future Work

In the first production iteration the site will be released for all Concordia students to be used to log usability issues related to university software and web sites. The usability data will be used by the various computer departments such as IITS and CS to fix various usability issues. The application reporting component will be implemented and used for mining the data that was collected.

The tool will be managed by the HCSE group. The indirect usage data as well as the direct feedback received through logged usability issues on USITRES itself will help shape the tool in an enterprise strength usability tool. Functionality implementation will be moved from the application to the database for consistency and efficiency.

The second production iteration will include bug fixes as well as a feature of customizable reports. With this feature, the testers will use wizards to create the reports that will the be best suited to the particular issue that they are logging or to the particular software that they are testing. The ontology will be modified to allow for custom usability ratings.

The full potential of USITRES will only be obviously expressed once it will be used by numerous users with various experiences and backgrounds. Such a study will be performed with the support of the HCSE group. The nature of its implementation and of the problems that it solves limits the type of testing that can be performed in isolation from a large user community. The full potential of this mass usability tool can only be manifested during its use by a sizeable user community. Additionally, any limited user group, that would have been selected for testing, would have been identified artificially. As confirmed by our literature review, we have to allow the end users to identify themselves as opposed to making assumption about who will they be. This model however incurs the added risk that only the very satisfied or the very unsatisfied users as well as "professional" testers will log in to record their opinions.

The purpose of usability testing is to determine the impression of a software product on a user, and it is much more effective to gather these impressions directly from the actual users, than put the usage through a microscope of formulas and measurements and then have usability experts extract meaning from the results of these formulas and measurements like other existing usability tools do. USITRES will be the inclusive tool that is needed to take usability testing out of the lab and into the real world where the actual users are. USITRES will focus usability measurements on users' lasting impressions of a software product and not on abstract formulas and arbitrarily chosen measurements subject to expert interpretation.

References

Antunes, HM *Improving and mediating usability-to-software engineering communication* 2001 Masters Thesis.

Ash, L *The Web Testing Companion: The Insider's Guide to Efficient and Effective Tests* John Wiley & Sons © 2003 ISBN:0471430218

Barrier, T *Human Computer Interaction Development and Management* (ed) Idea Group Publishing © 2002 ISBN:1931777136

Bauersfeld, P *Software by Design: Creating People Friendly Software* (New York, M&T Books, 1994)

Beyer, H, Holtzblatt, K *Contextual Design: Defining Customer-centered Systems* (ed) Morgan Kaufmann 1998

Brinck, T, Gergle, D & Wood, SD. *Usability for the Web: Designing Web Sites That Work* Morgan Kaufmann Publishers © 2002 ISBN:1558606580

Brust J, Leunissen S *Rapid Web Application Development With Oracle Application Express* Oracle

Buxton, W, Grudin, J, Greenberg, S, Baecker RM. *Readings in Human-Computer Interaction* (ed) Morgan Kaufmann 1995

Campbell, K *E-Ffective Writing for E-Learning Environments* By Published by Idea Group Inc (IGI) 2003

Castillo, JC, Hartson, HR & Hix, D *REMOTE USABILITY EVALUATION AT A GLANCE* Virginia Polytechnic Institute and State University Blacksburg 1997, VA 24061-0106 josec@hci.ise.vt.edu, {hartson, hix}@vt.edu

Cato, J *User-Centered Web Design* (2001) London: Addison-Wesley

Chen, Q *Human Computer Interaction: Issues and Challenges* Idea Group Publishing © 2001 ISBN:1878289918

Cogswell, J *Designing Highly Useable Software* Sybex © 2004 ISBN:0782143016

Colby, J *Practical Intranet Development* Glasshaus © 2003 ISBN:190415123X

DiMarco, J *Web Portfolio Design and Applications* (ed) Idea Group Publishing © 2006 ISBN:1591408547

Discount Usability; Usability Metrics, retrieved March 18th from http://www.userfocus.co.uk/articles/discount.html

Faulkner, C *The Essence of Human – Computer Interaction* Prentice Hall Europe 1998 ISBN 0-13-751975-3

Fazlollahi, B *Strategies for eCommerce Success* (ed) Idea Group Publishing © 2002 ISBN:193177708X

Dr Feher, P *Poceedings of the 7th European Conference on Knowledge Management* (ECKM 2006) Corvinus University of Budapest, Hungary Published by Academic Conferences Limited

Fink A, Kosecoff JB *How To Conduct Surveys: A Step-By-Step Guide* (ed) Sage Publications Inc 2005

Friedlein, A *Maintaining & Evolving Successful Commercial Web Sites: Managing Change, Content, Customer Relationships, and Site Measurement* Morgan Kaufmann Publishers © 2003 ISBN:1558608303

Greenbaum, TL *Moderating Focus Groups: a practical guide for group facilitation* (ed) Sage Publications Inc 2000

Hall, LM & Bodenhamer, BG *Figuring Out People: Design Engineering with Meta-Programs* Crown House Publishing © 2000 ISBN:1899836101

Hausman, A *Journal of Business & Industrial Marketing: Doing Business in a Digital World, Volume 20, Number 4/5, 2005* (ed) Emerald Group Publishing © 2005 ISBN:184544325X

Holmes, M *Web Usability & Navigation: A Beginner's Guide* McGraw-Hill/Osborne © 2002 ISBN:0072192615

Hom J, *The Usability Methods Toolbox* retrieved January 21, 2007 from http://jthom.best.vwh.net/usability/

Idea Bundle by Giga Information Group Giga Information Group, Inc. © 2001

International standards for HCI and usability retrieved January 21, 2007 from http://www.usabilitynet.org/tools/r_international.htm#9241-11

Jenz, DE *Strategic White Paper Ontology-Based Business Process Management The Vision Statement* President, Jenz & Partner Gmbh First Edition November, 2003 – Draft

Johnson, J *GUI Bloopers: Don'ts and Do's for Software Developers and Web Designers* by Morgan Kaufmann Publishers © 2000 ISBN:1558605827

Johnson, J *Web Bloopers: 60 Common Web Design Mistakes and How to Avoid Them* Morgan Kaufmann Publishers © 2003 ISBN:1558608400

Karwowski, W *International Encyclopedia Of Ergonomics And Human Factors* By, Published by CRC Press 2006

Khosrowpour, M *Web Applications—Web Usability Encyclopedia of Information Science and Technology*, Volume V (ed) IGI Publishing © 2005 from Books 24x7 database

Khosrowpour, M *Challenges of Information Technology Management in the 21st Century: 2000 Information Resources Management Association International Conference* (ed) Idea Group Publishing © 2000 ISBN:1878289845

Krishnamurthy, S *Contemporary Research in E-Marketing, Volume 2* Idea Group Publishing © 2006 ISBN:1591408245

Kuniavsky, M *Observing the User Experience: A Practitioner's Guide to User Research* Morgan Kaufmann Publishers © 2003 ISBN:1558609237

Kurosu, M *A Cultural Comparison of Web Site Design from a Usability Engineering Perspective* National Institute of Multimedia Education

Lazar J *User- Centered Web Development* By Published Jones and Bartlett Publishers 2001

Lindroth T, Nilsson S *Contextual Usability, Rigour meets relevance when usability goes mobile* , Laboratorium for Interaction Technology, University of Trollhättan 2001

Laroche, Michel *Journal of Services Marketing: Cultural Perspectives On Services Marketing, Volume 19, Number 3, 2005* (ed) Emerald Group Publishing © 2005 ISBN:1845441435

Mayhew, DJ *The Usability Engineering Lifecycle: A Practitioner's Handbook for User Interface Design* Morgan Kaufmann Publishers © 1999 ISBN:1558605614

Mohamed Z *International Journal of Quality & Reliability Management—Total quality management sustainability, Volume 19 Number 5 2002* (ed) Emerald Group Publishing © 2002 ISBN:0861767047

Mueller, J *Accessibility for Everybody: Understanding the Section 508 Accessibility Requirements* Apress © 2003 ISBN:1590590864

Nelson, SL. & Coleman, P *MBA's Guide to the Internet* Redmond Technology Press © 2000 ISBN:0967298164

Nguyen, HQ, Johnson, B and Hackett, M *Testing Applications on the Web: Test Planning for Mobile and Internet-Based Systems*, Second Edition by John Wiley & Sons © 2003 ISBN:0471201006

Nielsen, J *Usability Engineering* (ed) San Diego, Academic Press, 1993

Nielsen J *Usability Engineering* (ed) Morgan Kaufmann 1994

Nielsen, J *Coordinating User Interfaces for Consistency* (ed) Morgan Kaufmann Publishers © 1989 ISBN:1558608214

Nielsen, J *How to Conduct a Heuristic Evaluation* retrieved January 21, 2007 from http://www.useit.com/papers/heuristic/heuristic_evaluation.html

Nielsen, J *Usability 101: Introduction to Usability* retrieved January 21, 2007 from http://www.useit.com/alertbox/20030825.html

Oracle Internet Directory retrieved January 22nd 2007 from

en.wikipedia.org/wiki/Oracle_Internet_Directory

Oracle Application Server Single Sign-On retrieved January 22nd 2007 from http://www.oracle.com/technology/products/id_mgmt/osso/index.html

Oracle Real Application Clusters 10g retrieved January 24th 2007 from http://www.oracle.com/technology/products/database/clustering/pdf/twp_rac10gr2.pdf

Oracle Label Security 10g retrieved January 24th 2007 from http://www.oracle.com/technology/deploy/security/database-security/label-security/index.html

Oracle Spatial & Oracle Locator 10g retrieved January 24th from http://www.oracle.com/technology/products/spatial/index.html

Oracle Application Express retrieved January 24th from http://www.oracle.com/technology/products/database/application_express/pdf/Oracle_Application_Express_2.2_Datasheet.pdf

Oudshoorn, N & Pinch, T *How Users Matter—The Co-Construction of Users and Technology* edited by The MIT Press Cambridge, Massachusetts London, England

Oudshoorn, N & Trevor, P *How Users Matter: The Co-Construction of Users and Technology* (eds) The MIT Press © 2005 from Books 24x7 database ISBN:0262651092

Pavlichev, A & Garson, GD *Digital Government Principles and Best Practices* (eds) Idea Group Publishing © 2004 ISBN:1591402182

Pradeep, H *User-Centered Information Design for Improved Software Usability* Artech House © 1998 from Books 24x7 database

Ratner, J (ed) *Human Factors and Web Development* published by Lawrence Erlbaum Associates, London 2003

Redish, JC., Dumas JS. *A Practical Guide to Usability Testing* Intellect Books 1999

Riva G, Galimberti C *Towards Cyberpsychology: Mind, Cognition, and Society in the Internet Age* Published by IOS Press 2001

Rivero, LC, Doorn, JH & Ferraggine, VE *Encyclopaedia of Database Technologies and Applications - Ontologies and Their Practical Implementation* (eds) Idea Group Publishing © 2006 from Books 24x7 database ISBN:1591405602

Rosson, MB & Carroll, JM *Usability Engineering: Scenario-Based Development of Human-Computer Interaction* Morgan Kaufmann Publishers © 2002 ISBN:1558607129

Schultz, RA *Contemporary Issues in Ethics and Information Technology* Idea Group Publishing © 2006 ISBN:1591407796

Schwartz, DG *Encyclopaedia of Knowledge Management - Ontology* (ed) Idea Group Publishing © 2006 from Books 24x7 database. ISBN:1591405734

Seffah A., Donyaee M., Kline R., Usability and Quality in Use Measurments and Metrics: An Integrative Model, HCSE Group, Concordia University

Shneiderman, B *Leonardo's Laptop: Human Needs and the New Computing Technology* The MIT Press © 2002 ISBN:0262194767

Spool, JM *Web Site Usability – A designer's Guide* Morgan Kaufmann Publishers, Inc. 1998 ISBN 1-55860-569-X

Smith, C *PhotoShop and Dreamweaver Integration: Creating High-Impact Web Pages* (McGraw-Hill/Osborne © 2005) ISBN:0072255889

Snyder, C *Paper Prototyping: The Fast and Easy Way to Define and Refine User Interfaces* Morgan Kaufmann Publishers © 2003 ISBN:1558608702

Spolsky, J *User Interface Design for Programmers* Apress © 2001 ISBN:1893115941

Stefik, M *The Internet Edge: Social, Technical, and Legal Challenges for a Networked World* The MIT Press © 1999 ISBN:026219418X

Suh, W *Web Engineering: Principles and Techniques* (ed) Idea Group Publishing © 2005 ISBN:1591404320

Szewczak, E & Snodgrass, C *Human Factors in Information Systems* (eds) Idea Group Publishing © 2002 ISBN:1931777101

Travis, D *E-commerce Usability – tools and techniques to perfect the on-line experience;* Taylor and Francis (2002) 29 West 35th Street, New York, NY 100001 ISBN 0-415-25834-0

Tsui, E *Journal of Knowledge Management: Technology in Knowledge Management, Volume 9, Number 1, 2005* Emerald Group Publishing © 2005 ISBN:184544101X

Kowalski L *Web User Interface Development at Oracle: How the Evolution of the Platform Influenced Our Products' Human-Factors Efforts* in *Human Factors and Web Development* (ed) Julie Ratner Published by Lawrence Erlbaum Associates (2002)

Winters, CM., Norlin, E *Usability testing for library web sites: A Hands-On Guide* By Published by ALA Editions 2002

Wroblewski, L *Site-Seeing: A Visual Approach to Web Usability* by John Wiley & Sons
 ISBN:0764536745