

**USABILITY AND INTERNET-BASED BANKING**

by

**TOBIAS JAN VAN DYK**

**submitted in part fulfilment of the requirements  
for the degree of**

**MASTER OF SCIENCE**

**in the subject**

**INFORMATION SYSTEMS**

**at the**

**UNIVERSITY OF SOUTH AFRICA**

**SUPERVISOR: PROF P KOTZÉ**

**JUNE 1999**

## **Abstract**

Usability refers to the study and measurement of the quality of human-computer interaction. Internet-based banking is a task-oriented, goal-directed, Web-based, E-commerce application where users access and manipulate bank accounts remotely through a Web browser interface. The examination of the usability of this interaction through the distributed medium of the Internet formed the research basis.

The aims of the research are to review the application of general usability principles and properties to Internet-based banking, examine existing usability guidelines for hypermedia and Web design, and apply these during a structured evaluation of local and international online banking facilities. The research instrument is a combination of a checklist-based and heuristic evaluation, and is preceded by a task analysis.

Three research questions are addressed and solutions suggested. These are related to improving Internet banking usability, finding the best approach for measuring this, and creating an awareness that Web usability is important.

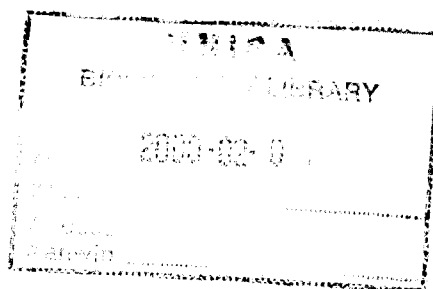
### **Key terms**

Usability, Human computer interaction, HCI, Hypermedia, Hypertext, Heuristic evaluation, Checklist evaluation, Task analysis, Task analysis for knowledge description, TAKD, Web page design, E-commerce, Internet banking, PC banking, Home banking, Online banking, Electronic banking, E-banking, Virtual banking.

I dedicate this to the loving memory of my parents Hannes and Mada van Dyk, who died tragically on the 9<sup>th</sup> of October 1998.

The author wishes to express his sincere appreciation to Prof Paula Kotzé for her patience, motivation, guidance, and support in formulating, executing, and completing this research project.

332.102854678 VAND



0001740161

# Contents

Chapter	Page
<b>Table of Contents</b>	1
<b>List of Figures</b>	5
<b>1 Introduction</b>	7
1.1 Introduction	7
1.2 The Research Dimensions	7
1.2.1 The Research Problem and its Importance	7
1.2.2 Research Objectives and Benefits	8
1.2.3 The Research Design, Instrument and Sample	8
1.2.4 Data Collection and Analysis	9
1.3 A Chapter Overview	9
1.4 Conclusion	11
<b>2 HCI and Usability</b>	12
2.1 Introduction	12
2.2 Human Computer Interaction	12
2.3 Usability	14
2.4 Usability Principles and Properties	17
2.4.1 Learnability	19
2.4.2 Flexibility	23
2.4.3 Robustness	24
2.5 The Graphical User Interface (GUI)	26
2.6 Dialog and Interaction Styles	26
2.6.1 Command Line Interaction Style	29
2.6.2 Windows Interaction Style	29
2.6.3 Direct Manipulation Interaction Style	31
2.6.4 Iconic Interface Interaction Style	32
2.6.5 Graphical Interaction Style	33
2.6.6 Menu Interaction Style	33
2.6.7 Form-filling Interaction Style	34
2.6.8 Programming Language Interaction Style	34
2.6.9 Natural Language Interaction Style	35
2.6.10 Function Key Interaction Style	35
2.7 Conclusion	36



<b>3</b>	<b>Usability for the Web</b>	37
3.1	Introduction	37
3.2	Web Usability	38
3.2.1	Learnability	42
3.2.2	Flexibility	42
3.2.3	Robustness	43
3.3	Web Usability in Practice	43
3.4	Conclusion	46
<b>4</b>	<b>Internet-based Banking</b>	48
4.1	Introduction	48
4.2	A Classification for Banks with a Web Presence	53
4.3	Banks offering Internet-based Banking	56
4.3.1	Standard Bank	56
4.3.2	First National Bank (FNB)	57
4.4	Conclusion	58
<b>5</b>	<b>Data Collection</b>	59
5.1	Introduction	59
5.2	Capturing Internet Banking Web Sites	60
5.2.1	Amalgamated Banks of South Africa	60
5.2.2	Bank of America (USA)	62
5.2.3	Nedbank and Netbank (South Africa)	64
5.2.4	First National Bank (South Africa)	65
5.2.5	Standard Bank (South Africa)	66
5.2.6	Citibank (USA)	68
5.2.7	Bank One (USA)	69
5.2.8	Other Banks	70
5.3	Conclusion	71
<b>6</b>	<b>Task Analysis</b>	72
6.1	Introduction	72
6.2	Task Analysis	72
6.2.1	Definition, Benefits and Application	72
6.2.2	Jobs, Roles, Projects, Tasks, Activities and Actions	74
6.2.3	A Model of Interaction for Task Analysis	75
6.2.4	Task Analysis Approaches	77
6.3	Task Analysis for Knowledge Description (TAKD)	78
6.4	Applying TAKD to an Internet-based Banking Task	80

6.4.1	The Problem Area and Task Problem Statement	80
6.4.2	Task Observation and Data Collection	80
6.4.3	The Task Activity List and Prose Description	81
6.4.4	Selection of Specific Objects and Specific Actions	82
6.4.5	Task Descriptive Hierarchy (TDH) Diagrams	83
6.4.6	Knowledge Representation Grammar (KRG) Sentences	90
6.4.7	Discussion of the Results from the Analysis	93
6.5	Conclusion	100
<b>7</b>	<b>Usability Evaluation</b>	101
7.1	Introduction	101
7.1.1	The Design and Measurement of Usability Evaluation	101
7.1.2	The Evaluation of Usability Properties	102
7.2	Heuristic Evaluation Methods	105
7.2.1	Nielsen's Ten Usability Heuristics	107
7.2.2	Instone's Heuristic Evaluation Approach	108
7.2.3	The Borges and Morales Usability Heuristics	109
7.2.4	The Levi and Conrad heuristic list	110
7.2.5	A Collection of other Usability Heuristics	111
7.2.6	The Experimental Usability Heuristics List	112
7.3	Checklist-based Usability Evaluation	115
7.3.1	Web-sourced Checklist-based Usability Evaluations	115
7.3.2	General Checklist-based Usability Evaluations	117
7.3.3	The Experimental Usability Evaluation Checklist	118
7.4	Existing Internet Banking Evaluation Approaches	119
7.4.1	The Journal of Internet Banking and Commerce checklist	119
7.4.2	The Unisys Internet Banking Barometer	120
7.4.3	The CNET Evaluation	120
7.4.4	AOL and the Bank Rate Monitor	121
7.4.5	MWeb-hosted Internet Banking Survey	122
7.4.6	Standard Bank's Survey	124
7.5	Conclusion	125
<b>8</b>	<b>Heuristic Usability Evaluation</b>	126
8.1	Introduction	126
8.2	Data Collection Methods for the Heuristic Evaluation	126
8.2.1	Standard Bank	128
8.2.2	Other Local (South African) Banks	138

8.2.3	Overseas Banks	139
8.3	A Heuristic Evaluation of Standard Bank's Internet-banking Web Site	140
8.3.1	Learnability	140
8.3.2	Flexibility	149
8.3.3	Robustness	152
8.3.4	Web-specific Usability Aspects	156
8.4	Conclusion	163
8.5	Appendix	165
<b>9</b>	<b>Checklist-based Evaluation</b>	<b>208</b>
9.1	Introduction	208
9.2	The Ravden and Johnson Checklist	208
9.2.1	Data Collection and Analysis Methods	208
9.2.2	Discussion of the Results	209
9.3	The JIBC Checklist	211
9.4	Conclusion	212
9.5	Appendix	213
<b>10</b>	<b>Conclusion: Usability and Internet-based banking</b>	<b>219</b>
10.1	Introduction	219
10.2	A Chapter Overview	221
10.3	Conclusion and Directions for Future Research	223
	<b>List of References</b>	<b>224</b>
	<b>List of Internet URL's</b>	<b>230</b>

## List of Figures

<b>Figure</b>		<b>Page</b>
2.1	The two views of HCI	14
2.2	The two components of usefulness	15
2.3	Standard Bank's Internet banking demonstration and online help facilities	20
2.4	IBM's 3D interaction models	22
2.5	Command line interaction with the Standard Bank Internet banking Web site	30
2.6	Form-fill interaction during a Standard Bank Internet banking session	35
3.1	A general view of Sun's usability lab	44
3.2	A view of the observer side and equipment	44
3.3	Card sorting of concepts	45
3.4	Paper prototype	45
3.5	Sun's first design	45
3.6	Sun's fifth design	45
3.7	Sun's final home page design	46
4.1	Internet banking evaluation matrix	56
5.1	ABSA's new site demonstration session using a frame-based page structure	61
5.2	ABSA's old site login screen using a no-frames page	61
5.3	BOFA's demonstration session single page structure	63
5.4	BOFA's help page four-framed page structure	63
5.5	Nedbank's demonstration session two-frame page structure	65
5.6	FNB's demonstration session using a three-frame-based page structure	66
5.7	FNB's demonstration session help using the same three-framed page	66
5.8	Standard Bank's demonstration session two-frame page structure	68
5.9	Standard Bank's demonstration using two additional menu's placed in frame 2	68
5.10	Citibank's transaction demonstration session using a two-frame page	69
5.11	Citibank's help session using a three-frame page structure	69
5.12	Bank One's help session using a second, separate window	70
5.13	Bank One's demonstration session using a two-frame page structure	70
6.1	Task analysis based on a simple iterative computer operating model	76
6.2	Enlarged screen dump cutouts taken during the task progression	84
6.3	Screen display sequence for the account statement task	85
6.4	Task descriptive hierarchy	89
6.5	Activity task descriptive hierarchy	90
6.6	The new homepage for Standard Bank with direct banking access	96
6.7	Error message displayed if navigation buttons used at the logoff page	98
7.1	Usability problems detected versus the number of evaluators used	107

7.2	General and Web-specific usability guidelines in the usability problem space	113
7.3	Royal Bank of Scotland evaluation by Unisys: pages 1 and 2	120
7.4	Internet banking survey hosted by MWeb	122
7.5	Standard Bank's online questionnaire: pages 1, 2, 3, and 4	124
8.1	Version 1: September 1997 Standard Bank home page	130
8.2	Version 2: May 1998 home page	130
8.3	Version 3: November 1998 home page	130
8.4	Version 4: January 1999 home page	130
8.5	Navigation option from the pull-down menu	130
8.6	Home page button	131
8.7	Internet banking button	131
8.8	User feedback acknowledgment	131
8.9	Version 1 banking option static button	133
8.10	Version 2 banking option static button	133
8.11	Version 3 banking option animated buttons	133
8.12	Version 4 banking option animated button	133
8.13	Version 2 prominent home page navigation button	133
8.14	Version 4 reduced prominence home page navigation button	133
8.15	Version 1: September 1997 Standard Bank Internet banking home page	134
8.16	Version 2: November 1997 Internet bank home page	134
8.17	Version 3: May 1998 Internet bank home page	134
8.18	Version 4: November 1998 - April 1999 Internet banking home page	134
8.19	Version 1: Old page transaction option buttons after sign-on	136
8.20	Version 2: New transaction option text links after sign-on	136
8.21	Feedback via two button states	139
8.22	BankOne's context sensitive help frame	141
8.23	Three stages in Standard Bank's Internet banking information brochure	143
8.24	BOFA's transaction selection feedback mechanism	144
8.25	Standard Bank's new demonstration facility	145
8.26	Citibank's use of mouse-over and tool-tip effects	147
8.27	The Bank of America's banking customization facility	151
8.28	Undo during an account transfer for ABSA, Citibank and Standard Bank	154
8.29	Navigational and user guidance diagram for Standard Bank's home page	158
8.30	Frames selection problem	161

## Chapter 1

# Introduction: Usability and Internet-based banking

The evaluation of usability for an Internet banking application is introduced. The research problem, questions, objectives, and benefits are formulated and described. An overview for each of the included chapters is presented.

## 1.1 Introduction

Usability refers to the improvement, study and measurement, of the quality of human-computer interaction. Internet-based banking is a task-oriented, goal-directed, Web-based application where users access and manipulate bank accounts through a Web browser interface. The study of the usability of this interaction through the distributed medium of the Internet formed the basis for this research.

## 1.2 The research dimensions

### 1.2.1 The research problem and its importance

When viewed as an application host and information, education, and entertainment delivery medium, the Internet dwarfs all other comparable software products and platforms in terms of its number and variety of users and client platforms.

Within this distributed environment interactive applications currently constitute a small but rapidly growing class of mainly Web browser based software products. Electronic commerce (E-commerce) is one important example of these Internet-based applications and banking through the Internet should be viewed as a significant category of e-commerce applications. No other type of Internet application currently receives more attention and resource allocation than e-commerce and it is undisputed that the very survival of both retail and commercial banking will depend on their enthusiastic and effective adoption of this kind of remote banking.

Across the world increased competition, deregulation, and the spread of electronic banking are reducing the number of banks and their branches through consolidation, mergers and even closure. Electronic banking in general, and Internet-based banking in particular, offers one route to the salvation and continued growth of these 'traditional' banks. These institutions believed in the past

that their (many) branches offered them distinct competitive advantages, but in most cases branch-saturation strategies have proved to be inefficient and costly solutions towards the achievement of excellence and convenience in client service.

Within this context the Internet and its hosted applications remain a remarkably immature technology and application domain, given the explosive growth in its user count. This conjuncture of an immature technology, a distributed delivery and interaction, an immense heterogeneous client base, and a usability-naïve Web developer community, has led to a number of unique and also common usability problems, but with an impact that surpasses all that has ever been experienced.

Three essential questions that will be addressed by this research are:

- (1) How can the quality of interaction (usability) during an Internet-based banking task be improved?
- (2) What is the best approach and method for measuring this usability?
- (3) How should an awareness be created that '*usability matters*' – even on the Web?

To conclude, the research problem is two-fold but interrelated – promoting the design of high usability Internet-based applications and services, and managing the growing emphasis placed on electronic banking services such as Internet banking.

### **1.2.2 The research objectives and benefits**

The aims of the research are to review the application of general usability principles and properties to Internet-based applications, examine those usability guidelines that currently exist for hypermedia design, and then apply these criteria during a structured evaluation of both local and international online-banking facilities. The proposed research will contribute towards finding solutions for the important problem of designing high usability Internet-based and security-sensitive interactive systems. More specifically, finding answers to the three research questions will be addressed. In addition an examination of the nature of, and support given to, the user's task during a banking session will be included.

### **1.2.3 The research design, instrument, and sample**

A two-stage approach had been used. The exploratory phase consisted of a preliminary research project that focused on the home-banking facilities provided by one local bank (Standard Bank). The formal phase expanded on this examination with the inclusion of other banks, and the application of the two main research instruments. For this type of application no significant control over the main usability variables can be exercised. For some of the local banking sites (especially Standard Bank), a longitudinal study was also conducted.

The research instrument is a combination of a checklist-based and heuristic evaluation, and is preceded by a task analysis. The five local banks that offer Internet banking facilities (ABSA, Nedbank, Standard Bank, First National Bank and Mercantile Bank), and a selection - based on a spread in observed usability - of six major international banks (such as the Bank of America), constituted the research sample.

### **1.2.4 Data collection and analysis**

Primary data collection consisted of capturing the results of interactive online banking sessions, which for Standard Bank had been as an actual client, but for the other sites using their demonstration facilities. The body of printed literature on usability, human computer interaction, Internet banking, hypermedia applications, and web page design, was the main source for secondary data. These sources were supplemented with applicable Internet-based resources.

## **1.3 A chapter overview**

The research project is presented as ten chapters. Four of these are directly concerned with the empirical usability evaluation and task analysis.

**Chapter 1. Introduction: Internet-based banking and Usability.** This is the current chapter and includes a description of the research dimensions such as the research problem, its aims, the research instrument and its importance and benefits. It also includes an overview of all ten chapters that constitute the body of the research effort.

**Chapter 2. HCI and Usability.** This chapter examines some of the important issues related to human-computer interaction. It includes a discussion of usability and its underlying principles and properties, continues with a brief examination of the modern graphical user interface, and concludes by examining the different classes of interaction styles. The aspects of usability and interaction as presented in this chapter form the foundation for all the subsequent chapters and discussions.

**Chapter 3. Usability for the Web.** The general usability principles and properties as examined in chapter 2 also apply to the design of Internet and Web-based interactive software products. Because of certain unique characteristics of Web-based applications (such as response, feedback, browser, bandwidth and graphics limitations), and the extremely wide range of client characteristics it is also appropriate to examine hypermedia-specific usability guidelines.

**Chapter 4. Internet-based banking.** An overview of Internet banking and an examination of a selection of banks that offer Internet-based banking are included in this chapter. A classification for banks with a Web presence is presented and commonly used terms such as home banking, PC



banking, and Internet banking are discussed. The advantages associated with Internet-based banking for both the bank and their clients are also presented.

**Chapter 5. Data collection.** This chapter describes the techniques used to capture a selection of local and overseas Internet-banking Web sites. Because of the transient nature of these online Web sites it was necessary to 'freeze' these Internet banks for evaluation by capturing them to local storage in a browsable and interaction-enabled form. The latter aspect proved to be difficult as most of the banks make use of dynamically generated Web pages. Page architectures used by these banks are also examined.

**Chapter 6. Task analysis.** This chapter examines task analysis concepts and approaches. It then applies one specific approach – TAKD (Task analysis for knowledge description) – to an Internet banking task where the user logs into the banking facility and displays a balance for a selected account. The advantages associated with the use of such a structured approach to the analysis of the banking task in uncovering usability problems are examined and suggestions for improving the interaction quality are offered.

**Chapter 7. Usability evaluation.** Although guidelines designed specifically for Web site evaluation do have merit it is considered more rigorous to base the evaluations on the general usability standards, principles, and properties as identified in chapter 2. Two evaluation methods were used, a heuristic evaluation on the overall Internet banking site and a closely related checklist-based technique on a specific task (as in chapter 6) within the site. This chapter discusses the choices available for these two evaluation approaches – implementation details for the actual evaluations are described during chapters 8 and 9.

**Chapter 8. Heuristic evaluation.** This chapter presents the results of applying a heuristic evaluation to four South African and six international Internet banking sites. Because Standard Bank (South Africa) was the only bank that was accessed as a full client as opposed to limited access through a banking demonstration facility, the main focus of the evaluation had been on this bank's Web site. The other banks studied were used to generate material supplementary to this main evaluation and are not extensively evaluated individually.

The guidelines used are the three categories of general usability principles (learnability, flexibility, and robustness) and their properties (such as consistency, user guidance, and task conformance). These three categories are supplemented by four Web-specific guideline groups namely navigation, the use of aesthetic and minimalist design, the degree of matching between the interactive system and the real world, consistent appearance and behaviour, and the adherence to, and implementation of, accepted (formalised or de facto) Web-related standards. Information for all the banks studied

had been collected through the use of their demonstration facilities – Standard Bank is the only exception to this.

**Chapter 9. Checklist-based evaluation.** This chapter describes the application of a checklist-based usability evaluation to a banking task executed within the Internet banking facility of a South African bank. Two usability checklists were scored for the banking task as identified and examined in chapter 6. The first checklist is the detailed Ravden and Johnson (1989) questionnaire, which although designed for a general usability evaluation is also suitable for use on a goal-oriented Web application. The second checklist is a brief single page questionnaire designed specifically for an Internet-based banking evaluation and is used by the JIBC (Journal of Internet Banking and Commerce).

**Chapter 10. Conclusion.** The answers to the three research questions are examined and the research objectives re-appraised. An overview of the conclusions reached for each of the chapters is presented and suggestions for future research offered.

## 1.4 Conclusion

The importance of improving usability for the Internet is indisputable. This is because of the potentially large impact poor quality interaction could have by virtue of the very large number of Internet users. Internet banking is one notable example in the rapidly expanding domain of Internet e-commerce applications. It is notable because it has many of the typical characteristics of traditional interactive GUI-based software products.

In order to improve usability it must first be measured. A heuristic and a checklist-based evaluation are considered to be suitable approaches for the measurement of Web-based applications.

## Chapter 2

# HCI and Usability

Human-computer interaction (HCI), usability, interaction styles and the modern WIMP-based graphical user interface are examined in this chapter.

## 2.1 Introduction

This chapter first examines some of the important issues related to human-computer interaction. This is followed by a discussion of usability and its underlying principles and properties. It continues with a brief examination of the modern graphical user interface, and concludes by examining the different classes of interaction styles.

The aspects of usability and interaction as presented in this chapter form the foundation for all the subsequent chapters and discussions. It is based on two earlier research projects by the author, namely an examination of the usability characteristics of three file managers (Van Dyk, 1996), and a study of usability laboratories (Van Dyk, 1998). It makes use of examples from the Internet-based banking application that will be evaluated for its usability later in the study.

## 2.2 Human computer interaction

Human-Computer Interaction (HCI) is concerned with the approaches, methods and tools for the development of the human-computer interface, assessing the usability of computer systems, and with broader issues about this interaction between humans and computers. It is based on theories about how humans process information and interact with other people, objects and computers, in an organizational and societal context. HCI borrows from the knowledge and skills as found in psychology, anthropology and sociology, in addition to those from the computer and information system sciences (Preece and Rombach, 1994:553). Note that the equivalent term (Computer-Human-Interaction), is also widely used (Nielsen, 1993:23) (Sullivan and Tyler, 1991:ix).

Gulliksen and Sandblad (1995:135) note that it is important that user interface design should not be based on optimizing the use of the computer only. It should rather be an optimization of the total work activity – they are of the view that the computer should be a ‘transparent’ tool, i.e. the interface should be ‘obvious’ (and intuitive) to the user. Winograd (1995:65) concludes that as the field of programming has matured over the years, emphasis has shifted first from programming to

programmers and then to users. Currently the attention has moved from the algorithms required by users to the cognitive structures of people who produce and use it, i.e. there is a shift from machines to people. In this regard the 1994 inaugural statement of the special interest group of the ACM on HCI reads as follows (Winograd, 1995:67):

*'We seem to have moved well beyond the idea that making a computer useful is simply to design a good interface between man and machine. Our ideas have evolved to the point where the richness of human experience comes to the foreground and computing sits in the background in the service of these experiences.'*

The director of systems technology and research at Silicon Graphics (Mashey, 1996:166) explains that to the specialization of hardware and software a third category should be added. This class of 'wetware' (i.e. people) may be indicative of the new trend in thinking about people as a previously neglected component of an information system. The term 'liveware' as used by Harrison (1996), is an even better description of this third and multi-modally rich, component. This new shift in thinking is taken a step further in recent HCI research through the use of artificial intelligence (AI) approaches, in 'Adaptive User Interfaces' (Brown, Totterdell and Norman, 1990:2), and 'Intelligent User Interfaces' (Sullivan and Tyler, 1991:vii).

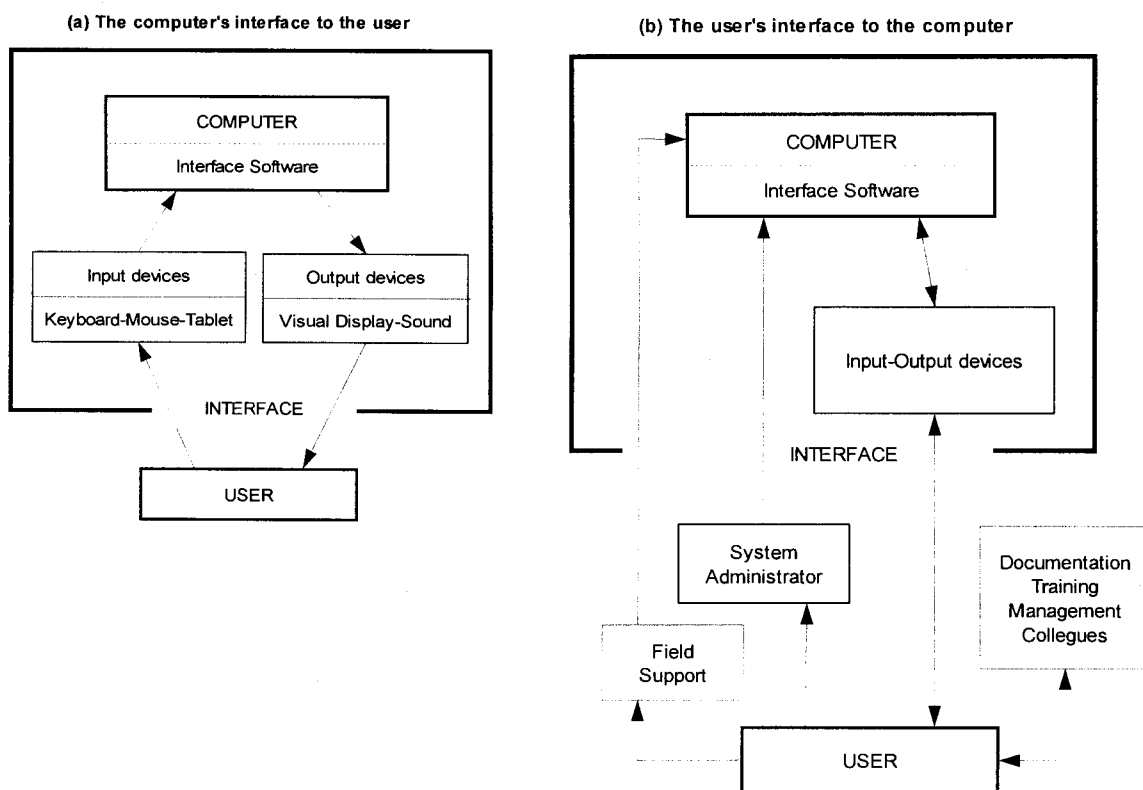
Dix, Finlay, Abowd and Beale (1998:110) refer to a comprehensive framework for HCI and its main components (such as ergonomics, dialog, the societal and organizational context (environment), input/output, system/user and the interface), which is sourced from the curriculum development subgroup of the same group (ACM SIGCHI) as above. The same authors (Dix et al., 1998:105) also discuss two other frameworks for HCI. The first is based on Norman's two phase uni-directional model for HCI, and the second an extended four step bi-directional interaction model proposed by Abowd and Beale, that addresses the limitations in Norman's unidirectional framework (user to system only), and includes the system's communication through the interface. In the excellent discourse by Weller and Hartson (1992:313) on the direct-manipulation paradigm in HCI, the aspect of empowering interaction environments is comprehensively addressed. They examine the problem of user cognition during problem solving through a computer, and achieving enhanced usability through user confidence and human-computer partnerships.

Grudin (1993:114) also examines the aspect of HCI directionality, and comments on the development of human-computer interfaces: from programmer interfaces to user interfaces (dangerous because of the implication that there is a 'typical user'), and recently to organizational interfaces (to indicate the shift towards collective/group support). Grudin continues (1993:116) by acknowledging the importance of recognizing directionality differences in the human-computer (as opposed to computer-human) interfaces, and notes that the two are not mirror images. The computer's interface to the user can be reasonably and simply described as being the software controlling the dialog via I/O devices, i.e. it is an engineering perspective. The opposite case, namely

the user's interface to the computer is considerably more complicated – users consult documentation, are trained, get advice from colleagues, system administrators, and have their system modified. A user's interaction is shaped by a multitude of surrounding factors – his work or task environment. These two interface views are shown in figure 2.1.

It can therefore be concluded that in order to design quality user interfaces, a user's model of the world – his mental, conceptual model or 'world view' (from Grudin (1993:115) on Moran's 1981 seminal work on mental models) – must be incorporated into the design. Grudin (1993:116-118) also discusses a number of barriers that prevent and discourage designers from basing their design on the a user's own mental model, and instead using their own often incorrect interpretation of the system user.

**Figure 2.1 The two views of HCI**



Source: Grudin, 1994:116.

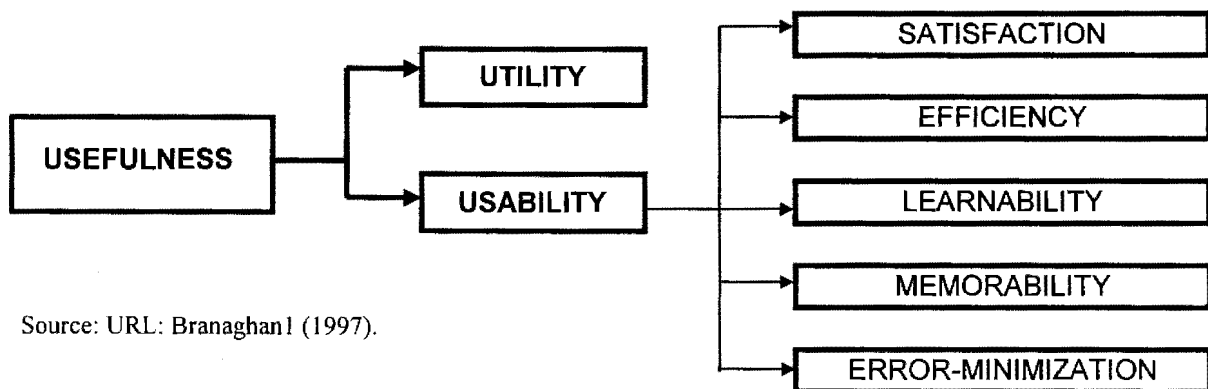
## 2.3 Usability

Usability (which is preferred to the less rigorous term '*user friendly*' – Nielsen (1993:23, 1996(1):12) comments interestingly on the difference), as an aspect of the study of human computer interaction, is in essence the systems ability to support the user's activities and tasks (Newmann and Lamming, 1995:166). For a software product, usability is the user's view of the software quality (De

Wet, 1994:75), and this quality may be operationally measured in terms of criteria such as learnability, flexibility, reliability, and consistency (Preece and Rombach, 1994:557). It is therefore a reflection of the interaction quality.

Branaghan (1997, URL: Branaghan1) offers a different perspective on usability (shown in figure 2.2). He regards usability as part of the broader quality of **usefulness** – i.e. that which refers to whether a (software) product helps the user to achieve his/her goals. One component of usefulness is **utility**, which refers to whether the product has the required functionality that the user needs in his task execution. The other component is then usability, which refers to whether the user can easily access that functionality. Note that his view is not a fundamental extension of the concept of usability (as task support is covered by aspects such as task conformance), but his view is nonetheless a useful alternative representation. Sutcliffe (1988:184), also considers utility – or task and system functionality – to be an essential usability ‘quality’. Nielsen (1993:25), earlier expanded on the interpretation from Branaghan (1997). He includes the dimensions of social and practical acceptability to the left-hand side of figure 2.2.

**Figure 2.2 The two components of usefulness**



Source: URL: Branaghan1 (1997).

Usability principles such as **learnability**, **flexibility**, and **robustness** (Dix et al., 1998:162), are often used to explain the concept of usability. These when applied to the design of interactive systems will enhance its usability. The principles are in turn composed of several usability properties (sub-principles), and are as listed in table 2.1.

There are a number of large commercial companies with usability laboratories (Van Dyk, 1998) that provide information as to what they regard as important usability principles. Microsoft and IBM are amongst the larger of these, and Microsoft's seven '**user-centered design principles**', and IBM's eleven '**usability design principles**' (URL: Microsoft1, URL: IBM1), are presented in table 2.2.

**Table 2.1 Usability principles and properties**

Learnability	Flexibility	Robustness	Language-Action paradigms
Synthesizability	Dialog Initiative	Observability	
Consistency	Multi-threading	Recoverability	
Familiarity	Task Migratability	Responsiveness	
Visual clarity	Substitutivity	Task Conformance	
User Support and Guidance	Customizability	WYSIWYG	
Predictability	Control		
Generalizability			
Enjoyability			
Interaction Metaphors			
Conceptual Models			

Source: De Wet (1994).

**Table 2.2 Usability principles from Microsoft and IBM**

Microsoft: User Centered Design Principles		IBM: Usability Design Principles	
1	User in control	1	Support
2	Aesthetics	2	Familiarity
3	Simplicity	3	Simplicity
4	Directness	4	Obviousness
5	Consistency	5	Satisfaction
6	Forgiveness	6	Encouragement
7	Feedback	7	Safety
		8	Versatility
		9	Personalization
		10	Affinity
		11	Accessibility

Source: URL: Microsoft1, IBM1 (1997).

Note that Microsoft's principles correspond closely to what are listed as properties (sub-principles) in table 2.1. Properties such as personalization (which includes VRML GUI's – IBM's real-world and real-places interfaces, and avatars, receive strong emphasis from IBM as an important part of their 'future' interfaces – as referred to by Tebbutt (1997:228) whilst examining Tony Temple's (IBM's VP of HCI) views on usable interfaces. But it is in essence closely related to the flexibility properties listed in table 2.1, such as customizability and migratability. According to Shneiderman (1992:203) the related principle of 'virtuality' – i.e. a user's mental model of a real world inside the computer, is an important aspect of modern direct manipulation interfaces.

As expected, many entertainment companies join Microsoft in another view of usability (URL: Microsoft2), by including *interactivity* as one of their main usability properties – this could be viewed as a combination of user engagement, enjoyability, consistency and familiarity (URL:

Andiron1). According to Garzotto, Mainetti and Paolini (1995:75), interactivity is usually considered to be a blend of dynamics – moving, navigation, and playing, and of presentation – i.e. it is supported through content and functionality.

It should be noted that there is to some extent disagreement on the naming and nature of these principles and properties. Newmann et al. (1995:7-8, 30, 76, 183), refer to ‘retention of acquired skills’, ‘user satisfaction’, ‘speaking the user's language’, ‘provision of shortcuts’, ‘reducing the user's memory load’, and ‘clearly marked exits’, as being major usability factors. Mayhew (1992:8-28), identifies aspects such as ‘invisible technology’, ‘error protection’, and ‘ease of learning’ as the major usability design guidelines. Sutcliffe (1988:184-185) identifies five usability qualities, namely Learnability, Utility, Coverage, Effectiveness, and Attitude. On further analysis most of these factors can be shown to be closely related to the principles and properties as listed in table 2.1. Others such as satisfaction are included in the second list (table 2.2).

## 2.4 Usability principles and properties

These should be attributes of the total work system, rather than just aspects of the software product or of the software-hardware system. It includes the user, the environment, the task, and the computer system, and it enhances the quality of interaction between user and computer.

In addition to the general usability principles and properties as discussed here, chapters 3 and 7 include an examination of Web-specific usability properties such as navigation and minimalist page design. Also included is an explanation of the relation between these general and Web-specific usability properties. Chapter 8 applies the properties and principles presented in this chapter to a heuristic evaluation of an Internet-based banking application.

Bastien and Scapin (1995:106) prefer the term ‘*usability dimensions*’, and describe these as dimensions aimed at explaining the concept of usability, evaluating/measuring interface features, and organizing sets of data on user interface design.

Three such principles which when applied to the design of interactive systems will enhance its usability, are identified by Dix et al. (1998):

- (1) **Learnability:** The ease with which maximal performance and effective interaction is achieved.
- (2) **Flexibility:** Exchanging information through a number of choices and options.
- (3) **Robustness:** Support in determining successful achievement and goal assessment.

De Wet (1994:76) adds a fourth principle:

- (4) **Language-Action paradigms:** Image projection through either action or language/words at the interface level.



Sutcliffe (1988:184-185) identifies five related usability qualities:

- (1) **Learnability:** This covers both learning and remembering, and can be measured via changes in error rates, task completion times, recall of facilities and names, and user knowledge.
- (2) **Utility:** This is linked to system functionality and matching the user's task requirement with perceived ways to do it (the task-fit).
- (3) **Coverage:** This is the number of system facilities actually used.
- (4) **Effectiveness:** This is how well the system performs when achieving user goals. It can be measured as error rates, task completion times, and system facilities usage.
- (5) **Attitude:** This is an indicator of user satisfaction and motivation.

Closely related to the previous is Nielsen's (1993:26) five measurement dimensions – learnability, efficiency, memorability, errors, and satisfaction. His comment on the choice of users for measuring these dimensions is interesting – an experienced user will yield a smaller volume of learnability data than a novice user.

Bastien and Scapin (1992:187, 1995:110) use the terms usability dimension, usability principle, and usability heuristic and usability criterion interchangeably. They also identify a set of 18 elementary ergonomic criteria, which includes guidance (prompting, grouping and distinction of items), immediate feedback, legibility, user workload (brevity (concision, and minimal action) and information density), user explicit control (explicit user actions, user control), adaptability (flexibility, user's experience management), error management (error protection, error messages quality, error correction), consistency, significance of codes, and compatibility.

Woodson, Tillman and Tillman (1992:ii), note that '*human factors engineering*' is the (only) correct equivalent for '*ergonomics*' – with the first term widely used in the United States. Alternatively the term ergonomics is the preferred term for industrial-related work, whereas the first is the preferred term by the U.S. military.

Newman and Lamming (1995:7-8,30,76), prefer the use of '*usability factors*', and identify retention of acquired skills, learning to use the system, error recovery by the user and error incidence, user's satisfaction, customizing and reorganizing the system, and speed performance, as the major factors – these are all covered by the previously presented classification scheme.

Newmann et al. (1995:183) are of the opinion that a different set of usability factors (taken from Nielsen and Molich, 1989), will be more suitable for use as guiding factors during a heuristic evaluation. These are simple and natural dialog, minimizing the user's memory load, consistency, feedback, provision of shortcuts, good error messages, clearly marked exits, error prevention, and '*speaking the user's language*' – they are perhaps closer related to the factors used during a cognitive walkthrough.

Mayhew (1992:8-28), lists the following as desirable usability design guidelines – user, product, task and work-flow compatibility, consistency, familiarity, simplicity, direct manipulation, control, WYSIWYG, flexibility, responsiveness, invisible technology, robustness, protection (against human error), and ease of learning and use.

The three main principles (Dix et al., 1998) are in turn composed of several sub-principles as listed in table 2.1. In order to distinguish between the main design principles and the sub-principles, the latter will be referred to as usability properties.

The addition of the enjoyment (pleasurable) property (De Wet, 1994) is interesting, especially when regarding its connection with the important notion of Internet-based user engagement and entertainment. Enjoyability is also listed by Card et al. (1983:409) as a usability performance variable.

### 2.4.1 Learnability

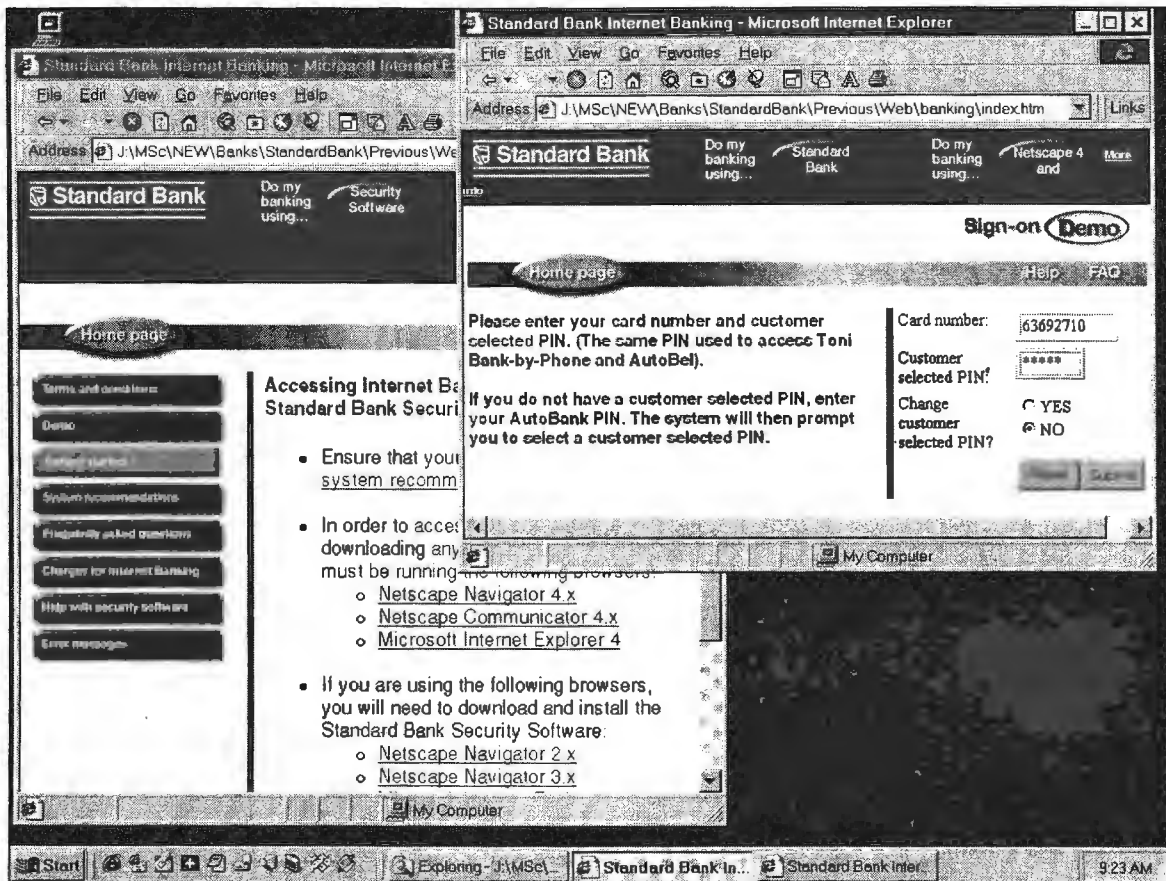
This is the ease with which maximal performance and effective interaction is achieved (Dix et al., 1998:162). It covers both learning and remembering and it can be measured via changes in error rates, task completion times, recall of facilities and names, and user knowledge (Sutcliffe, 1988:184). Ten of the specific usability properties that support the principle of learnability are discussed below. Learnability through user guidance is shown in figure 2.3 (www.sbic.co.za, 1998). This is an Internet-banking session using the bank's interactive demonstration facility in the right-hand browser window, and the bank's online-help facility in the other (left-hand) browser window.

**Predictability:** This is support of the user to determine the effect of future action based on past interaction history, and is the avoidance of surprises to the user (except to support the enjoyment/engagement property where appropriate). Predictability is a user-perspective-based deterministic concept: computers are inherently deterministic already. It is related to the ability of the user to determine the effect of operations on the system (based on the user's mental model of system behaviour), and the knowledge of which operations can be performed, i.e. a high degree of potential operation visibility must exist. It supports the human ability for higher recognition than just recall performance (Dix et al., 1998:163).

**Synthesizability:** This supports the user to assess the effect of previous operations on the current state of the system. It is concerned with the building of the user's mental model of how the system behaves (as used in behaviour prediction above). It is important that when the internal state is changed through an operation, this must be visible to the user in an informative, honest, and clearly observable manner. The notification of the change can be either immediate (preferred), or delayed

(eventually), which may lead to a problem of association of the earlier change with the later notice (Dix et al., 1998:164).

**Figure 2.3** Standard Bank's Internet banking demonstration and online help facilities



Source: www.sbic.co.za (1998).

**Familiarity:** This is the extent to which the user can apply his knowledge gained from other experiences (computer and real-world domain), to his interaction with a new system. It is therefore a measure of the gap between existing and required effective interaction knowledge. The use of metaphors supports this property, and it can also be referred to as guessability (Dix et al., 1998:165).

**Generalizability:** The user is supported in the extension of his knowledge of specific interaction techniques both within and across applications, to other similar situations, resulting in a higher level of predictability. It is used in situations with similar goals, and applying knowledge in achieving one to achieve the other. Standards and programming style guides will enhance generalizability (Dix et al., 1998:166).

**Consistency:** Similar situations and task objectives show a similarity in input/output behaviour. Consistency manifests itself on many levels and forms such as naming conventions and command-

argument invocation, i.e. it is always relative (Dix et al., 1998:166). De Wet (1994:79) distinguishes between semantic consistency (the meaning of the interface elements), syntactic consistency (the appearance of the interface elements), and physical consistency (the hardware and its use).

**Visual Clarity:** This is achieved if output information is displayed in a format that is clear, precise, sharp, well-organized, unambiguous and easy to read. The user's attention should be drawn to important aspects in the output display, possibly through visual stimulation – blinking, colour schemes, borders etc. (De Wet, 1994:80-81). Shneiderman (1992:80) lists nine categories of user 'attention-getters', but recommends restraints (as listed in brackets) in its use to avoid desktop cluttering. They are intensity (two levels), marking (underline, boxing, arrow pointer, bullet/asterisk, 'X'), size (four or less), fonts (three or less), inverse video, blinking (2 to 4 Herz), colour (4 standard and occasional other colours), colour blinking (two colour change), and audio (regular feedback through soft tones, harsh tones for emergencies).

**Enjoyability:** Marcus (1993:101) proposes using the term 'playstations' instead of 'workstations', to refer to the new type of interface that will 'engage and entertain' the user, i.e. the interaction will be of an enjoyable nature. In order to achieve this new level of positive communication in HCI aspects such as the user's mental model, metaphors (situational mapping agents), model navigation, and the appearance of the user interface are of prime importance.

Weller et al. (1992:318,325) also address this aspect of user engagement in detail through the use of 'seductive interfaces' where engagement is meant to be a high level of user involvement, attention and interest. (Note that engagement does not imply entertainment or enjoyment, though the opposite does apply: 'positive engagement' may be an appropriate term.) Engaging interfaces according to them are also related to those that encourage browsing, exploration and experimentation, and result in an eagerness to use the system. Weller et al. (1992:325) observe that the mimetic illusion (a dramatic and artistic representation) of the interface, promotes this idea of user engagement, such as is used in arcade-style games. An engaging interface should be multi-modal: visual, auditory, and lexical. De Wet (1994) associates enjoyment with pleasant and exiting interfaces.

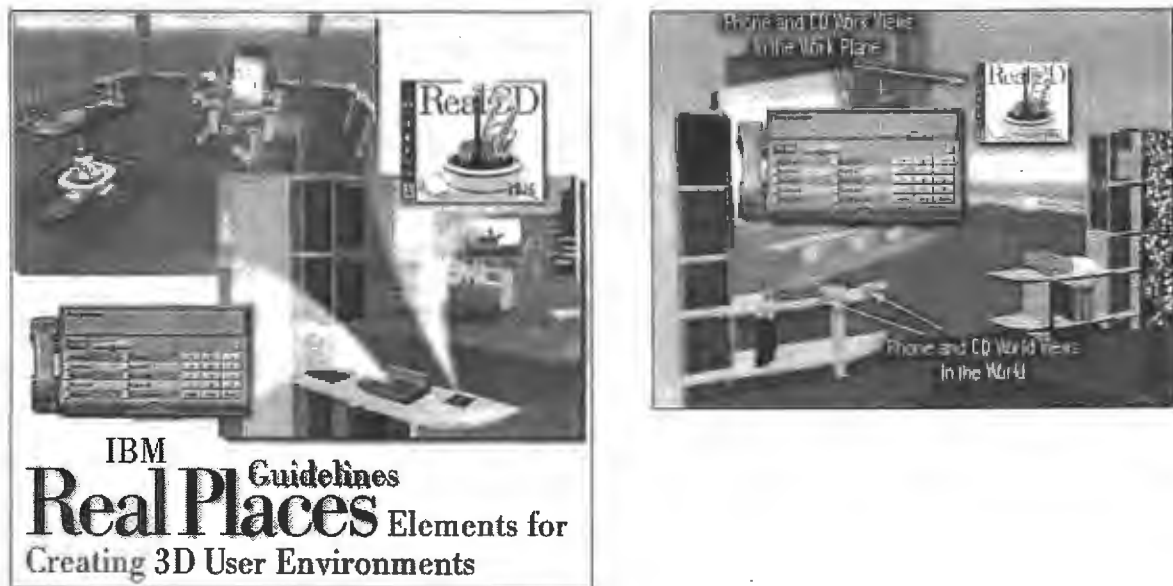
**User Support and Guidance:** This is the degree to which the system combines learning during using. It leads users by provision of informative, easy to use and relevant, guidance and support, such as through online and offline help facilities and meaningful error dialog messages. Trial-and-error (experimentation) learning should be supported and even encouraged (De Wet, 1994:81).

**Conceptual (Mental) Models:** This is the mind picture the user has about the system. The development of such a model is part of learning the new system, by starting with a familiar analogy and then shaping it through the differences inherent in the new system. Apart from a learning aid, mental models are also employed in predicting system behaviour. The developer or designer usually

employs a different conceptual model than that of the user, which will lead to translation, usability and comprehension problems for the user (De Wet, 1994:81-82).

Shneiderman (1992:203) speaks about the ‘principle of virtuality’ in regard to direct manipulation interfaces – the user forms a mental picture of the on-screen world that can be manipulated. Figure 2.4 shows the introductory page for IBM’s research facility into Internet-based 3D-world-view modeling. Although related to the 3D-interaction style as discussed by Dix et al. (1998:122) it should rather be regarded as a whole new approach for looking at human-computer interaction and user interface design. This attempts to create a computer-based world that matches the user’s view with extreme fidelity – it is almost certain that Internet-based banking applications will eventually evolve into this kind of virtual-banking interaction eventually.

**Figure 2.4 IBM’s 3D interaction models**



Source: [www.ibm.com/ibm/hci/guidelines/design/3d/toc.html](http://www.ibm.com/ibm/hci/guidelines/design/3d/toc.html) (199).

Newmann and Lamming (1995:326) state that the user, through theorizing about the inner workings of the system, gradually develops a working ‘model’ of the system around which they organize their knowledge and theories. It is therefore (mostly) based on observations on its external behaviour rather than knowledge about its internal workings. This ‘system image’ (1995:333) is largely presented through the user interface. They (1995:334-337) continue, by describing several types of user models, such as mapping models and analogical models. These refer to the user’s model being shaped by something that he/she is already familiar with, i.e. this could be considered a link between system metaphors and the user’s set of conceptual models. The difference between analogical models and metaphors is considered to be a case of degree rather than kind. Mayhew (1992:97) states that the design of a conceptual model into the user interface based on familiar metaphors will facilitate learning. Note however that the system can only reinforce, superimpose, or suggest an appropriate

mental model, but can never dictate it – the user will formulate and use his/her own individual mind picture.

**Interaction Metaphors:** This is a type of contextual or situational mapping agent, where a familiar object from one context is used to convey a similar property to the other non-literal context. It can be used to teach a new concept in terms of another that is already understood and familiar, and results in enhancing the consistency and learnability of the interface (De Wet, 1994:80).

Metaphors are common in HCI, such as the ‘virtual office’ and presenting the interface as a ‘desktop’. Mayhew (1992:97-98) examines the Apple Macintosh desktop metaphor and its Inbox, Outbox, Trashcan, Clock, Appointment Book, Folders and Documents.

The Wang Freestyle desktop is another prime example of this metaphor implemented in a true object-orientation – its desktop contains icons resembling office equipment such as a fax machine, a printer, a file cabinet, a stapler and unstapler, pads of paper, a trash can, and an inbox. It is different from most systems in that a graphics tablet and pen (rather than a mouse) is used to manipulate these objects, and was known for its ability to perform almost all operations without touching a keyboard. Note that this was an application rather than an operating system – although the distinction between the two is often blurred in modern GUI operating systems (Greenberg, 1989:156).

Other common metaphors include the ‘trashcan’ ‘paper bin’ and ‘recycle bin’. Marcus (1993:103-106) discusses the optimal use of metaphors in detail, and presents an extensive list of these elements.

### 2.4.2 Flexibility

This is the multiplicity of ways in which information can be exchanged between the user and system (Dix et al., 1998:167). Six of the specific usability properties that support the principle of flexibility are discussed below.

**Dialog Initiative:** This allows the user freedom from artificial constraints on the input dialog imposed by the system, aims to maximize the user's ability to pre-empt the system (user pre-emption enhances flexibility), and minimizes the system's ability to pre-empt the user (system pre-emption restricts the user to a fixed input dialog behaviour) (Dix et al., 1998:168). A pre-emptive system (also referred to as a modal system) may however be required in some cases for safety purposes, where it is possible to lose track of the tasks initiated or where data integrity may be compromised (De Wet, 1994:83).

**Multi-threading:** This is the ability of the system to support user interaction on more than one task at a time. Two type of multi-threading exist – concurrent (simultaneous communication of separate task communication) and interleaved (a temporal overlap between tasks, i.e. time-slicing) (Dix et al., 1998:169). A multi-modal system relates to this principle and has two dimensions – separate communication channels combined into a single input or output (an error message is announced with a beep sound), and multiple input channels which may be used for the same action (such as different ways to open a window) (De Wet, 1994:83). Windowing systems are a natural implementation of a multi-threading dialog, with each window representing a separate task.

**Task Migratability:** This refers to the transfer of control for task execution between the system and user partnership. For example an internal system task can become a cooperative and shared action if the situation, system or user requires this (Dix et al., 1998:170). Equal opportunity (Thimbleby, 1990) is related to this property and also to substitutivity, as discussed next (De Wet, 1994:83).

**Substitutivity:** This allows equivalent input and output values to be (arbitrarily) substituted, such as an input in different units, or an output (representation multiplicity) in graphic or text format (Dix et al., 1998:170). Equal opportunity is a blurring between input and output, the conversion of one into the other, and requires a user pre-emptive system (i.e. the user in control).

**Customizability:** This is the degree of modifiability (customization) of the user interface by the user or the system itself, i.e. the interface adaptability – user initiated, or adaptivity – system initiated (Dix et al., 1998:171). The advantage of such system flexibility can be an increase in user engagement and productivity, but can also lead to a loss in portability and consistency (De Wet, 1994:85).

**Control:** This is related to most of the other flexibility properties – dialog initiative, equal opportunity, task migratability, substitutivity, and user customization. System control (the system dictates) and user control (the user dictates), are both found in varying degrees in any system, and a balance between task and user should be found (De Wet, 1994:85).

### 2.4.3 Robustness

This is the support that is given to the successful achievement and assessment of the user's set of goals interacting with the system (Dix et al., 1998:172). It is the extent to which the system can continue to perform despite a violation of its specification assumptions (De Wet, 1994:90). Five of the specific usability properties that support this principle are discussed below.

**Observability:** This refers to the user's ability to evaluate the internal system state from its perceivable representation. Observability is related to the principles of *browsability* (exploring the

internal system state through its (limited) external viewport), *defaults* (a response is recognized rather than recalled – of which two type are found (static (initialized) defaults, and dynamic (session-evolving) defaults), *reachability* (the possibility of navigation through the observable internal system state), and *persistence* (the duration of a communication action, and the user's ability to make use of it) (Dix et al., 1998:172) (De Wet, 1994:86).

**Recoverability:** This is the user's ability to take corrective action after recognizing an error. Recoverability can be forward (when the error is uncorrectable and not revocable), or backward (undo the previous condition until a stable state is reached – i.e. roll the system state back). Recovery can also be system initiated (ensuring fault-tolerance, safety, reliability, and dependability), or user initiated. The principle of *commensurate effort* states that a difficult 'do' operation will be difficult to 'undo' as well, and this should be applied during recovery design (Dix et al., 1998:173).

**Responsiveness:** This is the user's perception of the communication rate with the system (Dix et al., 1998:174). It is the degree to which the system responds quickly and meaningfully to user requests to carry out various operations (De Wet, 1994:88). Informative feedback is related to this, as poor feedback is perceived to be poor responsiveness (no indication of the interaction). Response time (time required by the system to indicate a state change to the user) and response time stability (consistency) are important. The first should be as short as possible and the second should be consistent in order to enable the user to construct an accurate and dependable mental response picture of the system (De Wet, 1994:88).

**Task Conformance:** This is the degree to which the system supports the user's tasks of interest, and the way the user understands it. Task completeness refers to the coverage of the task (level of mapping between system services and the user tasks), and task adequacy to the user's understanding of the tasks. De Wet (1994:89) mentions that task conformance is enhanced by reducing the semantic distance between the form of an expression and its meaning. Semantic distance is a function of syntactical, lexical, and layout aspects.

**WYSIWYG:** As a principle ('What You See Is What You Get') refers to what is seen should 'tell the whole story' (De Wet, 1994:89). Dix et al. (1998:84) emphasize however that it should be remembered that the two media of paper and screen (hard and softcopy), are in some aspects such as resolution significantly different, and that WYSIWYG is not a panacea for all usability ills – 'What you see is ALL you get' (Dix et al., 1998:151). Dix et al. (1998:354) regard WYSIWYG as a generic usability principle (such as 'undo'), that strongly and positively impacts on the previously discussed aspects of predictability and observability.



## 2.5 The graphical user interface (GUI)

According to Microsoft a true GUI environment must satisfy six requirements (Seymour, 1989:98):

- (1) It is a graphically oriented interface, making extensive use of icons.
- (2) It has good screen aesthetics – it looks good and is a pleasure to work with.
- (3) It allows direct manipulation of on-screen elements, allowing for grab and drag rather than command execution.
- (4) It embraces the object-oriented paradigm, so that the user chooses an object first, then selects the action. This frees the user from an action sequence that must be completed before moving onto the next task.
- (5) It offers standard expected elements, such as menus, standard window elements, and dialog controls, to provide consistency across applications.
- (6) It explores bitmapped displays, offering true WYSIWYG representation of printed output.

(7) Additional elements that Microsoft feels that a GUI should incorporate are:

- (7.1) Application support: A strong set of user-interface controls tools to build applications.
- (7.2) Consistency across platforms as well as across applications.
- (7.3) Ease of use, visual appeal, ease of installation configuration, and high enjoyability
- (7.4) Productivity: WYSIWYG and information communicated through graphics
- (7.5) Flexibility: Support for a keyboard and mouse as well as for range of devices and peripherals.
- (7.6) User customization and personalization.
- (7.7) Multiple-application and inter-application support with tools such as dynamic data exchange, messages, and clipboards.

It is interesting to note that at that stage (1989), Microsoft's own Windows GUI's included very few of these elements, but that its 1995 and 1998 Windows releases did exhibit most of these elements. From this description, interface styles such as DM (direct manipulation), WIMP, Menus, Graphical interface, Form Fill-in (as in dialog boxes) and WYSIWYG, are evidently very much a part of a typical modern GUI. As such they are also found in the Windows-hosted and Internet-based banking applications examined.

## 2.6 Dialog and interaction styles

Dix et al. (1998:104, 195) appears to use the terms *interaction style*, *interface style* and *dialog style* interchangeably and synonymously. If interaction is the dialog between user and computer, then the manner (style) of this interaction has a strong effect on this dialog. According to De Wet (1994:44) a dialog style is the overall style of interaction, and these can vary from the very simple and highly constrained to highly complex and open-ended. De Wet (1994:5) considers an interface (dialog) style to be the type of human-computer interface. She (De Wet, 1994:45) also mentions that most

interfaces employs more than one dialog style, and that in order to standardize terminology a dialog style should be called an interface.

The definition by Dix et al. (1998:195) of a dialog style includes both input and output components: '*Dialog style refers to the means by which the user communicates input to the system, including how the system presents the communication device (output).*' Newman and Lamming (1995:180) simply consider *user dialog* to be '*that separate stream of interaction*', and prefer the use of *interaction style* to describe the range of alternative design strategies for the user interface (1995:294).

The term 'interactive interface' is also used to describe a certain type of architectural design framework (Rumbauch, Blaha, Premerlani, Eddy and Lorensen (1991:214). As a design type it is dominated by interaction between the system and independent (not controlled) external agents (humans, devices, and other systems/programs). Rumbauch et al. (1991:214) note that interactive interfaces are dominated by a dynamic model (as contrasted to a functional and object model), are concerned with external appearance rather than deep semantic structure, and are best implemented using concurrent control (multi-tasking), or event-driven control (via interrupts or call-backs), rather than procedure-driven control.

Although the focus of this research is on an Internet application running under a WIMP (Windows-Icons-Mouse-Pointer) type interface, a discussion of other categories is appropriate. It will be demonstrated that the Windows-based Internet-based interface includes elements from most of the other interaction styles – especially the windows, direct manipulation (or point-and-click), menu-based, and form-fill styles.

Van Dyk (1996:6) presents a detailed comparison in table format for interaction style categories as suggested by Dix et al. (1993:102, 1998:115), De Wet (1994:5), Shneiderman (1992:70), Smith and Mosier (1986), Baecker and Buxton (1987:427-433), Sutcliffe (1988:68-75), Davis and Bostrom (1992:148), Newmann and Lamming (1995:294), Weller et al. (1992:323), and Mayhew (1992). Dix et al. (1998:116) later added the point-and-click, and previously mentioned 3D-interfaces (virtual) styles to their earlier (1993) list. The comparison (Van Dyk, 1996) is presented here in adapted form as table 2.3.

The classification used by Baecker et al. (1987:427-433) is comprehensive, offers a finer level of classification relevant to the WIMP type interfaces examined, and will be used in the discussion below. Because their classification omits the standard function key style category, this is added to yield a total of 10 classes of interaction styles. Note that Dix et al. (1998:116) consider the function key interface to be part of the command line interface. Question and Answer dialog style interfaces are in essence a type of Form-Filling interaction style.

**Table 2.3 Interaction style classification**

Author	Dix	De Wet	Smith	Mayhew	Davis	Baecker	Sutcliffe	Newman Lamming	Weller	Shneiderman
Form-filling			1	1		1	1	1	1	1
Form-fills + Spreadsheets	1	1								
Question + Answer (Q+A)		2	2	2			2	2		
Q+A + Query Dialog	2									
Query Language (i)			3							
Prog. Language Dialog						2				
Command Language		3	4	3			3		2	2
Command Line Interface	3					3		3		
Command-based Interface					1					
Natural Language	4	4	5	4		4	4		3	3
Test-based Natural Lang.								4		
Voice-based Interaction								5		
Menus	5	5		5			5			4
Menu Systems						5				
Menu Selection			6						4	
Menu-based Interface					2			6		
Function Keys		6	7	6				7		
WIMP (iv)	6	7								
Graphics Selection			8							
Graphics Interaction						6				
Direct Manipulation		8		7	3	7			5	5
Graphical Direct Manip.								8		
Iconic Interface						8	6			
Window Systems						9				
	(v)	(ii)						(iii)		

Notes: Table 2.3

- (i) Query language could be considered a special case of a command language (Dix et al., 1993:166). Baecker et al. (1987) consider languages to be a special case of their programming interaction style.
- (ii) A more general, two category classification is used by De Wet (1994:5,45), and is also investigated by Davis and Bostrom (1992:143), in comparing direct manipulation and command-based interfaces:
  - (a) **Character-based user interface:** Includes Command-language, Menus, Form-filling and Spreadsheets, Question and Answer, Function Keys, and Natural language Interfaces..
  - (b) **Graphical based user interface:** Includes Direct-manipulation, and WIMP interfaces. Commonly, the term GUI or "Graphical User Interface" is used to refer to WIMP related interfaces such as Microsoft Windows.
- (iii) Newmann and Lamming (1995:294) use a three category interaction style framework, namely Key-Modal (Menu's, Q+A, Function Key, and Voice-based), Direct Manipulation (Graphical Direct Manipulation and Forms Fill-in), and Linguistic (Command-line and Text-based Natural Language): key-modal refers to a mainly keyboard interface.
- (iv) WIMP is an acronym for a Windows-Icons-Mouse-Pointers based interface. Interestingly, De Wet (1994:58) also mentions that WIMP is sometimes meant to refer to Windows-Icons-Mouse-Pulldown-Popup-Menus, or to Windows-Icons-Mouse-Pulldown-Menus. Both Dix (Dix et al., 1993:106) and Shneiderman (1992:207) prefer the latter, which is perhaps an indication of the widespread use of menus in WIMP styled systems.
- (v) Dix et al. (1998) add the categories of Point and Click and 3D Environments to their 1993 classification.
- (vi) Woodson et al. (1992:257) use the US Military classification which does not include any of the direct manipulation or graphical styles - it includes Question+Answer, Query-Language, Form-Filling, Menu-Selection, Function-Keys, Command-Language, and Constrained-Natural-Language styles only.
- (vii) Treu (1994:107) classify interaction styles as being:
  - (a) Location-based: multiple screens, templates, menus, windows, and forms and text.
  - (b) Drawing-based: plots and histograms, static diagrams, iconic images, digital images, and interactive graphics.
  - (c) Written/printed language: query languages, command line and natural language.
  - (d) Using other skills apart from vision: pattern recognition, speaking, and hearing.

A WIMP style interface usually includes most aspects of the Windows, Iconic, Direct Manipulation, and Graphic Selection style categories, and is not discussed as a separate category, although it can be treated as such. Dix et al. (1998:123) mention that there are however several interaction techniques or special WIMP elements that are collectively referred to as ‘widgets’, and are not windows, icons, pointers (direct manipulators) or menus. These include buttons, dialog boxes and palettes. Several authors point to the fact that the distinction between these dialog styles are not always clear, and in practice user interfaces and especially graphical user interfaces (GUI), use a combination of many of these styles – Davis and Bostrom (1992:148): *‘An application may exhibit characteristics of more than one interface type’*.

Gulliksen et al. (1995:138) note that applications developed using a general style guide does have certain disadvantages (such as a sometimes-inflexible match to user's requirements), and discuss an alternative approach, which they refer to as domain-specific interface design.

### 2.6.1 Command line interaction style

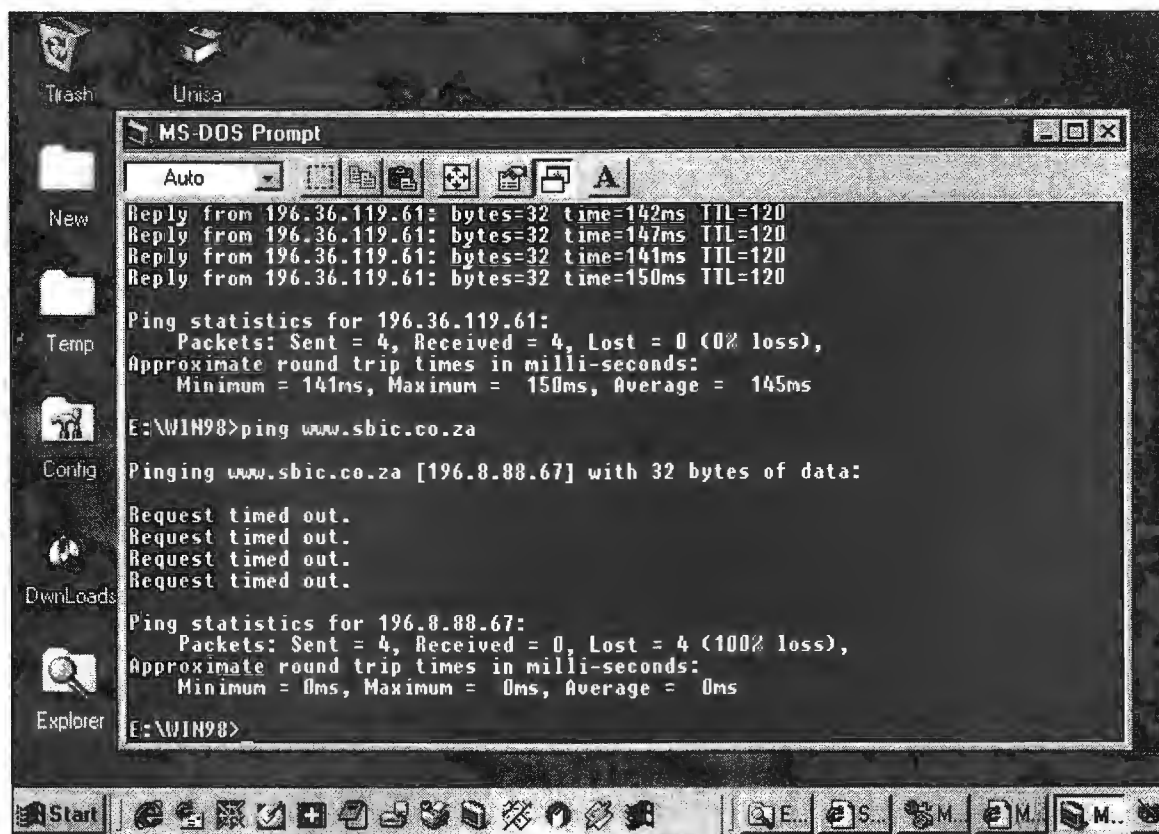
The user enters instructions to the computer through the use of a formally defined command language. This is the traditional method of issuing computer instructions. The sometimes bizarre (especially in its naming conventions) but powerful UNIX command language and the less flexible MSDOS command language are examples of this style (Baecker et al., 1987:428). Dix et al. (1998:116) consider this style to be very powerful (through its direct access to system functionality), but difficult to learn and use. Modern GUI's still use this style – for example in the text fill-in part of dialog and configuration boxes, as well as in their configuration and start-up scripts, and registries.

Figure 2.5 is an example of a command line-based interaction. The Standard Bank Web server is pinged from the DOS window under Microsoft Windows 98. It is interesting that for both Standard Bank and First National Bank ([www.fnb.co.za](http://www.fnb.co.za)) the server is configured to ignore such requests, but that the ABSA bank server ([www.absa.co.za](http://www.absa.co.za)) politely but insecurely returned the request – as is shown in the upper part of the DOS window.

### 2.6.2 Windows interaction style

The user's screen is divided into a number of possibly overlapping rectangular areas, each of which handles a specific function or is a virtual terminal. Because each window can in principle support its own human-computer dialog, the interleaving of multiple concurrent activities at a single workstation is facilitated. Aspects such as tiling versus overlapping, cascading (using a deck of cards metaphor), panning (scrolling in whole window units), and windows sets are of interest (Baecker et al., 1987:432) (Shneiderman, 1992:351-353). Tiled and overlapping windows implementations are sometimes (aptly) described as being respectively ‘tidy’ or ‘untidy’ desktop motifs. Sutcliffe (1988) also refers to this windowing aspect as ‘cluttered’ and ‘uncluttered’ desktops.

Figure 2.5 Command line interaction with the Standard Bank Internet banking Web site



Source: www.sbic.co.za (1999).

Shneiderman (1992:341) offers an approximate definition for an interface window taken from the Microsoft Windows 3.0 user's guide: *'A window is a rectangular area that contains a software application or a document file. They can be opened, and closed, resized and moved. Several of them can be opened on the desktop at the same time, and they can be shrunk to icons, or enlarged to fill the entire desktop.'*

Windows have various associated aspects that increase their usefulness, such as scrollbars (used for scrolling the text or graphics within the window), borders (frames), titlebars and various control boxes for resizing, closing or minimizing the window (Shneiderman, 1992:342) (Dix et al., 1998:124). Shrinking a window to its representative icon, expanding an icon into a window (iconifying and de-iconifying), and moving and resizing the window, are common windowing operations, all part of the system's 'windows housekeeping'.

Arranging windows into sets (called 'rooms' (Shneiderman, 1992:340), can aid in matching window strategies to user tasks – it is a type of windows macro that enables acting on several windows at once. Coordination of multiple window and window sets, through actions such as simultaneous

scrolling, hierarchical browsing, dependent windows (opening and closing), are also on occasion useful.

This appears to be the prominent (if not dominant) paradigm in dialog styling – note the use of the style identifier in the names of the various windows interface systems such as X-Windows and Windows 95.

*This style constitutes the Window aspect of the WIMP interface.*

### 2.6.3 Direct manipulation interaction style

The user manipulates through a language of buttons and pointing devices a graphic representation of the underlying data. This style is also known as a ‘point and select’, a ‘point and click’, or even a ‘drag and drop’ style (Mayhew, 1992:288). Dix et al. (1998:121) observe that this is the predominant interaction style found on the Internet – the user navigates by first pointing his/her mouse cursor at a hypertext link, followed by a mouse click action. For example in figure 2.6 the user selects transaction options by pointing to the options ‘menu-bar’ in the browser window, and activates it by clicking on the text hyperlink.

Typical features of this ‘graphical-cursor-oriented’ style are physical action (movement or selection by mouse, joystick, touch screen, trackball etc.), labeled button pushes, rapid, incremental, reversible operations whose impact on the object is immediately visible, and a continuous representation of the object of interest. Advantages of this style include rapid, incremental learning, diminishing the need for error messages, an immediate feedback of action, a reduction in anxiety because of easy reversibility, and better prediction of system response (Baecker et al., 1987:432).

Shneiderman (1992:181-229) gives this ‘spatial representation’ style extensive attention, and regards it to have exiting potential. According to him it supports the principles of virtuality: manipulatable reality representations, and the principle of transparency: applying intellect directly to the task, with the tool disappearing. Direct manipulation also facilitates bridging the ‘Gulfs of Execution and Explanation’ (from Norman's influential model of human-computer interaction), i.e. the focus should be on the task ‘What’ and not on the computer ‘How’ (Shneiderman, 1992:203). Weller et al. (1992:321) and Dix et al. (1998:106) also comment on these discrepancies between the user's goals and the computer's state. The *execution gap* is an indication of the difficulty of getting the computer to perform a task, and is decreased by using a high level and specialized interface language or style. The *evaluation gap* is a measure of the difficulty in determining whether a goal has been achieved. If the interface feedback language is different from that of the user this gap widens.

According to Sutcliffe (1988:166-167) Shneiderman coined the term direct manipulation in 1983 to refer to WIMP interfaces such as those found on the Apple Lisa. The foundation for this style can be summarized as a set of seven principles (Sutcliffe, 1988:167):

- (1) **Explicit action:** the user points at, and manipulates objects on the screen rather than addressing the objects through intervening code as in command languages or hierarchically as in menus.
- (2) **Immediate feedback:** the results of the user action are immediately visible.
- (3) **Incremental effect:** user actions have an analogue/sequential dimension, such as dragging an object with the cursor.
- (4) **Intuitive interaction:** the interaction matches the user's conceptual (mental) model of how the system should operate, and the display shows pictograms of familiar objects.
- (5) **Layered learning:** 'Onion peel learning'. The system complexity is gradually revealed in layers as the user explores system facilities.
- (6) **Reversible actions:** all actions can be undone by reversing the sequence of operations.
- (7) **Pre-validation:** only valid operations have an effect, and invalid operations show no effect.

Weller et al. (1992:323-325) discuss this style in detail, and note that its essence is '*an orienting notion*'. It succeeds as an '*empowering*' environment because it draws on the user's ability to recognize familiar comprehensive objects and processes of the interface's representation of real-world situations, and also transfers meaning metaphorically to the abstract, complex problems of the real world that the user wants to solve. This style exhibits a paradigm shift from a view of a language as description, towards one as a language of action. It shows what is meant instead of using a command to say it.

*This style represents both the Pointer and Mouse aspects of the WIMP interface.*

#### 2.6.4 Iconic interface interaction style

System feedback and user commands are expressed as graphical symbols or pictograms instead of words. Icon styled interfaces have advantages such as visual distinctiveness, a potential larger universality than text, and possessing an extensive set of image attributes such as colour, style, shape, size, and placement. A number of disadvantages such as the difficulty in finding close picture equivalents of computer concepts also exist (Baecker et al., 1987:431). The 'tool-tip' help or descriptive widget found in the Windows and browser interface is in essence also an icon.

Shneiderman (1992:207) mentions that semioticians (semiotics being the study of signs and symbols) embraced the concepts of visual language and visual thinking, and that the computer through its remarkable visual environment, challenged the traditional 'logical, linear, text-oriented, left-brained, compulsive, rational programmer'.

His explanation for the origins of 'WIMP' in this regard is amusing and may be partly correct – it refers to how these traditionalist programmers viewed the disciples of these new interfaces.

Applying semiotics guidelines yields the following five levels of icon design (Shneiderman, 1992:210):

- (1) **Lexical qualities:** Machine generated marks – pixel shape, colour, brightness, and blinking.
- (2) **Syntactics:** Appearance and movement – lines, patterns, modular parts, size, shape.
- (3) **Semantics:** Objects represented – concrete, abstract, part-whole.
- (4) **Pragmatics:** Overall legible, utility, identifiable, memorable, and pleasurable.
- (5) **Dynamics:** Receptivity to clicks – highlighting, dragging, and combining.

Marcus (1993:102) also discusses the use of visual semiotics in terms of four related dimensions: lexical, semantic, rhetoric and syntactic.

*This style constitutes the Icon aspect of the WIMP interface.*

### 2.6.5 Graphical interaction style

The user is modifying and defining sketches diagrams, renderings, and other two and three-dimensional images or pictures. Baecker et al. (1987) caution against blindly adapting this approach, and state that the design of any interactive dialog must be based on the examination of six factors – the dialog partners (usually the human and computer), the display technology (generally the output), the input technology, and the content and context of the dialog. The adoption of a graphics standard may be in conflict with the important goal of providing the most congenial and fluid interface possible, through the optimal positioning of the six factors (Baecker et al., 1987:433).

Sutcliffe (1988:168) is of the opinion that the direct manipulation style gave rise to the WYSIWYG acronym (What You See is What You Get), i.e. delivering an exact image of what the user expects to see, but it is perhaps more appropriately viewed as being sourced from the graphical interaction style. The 'visual interface style' has many elements in common with this style (Gulliksen et al., 1995:140).

### 2.6.6 Menu interaction style

The user issues commands by selecting, in sequence, choices from a menu of displayed alternatives. The menu-based approach makes computers more accessible to those with little or no computer skills ('naive' or 'casual' users). Issues usually considered are the height, width, and breadth in tree-structured menus, names versus numbers versus icons as menu constituents, and scrolling, paging, pop-up, and pull-down techniques. The standard Windows Internet browser relies to a moderate



extent on this type of style (Baecker et al., 1987:430). Dix et al. (1998:117), mention that the menu interface is designed to compensate for the fact that a human's ability for information recall is inferior to the ability to recognize it from a visual cue – menus provide information cues in the form of an ordered list of operations which can be scanned. Menus are often associated with keyboard accelerators ('shortcuts'), i.e. key combinations that have the same effect as selecting a menu item. An example of the menu interaction style is shown in figure 2.6 where the user can open the browser's file menu, and from there the user can then print the screen contents.

### **2.6.7 Form-filling interaction style**

The user issues commands by filling in fields in one or more forms displayed on the screen. It differs from the menu-based style where a display of alternatives are offered, in that a display of requirements are shown. These are usually integrated as various options and values on a single screen. The quality of these interfaces is primarily a function of the extent to which the form logic mirrors the system logic, the clarity of design and visual representation of the screens, and the degree to which reliable and correct data entry is facilitated (Baecker et al., 1987:431). Dix et al. (1998:119) consider spreadsheets to be a special case of this interaction style.

Dialog boxes with a text box fill-in are common in the GUI world. The Question and Answer (Q+A) interaction style shares many attributes with the form-fill interaction style. They are in general simple interfaces for providing a usually YES/NO type input to an application in a specific domain, although a list of choices is also sometimes used. They are easy to learn but of limited flexibility and power (Dix et al., 1998). Modern GUI's and their Internet applications frequently use a version of these Q+A interfaces in the form of radio-buttons and checkboxes – both can be viewed as graphical 'toggles'. A version of these toggles is also often found in the menu component of GUI's – it is used to indicate whether an option is set or reset (active-inactive).

The left-hand window in figure 2.3 is an example of a form-fill interaction style – the Internet-banking user has to enter his/her account number and a password (PIN number) to gain access to he transaction facilities. This window also has two radio-buttons to select an option to change his/her PIN number. Another example is in figure 2.6. The banking client can change his/her personal details from the customer services option on the menu-bar.

### **2.6.8 Programming language interaction style**

This command language style allows for auto-extension through the definition of procedures, macros, and functions. This could include database query and retrieval languages. By definition, through its extensibility in vocabulary and capabilities, both the UNIX and MSDOS command languages show characteristics of this style (Baecker et al., 1987:429).

**Figure 2.6 Form-fill interaction during a Standard Bank Internet banking session**

The screenshot shows a web browser window titled "Standard Bank Internet Banking - Microsoft Internet Explorer". The address bar contains the URL: <https://www2.encopt.standardsbank.co.za/cgi-bin/encopt/bankint/nc3636cal#a3302A7#b67#14/banint.html>. The page header includes "Standard Bank Internet Banking" and "Share and portfolio services". A navigation menu contains links for "Balance enquiry", "Statement details", "Account payments", "Interaccount transfer", "Beneficiary maintenance", "Customer services", and "Log off". A "Customer details" section is active, displaying a form with the following fields:

Card number:	21 738
Name of cardholder:	VAN DYK TJ
E-mail address:	<input type="text" value="vdykj@alpha.unisa.ac.za"/>
Telephone area code:	<input type="text" value="012"/>
Telephone number:	<input type="text" value="4296676"/>
Fax area code:	<input type="text"/>
Fax number:	<input type="text"/>
Occupation:	<input type="text" value="Lecturer"/>
Charge account for monthly subscription charge:	<input type="text" value="9-5 MENLYN BRANCH ACHIEVERPLAN"/>

At the bottom of the form, there is a link for "Registration for Internet Packages".

Source: [www.sbic.co.za](http://www.sbic.co.za) (1999).

### 2.6.9 Natural language interaction style

The user's command language is a significant, well-defined (structured) subset of some natural language (usually English). These avoid the highly stylized and grammatically limited nature of command and programming language styles, and therefore increase the expressiveness of user input. The first practical example of such systems are now appearing such as the command interfaces for IBM's ViaVoice system ([www.ibm.com](http://www.ibm.com)), and Dragon Systems Dictate as bundled with Corel Office ([www.corel.com](http://www.corel.com)). Weller et al. (1992:323) mention that the robot HAL in 2001:A Space Odyssey, exemplified this type of interface.

### 2.6.10 Function Key interaction style

Commands or objects/operations are specified through function buttons (special dedicated purpose or hard coded keys), on a keyboard or touch-screen. The membrane keyboards used in process control systems and appliances, and the limited functionality, parallel port driven toddler's keyboards (such as the 'Comfy Keyboard'), are examples of this interaction style, but it is also found frequently in many other applications such as word processing and spreadsheet programs. If the special purpose

keys are programmable and changeable, the term 'soft' function key is commonly used (De Wet, 1994:51). In the Internet-browser these keys are also extensively used – for example pressing the F5 function key will refresh the current page.

## 2.7 Conclusion

The three general usability principles and their properties as discussed in this chapter will form the basis for the rest of the research project and the heuristic evaluation as presented in chapters 7 and 8. These two chapters will add Web-specific properties to those identified here. Chapter 3 will examine the issue of usability as tailored for the Web.

It was shown that for the WIMP-based Windows-hosted Internet-banking application most of the interaction styles examined here are present. The windows, direct manipulation (point and click), menu, and form-fill styles are particularly important.

It should be noted that the research makes extensive use of software products from Microsoft. These include the operating systems used during the two year duration of the research namely MSDOS, Windows 3.11, Windows 95 and 98 and NT 4.0, and the Internet browser which progressed from version 3.0 to 5.0 during the period. (The Netscape 3.0, 4.0 and 4.5 versioned browsers were also used.) The author on two occasions commented favourably on the extent of Microsoft's commitment to usability testing and evaluation of its products (Van Dyk, 1996, 1998). Further information regarding Microsoft's usability-centered development approach can be found in Cusumano and Selby (1995, 1997), and in Dieli (1994).

## Chapter 3

# Usability for the Web

This chapter examines how the usability principles as described in chapter 2 apply to Web-based applications such as Internet banking.

## 3.1 Introduction

*'The smallest of usability problems, when multiplied across thousands or millions of users, becomes a source of massive inefficiency and untold frustration.'*

Cockburn and Jones (1996:105), quoting Sun's Jacob Nielsen (1993).

Buckingham-Shum and McKnight (1997), introduce a special issue on Web usability by stating that: *'there was something missing between the vast amounts of hypermedia and related human-computer interaction research that has been conducted, and the most popular hypermedia system in existence – the World Wide Web'*. They note that the *'Web community by and large'* seemed to be ignoring hypermedia and usability research.

Smith, Newman and Parks (1997), expand on this view and emphasize that *'a decade of research into the usability of hypertext and hypermedia systems'* can be directly applied to World Wide Web pages – which can be regarded as being predominantly hypertext with hypermedia appendages. Using the term Web *'page'* is somewhat misleading – the incorporation of interactivity and languages such as Java, JavaScript, ActiveX, and VBScript would imply that the *'page'* has now become a *'program'*. Thimbleby (1997) cautions that the enormous size, number and variety of users, and dynamic nature of the Web do mean that hypertext usability problems are likely to be magnified by a *'factor of millions'*.

Bieber, Vitali, Ashman, Balasubramanian and Oinas-Kukkonen (1997), distinguish between **hypertext** and **hypermedia** (which are often used interchangeably (Dix, Finlay, Abowd and Beale, 1998:593)), by stating that *'hypermedia nominally applies hypertext concepts to multiple media'*. Importantly Bieber and Isakowitz (1995:8) observe that *'hypermedia is the science of relationships'* – it is concerned with the structuring, presenting, and access to content within an information domain.

## 3.2 Web usability

The general usability principles and properties as examined in chapter 2 also apply to the design of Internet and Web-based interactive software products. Because of certain unique characteristics of Web-based applications (such as response, feedback, browser, bandwidth and graphics limitations, and the extremely wide range of client characteristics (Mayhew, 1998:4)), it is also appropriate to examine other more specific hypermedia-orientated usability guidelines.

IBM (URL: IBM3), makes a careful distinction between **principles**, **guidelines** – often referred to as heuristics ('rules-of-thumb', Nielsen (1993)), and **conventions**:

**Principles** are the goals that guide design decisions. They reflect knowledge about human perception, learning and behaviour.

**Guidelines** are based on principles, but are specific to a particular design domain (such as web page design).

**Conventions** dictate specific design decisions. A choice is made that aligns with the site goals and audience requirements.

Garzotto, Mainetti and Paolini (1995:74-75) present the following heuristic *evaluation* criteria (table 3.1), for hypermedia design, and also list (table 3.2) five usability *design* guidelines for hypermedia applications.

**Table 3.1 Usability evaluation criteria for hypermedia design**

1	<b>Richness</b>	Information, Reachability, Navigation
2	<b>Ease</b>	Accessibility, Understanding
3	<b>Consistency</b>	Regularity, Similarity
4	<b>Self-evidence</b>	Guessability
5	<b>Predictability</b>	Anticipation
6	<b>Readability</b>	Validity, Clarity
6	<b>Reuse</b>	Objects and Operations
Source Garzotto, Mainetti and Paolini (1995).		

**Table 3.2 Usability analysis guidelines for hypermedia design**

1	<b>Content</b>	Information, Static-Animation
2	<b>Structure</b>	Content Organization
3	<b>Presentation</b>	Visualization, Navigation
4	<b>Dynamics</b>	Interaction
5	<b>Interaction</b>	Dynamics and Presentation
Source Garzotto, Mainetti and Paolini (1995).		

Bowman (URL: MAST1) considers the elements as presented in table 3.3 to be the most important Web usability design aspects. He adds an iterative approach to design – which is also an accepted general design principle. Most of his elements correspond closely to those from IBM, as discussed next.

**Table 3.3 Bowman's usability guidelines for Web page design**

1	<b>Information Structures</b>	Sequential, hierarchical, associative
2	<b>Information Organization</b>	User categories, mental models, icons, and intuitiveness
3	<b>Short-term Memory and Information Mapping</b>	Functional blocks, chunking, relevance, labeling, consistency, integrated graphics, accessible detail, and hierarchy
4	<b>Iterative Design</b>	User testing
5	<b>Unified Design</b>	Constant design – orientation clues, predictable location, reinforced user expectations, enhanced usability, and user satisfaction Global controls – overpowering, visible differences, banners, and location clues
6	<b>Java</b>	Appropriate animation and use

Source: Bowman, URL: MAST1 (1997).

IBM continues (URL: IBM3) their discussion on guidelines and principles by listing six Web design usability guidelines as presented in table 3.4. Note their focus on hypermedia high-impact elements, such as navigation, HTML, and Java.

**Table 3.4 IBM's usability guidelines for Web page design**

1	<b>Structure</b>	Navigation structure Information space structure Page definition – Home, Navigation, Content, Input pages
2	<b>HTML</b>	Design strategy Table and Frame layout
3	<b>Visuals</b>	Navigation and Information content Affordance Visual feedback Layout Visual identity Continuity and consistency
4	<b>Engagement</b>	Content variation User feedback
5	<b>Effectiveness</b>	Messages Information prioritization Consistent style Use of graphics Design for printing
6	<b>Navigation</b>	Navigation cues
7	<b>Messages</b>	Clear error messages Keep users informed
8	<b>Java</b>	Use of Java applets

Source: URL: IBM3 (1997).

Bevan (1997, URL: NPL1) also examines some of the more important usability issues in Web site design. He regards the main reasons for the slow and confusing use of many Web sites to be due to the following:

- (1) The Web site has a content and structure that mirrors the internal concerns of the organization rather than that of the user.
- (2) The Web site contains material that is appropriate in printed form, but needs to be adapted for Web presentation. (The opposite case also frequently applies to Web sites.)
- (3) Web site publishing has now become very easy, and it is usually not subject to the same quality criteria used for other forms of publishing.

Reason (2) and especially reason (1) are considered to be major problem areas in the case of online banking Web sites. These two aspects are examined in more detail in chapter 9 (Heuristic evaluation).

Bevan (1997) then provides a number of guidelines for the design of highly usable Web sites (table 3.5). Note that '*site design*' should be considered an earlier, larger and higher (more abstract), design activity than the previously discussed '*page design*' guidelines.

**Table 3.5. Guidelines for high-usability Web-site design**

1.	<b>Planning</b>	<b>Provider requirements</b> – define the <b>business objectives</b> of the site. <b>User requirements</b> – specify in detail the intended context of use. Define <b>key scenarios</b> (user classes, task sequences-goals) of use.
2.	<b>Site Structure</b>	<b>Meaningful structure and information content.</b> Use card sorting.
3.	<b>Navigation Support</b>	<b>Help users find their way.</b> Highlight <b>important links.</b>
4.	<b>Page Design</b>	Design for <b>effectiveness.</b> Design for <b>efficiency.</b> Support <b>different browser</b> environments. Support <b>visually impaired</b> users and <b>text-only browsers.</b>
5.	<b>Evaluation</b>	Expert inspection. Early mock-ups. Functional prototypes.
6.	<b>Implementation</b>	Ensure pages meet <b>quality and usability requirements.</b>
7.	<b>Maintenance</b>	<b>Plan and review site growth.</b> <b>Monitor</b> user feedback, arrival points on the pages, and search parameters. Check <b>broken links.</b> Compare to other similar sites. Keep track of <b>browser and site design developments.</b>

Source: Adapted from Bevan (1997, URL NPL1).

Bevan (1997) continues by analysing the aspects listed in table 3.5 in more detail. Business objectives require attention to aspects such as the purpose of the site, user categories and goals, user attraction and attention, and quality and usability goals. User requirements imply attention to aspects such as who, why, how, the frequency of visit, and user demographics. For navigation, consistency and depth, meaningful names, and navigation buttons are amongst the aspects discussed, and for page design animation, graphics, interlacing, background patterns, colour, site identity, and overviews are important.

Thüring, Hannemann and Haake (1995), are of the opinion that two factors are crucial for improving the readability of Web pages, namely **coherence** – which supports the readers ability to understand and remember text, and **cognitive overhead** – which decrease the effort and concentration required by the reader. They then propose eight principles that can be used to support the cognitive design issues that stem from these two factors. Their principles are shown in table 3.6.

**Table 3.6 Cognitive design principles**

1	Indicate equivalencies between information units.
2	Preserve the context of information units
3	Use higher order information units
4	Visualize the structure of the document
5	Include cues for the structure visualization that show the readers current position, the path to the current position, and navigation options for moving on
6	Provide a set of complimentary navigation facilities which cover aspects of direction and distance
7	Use a stable screen layout with windows of fixed position and default size
8	Use typed link labels
Source: Thüring, Hannemann and Haake (1995).	

A comparison between the Web-oriented usability characteristics and the general usability properties listed earlier, indicates that most of these general properties are implied and embedded in the characteristics as listed by IBM (URL: IBM3) and Bowman (URL: MAST1). (Chapter 7 – Usability evaluation, reexamines this aspect.) The focus has now shifted towards user enjoyability, engagement, support, guidance and feedback, and interaction consistency and response. This is not surprising, given the (current) high focus on the Web as an entertainment medium and the immaturity of the technology. This should be combined with the only average degree of computer-sophistication of the typical Web user – as commented on by Katz and Aspden (1997:97), and Hoffman, Kalsbeek and Novak (1996:36), and the bandwidth limitations imposed by the communications technology (which is more of a problem locally than in the bandwidth-rich USA).



The correspondence between the earlier presented (chapter 2) general usability principles and those more specific to the design of hypermedia are illustrated on the basis of the three principles of learnability, flexibility and robustness.

### 3.2.1 Learnability

This is the ease with which maximal performance and effective interaction is achieved and covers both learning and remembering (Sutcliffe, 1988:184) (Dix et al., 1998:162).

This principle forms the bulk of Garzotto et al.'s (1995) evaluation criteria (table 3.1). Predictability and consistency are listed as properties of the learnability principle as presented in table 2.1. Visual clarity (table 2.1) requires readability (table 3.1). Self-evidence or guessability (table 3.1) is related to familiarity (table 2.1). Enjoyability and user guidance require richness (table 3.1). Ease and understanding (table 3.1) require the presence of all the learnability properties.

Bowman's (URL: MAST1) information organization (table 3.4) is covered through the properties of mental models, synthesizability (table 2.1), and guessability. Effective information mapping implies consistency, visual clarity, and user support and guidance.

Atkinson and Kydd (1997:53) present a study on playfulness and the Web – their findings are applicable to the related properties of engagement and enjoyability. They confirm that the presence of these properties tend to motivate the user to return to the Web site. Kim and Moon (1998) study the emotional aspects of 'cyber banking' systems – they suggest guidelines for inducing appropriate target emotions in the user.

Thüring et al.'s (1995), cognitive design principles also focus strongly on learnability issues – their principles cover consistency (stable layouts), information cues and visualization to aid in remembering, and a logical organization that will be consistent with the user's mental model(s).

### 3.2.2 Flexibility

This refers to the exchange of information through a number of choices and options.

With the exception of the use of programming languages such as Java and ActiveX on the Web page, which increase the options and power given to the user for customization (Dix et al., 1998:171), this principle seems to receive less attention, and is covered to a lesser extent by the previously presented hypermedia guidelines. Bevan's (1997) guidelines for effectiveness, efficiency and supporting multiple browser environments are in part related to this principle. Nielsen (1998) notes that one of the important differences between Web and traditional software design comes from the designer passing much of his/her control for the user interface to the Web user and the client hardware and

software. Rice, Farquhar, Piernot, and Gruber (1998) also comment on this aspect of passing control to the user, and examine some of the browser-related rendering and design aspects.

The two main Internet browsers (Netscape and Explorer) are customizable to a significant extent (such as the America Online (AOL) customized browser based on Internet Explorer), and the operating system usually offers support for multi-threading (for example running more than one browser window at the same time). What will be required from the Web document for effective support for this principle will be a user-initiated and controlled customization of the entire page presentation and layout. The recent (abortive?) focus on push-technology, i.e. only delivering the content required by the user, should enhance the support of this principle. This aspect of page customization for Internet-based banking is examined in more detail during chapters 7 and 8.

### 3.2.3 Robustness

This lends support for successful goal achievement and assessment.

Web design guidelines such as Bevan's (1997, URL: NPL1) navigation support, IBM's (URL: IBM3) feedback and meaningful messages, navigation cues, keeping users informed, and MIT's (URL: MAST1) orientation cues all support this principle. The properties of observability (through browsability, reachability, persistence and operation visibility), and recoverability (through reachability, and forward and backward recovery), are especially evident. Garzotto et al.'s (1995) accessibility is virtually synonymous with the aspect of reachability. The issue of responsiveness (through increased bandwidth, better compression techniques, careful graphics design, etc.) is the subject of intense research, and the guidelines relating to the judicious use of graphics as presented by IBM (URL: IBM3) are related to this. Johnson and Gray (1998) note that '*delays in response time can lead to frustration and error*' – i.e. it also impacts on the enjoyability aspect. Task conformance (how the user is supported in all his/her tasks), in terms of task completeness and task adequacy, also receives some attention – Bevan's (1997) planning stage focus on business and user requirements and key (task) scenarios.

## 3.3 Web Usability in practice

As an excellent (and well-known) real-life example of how a Web page had been designed and evaluated for usability, the procedure used by Sun for designing and developing their home page is presented below.

**Sun Microsystems.** Sun (URL: Sun1) (Nielsen, 1995(3):98) presents their approach to usability testing of WWW pages and its design, and specifically how this approach was applied to the development of their own home page. This is an often-used example in literature – Shneiderman

(1997) notes that this specific exercise has '*highlighted key principles*', Morville (1997:34) considers it '*intriguing*'.

The design procedure followed by them – executed in their own usability labs as shown in figures 3.1 and 3.2 – consists of a series of initial user reaction tests using paper prototyping (figure 3.4) (Nielsen, 1995(2):88). It also includes the observation of a selection of different users browsing a number of carefully selected (for both similarity and dissimilarity) Web pages from other companies (Nielsen 1995(1):75). A card sorting study (figure 3.3) was used to ascertain the user's classification of concepts and own view of the system (i.e. mental or conceptual model(s)). This development phase was mainly used for the grouping of icons on the Web page. This stage is followed by more traditional usability tests using a (partly functional) prototype or even just a skeletal HTML document mockup with dangling links (stubs).

It is interesting to note that Bowman (URL: MAST1) proposes a design and evaluation methodology very similar to that as used by Sun – his information organization stage is based on Nielsen's (previously employed by Sun) four stage procedure – to which he added an additional step (step 2):

- (1) Card Sorting to discover categories.
- (2) Icon assignment to categories.
- (3) Icon intuitiveness testing.
- (4) Card distribution to icons.
- (5) A think-aloud walkthrough of page mockup and rating icon aesthetics.

**Figure 3.1**  
A general view of Sun's usability lab



Source: URL:Sun1

**Figure 3.2**  
A view of the observer side and equipment



The development of Sun's home page went through more than 15 iterations, with the first, an intermediate and the last (final) design as shown in figures 3.5 to 3.7. Their first page was considered misleading and confusing by users, being very 'busy'. Their intermediate design was the first to promote Sun's new Java technology and their HotJava Web Browser. The Java mascot ('Duke') was not liked by users, and was replaced in the next iteration (not shown), with the now ubiquitous Java coffee cup and saucer. Their final design shows a number of clearly defined clickable areas, and had 'the steaming cup look less like a nuclear explosion' than the preceding version.

Figure 3.3 Card sorting of concepts

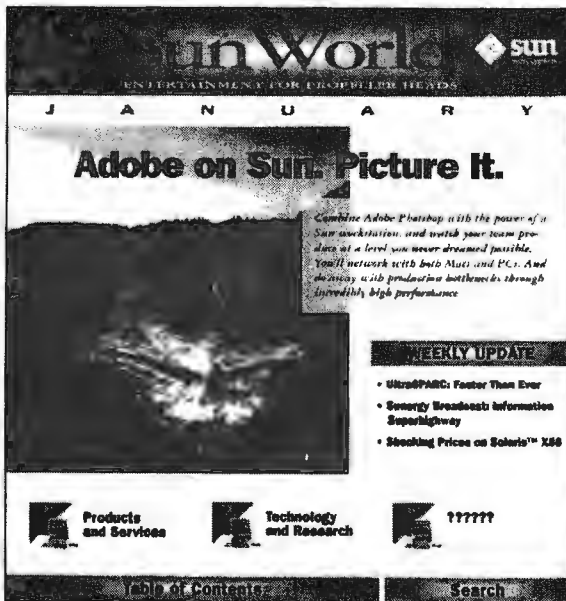


Source: URL: Sun1

Figure 3.4 Paper prototype



Figure 3.5 Sun's first design



Source: URL: Sun1

Figure 3.6 Sun's fifth design



Figure 3.7 Sun's final home page design



Source: URL: Sun1

### 3.4 Conclusion

All of the general usability principles and properties (as listed in table 2.1) can be applied with good effect to hypermedia-hypertext designs. Many of the Web-specific usability guidelines are either relabeled general usability properties, or a combination of these, and could also qualify for inclusion as general evaluation criteria. Certain of the Web-specific criteria combine usability properties through an overlap between the three classes of general usability principles – but the boundaries for these three are in any case not always distinct – as will be examined in chapter 7, figure 7.2. Importantly Mayhew (1998:4) observes that the difference between Web (interface) design and traditional software design is a matter of degree – the knowledge acquired in designing the latter systems can be effectively applied to the World Wide Web but seldom are.

It can therefore be concluded that what is significant is that the importance of some of the general criteria or usability properties is amplified (or prioritized) for Web-based or hypermedia applications. An aspect such as navigation which is related to general properties such as reachability, user guidance, and browsability, is one example of this. For many entertainment focused Web sites properties such as enjoyability and responsiveness (for streaming audio and video applications) are

disproportionately important. Other Web sites that focus on the presentation of their information content should be evaluated more strongly in terms of properties such as visual clarity, user guidance, and task conformance.

The discussion as presented here should be read in conjunction with the range of Web-specific heuristics as presented in chapter 7. The guidelines presented there are more tailored and specific than those examined here and will focus on the evaluation of Internet-based banking applications.

## Chapter 4

# Internet-based banking

An overview of Internet banking and an examination of a selection of banks that offer Internet-based banking are included in this chapter.

## 4.1 Introduction

*'First came ATM's where transactions could be made electronically. Then telephone banking. Now newer innovations such as Internet banking, video kiosks and interactive TV banking will enable banks to serve customers at reduced costs.'*

*'We will see smaller, more tailored branches that will be used mostly for advice and building relationships while transactions will be made by cheaper electronic methods. Increasingly people are using telephones and computers to manage their everyday lives'* – Charles Chemel of Standard Bank (South Africa).

New technology means other players can move into bank territory using cheaper electronic distribution channels.

Source: Financial Mail, 10 October 1997.

The 'Big Four' group of South African banks started offering Internet-based online banking services from the end of 1996. **ABSA** (Amalgamated Banks of South Africa) – which previously included United, Trust, Allied and Volkskas Banks – was the first to offer an initially limited in range online facility in December 1996. In the period January to May 1997 **Nedbank** (including Peoples Bank and Permanent Bank), **First National Bank** and lastly **Standard Bank** established their banking services. Initially all three offered a more complete product range than ABSA. In August 1997 a fifth bank (Mercantile Bank) joined the South African online banking community (Business Times, 1997) (Computer Week, 6 October 1997).

Accone (1999) presents interesting statistics on the current state of Internet banking in South Africa. According to the 1999 SA Web User survey less than half of South African Internet users are aware that they can bank through the Internet. A third make use of such a service – i.e. by implication almost two-thirds of those that are aware of the facility do make use of it. Accone (1999) also notes

that those banks that have marketed their online services extensively have reaped the rewards. ABSA and Standard Bank are used by 11% of users followed by Nedbank (8%) and FNB (3%). Confusingly these percentage values refer to the 33% Web users that use online banking facilities – when scaled to 100% they are 33% (ABSA), 33% (Standard Bank), 24% (Netbank), and 9% (FNB). Accone incorrectly states that there is little to distinguish the services from one bank to the other – this may be true for the features offered but not as far as usability is concerned (as discussed in chapters 7 and 8). It is likely that FNB's low adoption rate is in part due to its almost total absence of marketing – but note the comment made in section 4.3.2 on the validity of these user figures. The service fees charged by the banks (as during May 1999) vary between R22-80 p.m. (Nedbank) and R11-40 p.m. (ABSA.). Standard Bank charges R18-00 p.m. and FNB R12-00 p.m.

In the USA Internet-based online banking services are already well established (URL: Orc1) – more than 75 USA-based and credit unions were listed in late 1997. By middle 1998 the USA-based Netbanker's online banking report listed 167 USA banks that offered full Internet banking (URL: Netbanker1). Accounts can be paid through the Web, via a screen-phone or touch-tone telephone – similar to the Beltel service here – and users can import their transactions directly into Microsoft's Money or Intuit's Quicken. Netbanker also keeps a list of the 100 largest banks in the USA – four of the banks examined during this research are in the top 10 listed namely Citicorp/Citibank, BancOne/BankOne, Wells Fargo, and BOFA (Bank of America). Their (1998) total assets vary between \$311 billion (Citicorp) and \$98 billion (Wells Fargo). Out of the top 10 six banks offered full Internet banking – the other four only had a Web presence.

Marsden (1997:29) reports that IBM is working with 16 banks in North America to establish a pan-industry solution (the Integriion Financial Network) that will be used by about 60 million households for their banking services. Acceptance of the technology has been rapid in the USA. Since the report by Computer Week (18 November 1997), where it was noted that EC (electronic cash) initiatives in general and internet banking specifically, at large American banks such as ABN AMRO, Chase Manhattan, Citibank, and First Union are slow in the uptake, the growth in the USA has been explosive. The reason for the initial slow uptake was that banks have a fickle relationship with the Internet – they want to reach more customers but they also do not want to appear vulnerable to security breaches. (Both AMRO and Chase Banks are now members of the Integriion Network.)

✓ Diniz (1998) presents a survey on Web sites of banks in USA and identifies a number of Web banking models that are being adopted in the United States. American banks are using the Web to reach opportunities in three different categories – to market information, to deliver banking products and services and to improve customer relationship. Each one of these categories can be divided into three levels of interaction, in order to classify the different kinds of applications that are found on each of the banking Web sites.



Most of the recent banking products are Web and browser-based – the much earlier introduced Beltel (telephone-based) services are the exception to this – and they all require recent versions of Netscape Navigator/Communicator and Microsoft Internet Explorer. Some also support other browsers such as IBM's Web Explorer. The main requirement seems to be browser support for the Secure Socket Layer (SSL), which for example is not implemented by the older Netscape 2. Mercantile Bank's Internet service is not browser-based (Gordon, 1999) (Computer Week, 6 October 1997). It has the advantage in that much of the financial processing is done local and offline (client-based).

Phone-based services such as Telkom's Beltel have been in existence since the late 1980's. These have been very successful in catering for a wide range of home and corporate banking services for all four of the major South African banks mentioned at the start (Marsden, 1997). Webster (1997) reports that the end is in sight for Telkom's Beltel and (French-originated) Minitel services. He quotes Erasmus from Beltel as saying that they (pragmatically) accept the end of Beltel in its current form '*especially once perceptions about Internet security change*'. Most of the about 56 000 South African Beltel users will inevitably move to Internet-based banking, drawn by the faster, more convenient, and flexible functionality. Statistics on Beltel use show that about 80% of its users use the system at least once a week, and about 50% use it daily. If these access patterns are translated to the (much) larger group of potential internet users, South Africa's Internet service providers stand to gain significantly from the general acceptance of this newer technology.

- ✓ These Internet-based services allow banking clients to obtain account information, balance enquiries, and execute account payments and inter-account fund transfers. Additionally, by linking their accounts to personal finance software (such as Intuit Quicken and Microsoft Money), they will be able to track their spending offline, and later reconcile that with their bank statements online. The *South African Banks Act* restricts a bank's ability to provide certain online services such as applications for a credit card, as original paper documentation is required.
- ✓ Marsden (1997:30) mentions that (apart from cost reduction and client expansion), the big factor driving these banks in offering these services seems to be that they do not want to miss the boat by not offering customers what they want. The possibility also exists that software companies such as Intuit and Microsoft may decide to work together with brokerage houses and credit companies to offer this class of service themselves.
- ✓ The Financial Mail (October 10, 1997) reports that in the USA increased competition, deregulation, and the spread of electronic banking reduced the number of banks by more than a quarter between 1985 and 1994. Such a banking consolidation is now evident in Europe and has also been observed here – as witnessed by the local mergers that led to the ABSA, Nedcor, and NBS-Boland groups of

✓ banks, as well as the declining fortunes of niche banks such as the African Bank and Islamic Bank of South Africa, and the emergence of the availability of 'in-house' banking services from certain of the larger retail groups (The Star, 1997). Electronic banking offers a route to salvation for this group of 'traditional' banks, which believed in the past that their (many) branches offered them distinct advantages, but these have now proved themselves nothing more than costly millstones.

✓ For the banks, one of the most obvious advantages is cost. Table 4.1 shows the relative costs to the bank per transaction for the various channels. Although the table was compiled for US Banks the '*figures should be enough to make any bank, in any country, sit up and take notice*' (Hennigan and Gourvennec, 1998).

✓ **Table 4.1 Channels and transaction costs**

**Channel - Cost/Transaction**

1. Branch Full Service : \$ 1.07
2. Telephone Average : \$ 0.54
3. ATM-full service : \$ 0.27
4. PC Banking (3rd party) : \$ 0.015
5. Internet Banking : \$ 0.010

✓ Other surveys have come up with widely different figures. The ABA (American Banking Association) 1998 figures are \$ 00-12 for a branch transaction and \$ 00-01 for an Internet transaction (Pederson, 1998). But there is consistency in one area – they all agree that there are huge (order of magnitude) potential cost savings if the banks manage to carry out a higher percentage of their transactions over the Internet.

✓ Security concerns are still a major stumbling block in the general acceptance of electronic commerce and banking. The main problem in the world outside of the USA is how to deal with the US export ban on strong encryption. Within the USA Web users can use 128bit encryption that is effectively unbreakable. The rest of the world is limited to using only 40bit in standard browser products such as Netscape and Internet Explorer's SSL (Secure Sockets Layer) protocol. This export restriction has recently been relaxed by the USA authorities to 56bit encryption for all exports, and 128bit for certain classes of users such as banks (Melliari, 1998). The encryption strength of the secure data communications link can be increased through the use of Java applets to above 128bit, but this may give rise to another set of usability and security problems (Baguley, 1998).

✓ Initially Standard Bank used a 128bit security socket (layer) developed by Centurion-based Nanoteq. Nanoteq was established in 1988 and is South Africa's leading developer of advanced data and communication security technology. They were recently acquired by Comparex (previously Persetel and Persetel-QData or PQData), as were two other local companies involved in the electronic security market namely Mosaic Software, and CAT (Consultancy in Advanced Technology) (Computer Week, 26 August 1996). Marnitz, CEO of the then Persetel, is quoted as saying:

*'Electronic commerce over local, wide area and open networks, such as the Internet is growing fast, despite concerns about security'. Developments in this area are very rapid, and these coupled with the market forces already discussed, will lead to a less negative perception on security issues during Internet banking in the near future. Wright of the Internet Solution (The Star, 4 November 1997), notes that 'banking on the Internet is alive and well in South Africa', and she recommends using the services offered by the local banks.*

Lagoutte (1996:6) comments on the situation in Western Europe during the middle 1990's – particularly salient points are bold-printed. *'Traditional retail banking is typically characterised by "brick and mortar" branches. Competitive advantage in the banking industry in earlier days came from careful planning of the branch network. Accessibility of banking services to the customer meant branches on every street corner. Expectations of bank customers have risen sharply and they now request banking services to be available any time. The boom of Automatic Teller Machines has shown that customers are willing and able to perform standard banking activities via self-service machines if this means that they have access to their bank accounts at any time they want.'*

*'Convenience seems to be the keyword for the banking industry in the future. Accordingly, most of the larger financial service institutions are experimenting with the use of services such as telephone and home banking. At the same time banks have recognised that traditional banking is costly for the organisation and competitive pressure leads to more cost-awareness. Staff in "brick and mortar" branches is heavily occupied with paper work regarding standard transactions that could be automated. Automation not only means major cost reductions for the banks but also a high increase of accuracy and reliability, since technology, provided that it is set up correctly, is able to perform routine tasks more accurately than humans.'*

*'With deregulation in all of the Western European countries, barriers of entry to providing banking services have declined. Furthermore, insurance companies and other non-financial institutions are entering the market taking away large numbers of traditional retail banking customers. The intangibility of services facilitates the swift copy of any innovative ideas and the number of financial products and services on offer is constantly growing.'*

This comment proved to be a prophetic – the list of European banks presented on the Unisys Banking Barometer Web site (URL: Unisys1) has grown dramatically. Note however that European banks regard Internet-based banking as mainly an extra channel rather than a replacement for their branches (Wallace, 1998:67). One example of this is the Royal Bank of Scotland (RBS) which has 600 000 telephone-banking customers. In June 1997 they were the first to offer fully-fledged Internet-banking in Britain. Towards the end of 1998 they still only had 21 000 Internet-banking

customers – which could be compared with the 40 000 figure for Standard Bank (South Africa) with a similar start-up date.

In contrast one of the larger German banks (HypoVereinsbank) has about 400 000 Internet-banking customers. The difference may also be in part due to the high online charges from UK banks compared to the (almost) free German online banking services as such as Deutsche Bank (Sunday Times, 14 March 1999) (Wallace, 1998) (Pederson 1998).

Baguley (1998) presents a regular column on European Internet banks, and gives a more detailed description of the RBS facilities and Web site. He also presents a summary of the Ernest and Young (1998 UK) annual survey on the impact of technology on banking ([www.ernsty.co.uk](http://www.ernsty.co.uk)). This supports the previous (strategically short-sighted) view that the European (and especially UK) banks do not think that the availability of Internet-banking facilities will gain them any additional customers – they view it as a means of satisfying existing customer needs only.

## 4.2 A classification for banks with a Web presence

According to Unisys (URL: [Unisys1](http://Unisys1)) European Banking Barometer Web site banks and building societies with a Web presence can be placed into one of four categories:

(1) **Net presence.** This is the most basic category, describing sites that concentrate on providing information but fail to recognize the full advantages that the Internet provides in comparison to paper-based information.

Most of these 'Internet-Banks' are PR experiments (albeit sometimes very aesthetic ones) in order to occupy the space which, eventually, will be the primary location for the Bank's soon-to-come online services. The financial institutions also use this type of site to generate awareness or improve its image, for example by printing the site URL on its stationery. In the eyes of the general public the fact that the Net presence is limited does not matter. At the moment (1997) the average user thinks that the Bank is really at the cutting-edge of technology, and for now many banks seem happy enough with this state of affairs. However, when bank customers eventually get on-line and actually try to do something at the bank's site then they will discover that in many cases they can't get any useful services directly from the Web page, let alone open a new account if they have not got one yet. In spite of the limitations this type of site is a useful first step for banks to get a feel for life online.

(2) **Interactivity.** This is the next step up, describing sites that attempt to be interactive with the user. To fall into this category it is not enough simply to supply a feedback form and an e-mail address. The sites need to be well designed and easily navigable, and take advantage of a combination of features such as Java, Java Script, animated graphics, sound and video in order to make the user's visit more interesting and useful. Many banks have included these features, a good example of which is a Java-based mortgage repayment calculator.

Hennigan et al. (1996) comment on this category. *'More and more banks have understood that in the wake of the post-industrialist Society, individuals require interaction, and personalised contact. This is what some sociologists and marketers have called 'mass-customisation'. Each bank customer wants to be perceived as unique as opposed to 'one in the crowd'. As a consequence, personal contact is crucial, and this is certainly one of the areas where the Internet, with features like Email and custom fill-out forms, is more efficient than any other channel. A number of financial institutions have understood this, and they have tried to generate some interactivity with their Web users. Lloyds has gone one step beyond the use of simple requests forms, by providing its Web users with an electronic forum (<http://www.homeworker.co.uk>) on the subject of remote working. As for Natwest, they describe their project entitled 'Banking on the Future' at ([www.botf.natwest.co.uk](http://www.botf.natwest.co.uk)).'*

It is not sufficient solely to present a fill-out form or offer a bit of interactivity. What matters to the potential Internet customer is also that the bank or the building society is capable of processing the request instantly (otherwise, why not use traditional – paper or branch – means of communication).

Although more and more financial institutions are falling into this category as they aim to become more interactive, generating interactivity does not stop with the creation of a CGI (common gateway interface) form. It also requires a lot of work from a marketing point of view. Customers are becoming more and more demanding and they are expecting ever-higher levels of service from their bank.

(3) **Home Banking.** This is defined as the use of proprietary financial software running on a computer in the home, to perform transactions such as fund transfer and bill payment. Banks and building societies usually allow their proprietary software to be downloaded from their Web pages.

This third category has emerged as the need for full convenience banking services emerged. The institutions that belong to it are keen on providing online services to their customers, although not necessarily with the aim of grabbing new clients from the Internet itself, and have done so by writing their own software which they distribute to their users. These software packages are private, and run outside the browser as separate applications which is why they do not come under the category of full Internet Banking.

✓(4) **Internet Banking.** Internet Banking is similar in functionality to Home Banking, with the important difference that it allows access to accounts from a standard Web browser, eliminating the need for proprietary software.

There are two main categories of full Internet banks. On the one hand there are a handful of respectable banks like Wells Fargo, Bank of America or Citibank. They will allow the user to open and manage accounts, pay bills, and transfer money between accounts online. Accounts can be accessed from any browser, any platform, anywhere, and anytime. However non-USA citizens will not be able to open an account with these banks. To open an account in America (whether it happens through the Internet or not), they currently need to have social security numbers. This is not a legal restriction but is just common usage among American banking institutions. This situation is however changing. In a detailed comment on the state of international banking and the Internet, Crede (1997:271, 303) observes that American banks are rapidly internationalizing. They are also adopting the Internet as an effective means for intraorganizational communication by using a combination of e-mail, Web server, and GroupWare technologies, and as a means for greater direct customer contact. Citibank Direct is now accessible from anywhere in the world to non-USA citizens (Baguley (1998).

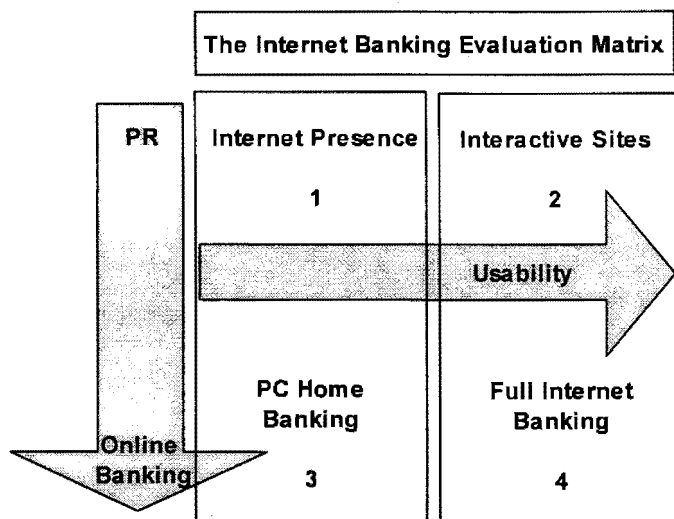
On the other hand, pure online banks have cropped up on the Internet, which exist only in cyberspace and have no physical branches. Security First Network Bank (SFNB) was the first of such to be online. Banking software is their primary activity and the bank was created in order to show that the concept was working. Humphreys (1997:75) provides a detailed description of the important banking trends that were set in motion by this unusual bank. Ohio-based Huntington Bancshares ([www.huntington.com](http://www.huntington.com)) used the same software ('5-Paces') to implement their online bank and they subsequently bought shares in SFNB. Other online outfits include Antigua's European Union Bank and the UK pioneer Banknet

Hennigan et al. (1998) again note that one common confusion is between *Internet Banking* and so-called *PC Home Banking*, which has been around for some time. With PC Home Banking, banks (or building societies) usually send out a financial software package on disks, allowing users to fill in details off-line and then to send them into the bank over the bank's private network. The main difference is that full Internet Banking allows users to access their accounts from any PC (with a browser), without having to carry special software around everywhere.

They continue and offer the following interesting comparison between the four classes of banks with a Web presence (figure 4.1). They compare the site purpose (public relations to full online banking) on one dimension to 'user-friendliness' (usability) on the other dimension – correctly Internet browser-based banking is indicated by them to be the most usable of the four classes.

Pecenik (1998) describes the normal evolutionary stages that a bank follows to reach full-service Internet banking – based on the history of Nuova Banco di Credito di Trieste ([www.nbctkb.it](http://www.nbctkb.it)). The first two steps are the establishment of a link to the Net through a mailbox, which is followed by the creation of a homepage with information. Setting up interactive pages to communicate with clients and integration of the bank's databases follow this. Stages 5 and 6 consist of allowing transactions on the Web and the launching of an e-commerce site. The last stage is the full integration of applications leading to a 'virtual' banking enterprise.

**Figure 4.1 Internet banking evaluation matrix**



Source: Adapted from Hennigan et al. (1996).

### 4.3 Banks offering Internet-based banking

Two (Standard Bank and First National Bank) of the five local banks that offer Internet banking services are examined in more detail below. ABSA's Direct banking is studied through a series of articles in Intelligence magazine (1998). Netbank (evaluated by Melliar, 1998), and Mercantile Bank are the other local banks and they are examined – together with the group of overseas banks – during the discussions included in chapter 5 (Data collection), chapter 7 (Usability Evaluation) and chapter 8 (Heuristic Evaluation).

#### 4.3.1 Standard Bank

Standard Bank introduced their internet-based banking service during the May 1997 Computer Faire at Gallagher Estates in Midrand (South Africa), about two years after their first (non-interactive) presence on the Internet. Their 1997 advertising slogan for their online banking site read: '*One small click for you – one big click for banking*' – which in part reveals the bank's commitment to the importance of the technology. Standard Bank claims in their 1997 information brochure that within

only eight days of going live, the 1000th client registered for their Internet banking service. Currently Standard Bank has about 40 000 Internet-banking clients (Gordon, 1999) – this would translate to about one third of their Internet-enabled potential Internet-banking users.

Standard Bank also offers other electronic banking services. The first is their somewhat outdated **AutoBel** technology. This is based on Telkom's **Beltel** home information service which requires a personal computer with a modem and software (such as PCBel), or a dedicated (Telkom supplied) **Minitel** terminal. They also have a much less flexible phone-based service called **Toni-Bank-by-Phone**. The services offered through AutoBel are similar to those available from their Internet service and include bills payment, balance enquiries, obtaining provisional statements, unit trust purchases, naming beneficiaries, cheque book requests, stop and debit order configuration, loans application, email communication with bank officials, and a change of address, as well as links to information on credit cards, interest rates, and investments.

Their current integration with Soho (Small-Office-Home-Office) accounting software is limited to Pastel accounting ([www.pastel.com](http://www.pastel.com)), but they do plan for easy integration with Microsoft's Money and Intuit's Quicken accounting packages.

#### **4.3.2 First National Bank (FNB)**

Van den Bergh, MD of FirstNet at First National Bank (FNB) (Marsden, 1997:30), states that FNB's attitude to the Internet is that it is far more than just a place for advertising and electronic brochures, but is rather a vehicle for the delivery of real value-added products. Apart from their internet-banking site FNB also has a number of online sites that provide a great deal of different functionality, such as WesBank (at [www.wesbank.co.za](http://www.wesbank.co.za)) for automotive financing, and their Online Investment Service (found at [www.fnbinvest.co.za](http://www.fnbinvest.co.za)). Currently FNB has about 20 000 Internet-banking clients and extrapolation of their present joining rate leads to a projection of 40 000 users by November 1999 (Gordon, 1999). Note the discrepancy between this figure when compared to Standard Bank's 40 000 users, and the 3% and 11% as found in the Web survey presented by Accone (1999).

Getz (The Star, 19 May 1997 and 22 September 1997) explains FNB's commitment to this new technology as a means for its planned revolution in customer care. These are: being conveniently available for 24 hours a day, recognizing individual differences in customers, and delivering more than just the online equivalent to traditional banking services. FNB's also offers (other) telephone-based banking services – as a response to complaints about the impersonal nature of Standard Bank's Toni it has a service called Direct Line which offers the same access method but has people on the other side of the telephone. FNB's Internet banking service is called First Online, and went online in early May 1997 – about a week before the Standard Bank site.



## 4.4 Conclusion

Lagoutte (1996:58) concludes that *'We are witnessing the beginnings of a fundamental shift in retail banking. Tomorrow, the bank will not be concentrating on achieving competitiveness across the board. Rather, fostered by deregulation and technological change, it will revolve around achieving excellence in only a small number of areas in which it genuinely believes it can achieve advantage. Retail banks today depend heavily on their branch networks and see it as a core of their business. In the future the branch network is but one of the several channels with which the bank accesses its stand-alone client base. Indeed, the branch network is in the future a privileged channel, used only for accessing that small percentage of the client base that offers the potential for a genuinely profitable relationship. The rest of humanity will deal with technology to transact its banking. Alternatively, it will use the telephone or a home TV based system as its access to retail banks.'*

Although it may be considered to be difficult to predict the future in the volatile Internet environment, the amount of activity in the area of Internet banking means that the situation is evolving, growing, and gaining acceptance amongst banking clients very rapidly. Banks are starting to understand that the Internet can take them beyond their own borders in order to expand their customer base. They should however combine this expansion of their client base with a geographical strategy, which by targeting specific countries will help them minimise risk and maximise revenue.

Although still in its infancy – Wells Fargo Bank (Intelligence, 1998) estimates that the composition of routine transaction delivery by the year 2000 will be 13% through PC banking/Internet banking, 35% through telephone banking, 30% through ATM's, and 22% through the bank's branches – Cronin (1997:vii) observes that in essence the debate about security and Web banking on the Internet is over – the Internet has emerged as *the* key area for the future of financial services.

## Chapter 5

# Data Collection

**This chapter describes the techniques used to capture a selection of local and overseas Internet-banking Web sites as required for their offline browsing and evaluation.**

## 5.1 Introduction

Perhaps one of the most notable characteristics of the Internet is its evanescence and volatility. This aspect proved to be problematic during a usability evaluation as no guarantee exists that the Web site examined today will be the same site available for study tomorrow. Because of their transient nature it was necessary to 'freeze' these Internet-banking Web sites by capturing them to local storage in a browsable and interaction-enabled form. The latter aspect proved to be difficult as most of the banks make use of dynamically generated Web pages – these are HTML (hypertext markup language) pages that are constructed on the server-side depending on the nature of the client's browser query.

The capturing was done using a collection of tools and techniques of varying sophistication. The simple techniques included direct copying of a generated HTML source document from the browser window and/or direct copying from the browser's local disk cache. Use was also made of automated tools such as the browser's subscription engine and some of the large number of offline browsing or site-capturing tools that are freely available. These automated tools are not really suitable for capturing dynamically generated Web pages but were used to download some of the demonstration sites based on static Web-page storage.

This chapter describes some of the techniques used to capture the banking Web sites to CDR (Compact Disk Recordable), and includes an analysis of the page architectures used for these sites (such as the use of frames). The latter aspect tends to have an insidious impact on usability and will be examined in more detail in later chapters. Additionally, a number of the more interesting help-page designs and their construction are described in this chapter.

This chapter does not include a description of the techniques used to capture usability evaluation data – data collection methods for the evaluations are described within their respective sections later in this study (chapters 6, 7, 8, and 9).

## 5.2 Capturing Internet banking Web sites

Most of the banks studied tend to use absolute (link) references extensively in their HTML pages. This makes the construction of a portable – i.e. movable to another directory – offline Web site much more difficult. In addition, WebDownloader (a Web-site capturing utility), replaces these absolute URL references in the HTML source code with absolute file references, as shown in the example below. This behaviour necessitated a tedious amount of manual conversion of the absolute references to relative references.

### Absolute references generated by WebDownloader:

```
<a href="file:///c:/INetBank/www.bankone.com/mapfiles/bl_persheadmap.map">  
    
</a>
```

### Relative references

```
<a href="../mapfiles/bl_persheadmap.map">  
    
</a>
```

### 5.2.1 Amalgamated Banks of South Africa

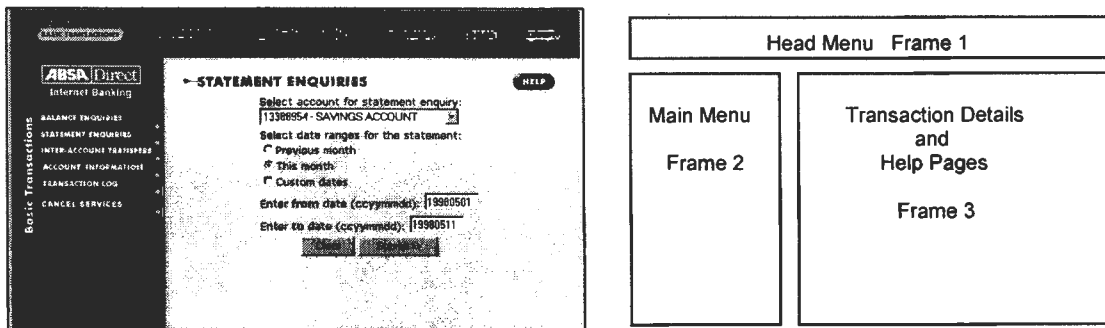
Because ABSA uses dynamic scripting for both their demonstration (and actual or live) Internet banking sessions, it was neither possible to capture the demonstration session in the browser's cache, nor was it possible to capture it through custom caching software for later offline browsing. ABSA's older home-banking facility, which requires special security software, does not have a demonstration function and could be captured conveniently from the browser's cache up to its login screen. A mirror image of a part of their Web site was constructed as described below.

A combination of Microsoft's Internet Explorer 4's subscription facility and the freeware offline cache builder WebDownloader ([www.saransk.sitek.net/pages/army/](http://www.saransk.sitek.net/pages/army/)), was used to first construct a two-level Web site image (i.e. following all on-site links up to 2 levels away), of the main ABSA banking site ([www.absa.co.za](http://www.absa.co.za)). Because the Internet bank and demonstration site ([www.absadirect.co.za](http://www.absadirect.co.za)), are secure (and separate) server sites, and also because of their extensive use of dynamically generated HTML pages – note the .asp extension in the listings, which indicates the use of Microsoft's active server pages – further links into the demonstration session had to be (laboriously) constructed by capturing all of the more important individually active server generated pages, and then hand-modifying part of the original HTML source code.

ABSA uses a three-framed structure for its Internet-based banking session as shown in figure 5.1 below. It is interesting that their old banking site used a no-frames page structure (at least) up to its login page – as shown in figure 5.2 – and that ABSA when converting to their new (direct) banking

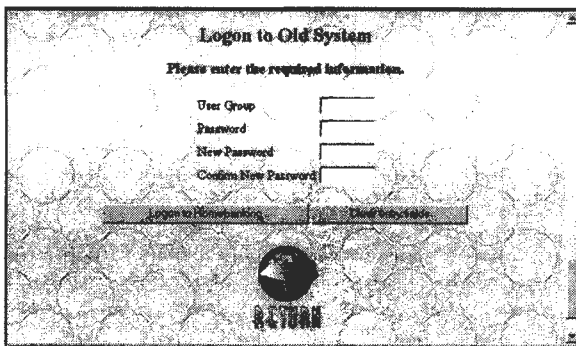
site was the only bank to completely revamp their site's structure and appearance – Nedbank, Standard Bank, and First National Bank's new sites all display almost identical when compared to their older (security software required) sites.

**Figure 5.1** ABSA's new site demonstration session using a frame-based page structure



Source: [www.absadirect.co.za/ib](http://www.absadirect.co.za/ib)

**Figure 5.2** ABSA's old site login screen using a no-frames page



Source: [www.absa.co.za](http://www.absa.co.za)

As an example, listings 5.1.1 and 5.1.2 show part of the HTML source code for the main-menu (frame 2) as used by ABSA, and after modification for offline browsing. (The modified sections are shown in bold.) This was necessary because the code for frame 3 (the transaction frame) is generated dynamically, had to be captured manually during each transaction, and was then saved as individual files (identified here as demonstration2a.html through to demonstration7a.html). If a transaction had more than one page (as for inter-account transfers) then up to three of these pages were captured (such as for demonstration6a.html to demonstration6c.html). A similar exercise was necessary for the context-sensitive help pages – notice the help button in the top right-hand corner of frame 3 in figure 5.1 – which are also dynamically generated during the interactive session. For example the code section:

```
<a href =
"help.asp?FromTran=Y&Filename=demonstration_Balance.asp&Page=Request&Section=Transaction&PageNumber=Balance#Request" onMouseOver="window.status='Help';
return true">
```

which generates the help page for the Accounts-Balance query, had to be modified to load the session-captured help page (and then saved as demonstrationBalanceHelp.html) as follows:

```
<a href="demonstrationBalanceHelp.html" onMouseOver="window.status='Help';
return true">
```

Another class of source code modification was required because of the use of absolute (rather than relative) URL references in ABSA's HTML source code – these usually consisted of a full pathname from the server root (such as '/ib/images/help.gif' which had to be converted to either './ib/images/help.gif' or 'images/help.gif', as appropriate).

### Listing 5.1.1 Original source code listing for part of the main menu frame in figure 5.1

```
<td valign=top>
  <a href="demonstration_Balance.asp?PageNumber=Request" target="trandetail">
    </a><br>
  <a href="demonstration_statement.asp?PageNumber=Request" target="trandetail">
    </a><br>
  <a href="demonstration_Transfer.asp?PageNumber=Request" target="trandetail">
    </a><br>
  <a href="demonstration_AccInfo.asp?PageNumber=Request" target="trandetail">
    </a><br>
  <a href="demonstration_TranLog.asp?PageNumber=Request" target="trandetail">
    </a>
  <a href="demonstration_absa_cancel.asp?PageNumber=Request" target="trandetail">
    </a>
</td>
```

Source: www.absadirect.co.za/ib/demonstration\_BasicTranMenu.html.

### Listing 5.1.2 Modified source code listing for part of the main menu frame in figure 5.1

```
<td valign=top>
  <a href="demonstration2a.html" target="trandetail">
    </a><br>
  <a href="demonstration3a.html" target="trandetail">
    </a><br>
  <a href="demonstration4a.html" target="trandetail">
    </a><br>
  <a href="demonstration5a.html" target="trandetail">
    </a><br>
  <a href="demonstration6a.html" target="trandetail">
    </a>
  <a href="demonstration7a.html" target="trandetail">
    </a>
</td>
```

## 5.2.2 Bank of America (USA)

Similar to ABSA, the Bank of America (BOFA) also uses scripting to dynamically create Web pages used in their demonstration session. This had the same consequence as far as caching and offline browsing are concerned, and also required the same type of modification to the HTML source code. Listings 5.2.1 and 5.2.2 show typical changes that were made. Similarly changes were made to the

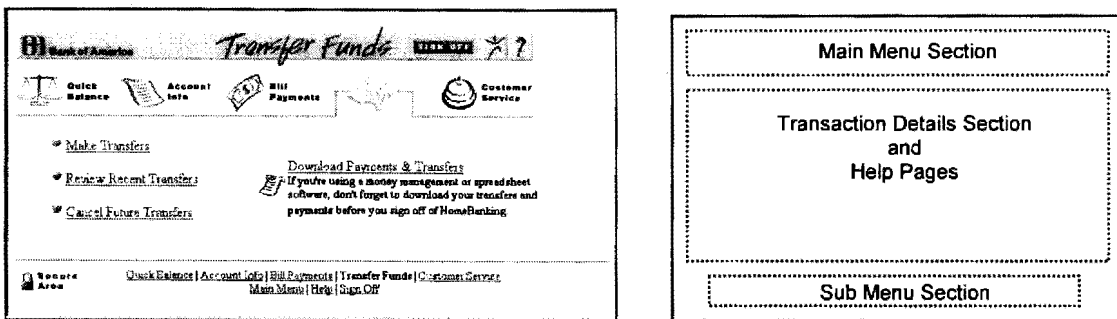
(context-sensitive) help system as shown below where the unmodified (captured) help-generating script is shown, followed by the changed code to call the captured static help page:

```
<A HREF="javascript:OpenHelp
('../ihb-bin/ihbcgi?popup_help&HelpRef=./html/help/qbhelp1.html')">
```

```
<A HREF="javascript:OpenHelp
('../html/help/gbhelp1.html')">
```

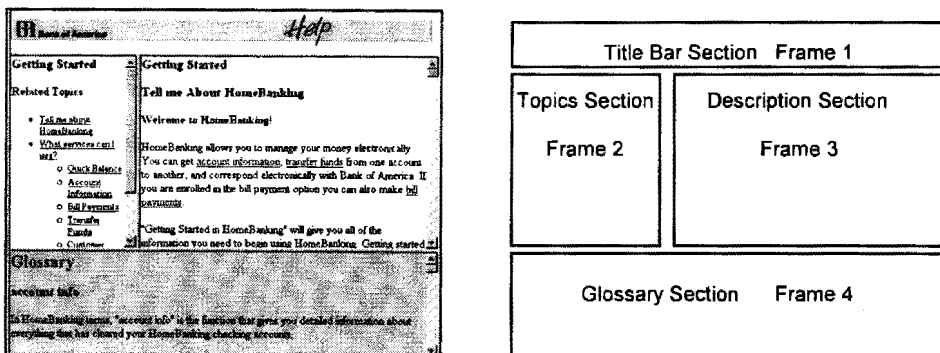
BOFA's online help pages are interesting because they open as a window separate from the main browser window. In addition they make use of a four-frame page structure – shown in figure 5.4 below – which is in contrast to their main demonstration page, which uses a very simple single page (no-frame) layout as shown in figure 5.3. For this the menu section is at the top and this code for this section had to be modified. These modifications are shown in part as listings 5.2.1 and 5.2.2.

Figure 5.3 BOFA's demonstration session single page structure



Source: www.bofa.com

Figure 5.4 BOFA's help page four-framed page structure



Source: www.bofa.com

**Listing 5.2.1** Original source code listing for part of the main menu frame in figure 5.4

```

<TD>
  <A HREF="../ihb-bin/ihbcgi?balance_inquiry">
    <IMG BORDER=0 SRC="../html/gifs/quick1.gif" WIDTH=118 HEIGHT=52
    ALT="[Quick Balance]"></A>
  <A HREF="../ihb-bin/ihbcgi?balances_home">
    <IMG BORDER=0 SRC="../html/gifs/acct2.gif" WIDTH=118 HEIGHT=52
    ALT="[Account Info]"></A>
  <A HREF="../ihb-bin/ihbcgi?bills_home">
    <IMG BORDER=0 SRC="../html/gifs/billpay1.gif" WIDTH=118 HEIGHT=52
    ALT="[Bill Payments]"></A>
  <A HREF="../ihb-bin/ihbcgi?transfers_home">
    <IMG BORDER=0 SRC="../html/gifs/transfer1.gif" WIDTH=118 HEIGHT=52
    ALT="[Transfer Funds]"></A>
  <A HREF="../ihb-bin/ihbcgi?service_home">
    <IMG BORDER=0 SRC="../html/gifs/cust1.gif" WIDTH=118 HEIGHT=52
    ALT="[Customer Service]"></A>
</TD>
Source: www.bofa.com

```

**Listing 5.2.2** Modified source code listing for part of the main menu frame in figure 5.4

```

<TD>
  <A HREF="../ihb-bin/ihbcgi1.html">
    <IMG BORDER=0 SELECTED SRC="../html/gifs/quick1.gif" WIDTH=118 HEIGHT=52
    ALT="[Quick Balance]"></A>
  <A HREF="../ihb-bin/ihbcgi2.html">
    <IMG BORDER=0 SRC="../html/gifs/acct1.gif" WIDTH=118 HEIGHT=52
    ALT="[Account Info]"></A>
  <A HREF="../ihb-bin/ihbcgi3.html">
    <IMG BORDER=0 SRC="../html/gifs/billpay1.gif" WIDTH=118 HEIGHT=52
    ALT="[Bill Payments]"></A>
  <A HREF="../ihb-bin/ihbcgi4.html">
    <IMG BORDER=0 SRC="../html/gifs/transfer1.gif" WIDTH=118 HEIGHT=52
    ALT="[Transfer Funds]"></A>
  <A HREF="../ihb-bin/ihbcgi5.html">
    <IMG BORDER=0 SRC="../html/gifs/cust1.gif" WIDTH=118 HEIGHT=52
    ALT="[Customer Service]"></A>
</TD>

```

### 5.2.3 Nedbank and Netbank (South Africa)

Nedbank's demonstration facility is cacheable in its entirety. The reason for this is probably because their demonstration source code had been pre-modified from their actual banking source code – i.e. Nedbank in effect already prepared their code similar to the way the ABSA and BOFA Web pages had to be prepared here (as described previously) for their offline Web sites.

A section of the menu frame source code (frame 1) reveals this similarity with the modified code from the ABSA and BOFA listings – notice how individual (separate) files are called from the menu options rather than being script-generated HTML:

```

<tr>
  <td> <a href="nmain.htm" target="NedMain"></a>
</td>

```

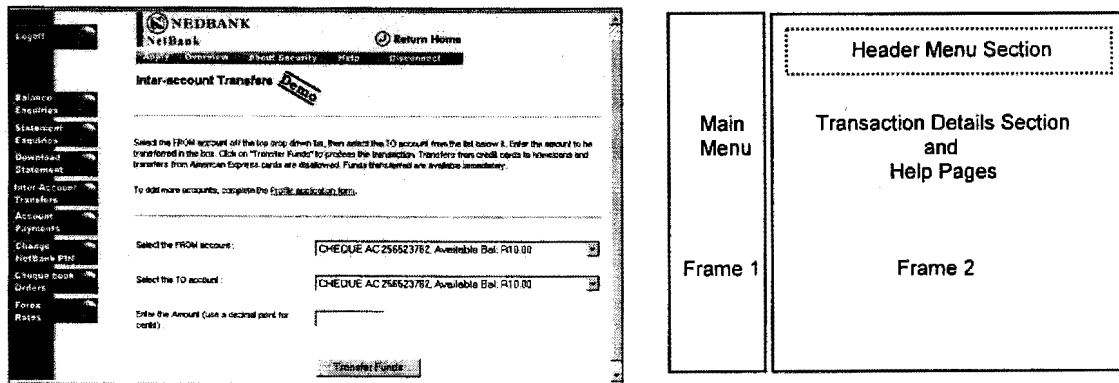
```

</tr>
<tr>
  <td> <a href="nstmt.htm" target="NedMain"></a>
  </td>
</tr>
<tr>
  <td> <a href="StatementDownload.htm" target="NedMain"></a>
  </td>
</tr>

```

Their demonstration session uses a two-frames structure as shown in figure 5.5, with the left hand frame acting as a static (non-scrollable) menu frame, and the right hand frame acting as a container for the transaction detail and help page display.

**Figure 5.5 Nedbank's demonstration session two-frame page structure**



Source: www.netbank.co.za

## 5.2.4 First National Bank (South Africa)

First National Bank's (FNB) demonstration session required only a moderate degree of modification to make it suitable for offline browsing. The link to the demonstration facility is a static link (to frame03.html as shown below) and required no modification. A section of this menu frame (frame 2 in figure 5.6) is shown in listing 5.3 – note the secure protocol (https) link to the actual banking site in the same listing.

Their demonstration session uses a three-frame structure similar to that of ABSA and is presented in figure 5.6. The top frame is used for the account selection rather than an additional menu. No online help is available during their transaction sessions – help is only available from the main pre-login page as shown in figure 5.7, which also uses a three-framed page layout.

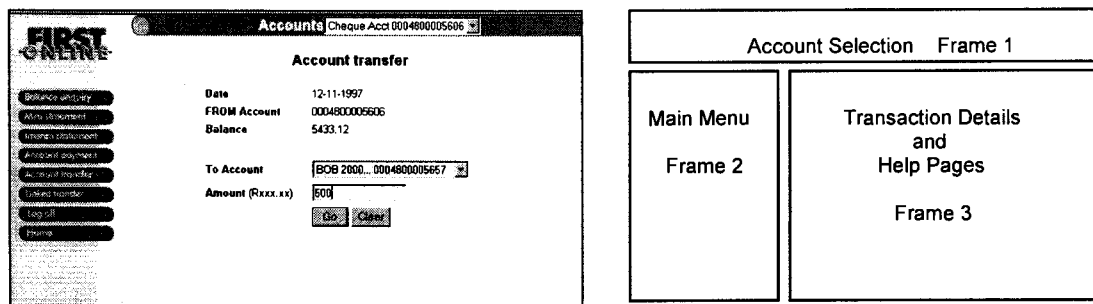


Listing 5.3 Menu frame source code for figure 5.6

```

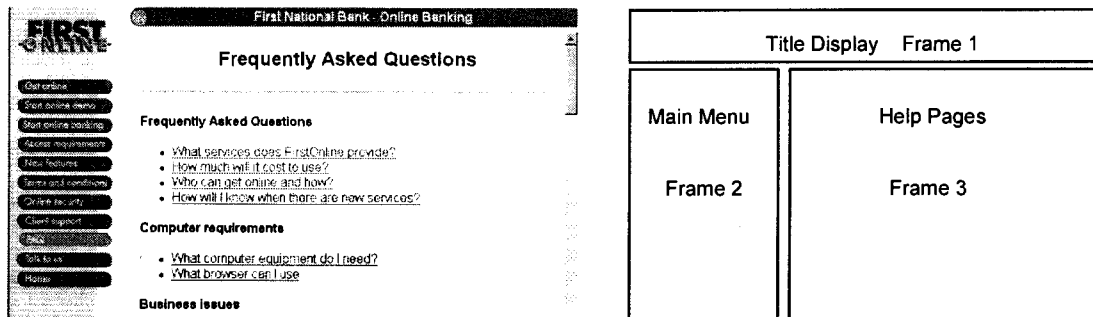
<p align="center">
  <a href="frame02.htm" target="_top">
  </a><br>
  <a href="frame03.htm" target="_top">
  </a><br>
  <a href="https://www.firstonline.co.za" target="_top">
  </a><br>
  
```

Figure 5.6 FNB's demonstration session using a three-frame-based page structure



Source: www.firstonline.co.za

Figure 5.7 FNB's demonstration session help using the same three-framed page



Source: www.firstonline.co.za

### 5.2.5 Standard Bank (South Africa)

Standard Bank effectively has three Internet banking sites in operation – the demonstration site which uses static HTML pages (and which can be found at [www.sbic.co.za/banking/demo/](http://www.sbic.co.za/banking/demo/)), the older site which required special security software (at [www.secure.standardbank.co.za](http://www.secure.standardbank.co.za)), and the new (direct browser), site (at <https://www.encrypt.standardbank.co.za>). For the reasons as discussed below only the demonstration site could be locally mirrored.

Similar to FNB, Standard Bank's demonstration session also required only moderate modification to make it suitable for offline browsing, and because of reasons similar to those mentioned for Nedbank: script-generated HTML pages in the actual banking session had been replaced with links to static pages, which are easily cacheable.

An attempt was made to construct an offline Web site mirror for the actual banking session, but this would have required the same amount of extensive page modification and individual script-output capture that were necessary for the ABSA and BOFA offline sites. As the (1998) demonstration session proved itself to be very faithful to the actual session, such an exercise was deemed to be unnecessary. In addition it was thought necessary for a more complete reflection of interactivity and usability, to evaluate the actual banking site online at all times, even though this could be an expensive exercise because of connection charges. This online actual banking evaluation was done for both the older site and the new site, as access to the old site was gained through the use of Windows 3.1, 16bit browsers (Internet Explorer 3.01 and Netscape Navigator 3) and Standard Bank's security software, and access to the new site through the use of 40bit encryption capable 32bit browsers (Netscape 4.03, and Internet Explorer 4.01), operating under Windows 95 and (later) Windows 98 and Windows NT.

Standard Bank still maintains its older Internet banking site – which requires additional security software – but for the reason mentioned above – the demonstration session being a very close simulation of the (older) site – the old site during actual use was not captured for offline browsing.

The difference between the demonstration and actual version is illustrated by listings 5.4.1 and 5.4.2, which shows the section of code associated with the login button for the two sessions. (The login codes for the actual session had been changed in listing 5.4.1.) Note the link reference in the demonstration session to a static page and the script parameters used for the actual session.

#### **Listing 5.4.1 Login button source code for the actual banking login session**

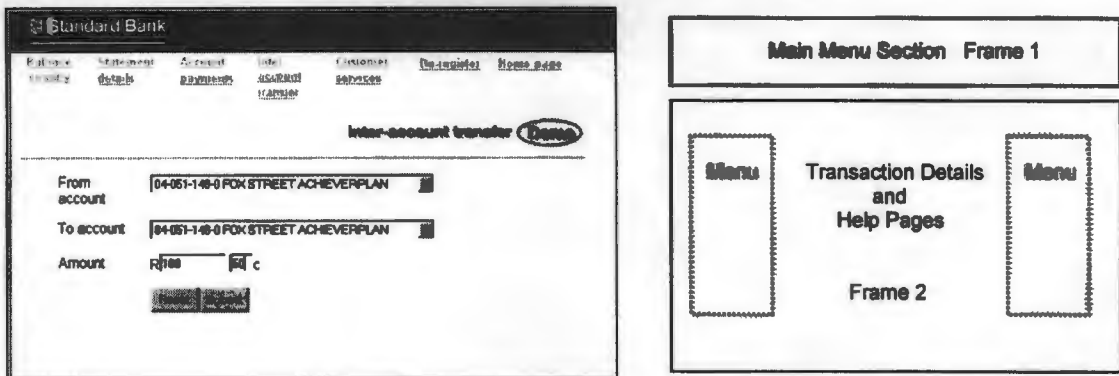
```
<A HREF="https://
www.encrypt.standardbank.co.za:443/cgi-bin/vpecgi/ibank.a042730397020c23be"
TARGET="_self">
<IMG SRC="/vpe/signon.gif"
ALT="Sign on" WIDTH=90 HEIGHT=24 BORDER=0 ALIGN=bottom naturalsizeflag="3">
</A>
```

#### **Listing 5.4.2 Login button source code for the demonstration banking login session**

```
<A HREF="signdemonstration.htm"
TARGET="_self">
<IMG SRC="signon.gif"
ALT="Sign on" WIDTH=90 HEIGHT=24 BORDER=0 ALIGN=bottom naturalsizeflag="3">
</A>
```

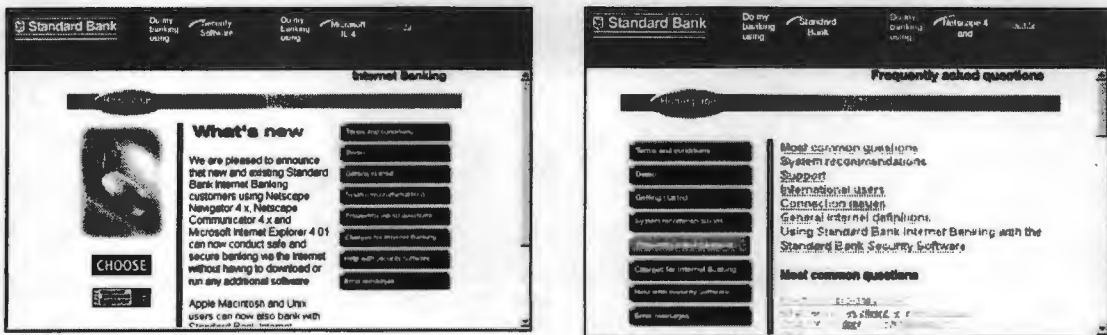
Their demonstration session uses a two-framed structure similar to that of Nedbank, but with a top horizontal menu frame. It is shown in figure 5.8 below with the top frame acting as a static (non-scrollable) menu frame, and the bottom frame acting as a container for the transaction detail and help page displays. The two additional menu regions shown in figure 5.8 are used for choice selection during the help page display (no help is available during the actual transaction), and to display additional choices in the pre-login screen. Both these additional menus are shown in figure 5.9.

**Figure 5.8 Standard Bank’s demonstration session two-frame page structure**



Source: www.sbic.co.za

**Figure 5.9 Standard Bank’s demonstration using two additional menu’s placed in frame 2**



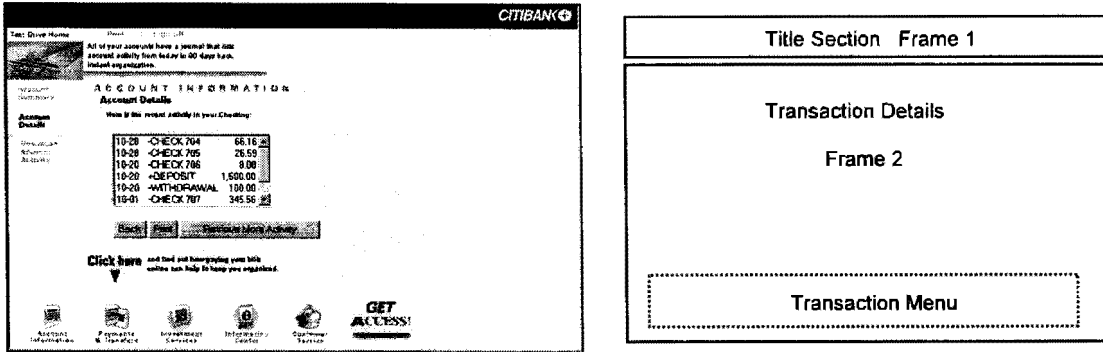
Source: www.sbic.co.za

### 5.2.6 Citibank (USA)

Citibank’s Internet-banking homepage uses a separate static demonstration facility which was cached with WebDownloader, and which required only minimal modification for offline browsing. In contrast Citibank’s main homepage makes extensive use of Java applets to display a large central advertising banner and stock exchange information in a separate bottom frame. This made caching of the page difficult without removing the Java-generated images and replacing them with static images.

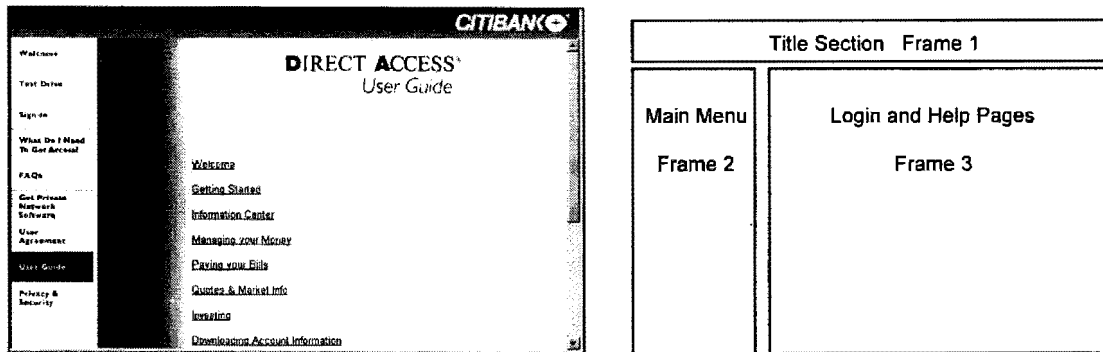
No help is available during the transaction session, which uses a two-framed layout as shown in figure 5.10. The separate three-framed help session is shown in figure 5.11.

**Figure 5.10 Citibank’s transaction demonstration session using a two-frame page**



Source: www.citibank.com

**Figure 5.11 Citibank’s help session using a three-frame page structure**



Source: www.citibank.com

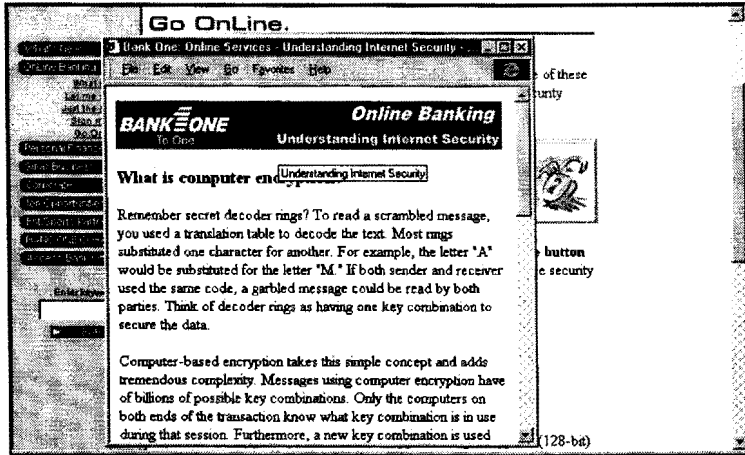
### 5.2.7 BankOne (USA)

BankOne (BancOne) makes use of both a separate (JavaScript scripted) window (shown in figure 5.12), and a separate frame (figure 5.13) to display its online help. The demonstration (seemingly) lacks some of the functionality that the actual session would have – the help button in figure 5.13 is non-operational during the demonstration. It is likely that the actual (online) session help would be displayed in the form of a separate window as in figure 5.12, and that the separate help frame in figure 5.13 is only present during the demonstration session.

The BankOne offline Web site was constructed using a combination of WebDownloader and Internet Explorer’s cache-files, with some modification because of BankOne’s frequent use of absolute URL references in their HTML source code documents. WebDownloader modified the

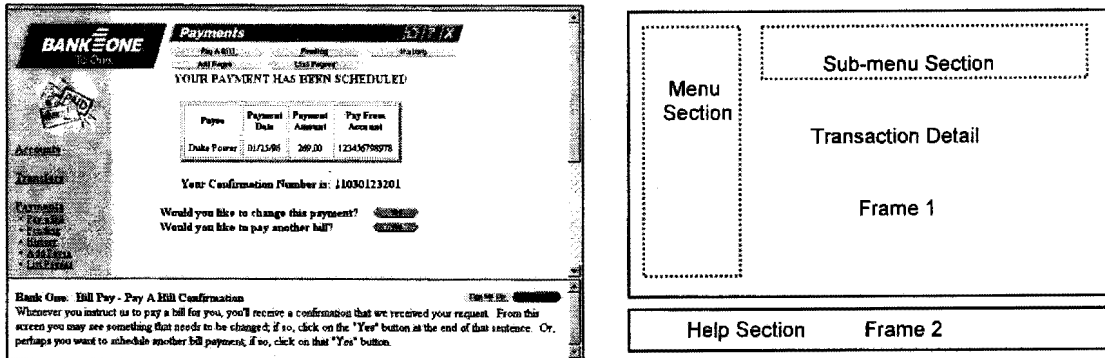
latter to file:// absolute references, and these were then manually replaced by relative references, to make the site portable.

Figure 5.12 Bank One's help session using a second, separate window



Source: www.bankone.com

Figure 5.13 Bank One's demonstration session using a two-frame page structure



Source: www.bankone.com

### 5.2.8 Other Banks

Two other bank Web sites were cached with no or minor modification using a combination of WebDownloader and manually copying Internet Explorer's cache files to a new location. They are:

**Mercantile Lisbon:** This is essentially a pure home banking site – access is only possible through special non-browser-based software. The cached pages are as a result mainly information pages. The special software is large download (3.8 MB) and includes a demonstration facility that was not extensively examined as it is not Web browser-based.

**Royal Bank of Scotland:** The demonstration site was easily cached because of the simple and static nature of the demonstration facilities offered – it is in essence a screen slide show.

### **5.3 Conclusion**

Storing the Internet-banking Web sites offline to local storage proved to be an absolute requirement for conducting the usability evaluation exercises. These evaluations would have been compromised and inaccurate if moving target source documents had been an additional factor. It should be noted that the local capture of these sites were done towards the middle part of 1998 – it is likely (certain!) that all of these sites have undergone major revisions since then which are not reflected in this research project. This aspect is not a consideration for the main banking site evaluated (Standard Bank), as the evaluation for this bank was conducted live (online) as well as offline, and the conclusions made based on its Web site reflect all changes to the site up to the early part of 1999.

## Chapter 6

# Task Analysis

This chapter introduces task analysis concepts and presents an application of one specific approach (Task Analysis for Knowledge Description or TAKD) to an Internet-based banking task.

## 6.1 Introduction

This chapter examines task analysis concepts and approaches. It then applies one specific approach – TAKD (Task analysis for knowledge description) – to an Internet banking task where the user logs into the banking facility and displays a balance for a selected account. The advantages associated with the use of such a structured approach to the analysis of the banking task in uncovering usability problems are then examined and suggestions for improvement offered.

## 6.2 Task analysis

A definition for task analysis is offered followed by an examination of the meaning of related concepts such as a job, task and an activity. A simple computer operating model may serve as an effective basis for an understanding of the goal-directed nature of the process of computer-based task execution. This is followed by a brief discussion on three major classes of task analysis approaches.

### 6.2.1 Definition, benefits and application

**Task analysis** may be defined as *the study of what a user is required to do in terms of actions and/or cognitive processes to achieve a task*. A detailed task analysis can be conducted to understand the current system and the information flows within it. These flows are important to the maintenance of the existing system and should be incorporated in any new or replacement, system (Maguire, 1997). Kirwan and Ainsworth (1992:1) present a similar definition as the previous but they substitute 'to achieve a task' with '*to achieve a system goal*'. Newman and Lamming (1995:20) also emphasize the goal-directed nature of tasks when they state succinctly that '*a task is a unit of human goal-directed activity*'.

Task analysis yields the following benefits: (1) It provides knowledge of the tasks that the user wishes to perform. (2) It is a reference against which the value of the system functions and features can be tested. (3) It is a cost-saving exercise because failure to allocate sufficient resources to the

task analysis activity increases the potential for costly problems arising in later phases of development. (4) Task analysis makes it possible to design and allocate tasks appropriately and efficiently within the new system. (5) The functions to be included within the system and the user interface can be more accurately specified (Maguire, 1997).

Earlier, Johnson, Diaper and Long (1984:500) gave a motivation for the use of task analysis: '*The evolution of computer applications and the need for good human-computer interaction has created a demand for knowledge-based descriptions of human task performance*'. Further support is given by Diaper (1997:41), when he argues that one of the areas of tension between Human-computer interaction (HCI) and Software Engineering has always been the assumption made in the latter that obtaining requirements analysis data is not a problem – but that such data should include what is done, by whom, and for what purpose – and that these questions can be answered by task analysis. Diaper and Addison (1992:124) also offer a strong commentary on the need for the incorporation of task analysis during the early stages of software development.

Task analysis has as its only major limitation that a formal task analysis can be time consuming and produce a large volume of data which may require considerable effort (and skill) to analyze. Johnson (1989:119) is more specific in his examination of other problems associated with the use of task analysis during phased design practice.

Dix et al. (1998:104) expand on Maguire's (1997) definition – they describe task analysis as '*the identification and description of the interactive system user's problem space, in terms of domain, goals, intentions, and tasks*'. They continue: It is the study of the way people perform tasks with existing systems, i.e. analyzing the way people perform their jobs. Apart from those tasks that directly involve a computer, its scope can also include modeling aspects of the world that are not part of the computer system. Task analysis can be used during the production of manuals, high-level system design, requirements capture, and detailed interface design. The latter is mainly aimed at the analysis of an existing system in order to suggest the design of a new improved interface design – such as menus and system dialog design (Dix et al., 1998: 260-263, 287).

It is interesting to note that during most discussions of task analysis, it is seldom that the research and practice from the perhaps more established human resource disciplines are referred to – and more specifically **job analysis** and **organizational design**. The exception to this is Johnson et al. (1985:500). They observe that job analysis has as its major deliverable a means of selecting people for a specific task. In principle there is little conceptual difference between task and job analysis – the latter being defined as '*developing a detailed description of the tasks involved in a job, determining job interrelationships, and ascertaining the knowledge, skills, and abilities necessary for an employee to successfully perform the job*' (Robbins, 1991:532). Even though the purpose and the scope of application between the two may differ, some of the modern job analysis techniques



(and even some of the established methods such as functional job analysis (FJA)), which includes McCormick's Position Analysis Questionnaire (PAQ) (Gibson, Ivancevich, and Donnelly, 1991:474), which emphasizes information processing and information sources, are closely aligned to the needs of the human-computer interaction practitioner. Even during their (well) argued dismissal of job analysis Johnson et al. (1985:500) do admit that it '*focuses on the job components (tasks), and requisite skills by classifying the basic or generic skills of a particular job*'.

Traditionally, job analysis deals with the three aspects of **job content** (addressed through FJA methods), **job context**, and **job requirements** (where PAQ is widely used). All three of these have direct application to all of the aspects of task analysis as discussed here, and better use should be made of these proven techniques.

### 6.2.2 Jobs, roles, projects, tasks, activities and actions

Diaper (1989:217) argues that a classic problem in task analysis has been that of an adequate definition of a **task**. He uses a four-level hierarchy of projects, tasks, subtasks and activities to alert the practitioner to the differences between these concepts, and emphasizes that the transition (and therefore distinction), between these are often vague. For the purpose of this analysis the (perhaps inexact) term 'subtask' will not be used and is replaced by the term **activity**. An activity has components ('subactivities'), and these micro-level activities will be referred to as **actions**. Diaper also examines the use of the concept 'job' in some length, and this will be included as the highest level of a five-level hierarchy.

As noted the difference between these levels are not exact, and Diaper suggests that the main distinguishing characteristic will be related to the *duration* of each of the levels. For example a major and lengthy task may be better viewed as being a '**project**'. The term 'process' or more correctly 'business process', is also commonly used in this context – Newman and Lamming (1995:20) regard a '*process as being a linked set of tasks*'. The notion of a **job** is likely to be the exception in this classification method as this could be described as being a person-oriented concept with a title and contractual obligations. A job can also be defined through the set of tasks assigned to the jobholder, and it is also common to find the term '**role**' being used at an intermediate level (between job and task).

Classically the relationship between a job and its constituent tasks is quantified by the use of the phrase '**job range**', and qualified through the use of the phrase '**job depth**' (Gibson et al., 1991:478).

Johnson (1989:121) defines a task as: '*An action or combination of actions that when carried out achieve some goal which results in a change of state in a given domain*'. Diaper's (1989:216) nearly equivalent reinterpretation of this as '*a task is a set of activities to be carried out which will change*

*the situation from an initial state to some specified goal state*' is preferred as it corresponds to the view of activities as will be used here.

An unusual definition of a task is given by Diaper and Addison (1991:388) based on Nielsen's (1990) classical paper on agent modeling. Diaper et al. (1991) note that an (albeit imperfect due to taxonomy and notational restrictions), approximation of a task may be given by the relation:

$$T = W + ( U + ( C + M ) )$$

T refers to the task, W the world task environment and application domain, U is the user agent, C the computer agent, and M represents documentation such as manuals. The + is a combination operator – so that the other expression U(C+M), refers to the user's model of the combination of a computer (system) and documentation, W(U+(C+M) refers to the task domain (world) model of the user interacting with a computer and documentation, and the task  $T = W+(U+(C+M))$  is then the combination of both these models.

This view of a task is useful because it: (a) Emphasizes the complex environment in which a task is executed. (b) Enumerates all the important agents impacting on such a task. (c) Correctly describes a task as being dependent on the user's mental model of the computer and documentation. (d) Argues that the actual task is a part of the larger world model filtered by the user's model of the latter.

It is useful to note that in practice (such as during observation) tasks are perhaps best distinguished from each other through an observation of a **goal redirection** and **goal separation** (Newman et al., 1995:20). These goal changes and achievement are reflected through system state changes, and Diaper et al. (1991:394), make use of this: '*a task is defined in terms of the changes of state that an interactive work-system causes in the (task) domain*'. Sheperd (1989:21) discusses the relationship between a task and its goal: '*A goal, which has to be attained by a person, has an associated task, which offers facilities and constraints on how the goal may be attained*'. He prefers the use of the term **operation** to the terms action and activities as used here.

### 6.2.3 A model of interaction for task analysis

An often-encountered, and simple computer operation model is presented in figure 6.1 – this specific example being adapted from Woodson et al. (1992:255), which is in turn based on the set of USA Military User Interface Design Guidelines. This type of model is eminently suitable to be used as the basis for formulating a strategy for task analysis. It is noted that Norman's execution-evaluation model (as discussed by Dix et al., 1998:105-106), has stages effectively similar to this model.

Norman's model consists of an **execution cycle**:

*[goal-establishment]-[intention-formulation]-[action-sequence-specification]-[action-execution]*

and an evaluation cycle:

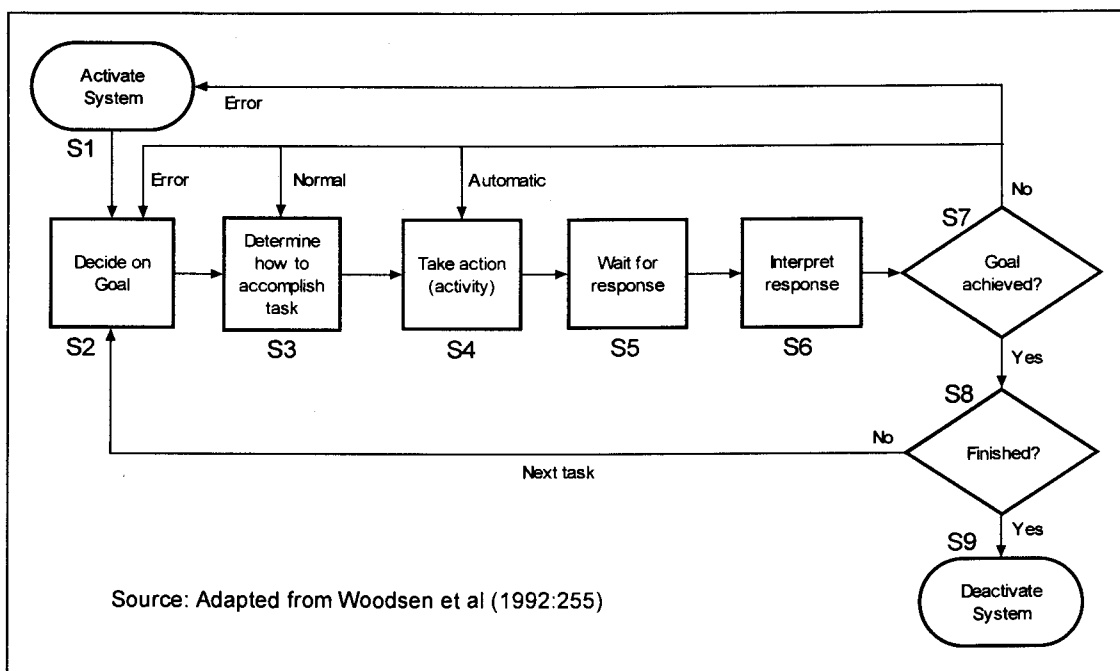
*[state-perception]-[state-interpretation]-[state-and-goals-comparison]*

Of particular interest is the reduction of Norman's seven stages of user activity into the three as presented by Diaper et al. (1991:389). These three are:

- (1) **Establishing goals.** For the task.
- (2) **Behaving.** This constitutes both physical behaviour and actions and also an expectation regarding the consequences of the physical action.
- (3) **Perceiving.** It includes an evaluation based on the expectation above, and a perception of the behaviour.

The correspondence of the above to both Norman's model and the simple computer operating model is immediately obvious – as a result it has an immediate and direct application to task analysis, of which task sequence construction, and Diaper's object/action pairs task 'generification' (discussed later), are but two examples.

**Figure 6.1 Task analysis based on a simple iterative computer operating model**



**Notes: Figure 6.1**

- (1) The S7-S4 feedback loop labeled 'automatic' corresponds to Johnson's (1989:121) notion of 'procedures' or well-practiced behaviours.
- (2) The error feedback loops S7-S2 and S7-S1 are included, as they are especially appropriate viewed within the unreliable communication context of Internet-based interaction. The 'normal' assumption of system deactivation only on completion of the task(s) is then often untrue.

- (3) The original (Woodsen's) term 'entry' was replaced by 'action' in S4 to correspond to the five-level hierarchy from job to action – a set of these (i.e. multiple loopbacks), would be an activity.
- (4) Johnson's (1989:121) definition of a goal: ' a purpose or objective to be achieved, it provides the purpose for which a task is undertaken', is appropriate within the context of this model.
- (5) The S9-S2 loop represents the execution of different tasks within single system activation – it corresponds to the previously discussed notion of a task-procedure or project.
- (6) S2 could be relabeled as 'Goal adjustment' if it is reached via an error feedback loop.

Woodsen's (1992:255-256), software design guidelines and descriptions for each of the nine phases (S1-S9), in the model, can also serve as a basis for an evaluation of the user interface. As an example phases 1, 2, and 3 use the following guidelines (table 6.1):

**Table 6.1 Guidelines for the phases of the simple computer operating model**

(1)	Activate system	1.1 Separate and distinct step. 1.2 Rapid as possible – if not possible advise the user of the status. 1.3 Automate whenever possible. 1.4 Do not require user knowledge of internal system mechanisms.
(2)	Decide on goal	2.1 Provide users with task-oriented guidance for every transaction. 2.2 Guidance should describe system capabilities and available procedures. 2.3 Provide guidance: separate manuals, or online within program (preferred).
(3)	Determine how to accomplish task	Design a dialog between the user and the computer that is compatible with the user and the task requirements – these dialogs include styles such as menu selection, command-line, and WIMP.

Source: Woodsen (1992:255).

## 6.2.4 Task analysis approaches

Weir and Alty (1991:148) comment on the cognitive aspects found in task analysis, and present a three category classification of task analysis approaches which includes the popular task decomposition methods as examined extensively by Dix et al. (1998). To this they add two cognitive approaches specifically aimed at the evaluation of usability properties. Sutcliffe (1988:49-50) notes that task analysis in HCI is not well defined, but due to its close relationship with systems analysis it is appropriate to use methodologies borrowed from the latter, such as De Marco's functional decomposition and structured analysis methods. Baecker et al. (1987:487-488) agree with this view, and they identify task analysis to be the first phase of a proposed user interface design methodology, which consists of a study of user needs and defining a set of functional requirements.

Task analysis differs from goal-oriented cognitive methodologies in that the user is observed from the outside while 'in action' (i.e. it is an observable behavioural study), rather than having its major focus on the user's internal mental state (Dix et al., 1998:262).

Task decomposition, knowledge based techniques, and entity-relationship based analysis, are three different but overlapping approaches to task analysis (Dix et al., 1998:261), and are presented in table 6.2.

**Table 6.2 Three approaches to task analysis**

- (1) **Task Decomposition:** This consists of defining the overall task i.t.o. subtasks and their sequence. Hierarchical task analysis (HTA), is such an approach and produces a hierarchy of tasks, subtasks and plans, or a description of task conditions (when a subtask is performed), and a task sequence or order of execution (Dix et al., 1998:262).
- (2) **Knowledge-based techniques:** This builds a conceptual model of the way the user views the system and the task, and what a user needs to know about task objects and actions. One technique (TAKD or task analysis for knowledge description) uses an either/or (AND, OR and XOR) branch based taxonomy (TDH or task descriptive hierarchy), of all objects and actions in the task, which differs from the HTA methods (based on 'how to'), in that it is based on task and object similarities (i.e. *genericity*) (Dix et al., 1998:268).
- (3) **Entity-relationship based analysis:** The emphasis is on identifying objects and actors, and their relationships and operations, rather than on object similarities. Objects are classified as being either actors (usually human entities), concrete objects (all the other 'things'), and composite objects (sets or combinations of the previous two groups). Attributes of objects and actors are listed only when it is relevant to a human or computer task. Actions are performed by an object/actor (the 'agent'), and change the state of an object/actor (the 'patient'). Allied to actions are messages (a double action), and events (all actions are events, but some events can happen spontaneously). Relationships can be *object<->object*, and *action->object* (the *object->action* relationship is implicit). This type of analysis combines well with the previous two taxonomy-based approaches, as its object-oriented nature fits in well with class-inheritance taxonomy (Dix et al., 1998:274).

### 6.3 Task analysis for knowledge description (TAKD)

TAKD is a method for task analysis that generates a hierarchical description of tasks – *the task descriptive hierarchy (TDH)* – and translates these into a set of *knowledge representation grammar (KRG)* sentences. Its original purpose was to analyze data from the observation of relevant tasks and then redescribe them using a single consistent representation of specified information technology knowledge that the user should possess (Diaper, 1989:108). Importantly, Johnson, Diaper and Long (1985:499), originally described TAKD as a '*method for producing descriptions of knowledge*'. It is therefore a *descriptive* rather than *normative* approach to activity modeling for task analysis (Newman et al., 1995:114). TAKD is also suitable for the evaluation of existing systems – provided the focus is on behaviours that can be measured and described rather than being broadly psychology-focused, and it can also be used to include a task analysis stage during the production of requirements specifications.

**Knowledge-based descriptions of tasks** are important because it identifies (1) the plan for carrying out the task, (2) the knowledge or concepts required, and (3) the interaction between different kinds of knowledge. The designer therefore gains understanding as to the commonalities that exist between tasks in terms of their knowledge requirements and their plans for execution (Johnson et al., 1985:500).

Importantly Diaper et al. (1992:127) emphasize that: '*TAKD places few constraints on analysts, but it does force a systematic, explicit way of thinking about tasks and their more general context*'. They continue: '*TAKD (expressly intends to be limiting in that it) forces an inherently user-centered perspective, because TAKD actions are always those of people, (and) not of machines*'. The focus of this analysis is then on the observable behaviours of the user rather than the computer, and the value of the analysis was proved to be realized as much from simply completing the exercise as it was from TAKD's formal outputs (such as the TDH diagrams and KRG sentences). This does not mean that TAKD ignores agents other than the user – indeed Diaper (1989:111) states that TAKD concentrates on the complete system, but it only includes these other aspects where they have an impact on the user behaviour.

A graphical representation of the stages of a TAKD analysis is presented by Diaper and Addison (1991:396). (Being a strongly iterative process and therefore somewhat convoluted it is not reproduced here.) This shows TAKD to have four major analysis cycles (table 6.3), with a controlled iteration between these (Diaper, 1991:396-397, 1997:43). The last sub-stage listed, making sequencing knowledge part of TAKD, is a fairly recent addition to the approach.

**Table 6.3 The four stages of a TAKD analysis cycle**

<p>(1) <b>Construction of a activity list (AL).</b></p> <p>This is a prose description of the set of tasks that are observed. It is also sometimes referred to as a task protocol. The input to this is normally a video recording of task execution and a post-task walkthrough although task-oriented interviews can also be used.</p> <p>(2) <b>Selection of Specific Objects (SO) and Specific Actions (SA).</b></p> <p>These are the lowest levels of objects and actions that will be subjected to analysis. They are directly selected from the AL where the SA are shown in square brackets [ ], and SO in round brackets ( ).</p> <p>(3) <b>Construction of the Task Descriptive Hierarchy (TDH).</b></p> <p>This represents each line from the AL by a single depth first traversal from a single leaf node (with at least one SO), to the single top node.</p> <p>(4) <b>(a) Specification of Specific and Generic Knowledge Representation Grammar.</b></p> <p>This is the main output from a TAKD analysis and describes a single traversal from the leaf node to the top node. A uniqueness rule applies to each sentence.</p> <p><b>(b) Specification of Task Sequence Representation.</b></p> <p>These are lists of repeated sequences from all levels of generification.</p>
---

## 6.4 Applying TAKD to an Internet-based banking task

The stages as described in table 6.3 form the basis of an application of TAKD to an actual Internet-based banking application. (Actual refers to 'real' or 'live' banking accounts and clients as opposed to using the Internet-based banking demonstration Web site.)

### 6.4.1 The problem area and task problem statement

The problem area is the execution of a simple banking task – obtaining a detailed statement for a (current) account – via the Internet. This task is preceded by other tasks: obtaining Internet access, navigating to the Bank's (Standard Bank South Africa) Internet banking Web site, and logging into the site. The task is concluded (the goal achieved), when the statement has been viewed, and the user logged off from the Bank's Web site. During this analysis the three tasks of logging in, obtaining the statement, and logging off will be analyzed as a single task with component activities (sub-tasks).

The **concise problem statement** is:

*'Display a statement for a current account using an Internet banking service'*

A more **detailed problem statement** reads as follows:

*'Login to Standard Bank's Internet Banking service using the account and password number, obtain a detailed statement for a current account, study this visually, and thereafter logoff from the site.'*

Again note that the precondition for the task execution is that the user has already obtained Internet access (either through a dialup account and logging into his/her ISP, or by using the company network), and navigated to the main Internet banking Web page – the sequence of these pre-task and task activities is detailed in the activity list (table 6.4), and the series of twelve screen dumps as presented in figure 6.3. After the completion of this part of the study, Standard Bank changed their home page (figure 6.3, screen 1) during January 1999 – the user can now directly choose an Internet banking option from this page, and therefore skip the second page (screen 2) figure 6.3. This change means that the server dialog box (screen 3) is still present, but shown at the top of screen 1 rather than screen 2.

### 6.4.2 Task observation and data collection

Diaper (1989:211-237) comprehensively examines various task observation and data collection methods that are suitable as inputs to the task analysis process. In the absence of suitable video recording equipment (aspects of these for laboratory-based usability evaluation studies were discussed by van Dyk, (1998)), a series of twelve screen dumps (figure 6.3) was used as visual basis for the analysis. These were accompanied by structured note taking (such as listing the physical objects used, and the screen elements accessed), during the task execution. There are disadvantages

to being both the observer and the observed, but it does ensure that the requirement of an observer that is completely familiar with the observed task is being reasonably met.

By implication, Diaper et al. (1991:390) argue that there are essentially only two useful methods for HCI data collection – through questioning and through observation. Regarding the latter they continue and note that: ‘(one) *should not to be too concerned with supposed mental states, but use observable behaviours to stand for mental states*’, as being the preferred approach to data collection. The previous quote is exemplary in explaining the *embedded attitude* in a (Diaper-executed) TAKD analysis – it always seems to be a firmly empirically based approach and implementation.

### 6.4.3 The task activity list and prose description

The task **prose description** (when presented in Diaper’s structured format this is referred to as the **activity list** or AL), is shown in table 6.4. This is a final version resulting from a number of refinements or passes. It should be noted that strictly speaking there are never ‘final’ versions of these TAKD analyses aids – i.e. activity lists, TDK diagrams, or KRG sentence sets – there are only suitably refined versions.

In earlier versions all detailed actions originated by the computer were included such as ‘(computer) [*accepts keystrokes*] from (keyboard)’. These were included based on the AL used by Diaper (1997:48), but were removed when it became clear that the AL has become unmanageably long and complicated, and stopping rules (Sheperd, 1989:22), were applied to these levels of activity. In addition browser-based computer actions were not included for the same reason – these could be ‘(computer) [*displays*] (arrows-for-navigation)’. In general all of the *visible* page-based, computer generated actions were included, even when these were not used in the specific task – an example of this being the screen prompts for clearing entries and changing of a password displayed on the main login page. Because this is an observational method only the actual (observed or recorded), human actions are listed. The same exclusion rule was applied to the computer agent – processing actions are not listed in the activity list (as mentioned above). Even with all these exclusions the resulting list as presented in table 6.4 is still surprisingly extensive for such an apparently simple task – it consists of thirty-seven separate actions on thirty different objects.

The activity list follows the TAKD convention of showing only one action on each line, but it differs in its format from that used by Diaper (1997). A more readable natural language sentence is followed by the compact specific object and activity sentence. In this latter sentence a degree of object hierarchy preordering is included, for example all page objects start with ‘page’ followed by the name of the specific page such as ‘transactions’, i.e. (page-transaction), or (page-accountstatement). Similarly buttons are identified as (button-submit) or (button-dialogbox-ok) – the latter could be perhaps better described as (dialogbox-button-ok), but as discussed later, dialog boxes will be treated the same as (Web) pages for the purpose of this task analysis. Where there are more



than one object instance, which only differs in its location (such as the submit buttons found on the login and account selection pages, these are numbered as submit1 and submit2. A similar numbering convention was used for repeated actions such as [display], [prompt], [press], [select], and [enter]. Doing it this way eased the construction of the TDH diagram and KRG sentences considerably.

All activities and actions executed by the computer for a specific page were first listed in their entirety, before the human actions were enumerated. For example the computer prompts for an account number, a password number, a password change flag, and a clear and submit option *before* the user takes any action. The conventions used in listing the sequences of computer-generated actions were based on their page positioning – the page is read top to bottom and left to right, and the actions required to generate the page elements are then listed accordingly.

Even at this early stage of analysis a number of usability problems already presented themselves as the result of completing the activity list. These are discussed in more detail later but they include aspects such as object page-positioning (shown through the sequence of the action list in table 6.2), and the display of irrelevant options (being present as an object in the AL). Other problems are the ‘double’ logoff, and the post-task navigational return to the Bank’s main home page, rather than to the Internet-banking home page from where the task was initiated.

A sequence of twelve screen dumps which captures the progression of observation for the task execution is presented in figure 6.3 – the screen and step numbers given in table 6.4 correspond to those in figure 6.3.

#### **6.4.4 Selection of specific objects and specific actions**

The specific actions and objects are the essential task elements, which are then restructured into the TDH – doing this has the effect of (temporarily) destroying knowledge about the relationships between these task-action pairs. This knowledge is regained in a more general manner during the construction of the KRG sentences.

The specific object list is presented in table 6.5, first as the raw list of all 30 objects from the activity list in figure 6.4, and then as a filtered list (where some of the objects not relevant to the task were removed), and refined (where the objects were rearranged in a pre-TDH hierarchy). The filtered and refined list reduced the number of objects to 18. During the refinement it was decided that the dialog box for leaving the secure server (displayed on a blank page), could be treated during this analysis as a ‘page’, and it is accordingly grouped with the set of page objects. During earlier passes the pull-down menu from where the account is selected had been grouped with the parent object account, as were the account number, password number and account statement. This was later changed and the pull-down menu list became a less specific ‘button(select)’ object. Statement and statementoff are two different states of the same object, and the latter was then also removed.

Similarly, figure 6.6 presents the specific activity list, arranged in descending frequency of occurrence. An initial and a final activity list are shown in figure 6.6. Both tables 6.5 and 6.6 clearly separate the two major agents namely the user and the computer. (There *are* other important agents such as the 'bank' and the 'bank-computer', which if included would have yielded a different set of TAKD outputs.)

Enlarged sections taken from the screen dumps in figure 6.3, which shows these specific objects, are presented in figure 6.2.1 through to figure 6.2.6.

### 6.4.5 Task descriptive hierarchy (TDH) diagrams

This is the principal analytical stage of TAKD – all available knowledge about the tasks is used to build a hierarchy of general and specific actions and objects. The nodes of the hierarchy represent actions and objects which progress from the general (on the left-hand side), to the specific (on the right side as leaf nodes) (Frascina and Steele, 1993:227). Note that there are nodes with observed data (specific objects) which were not obtained as the result of direct observation. These are considered to be either logically possible nodes, or known nodes, and the presence of these objects could suggest further investigation. Similarly Diaper et al. (1991:397) and Diaper (1989:117), emphasize that the presence of plausible empty leaf nodes is also useful and could suggest further examination as to the reasons why they do not have any specific objects attached to them.

It is important to recognize that the TDH is a non-sequential structure, and the sequence information inherent in the activity list is lost in the TDH. Similarly, as mentioned before, knowledge of the action or object relationships is also not included here. Both these categories of knowledge can be regained in a more critical and structured way when constructing the KRG sentences from the TDH.

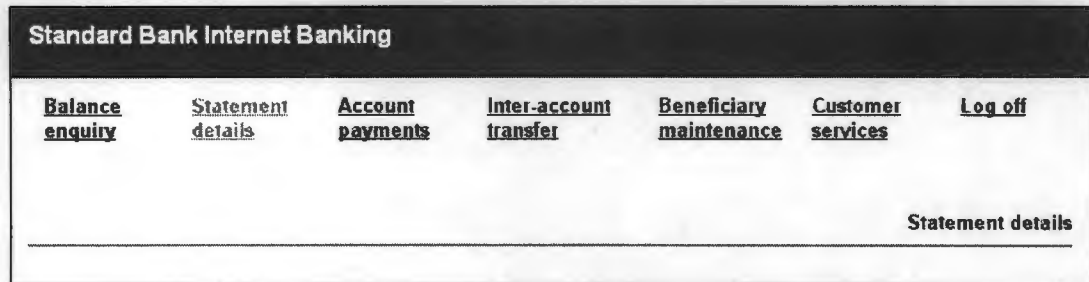
Building the TDH is a combined inductive (bottom-up and starting with the specific objects), and deductive (top-down and identifying suitable high level nodes), process (Diaper, 1989:120). This offers perhaps the best explanation as to why TAKD is such a strongly iterative method as these two construction approaches have to be matched by experimentation and refinement until they match and meet in a logically acceptable way in the middle level nodes.

Diaper (1989) discusses a number of heuristics that should be used during the construction of the TDH – these include making the diagram shallow and flat rather than deep by limiting the child (lower) nodes to no more than five or six, and discarding any single level lower-level nodes – the latter case would be merely a restatement of the same information.

The two TDH diagrams derived from the specific object and activity lists are presented as figure 6.4 for the specific objects, and figure 6.5 for the specific actions. Constructing a TDH for specific

actions is not normally done as it is often the case that there are very few of these which can be conveniently described as a set of generic, higher level actions. The results as presented in figure 6.5 and table 6.6 are proof of this general observation, but are nonetheless useful as it offer structured views of these actions.

**Figure 6.2 Enlarged screen dump cutouts taken during the task progression**



**Figure 6.2.1** The menubar and menubar options objects. Navigational position feedback is given via the hyperlink colour for the statement option, from the browser title bar (not shown), and the page title shown immediately below the menubar on the left hand side. The next figure (figure 6.2.2)), shows that positional feedback given via these text hyperlinks could be confusing – in this case both the balance enquiry AND the statement details options are shown in selected colours – although reaching such a state was not part of the observed task



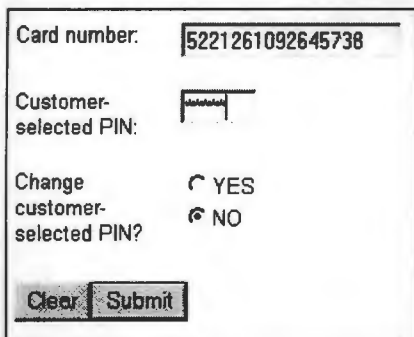
**Figure 6.2.2** This shows that positional feedback given via hyperlinks could be confusing – both the balance enquiry AND the statement details options are shown in selected colours – this state was not part of the task analyzed.



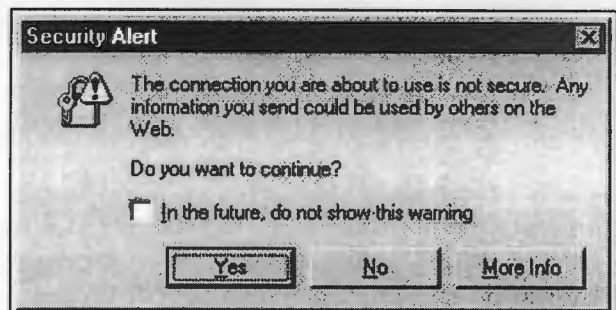
**Figure 6.2.3** The sign-on (logon) button object – this is a graphics picture associated with a hyperlink.



**Figure 6.2.4** The logoff button object. This is form generated and not a graphics object.



**Figure 6.2.5** The login panel showing the two login account objects, the form generated radio button objects, and the clear and submit button objects.

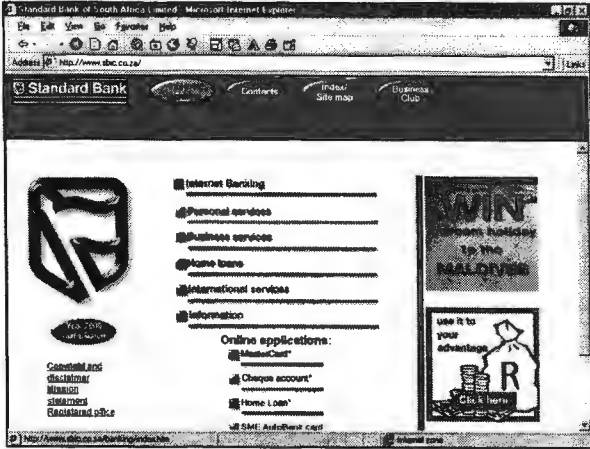


**Figure 6.2.6** The secure server dialog box object and its button objects.

Figure 6.3 Screen display sequence for the account statement task

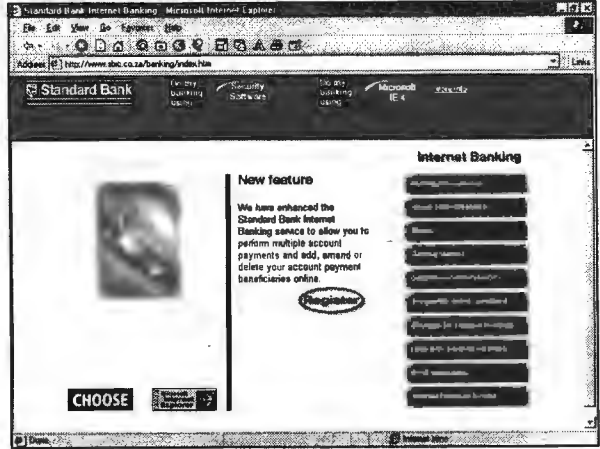
Step 1

Screen 1: Standard Bank home page



Step 2

Screen 2: Standard Bank Internet-banking home page



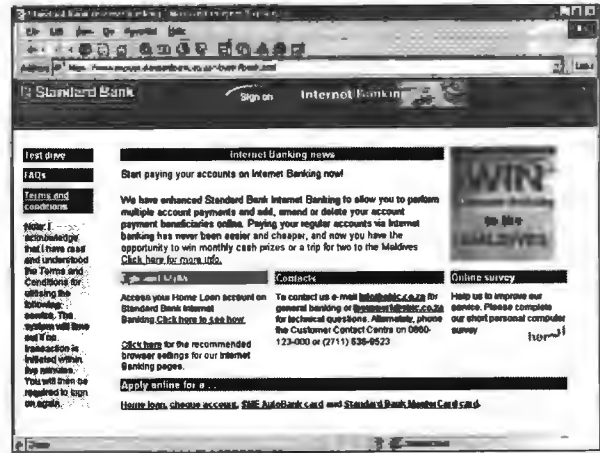
Step 3

Screen 3: Entering the secure server dialog



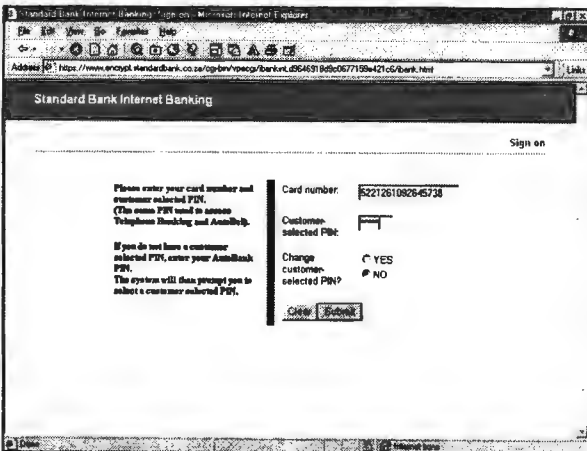
Step 4.1

Screen 4: Pre-login page with login button



Step 4.2

Screen 5: Main login page



Step 4.3

Screen 6: Login verification page

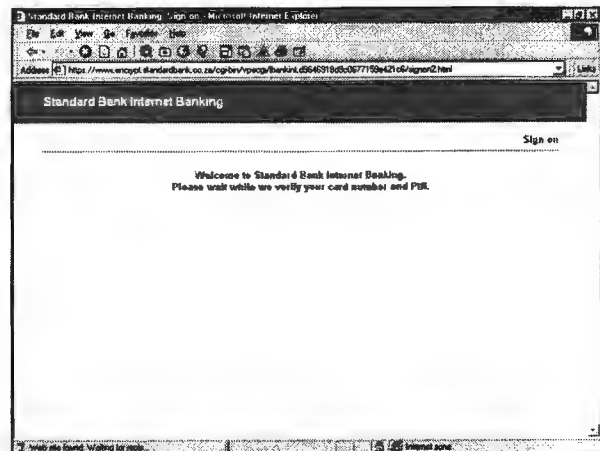
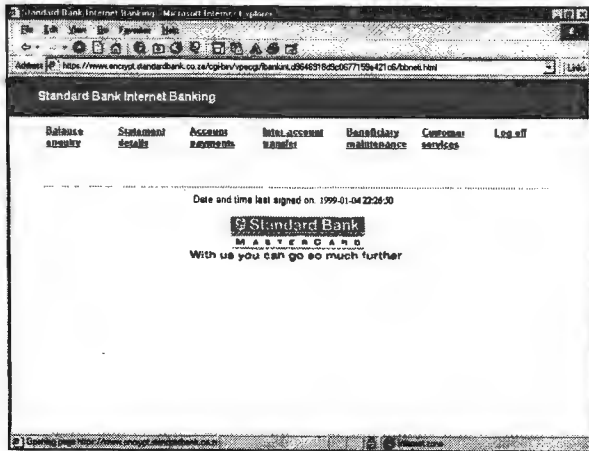


Figure 6.3 (cont.) Screen sequence for the account statement task

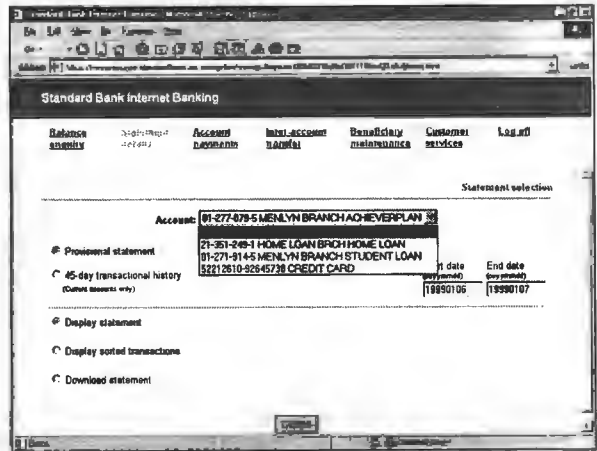
Step 4.4

Screen 7: Transaction options and welcome page



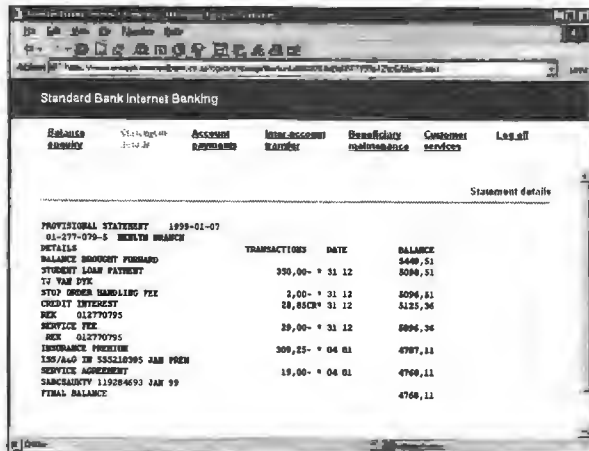
Step 4.5

Screen 8: Statement details account selection page



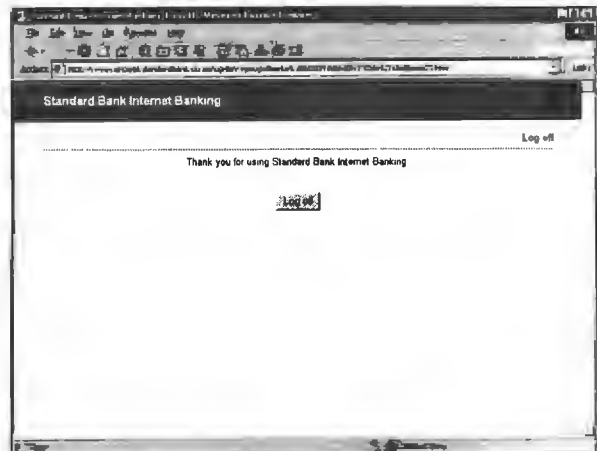
Step 4.6

Screen 9: Statement details display page



Step 4.7

Screen 10: Post-logout page



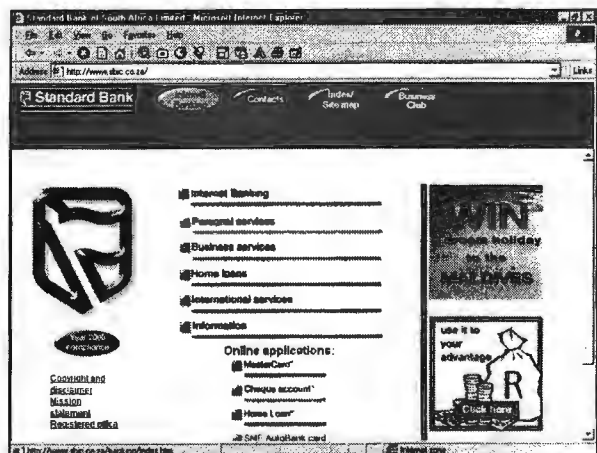
Step 5

Screen 11: Leaving secure server dialog page



Step 6

Screen 12: Standard Bank home page



**Table 6.4 Activity list for the account statement task**

Line	Screen	Step	Prose description	Specific Object ( ) and Action [ ]
1			Title: Internet-based banking statement task.	
2				
3	1-3	1-3	Pre-tasks (Steps 1, 2, 3): User navigated to the Internet-banking pre-login page. Computer displays Standard bank home page, prompts for navigation to Internet-banking site, prompts for login, displays secure server dialog box and prompts for ok button.	
4				
5	4	4.1	Computer displays pre-login page	(computer) [display1] (page-prelogin)
6	4	4.1	Computer prompts for navigation to login page	(computer) [prompt1] (page-login)
7	4	4.1	User presses login button	(user) [press1] (button-login)
8	5	4.2	Computer displays main login page	(computer) [display2] (page-login)
9	5	4.2	Computer prompts for account number	(computer) [prompt2] (number-account)
10	5	4.2	Computer prompts for password number	(computer) [prompt3] (number-password)
11	5	4.2	Computer prompts for change password	(computer) [prompt4] (radiobutton-passwordchange)
12	5	4.2	Computer prompts for clear entry button press	(computer) [prompt5] (button-clear)
13	5	4.2	Computer prompts for navigation to post-login page via submit button	(computer) [prompt6] (page-postlogin) (button-submit1)
14	5	4.2	User enters account number	(user) [enter1] (number-account)
15	5	4.2	User enters password number	(user) [enter2] (number-password)
16	5	4.2	User presses submit button	(user) [press2] (button-submit1)
17	6	4.3	Computer displays login verification page	(computer) [display3] (page-loginverify)
18	7	4.4	Computer displays transaction options page	(computer) [display4] (page-transactionoptions)
19	7	4.4	Computer prompts for transaction on menubar	(computer) [prompt7] (menubar) (menubar-other) (menubar-statement) (menubar-logoff)
20	7	4.4	User selects menubar statement option	(user) [select1] (menubar-statement)
21	8	4.5	Computer displays statement acc. select page	(computer) [display5] (page-statementaccselect)
22	8	4.5	Computer displays current menubar option	(computer) [display6] (menubar) (menubar-statementoff)
23	8	4.5	Computer prompts for transaction on menubar	(computer) [prompt8] (menubar) (menubar-other) (menubar-statementoff) (menubar-logoff)
24	8	4.5	Computer prompts for account selection on pull-down list	(computer) [prompt9] (pulldownlist-accounts)
25	8	4.5	Computer prompts for navigation to statement display page via submit button	(computer) [prompt10] (page-accountstatement) (button-submit2)
26	8	4.5	User opens account-selection pull-down list	(user) [open] (pulldownlist-accounts)
27	8	4.5	User selects account from pull-down list	(user) [select2] (pulldownlist-accounts) (account)
28	8	4.5	User presses submit button	(user) [press3] (button-submit2)
29	9	4.6	Computer displays account statement page	(computer) [display7] (page-accountstatement)
30	9	4.6	Computer displays current menubar option	(computer) [display8] (menubar) (menubar-statementoff)
31	9	4.6	Computer prompts for transaction on menubar	(computer) [prompt11] (menubar) (menubar-other) (menubar-statementoff) (menubar-logoff)
32	9	4.6	User reads account statement	(user) [read] (account-statement)
33	9	4.6	User selects menu-bar logoff option	(user) [select3] (menubar-logoff)
34	10	4.7	Computer displays post-logoff page	(computer) [display9] (page-postlogoff)
35	10	4.7	Computer prompts for final logoff button press	(computer) [prompt12] (button-logoff)
36	10	4.7	User presses logoff button	(user) [press4] (button-logoff)
37	11	4.8	Computer displays blank page	(computer) [display10] (page-blank)
38	11	4.8	Computer displays secure server warning dialog box	(computer) [display11] (dialogbox-secureserver)
39	11	4.8	Computer prompts for dialog box buttons press	(computer) [prompt13] (button-dialogbox-ok) (button-dialogbox-moreinfo) (button-dialogbox-no)
40	11	4.8	User presses dialog box ok button	(user) [press5] (button-dialogbox-ok)
41	12	4.9	Computer displays Standard Bank home page	(computer) [display12] (page-bankhome)

**Notes: Table 6.4**

- (1) The screen column corresponds to the series of screen dumps in figure 6.3.
- (2) The steps column corresponds to the steps listed in figure 6.3
- (3) The login button is referred to a 'login' rather than the 'sign-in' used on the Web page. This is done to be consistent with Standard Bank's use of 'logoff' rather than 'sign-off'.

**Table 6.5 Specific object list**

Unfiltered specific object list	Filtered and refined specific object list		
computer	agent-computer	page	login
user			transactionoptions
page-login			statementaccselect
page-loginverify			accountstatement
page-transactionoptions			bankhome
page-statementaccselect			dialogbox-secureserver
page-accountstatement			
page-postlogoff		button	login
page-blank			submit
page-bankhome			logoff
button-login			dialogbox-ok
button-clear			
button-submit1		menubar	menubar
button-submit2			statement
button-logoff			logoff
button-dialogbox-ok			
button-dialogbox-no	agent-user	account	account
button-dialogbox-moreinfo			statement
number-account			number
number-password			password
radiobutton-passwordchange			select-pulldownlist
menubar			
menubar-logoff			
menubar-statement			
menubar-other			
menubar-statementoff			
pulldownlist-accounts			
account			
account-statement			
dialogbox-secureserver			

**Table 6.6 Specific activity list**

Agent	Initial list		Final list	
	Activity	Frequency	Activity	Frequency
computer	prompt	13	prompt	13
	display	12	display	12
user	press	5	press	6
	select	3	select	3
	enter	2	enter	2
	open	1		
	read	1		

Figure 6.4 Task descriptive hierarchy

```

BANKING-WEB-PAGE
/
/-----COMPONENT (ELEMENT)
/
/ |
/ |-----PASSIVE
/ | /
/ | /-----TITLE page-title
/ | /-----OTHER background, text, pictures, animation, style, metatags
/ |
/ |-----ACTIVE
/ |
/ | |-----BUTTON
/ | | /-----SHAPE
/ | | / |-----RECTANGLE logoff, submit, dialog-ok
/ | | / |-----NON-RECTANGLE login (sign-in), password-change
/ | | /
/ | | /-----FUNCTION
/ | | / |-----PROCEED
/ | | / | |-----ACTIVATE login, submit, password-change, dialog-ok
/ | | / | |-----TERMINATE logoff
/ | | / |
/ | | / |-----UNDO clear
/ | | / |-----SELECT password-change (radiobutton)
/ | | / | account-select (pulldown list)
/ | | /-----GENERATION-SOURCE
/ | | |-----PASSIVE login (picture-button)
/ | | |-----ACTIVE submit, logoff, dialog-ok, change-password
/ | | | (script or form generated buttons)
/ |
/ |-----MENUBAR (HYPERLINK)
/ |
/ | |-----OPTION
/ | | {-----TRANSACTION statement, other (acc-transfer etc.)
/ | | {-----HELP (INFO) (not available as an option)
/ | | {-----TERMINATE logoff
/ | |
/ | |-----STATE
/ | | |-----SELECTED statementoff
/ | | |-----NOT-SELECTED statement, logoff, other-options
/ | |
/ | |-----APPEARANCE
/ | | {-----COLOUR statement, logoff, statementoff
/ | | {-----UNDERLINE statement, logoff, statement-off
/ | | {-----FONT TYPE/SIZE statement, logoff, statement-off
/
/-----ACCOUNT (OWNER)
/
/-----AUTHORIZATION (LOGIN ACCOUNT)
/ /-----NUMBER login-account-number
/ /-----PASSWORD login-password
/
/-----SELECTION (TRANSACTION ACCOUNT)
/ {-----DEFAULT first-account-listed
/ {-----SELECTED account-selected

```

KEY: / AND (all of the levels must be included)

| XOR (either/or, at least one level, but not more than one)

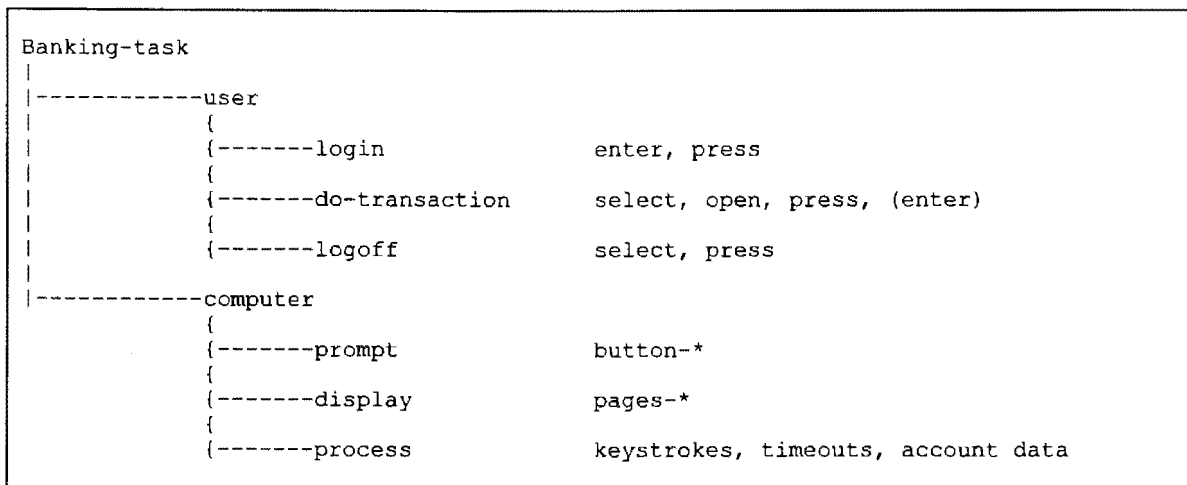
{ OR (could include more than one level, or all of them, but at least one)



**Notes: Figure 6.4**

- (1) A banking Web page does not always have an account – for example during pre-login and post-logoff
- (2) The specific Web pages will always have both passive and active elements. (A lack of active elements implies that the browser controls must be used for a proceeding action). For example even though the logoff page displays just one active element (the logoff button) it still has a passive page title displayed in the browser title bar. Similarly the login verification page only contains a line of passive text but its active element is in this case the action taken from the previous page – this page also contains JavaScript code to control a timeout setting on the verification period – this is also an active element.
- (3) Bracketed node names are possible aliases such as (page) ‘element’ being an alias for ‘component’.
- (4) The rectangle/non-rectangle node may be somewhat artificial for more general cases but it is acceptable for this specific task.

**Figure 6.5 Activity task descriptive hierarchy**



**6.4.6 Knowledge representation grammar (KRG) sentences**

These provide a means for describing the route for each specific object through the TDH. Constructing the TDH from the AL destroyed activity-object pairings – during this phase these relationships will be restored but they are then eventually re-linked in a more general way (Diaper (1989) use the term *generification* for this process). A convention is used for the KRG sentence where / / is used for AND branches, ( ) for XOR branches, and { } for OR branches.

A *uniqueness rule* applies: a completed TDH can distinguish between any two specific objects (and actions), through their set of KRG’s. For example the logoff button has the following set of object KRG’s:

```

component(active(button/shape(rectangle)/))
component(active(button/function(proceed(terminate)/))
component(active(button/generation-source(active)/))
    
```

whilst the login (labeled 'sign-in' on the Web page), button yields:

```
component(active(button/shape(non-rectangle)/))
component(active(button/function(proceed(activate)/))
component(active(button/generation-source(passive)/))
```

I.e. any one of these three partial KRG sentences would be sufficient to distinguish between these two buttons.

However to distinguish between the submit and logoff buttons the */function(proceed)/* KRG path is required. (Dix et al. (1998:271) note that the uniqueness rule should not be implemented too strictly – its use depends on the value (information), gained from implementing it.)

The object and action TDH's are then combined to yield a list of complete KRG sentences, each describing a simple task in general terms:

Before generalizing the following sentences may be derived to describe the following four (sub-) tasks:

(1) For the task of navigating to the login page there is only one KRG sentence:

```
Internet-banking-task (pre-login)
    press button/function(proceed(activate))/    sign-in button
```

(2) For the task of logging into the account we have the KRG sentences:

```
Internet-banking-task (login)
    enter account/number/
Internet-banking-task (login)
    enter account/password/
Internet-banking-task (login)
    press button/function(proceed(activate))/    submit button
```

(3) And for the task of displaying a statement for an account:

```
Internet-banking-task (do-transaction)
    select menubar(option)
Internet-banking-task (do-transaction)
    press button/function(select)/
Internet-banking-task (do-transaction)
    press button/function(proceed(activate))/    display statement
                                                    statement
                                                    account
                                                    submit
```

For the section above the action *button/function(open)/* was removed from the specific activity list and combined with the more general *button/function(select)/* action.

As discussed before two earlier sentences:

```
Internet-banking-task (do-transaction) open button/function(select)/ and
Internet-banking-task (do-transaction) select account/selected were generalized as:
```

*Internet-banking-task (do-transaction) press button/function(select)/*

Earlier in the analysis the sentence:

*Internet-banking-task (do-transaction) select menubar (not-selected)* seemed to be reasonable, but it is wrong – when an obtain-statement is the current task it is reasonable to select a statement for another account from the same position. This sentence was then changed to: *select menubar(option)*.

(4) And fourthly for terminating the session:

<i>Internet-banking-task (logoff)</i>	
<i>select menubar (option)</i>	<i>menubar-option- logoff</i>
<i>Internet-banking-task (logoff)</i>	
<i>press button/function(proceed(terminate))//</i>	<i>button-logoff</i>
<i>Internet-banking-task (logoff)</i>	
<i>press button/function(proceed(activate))//</i>	<i>button-dialogbox-ok</i>

During generification – which Diaper (1989:125-126) describes as a process of lower-node removal with the aim of identifying common classes of activity associated with a task, some of the KRG sentences above were combined (generalized), which then yielded the following set of three tasks (the pre-login task is combined with the login task), which are similar except for the login task and lower node removal:

#### **LOGIN:**

<i>Internet-banking-task (login)</i>	
<i>press button/function(proceed)/</i>	<i>sign-in button (*)</i>
<i>Internet-banking-task (login)</i>	
<i>enter (account/authorization/)</i>	<i>enter login-account and password</i>
<i>Internet-banking-task (login)</i>	
<i>press button/function(proceed)/</i>	<i>submit button (*)</i>

#### **DO-TRANSACTION:**

<i>Internet-banking-task (do-transaction)</i>	<i>display statement</i>
<i>select menubar (option)</i>	<i>account statement option</i>
<i>Internet-banking-task (do-transaction)</i>	
<i>press button/function(select)/</i>	<i>account for statement</i>
<i>Internet-banking-task (do-transaction)</i>	
<i>press button/function(proceed)/</i>	<i>submit button (*)</i>

#### **LOGOFF:**

<i>Internet-banking-task (logoff)</i>	
<i>select menubar (option)</i>	<i>logoff option</i>
<i>Internet-banking-task (logoff)</i>	
<i>press button/function(proceed)/</i>	<i>logoff button (*)</i>
<i>Internet-banking-task (logoff)</i>	
<i>press button/function(proceed)/</i>	<i>dialogbox-ok button (*)</i>

(\*) These five sentences are the same even though they act on four different objects – this does suggest that using fewer objects here would be appropriate.

The above then yields only four distinct sentences:

**Table 6.7 Final set of user KRG's**

<i>Internet-banking-task (*)</i> <i>press button/function(proceed)/</i>
<i>Internet-banking-task (do-transaction)</i> <i>press button/function(select)/</i>
<i>Internet-banking-task (*)</i> <i>select menubar (option)</i>
<i>Internet-banking-task (login)</i> <i>enter (account/authorization/)</i>

(Note that these four KRG's do apply to the specific task that had been observed – a more complete set of tasks such as an account transfer could include other KRG's to cater for entering of account amounts.)

For the computer agent only two specific actions are relevant, which yields the KRG sentences:

**Table 6.8 Final set of computer-agent KRG's**

<i>Internet-banking-task (*)</i> <i>display (page)</i>
<i>Internet-banking-task (*)</i> <i>prompt (one-of-four-user-agent-actions)</i>

### 6.4.7 Discussion of the results from the analysis

Ultimately, in order to design a human-computer interface, a well-defined description of the HCI is required – this should be based on what the user and/or system must know in order to accomplish the task goal (Johnson et al., 1985:502).

Action/object pairs are the main output from the TAKD analysis – when operationally combined they become knowledge items – and which describes explicitly the knowledge requirements for each task statement, presented in a grammar-like expression (the KRG sentences) (Johnson et al., 1984:499, 502). *The resultant combination of all the KRG's then constitute the complete knowledge requirement for the successful completion of the task.*

Apart from the main tangible and measurable outputs from the TAKD exercise (which are the TDH diagrams and KRG sentences), TAKD also produces useful intangible outputs such as the insights gained during the analysis into the task environment and execution. In the case of this specific analysis, where the number of actions and tasks are fairly small, and less benefit is therefore derived from the reduction offered through the KRG generification process, this intangible aspect of the

analysis proved to be the most useful. As a result a large section of the discussion that follows is not connected to the strict mechanical routines of the TAKD analysis in an obvious way, but the conclusions obtained are the result of the insights gained through completing the analysis.

What did become obvious during the analysis is that even for a short task as this one, that some form of automated tool support for the method would be advantageous. TAKD is a strongly iterative method and requires many refinements before a satisfactory output is obtained – as soon as these changes were made to the TAKD outputs such as the TDH diagrams and activity list the need for such as tool became especially apparent. Such a tool is available in the form of the LUTAKD toolkit (from Liverpool University), but this is a commercial product and could therefore not be used (Diaper, 1997:43) (Uddin, 1999)

Sequencing information was not studied during this analysis exercise. This may be included by constructing sets made from the list of final KRG's. For example, the sequence of actions for the login task would be 'enter (account)' and 'press (proceed)' when viewed at a fairly high level. A degree of re-ordering based on sequencing information had already been used in the analysis: The folding of the pull-down list actions 'open' and 'select' into the single 'press select' is an example of this. Any further use of the sequencing stage of TAKD would be more appropriate if a larger task had been chosen such as transferring between two accounts after the detailed statements for both accounts had been obtained. This would then yield a number of similar sequences of KRG sets (referred to as SRG's or sequence representation grammars by Diaper (1989:127)), which could for example, indicate occurrences of task repetition at a higher level than the KRG's presented here.

What are considered to be the most important results from the analysis are presented below:

- (1) **The small number of generic KRG's.** Apart from showing this to be a simple task, it more importantly also emphasizes that:
  - (a) Internet-based HCI is at present operating within a very (technology) restrictive task environment.
  - (b) It is a relatively new technology that has yet to evolve towards complexity. It has few interaction variations. These mainly consist of a few simple actions executed repetitiously. Within this new application domain Internet-based banking is even newer.

The above observations are supported by the frequency analysis of the final KRG's – for the user there are only four final KRG's with the following frequencies of occurrence:

Partial KRG	Frequency
<i>press button/function(proceed)/</i>	5
<i>select menubar (option)</i>	2
<i>enter (account/authorization/)</i>	2
<i>press button/function(select)/</i>	1

Similarly for the computer the actions are of two types only (excluding the invisible computer processing actions as previously discussed):

<i>prompt user</i> (do one of four generic actions)	13
<i>display page</i>	12

Human-computer interaction within this specific task environment is therefore characterized by a few simple actions that may be repeated many times. This has obvious favourable implications for task knowledge requirements and for usability learnability, but it does post some usability warnings: the large number of similar actions will tend to make the interaction boring, and the inherent delayed response resulting from unnecessary communication through these extra actions will further decrease the usability and reliability of the interaction.

Interestingly, Diaper et al. (1991:397) suggest that frequently occurring task elements as described in the TDH are more likely to be treated in a cognitively automatic manner than infrequent task elements. These actions are then as shown in the simple computer-operating model (figure 6.1), through the feedback loop S7-S4, as discussed earlier. These automatic actions would include the final KRG's 'press' and possibly also 'select'. An action that would not be automatic (in part because of its infrequency, but more likely because of the importance derived from the consequence of its activation), would be the transfer of funds, and even the act of logging in – i.e. all tasks that require a form of data entry (the 'enter' KRG).

Coincidentally, the small set of final generic KRG's is a vindication of TAKD's appropriateness for this type of analysis – any other result would have been counter-intuitive and confusing. This conclusion is perhaps in contradiction to what Diaper (1989:148), describes as a test for the correct node-level choice for the TDH: he notes that a result that shows the total number of KRG's to be only slightly larger than the number of different ones indicate the correct choice of low-level node detail. The conclusion above is however based on the generalized KRG sentences – there are nine KRG's in total, and four different KRG's. The conclusion reached when viewed within the context of the small task studied, and the explanations offered during generalizing, is therefore considered to be acceptable.

An examination of the four user-agent KRG's show two of these to be related to a navigation action (*press button/function(proceed)/* and *select menubar (option)* ), with a frequency of

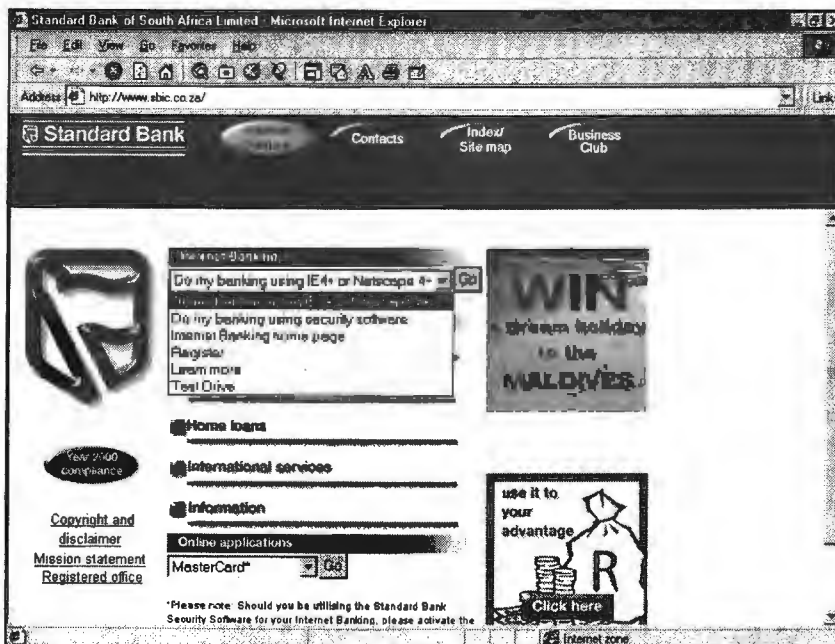
seven, and only three other action/object pair instances being part of the task. This result, given the nature of the task is hardly surprising: Interacting with an Internet-based application (and hypertext structures in general), is mainly about navigation, and this aspect of usability deserves the attention it usually gets. The primary functionality of the system is therefore related to these navigation actions and the focus should be on making it possible for the user to access these functions safely and easily.

The frequency results also indicate where an unreasonably large user effort is required. This is also related to reducing the need for navigation and will be discussed below.

As discussed in the beginning of this section, Standard Bank did implement changes to its banking Web site during early 1999, which did reduce the amount of navigation. It is now possible to reach the Internet banking sign-on page directly from the home page. This change is shown in figure 6.6 and should be compared with screen 1 in figure 6.3. The page that is skipped is screen 2 although the warning (alert) about entering the secure server obviously remains.

**Recommendation:** Reduce the unnecessary navigational button pressing – in particular the double logoff (discussed below), the frequent use of the submit button (also discussed below), and the display of intermediate pages as the result of these actions.

**Figure 6.6** The new homepage for Standard Bank with direct banking access



Source: www.sbc.co.za (1999).

- (2) **The submit button.** In both cases where submit buttons were used, their use should be bypassed. During the login process it should be sufficient if enter is pressed after the second input field (the password number) had been entered. It was noted that pressing enter here *is* accepted by the system, but in an inconsistent manner – pressing enter after an account had been selected from the pull-down list produces no reaction – both pages show the submit button to be in the same state – i.e. selected. The presence of the submit button, and its placement on the page (after the option for changing the password), could confuse the user rather than guiding him/her towards completing the task.

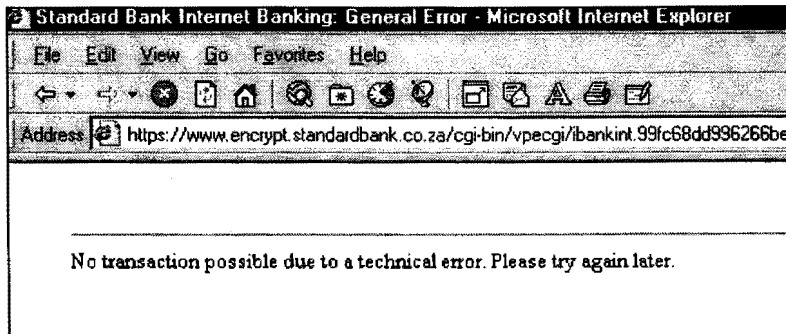
The second submit button is used to navigate to the page displaying the statement detail after the account had been selected from the pull-down list. The action of selection in the list should be sufficient to trigger the event of displaying the statement page. Once again the positioning of this button at the very bottom of the page is wrong – it would be better to display it next to the account selection element. The labeling of this button is perhaps also to be questioned – being form-generated it is probably a data processing remnant. It would be better to call this button a ‘proceed’ button for both cases where it had been used.

- (3) **The double logoff.** At the end of the task sequence a logoff option is available from the menubar displayed across the top of the page. When this (hyperlink) is selected the user is shown a second logoff page containing a single (forlorn) logoff button. A study of cognitive processes is not part of this specific analysis approach, but the result of this behaviour is bound to cause confusion and concern within the user, and is a major usability detractor. The first action should be sufficient for the logoff. Changing the label of this button to ‘Done’, ‘Finished’, ‘Leave bank’, or even ‘Bye’, would also be an improvement. To be consistent with the login button labeling namely ‘sign-in’, labeling the logoff button as ‘sign-off’ would also be acceptable.

If the browser’s navigation buttons are used at this stage to navigate back to the pages from where transactions may be done a server error occurs. This error will be examined in more detail in later chapters but the result of the user changing his mind at this stage eventually leads to the following page and message being displayed (figure 6.7):



**Figure 6.7** Error message displayed if navigation buttons used at the logoff page



Source: [www.sbic.co.za](http://www.sbic.co.za) (1998).

- (4) **The display of intermediate pages.** These include the temporary login validation page, the logoff page displaying a single element (a logoff button), and the secure server dialog box.

Usability demands effective feedback as a result of taking an action. For the login validation page it would have been sufficient to display the feedback message in the status bar at the bottom of the browser window, and so avoid the extra communication overhead associated with this (albeit simple) page. Note that appearances can be deceptive – an examination of the source code for this apparently almost blank ‘simple’ page reveals it be a hidden form-based page which also contains a small section of JavaScript to control a timeout period for the verification process:

```
<SCRIPT LANGUAGE="JavaScript">  
    timerID=setTimeout("document.signon2.submit()",1000)  
</SCRIPT>
```

The previous shows another important difference between evaluating an Internet-based application and a more conventional local-hosted computer program where this type of unrestricted and easy access to the source code of the application would have been unusual for the evaluator. This would be normal for the usability evaluations conducted in-house by the companies themselves (such as Microsoft doing a usability evaluation in their laboratories), but it is seldom the case that the independent evaluator has the opportunity to study the ultimate reasons for usability behaviour from the application’s source code. (Should this be forwarded as another reason for supporting the open source movement, Linux, and GNU licensing?)

The generation of a separate logoff page after this option had been chosen from the menubar is plain silly. The first logoff action should return the user to the origin page (see below). The

display of the secure server is a browser default that can be turned off, but it is perhaps the wrong default choice. The user does get feedback as to the security level of the current page through the display of the lock icon in the browser taskbar, and a less overwhelming reminder of this, requiring no user action such as the pressing of the OK button, would have been sufficient.

- (5) **Return to the page where the task was started.** The task is started when the login button is pressed from the Internet-banking home page. After the (double) logoff, the user is not returned to this page but to the Bank's main home page. It may be argued that the user has now completed his/her banking task (he/she did logoff after all), and has therefore no further interest in doing another banking transaction. This is probably true but the Internet-banking page does contain other items (links) that may be of interest to the user and these do not require him/her to be logged in. A return to the task's starting page would be preferred.
- (6) **Avoid complicated data entry.** During the login process the user is required to enter two numbers. The first is an account number that gives him/her access to that account and all other linked accounts over the Internet, and the second number is a PIN (personal identification number), that is used to authenticate the user. Both these numbers are exactly that – they are restricted to numerics only. It would be far more usable if the user could enter an alphanumeric ID for his account, and a password instead of the PIN. These are probably back-end restrictions imposed by the Bank's current (database) technology-implementation, but it should be possible to overcome these limitations fairly easily.
- (7) **The menubar fails as a navigational aid.** The menubar displays a number of transaction options. The user is reminded where he/she is (i.e. with what transaction he/she is busy) through a title displayed in the upper left-hand corner of the page (apart from the standard page title displayed in the browser's title bar and the URL shown in the browser's address box). The menubar is a simple list of text hyperlinks, which means that *all* options that had been previously selected are shown in a different colour. This hyperlink behaviour (which can be changed through style sheets and scripting), is shown in figure 6.2.2. The preferred implementation would have been to show only the current option differently. To introduce this would only entail some reasonably basic page scripting. Note that it is possible that the use of scripting may under certain circumstances compromise the site's security.

## 6.5 Conclusion

The use of a structured approach to the analysis of the banking task has distinct advantages, and proved itself to be most beneficial. The mere act of conducting the exercise was revealing, and yielded pointers to a number of usability problem areas. In this regard Diaper (1989:154) notes that the claim that many of the results obtained from a TAKD analysis are in fact self-evident is false, and do not take into account the short-lived nature of human memory when dealing with data of this type. The construction of the activity list, the TDH, and the derivation of the KRG sentences from them, yielded important design and requirements specification information such as justification for the inclusion or exclusion of application elements. What did become apparent is that a TAKD exercise is never completed – the author is convinced that after a delay of some months a revisit of the TAKD result elements as presented here would yield many new insights.

In addition Diaper and Addison (1992:137) stress that TAKD is even after a decade of refinement and use, still under development, and its current (considerable) utility should not be confused with its (future) potential value.

## Chapter 7

# Usability Evaluation

**This chapter examines usability evaluation approaches, including a heuristic evaluation and a checklist-based evaluation. The application of these techniques to an Internet-based banking task is introduced.**

## 7.1 Introduction

There are a number of online evaluation services for Internet banking Web sites. The criteria used during these evaluations tend to be feature-oriented (i.e. what services and facilities are offered on the bank's Web-site), but also include some of the other more important usability properties as examined in previous chapters. These Web-based evaluation services and their approaches are examined in more detail later in this chapter. Because of their narrow focus the evaluations conducted in this and the following two chapters followed a more conventional route by making use of general (not necessarily Web-oriented only), usability evaluation guidelines.

Two inspection-based evaluation methods were used: a *heuristic evaluation* on the *overall* Internet banking site and a closely related *checklist-based technique* on a *specific task* within the site. This chapter discusses the choices available for these two evaluation approaches – implementation details for the actual evaluations are described in the next two chapters.

### 7.1.1 The design and measurement of usability evaluation

De Wet (1994:74-75) identifies a number of measurement criteria, classes of performance measurement and methods for usability evaluation and measurement from published literature. These are concerned with assessing the user's performance, satisfaction and acceptance, within a specific situational context, and include:

- (1) **Performance aspects:** Goal achievement (accuracy, effectiveness), work rate (productivity, efficiency), knowledge acquisition (learnability, learning rate), and operability (error rate, function usage).
- (2) **Satisfaction aspects:** Success rate and needs satisfaction, ease of use, learning, using, remembering, convenience, effort, tiredness, time, errors, and the number, repetitiveness and sequence of activities.

More specific usability measurement aspects are: versatility and functionality, error scores (errors committed and correction time), performance (task execution times), ease-of-learning (time to learn a task), and start-up ease (installation and activation times).

De Wet (1994:75) mentions that usability could be measured and evaluated by identifying usability properties to which the system should in general adhere to, and then determining the degree and manner in which it complies to these properties. Any change in the product, user, task or environment, may induce a change in the system's usability.

This is in support of the view taken earlier by Card, Moran and Newell (1983:407-410). They stress that three structural variables – the **TASK**, the **USER**, and the **COMPUTER** – determine its performance variability. This implies that the evaluation of usability principles and properties (part of the computer domain) is strictly speaking an incomplete system evaluation. Card et al. (1983) elaborate, and add detail to their *computer* domain (dialog style, command syntax, naming conventions, display layout, input devices, and response time), the *user* (intellectual abilities (technical and intelligence), cognitive style (risk preference, curiosity, persistence), experience, knowledge (method, conceptual, task expertise), and perceptual-motor skills), and the *task* (task domain and task model)). They (1983:409) present an alternative classification based on performance variables, which appears to be a combination of the previously discussed usability properties and cognitive evaluation variables – it includes aspects such as enjoyability, acceptability, quality, robustness, error, time, learning, functionality, fatigue, stress, and memory overload (short-term or STM) and recall (long-term or LTM).

### 7.1.2 The evaluation of usability properties

Evaluation techniques are used to test the usability and functionality of the design and implementation of an interactive system, and the evaluation style can be either *field-based* (under operational conditions), or *laboratory-based* (under controlled and/or simulation conditions) (Dix et al., 1998:407). It differs from task analysis (which examines the broad total task and its components), in that evaluation focuses on the human-computer interface only. Evaluation has three main goals (Dix et al., 1998:406-407):

- (1) **Assessment of the system's functionality**, its match with the user's requirements, and the ease of task performance, i.e. evaluating both the range and reachability of functionality.
- (2) **Assessment of the effect or impact of the interface on the user** (and vice versa). This would include aspects such as learnability, usability, user overloading and attitude. The usability properties as later identified can be used as an evaluation basis.
- (3) **Identifying specific system problems**. These are aspects which cause unexpected results or user confusion, discomfort or effect his confidence, i.e. the negative aspects of the design.

Dix et al. (1998:436) identify eight factors that may be used when choosing an evaluation method. These include the development stage, the evaluation style, the degree-of-technique subjectivity or objectivity, the resources available, the information requirements, the response immediacy, and the degree of interference.

Preece and Rombach (1994:569-571) expand on the evaluation categories by Dix et al. above, and identify *five* approaches to usability evaluation and measurement as presented in table 7.1:

**Table 7.1 Five approaches to usability evaluation and measurement**

(1) <b>Laboratory testing methods:</b> These are traditional methods predominant in the 1970's and early 1980's, and are used to examine differences in one or more independent variables. Issues such as menu and command names and menu organization were typical issues, based on how people perform during a short and tightly controlled task.
(2) <b>Usability engineering:</b> This is a process where the usability of a product is specified in advance. During development testing takes place to check whether the planned-for levels of usability have been achieved. Usability criteria and acceptance levels are defined, and it is typically used in an iterative development fashion. In addition to quantitative benchmarking of people interacting with a computer system at task level in specially designed 'usability laboratories', a degree of field testing is also used, and provides a degree of qualitative data. (Microsoft makes extensive use of their own usability laboratories, and their beta (field) testing programs for products such as Windows 95, Explorer and Windows NT 4, were (and are) unequalled in their scope.)
(3) <b>Heuristic evaluation:</b> Heuristics (such as feedback, error messages, 'speak-the-user's-language' and short cuts) are advanced to help evaluators – either HCI specialists or 'expert' end-users – identify usability problems when inspecting a software system.
(4) <b>Ethnographic methods:</b> This is based on contextual enquiry, i.e. the usability of systems in their normal working context. This approach adds extra dimension to the above approaches in that the total task system is included in the study.
(5) <b>Cognitive modeling:</b> These focus on human processes, and approaches and methods such as the Keystroke Level Analysis (Card, Moran and Newell, 1983:259), GOMS (Goals-Operators-Methods-Selection rules) (Card et al., 1983:139), and Cognitive Complexity Theory. It is used to model the user's cognitive and physical behaviour. The Cognitive Walkthrough method has similar aims, and the lure behind these approaches is the potential for predicting usability problems very early during the development process.
Source: Preece and Rombach (1994).

Bastien and Scapin (1995:184) add a third category to the two used by Dix et al., and identify *three* main categories of approaches for the evaluation of human-computer interfaces. Their three approaches are presented in table 7.2. Preece et al.'s (1994) heuristic evaluation slots in comfortably with Bastien et al.'s (1995) expert evaluation category. A checklist-based evaluation does not however necessarily require a usability-expert evaluator – it is sometimes designed to be used either by the (inexperienced) end user, or the usability-naïve Web designer.

**Table 7.2 Three categories of approaches to usability evaluation**

- (1) **User performance based:** These allow for the collection of qualitative and quantitative data which are especially useful when dealing with new interface features or new tasks.
- (2) **Theoretical and formal model-based methods:** These approaches allow for predictions on the usability of interfaces and user's performance. They are useful when 'real' users (field-testing) is not an option, and also tend to be used mainly for comparisons among several different interfaces rather than in judgements on the overall quality of an interface. They also tend to have limited predictive power in complex situations.
- (3) **Expert Judgement Methods:** The problems encountered with the previous two categories may explain why this is probably the most widely used evaluation approach. It consists of either comparing interface features to existing recommendations and standards, or performing 'subjective' evaluations (based on training, field experience, and examination of human factors (ergonomics) data). Bastien et al. (1992:184), consider this approach to be superior to cognitive walkthroughs, guideline-based, and usability-testing methods, *in terms of the amount and importance of problems found*, and in its benefit-cost ratio (problems found per evaluator). The tools and knowledge currently available to evaluators are primarily standards, guides (which may be based on human factors criteria), recommendations, algorithms, and checklists. In a later study (Bastien et al., 1995:106), this expert-based evaluation method is expanded on through the use of a set of ergonomic criteria.

Source: Bastien and Scapin (1995)

Note that analytical evaluation techniques such as a cognitive walkthrough and a heuristic evaluation can be used in both the design and implementation stages of the interface (Dix et al., 1998:439). Being analytic it usually requires a *usability expert* conducting the evaluation. Experimental, query and observational techniques are aimed at the fully implemented system (or prototype), and will typically involve (*end*) users with the right population characteristics.

According to Dix et al. (1998:407) an evaluation under controlled laboratory conditions is particularly suitable for the comparison of alternative designs and implementations – it suits the deliberate manipulation of the context in order to uncover problems and advantages. This will be the preferred style used during this project wherever possible. It is important to note that the situational context of the evaluation also implies a realistic mapping to typical operational conditions. Both a heuristic evaluation (table 7.1 category 3, table 7.2 category 3), and a checklist-based expert evaluation (the same categories), will be used during the evaluation. Although cognitive walkthrough methods should be regarded as complementary to these two empirical evaluation methods, they are even more subjective, and may therefore suffer from significant evaluator bias (Dix et al., 1998:437).

## 7.2 Heuristic evaluation methods

This is based on a set of usability criteria (the 'heuristics'), and the interface is examined for cases where these criteria are violated (diminished) or promoted (adhered to). These usability criteria have been discussed previously and should include factors such as predictability, consistency, feedback, user overloading, and task oriented dialog (Dix et al., 1993:373). (Later Dix et al. (1998:413-414), replace these factors by Nielsen's (1993) list as shown in table 7.4.) As a criterion delineation is used for the evaluation it is less subjective than a cognitive walkthrough.

Nielsen (1998, URL: UseIt2) observes that: '*Heuristic evaluation aims at explaining each observed usability problem with reference to established usability principles, and it will be fairly easy to generate a devised design according to the guidelines provided by the violated principles for good interactive systems.*'

Borges, Morales and Rodriguez (1998:138) note that heuristic evaluation approaches are eminently suitable for detecting Web page usability problems. This is because they do not require expensive user testing facilities and can be used effectively and quickly by the less experienced in usability, Web page designer.

Levi and Conrad (1996:52) comment on an interesting application of Nielsen and Mack's (1994) heuristic evaluation guidelines to a set of Web page 'prototypes' (initial versions). They regard a heuristic evaluation to be the least formal of the usability inspection methods as proposed by Nielsen et al. (1994), and state that it differs from conventional empirical testing methods in three significant ways – '*it is easy, fast, and cheap*'. I.e. although it requires a trained evaluator (*not* a 'typical' end-user) the training required is minor, it is easy to set up and run, and it requires minimal other resources.

They applied a heuristic evaluation to the USA Bureau of Labour Statistics' Web site (stats.bls.gov), and used an interesting and novel approach to categorize the usability problems as identified by the evaluation exercise. They (1996:52) conclude that the design of a Web site could follow the phases as used for building conventional software systems, starting with requirements gathering and ending with distribution (publishing?). During the testing phase (activity within a phase?) they produce a Web site 'prototype' (initial design). This is then evaluated heuristically in six stages. These stages are a project overview (objectives, intended audience, and expected usage patterns), defining a list of usability heuristics or principles, identifying and training the evaluator teams, executing the evaluation individually and as a team, assigning severity ratings to the usability violations found, and formulating recommendations for removing these violations.



It is likely that a heuristic evaluation conducted on an existing Web site will be less cost-effective and rigorous (in terms of problems detected), than when it is integrated with the overall Web page design process. Instone (1998, URL: W3J1) presents a methodology for the page design process which consists of the stages listed in table 7.3, and includes a heuristic evaluation as stage 7.

Vora (1998:153) presents another view of such a Web site design methodology based on human factors and usability engineering principles. He notes that although his approach includes the usual system development stages of planning, analysis, design/development, implementation and maintenance, there are clear differences in the way Web sites should be designed when compared to traditional software systems. Some of these include designing for both exploratory and goal-oriented interaction, navigation issues such as entering a site at almost any page, and the wide and unpredictable variety of platforms and browsers. Vora (1998:164) also examines a number of different testing approaches during the site development, and considers checklist-based evaluations to be suitable for a remote, live, Web site testing.

**Table 7.3 Web usability engineering life cycle**

Stage	Description
1	Know the user: Individual user characteristics, the user's current and desired tasks, functional analysis, and the evolution of the user and the job.
2	Competitive analysis.
3	Setting usability goals and financial impact analysis.
4	Parallel design.
5	Participatory design.
6	Coordinated design of the total interface.
7	<b>Applying guidelines and heuristic analysis.</b>
8	Prototyping.
9	Empirical testing.
10	Iterative design and capturing design rationale.
11	Collecting feedback from field use.

Source: Instone, URL: W3J1 (1998).

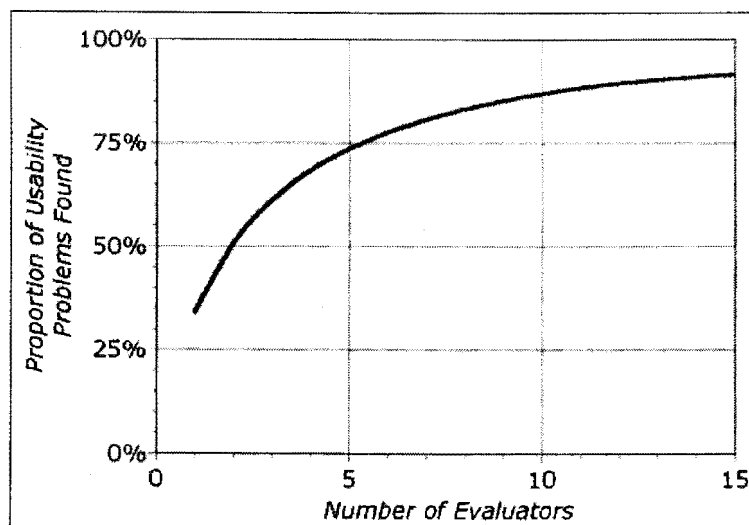
The author was the sole evaluator for the heuristic evaluation conducted on the Standard Bank Web site. Figure 7.1 is taken from Nielsen (URL: Useit2) and shows the proportion of interface usability problems found (using a heuristic evaluation) versus the numbers of evaluators. The curve represents the average from six heuristic evaluation case studies. Based on this data Nielsen recommends the use of at least three evaluators.

Levi and Conrad (1996:54) document a novel variant of the evaluation team make-up as proposed by Nielsen and Mack (1994). They used two separate teams, one consisting of four Web page developers, and another composed of four HCI experts. A comparison of the results for the type and

number of usability problems identified by these two different groups makes very interesting reading, and will be commented on later.

In defense of this project it should be noted that although this study is based on the use of a single evaluator, the evaluation was carried out over a period of about two years. During this time the evaluator effectively 'lived' with the Web site and its evolution – it was used for actual banking tasks in addition to conducting the simulated evaluations. It is therefore likely that the proportion of problems detected would be much higher than indicated by Nielsen's data for a single exercise evaluation, and could approach an (acceptable) 75% proportion of problems found. The significant number and variety of problems detected – as presented during the next two chapters – support this view. The case study used by Nielsen (URL: Useit2) is interesting because it refers to a voice response system allowing customers access to their bank accounts (Nielsen 1992).

**Figure 7.1 Usability problems detected versus the number of evaluators used**



Source: [www.useit.com/papers/heuristic/heuristic\\_evaluation.html](http://www.useit.com/papers/heuristic/heuristic_evaluation.html)

There are a number of documents available on the Internet that offer guidelines as to what should be used as the heuristics when evaluating a Web page. These will be examined in the next section.

### 7.2.1 Nielsen's ten usability heuristics

Nielsen (1998, URL: UseIt1) presents a list of ten Usability Heuristics (shown in table 7.4), based on an analysis of 249 usability problems. Dix et al. (1998:414) also present this list in the same format.

**Table 7.4 Nielsen's ten usability heuristics**

1	<b>Visibility of system status:</b> The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.
2	<b>Match between system and the real world:</b> The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.
3	<b>User control and freedom:</b> Users often choose system functions by mistake and will need a clearly marked 'emergency exit' to leave the unwanted state without having to go through an extended dialog. Support undo and redo.
4	<b>Consistency and standards:</b> Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.
5	<b>Error prevention:</b> Even better than good error messages is a careful design which prevents a problem from occurring in the first place.
6	<b>Recognition rather than recall:</b> Make objects, actions, and options visible. The user should not have to remember information from one part of the dialog to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.
7	<b>Flexibility and efficiency of use:</b> Accelerators – unseen by the novice user – may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.
8	<b>Aesthetic and minimalist design:</b> Dialogs should not contain information that is irrelevant or rarely needed. Every extra unit of information in a dialog competes with the relevant units of information and diminishes their relative visibility.
9	<b>Help users recognize, diagnose, and recover from errors:</b> Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.
10	<b>Help and documentation:</b> Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.
Source: Nielsen (1993) (URL: UseIt1, 1998).	

Previously Nielsen (1993:115) examined each component of this list in detail. Note however that this is a general usability list, and had not been changed or adapted for Web page evaluation.

### 7.2.2 Instone's heuristic evaluation approach

Instone (1998, URL: Webreview1) notes that heuristic evaluation is well suited for the Web because it can be constructed to be fast, easy and inexpensive – and importantly the evaluators can be spread across the globe with no loss in efficiency. He suggests that the stages involved during a heuristic evaluation should consist of: (1) identifying your heuristics, (2) gathering opinions about the site usability (based on the heuristics), (3) merging and rating the problems that were identified, and (4) work towards solutions.

In another article Instone (1998, URL: W3J1) also presents a list of sixteen heuristics that are specifically designed to be suitable for the evaluation of Web pages (shown in table 7.5). There is a

close correspondence between this list and the Nielsen (1993) list as presented previously: Items 1, 2, 4, 7, 9, and 10 relate directly to the same-numbered items from Nielsen's list, and items 3, 5, 6, and 8 are either derived, or distilled from Nielsen's list. Items 11 and 16 are also present in Nielsen's list. Dix et al. (1998:413) present the first nine items from this list in the same format.

Instone (URL: [Webreview2](#)) adapts Nielsen's ten usability heuristics for the Web. (Chapter 8 examines an illuminating comment he makes in the introduction to this document.) For example under system visibility he focuses on navigation and adds 'Probably the two most important things that users need to know at your site are *'Where am I?'* and *'Where can I go next?'*'. This is similar to his comments as presented in table 7.4, but he expands on each of Nielsen's guidelines with a suitably Web-pragmatic emphasis.

**Table 7.5 Instone's Usability Heuristics**

1	Simple and natural dialog	9	Prevent errors
2	Speak the user's language	10	Help and documentation
3	Minimize the user's memory load	11	User control
4	Consistency	12	Site structure
5	Feedback	13	Design for change
6	Clearly marked exits	14	Bandwidth
7	Shortcuts	15	Browsers
8	Good error messages	16	Standards
Source: <a href="http://www.w3j.com/5/s3.instone.html">www.w3j.com/5/s3.instone.html</a>			

### 7.2.3 The Borges and Morales usability heuristics

Borges and Morales (1998, URL: [ACM1](#)) constructed a list (table 7.6) of Web-specific usability heuristics based on the earlier work by Nielsen (1993). Borges, Morales and Rodriguez (1998:141) later make use of this list during a heuristic evaluation of six university Web pages. Their list is considered to be perhaps too specific for direct use here – it would be more effective when formatted into a checklist, or when used in conjunction with the lists as presented by Nielsen and/or Instone (as presented previously).

Interestingly Borges et al. (1998:138) select the following three usability heuristics as the major contributors to (empirically) proven Web page usability problems. These three are Nielsen's heuristics numbers 2, 4, and 8 from table 7.4 namely (1) **Aesthetic and minimalist design**, (2) **Match between system and the real world**, and (3) **Consistency and standards**. Non-adherence to the first of these is considered by them to be the most important in terms of usability problems caused.

**Table 7.6 The Borges and Morales list of usability heuristics**

<b>For any page:</b>	
1	Headers should not take more than 25% of a letter size page
2	Headers and footers should be clearly separated from the body of the page. (One way of achieving this is by placing bold lines or bars between them and the body.)
3	Names of links should be concise and provide a hint on the content of the page that they link to. (Avoid using technical words such as servers, links, Web server, etc.)
4	Avoid adding explanatory comments to textual links
5	Avoid 'linking-mania' (making a link every time a keyword of a page is mentioned in a text)
6	Verify that links connect to existing pages
7	Linking icons should have a distinctive feature of the page they are linking to
8	Maintain consistency when using icons. The same icon should be used for the same intended purpose
9	Colors should be selected so that the pages can be clearly displayed and reproduced on black and white displays and printers
10	It is desirable to include the date the page was last modified, the mail address of the person that maintains the page and the URL address of the page on a footer
<b>For the home page of the repository:</b>	
11	Pages should not be overcrowded with links
12	Pages should be short (about a letter size page)
13	Links should be to primary aspects or characteristics of the institution. Textual information should be left for secondary pages
14	Organize links as primary and secondary topics
15	Links to resources or other repositories on the Internet should be placed on a secondary page. (This page should be reached with a link on the primary page.)
16	A more extensive index of links, properly grouped, can be provided on a secondary page for fast access to a wide range of the institution's repositories
Source: <a href="http://www.acm.org/sigchi/chi96/proceedings/shortpap/Rodriguez/rn_txt.htm">www.acm.org/sigchi/chi96/proceedings/shortpap/Rodriguez/rn_txt.htm</a>	

### 7.2.4 The Levi and Conrad heuristic list

In a very well designed usability inspection project, Levi and Conrad (1996:54) describe the application of Nielsen and Mack's (1994) usability guidelines to the evaluation of a set of Web pages. Their eight guidelines are very similar to the list presented by Instone (1998, URL: W3J1). After the evaluation they then modified the list based on feedback from their two different evaluation teams, and produce a new list with nine guidelines. Table 7.7 presents both sets of guidelines. Note that in their second list items 6, 7, 8, and 9 (and to a lesser extent also item 4), are strongly related to navigational issues – the results from their evaluation indicate this to be a major cause of usability problems.

Levi et al. (1996) continue, and analyze the results from the evaluation by assigning severity ratings to each usability violation found on a five point scale. (Note that Bastien and Scapin (1995) (table 7.2, category 3), refer to the '*the amount and importance of problems found*'.) The scale varied from 0: Not a usability problem, 1: Cosmetic, 2: Minor, 3: Major, to 4: Catastrophic problem. Accordingly they produce a list of usability violations which contains both frequency and severity information.

**Table 7.7 The Levi and Conrad original and modified heuristic evaluation guidelines**

Original list of guidelines	Modified list of guidelines
1 Speak the user's language	<b>1 Speak the user's language.</b> (Use words, phrases, and concepts familiar to the user. Present information in a natural and logical order.)
2 Consistency	<b>2 Be consistent.</b> (Indicate similar concepts through identical terminology and graphics. Adhere to uniform conventions for layout, formatting, labeling, etc.)
3 Minimize the user's memory load	<b>3 Minimize the user's memory load.</b> (Take advantage of recognition rather than recall. Do not force users to remember key information across pages.)
4 Flexibility and efficiency of use	<b>4 Build flexible and efficient systems.</b> (Accommodate a range of user sophistication and diverse user goals. Provide instructions where useful. Lay out screens so that frequently accessed information is easily found.)
5 Aesthetic and minimalist design	<b>5 Design aesthetic and minimalist systems.</b> (Create visually pleasing displays. Eliminate information that is irrelevant or distracting.)
6 Chunking	<b>6 Use chunking.</b> (Write material so that documents are short and contain exactly one topic. Do not force the user to access multiple documents to complete a single thought.)
7 Progressive levels of detail	<b>7 Provide progressive levels of detail.</b> (Organize information hierarchically, with more general information appearing before more specific detail. Encourage the user to delve as deeply as needed, but stop whenever sufficient information has been reached.)
8 Navigational feedback	<b>8 Give navigational feedback.</b> (Facilitate jumping between related topics. Allow the user to determine his/her current position in the document structure. Make it is easy to return to an initial state.)
	<b>9 Do not lie to user.</b> (Eliminate erroneous or misleading links. Do not refer to missing (or inaccessible) information.)
Source: Levi and Conrad (1996).	

### 7.2.5 A collection of other usability heuristics

Scholtz (1998, URL: [Uswest1](#)) presents a collection of other usability heuristics collected from four institutions represented by their Web designers. The institutions were Ameritech, DePaul University, Intel, and NIST. They had to answer the questions 'What are the top two criteria you use in designing and developing Web sites?' and 'How have you tested these criteria during design and development?'

The results from this survey are presented in table 7.8 below. Ameritech makes use of evaluation techniques that include checklists, focus groups, and card sorting techniques. DePaul University uses automated tools for navigation link-checks and response times loggers. Intel employs general usability testing and statistical tools, and NIST makes use of standard heuristic evaluation criteria (adapted from Nielsen, 1993), and usability goals. For the latter they use a simple binary test i.e. yes/no.

**Table 7.8 Usability criteria used in designing Web sites**

<b>Ameritech</b>	(1) Is the content organized in a logical and use-sensitive manner with clear navigation? (2) Is the interface easy enough to use without documentation or training?
<b>DePaul University</b>	(1) Design effective documents and effectively manages user expectations so that delays in response time or variable response times do not adversely affect users. (2) Design Web sites that support both searching behavior and link traversal by users.
<b>Intel</b>	(1) <b>Intuitive navigation</b> for a wide audience of Intel products and technologies (developers, end users, investors, job seekers, channel, distributors, etc.). With everyone from software developers wishing to enhance their applications to include MMX (r) technology to small investors wishing to purchase Intel stock, our Web site must meet all of these diverse user requirements. (2) <b>Enhance the Intel brand.</b> We are a company that represents innovation, technology as well as high quality. Our Web site needs to reflect and reinforce that image in every element. That includes everything from <b>consistent look and feel</b> of each Web page all the way through outstanding server performance to eliminate downtime. (3) <b>Compatibility</b> with a wide range of desktop configurations. Our audience base consists of Win 3.1, through Win NT users, UNIX users and 14.4 dial-up access through to dedicated T1 access. We have a mix of browsers and browser versions. One of our design requirements is that the user experience is consistent as much as possible across all platforms and plug-ins.
<b>NIST</b>	The majority of Web sites at NIST are concerned with providing information. In other words, the focus is on content and not task. This I believe is true for much of the Web, but may change as more transaction processing is performed on the Web and other task-based activities are introduced. From this content perspective, I usually develop a set of usability goals with the site owner. These goals are based on information location/retrieval of the content. The other criteria I establish involves consistency in terms of site identity which is done by developing a page template involving contact information, layout, logos, etc.
Source: <a href="http://www.uswest.com/Web-conference/proceedings/scholtz.html">www.uswest.com/Web-conference/proceedings/scholtz.html</a>	

### 7.2.6 The experimental usability heuristics list

Constructing the list to be used as a framework for the evaluation of the banking Web site proved to be somewhat problematic. Initially it was deemed sufficient to make use of the work by Nielsen and Instone's adaptation thereof for the Web, and use this as the basis for the evaluation.

Finally it was decided to make use of the author's own earlier experience (Van Dyk, 1996), with such an evaluation (a heuristic evaluation of file-managers running under X-Windows, Windows 3.1 and Windows 95), and employ the general set of usability principles and properties as a guideline. The advantage of doing it this way is that the evaluation is likely to be more comprehensive than when using the more limited set from Nielsen and Instone, but it has the (apparent) disadvantage that the guidelines are not specifically adapted (or more correctly **prioritised**), for the Web.

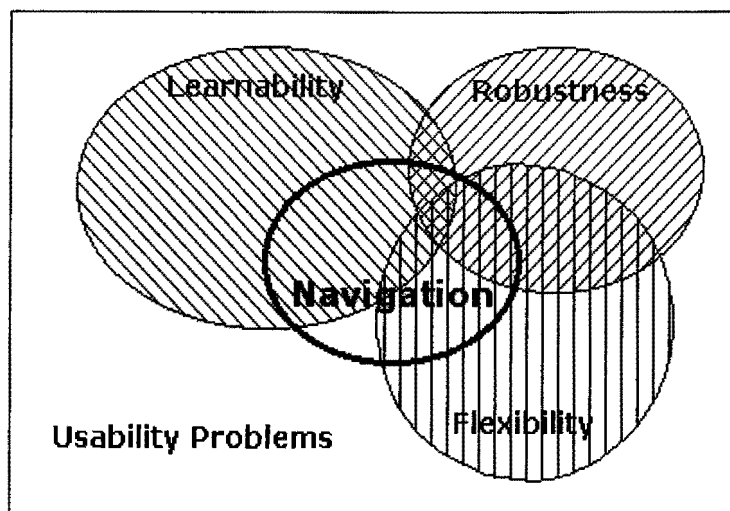
In support of this view, Borges, Morales and Rodriguez (1998) strongly suggest that heuristic guidelines should be based on the fundamental principles of UI design, but should also be translated into a simpler form usable by relatively inexperienced usability evaluators (as would be the case for most Web page designers). In addition Grose, Forsythe and Ratner (1998:121) proved that the use of

existing Web-specific style guidelines (which are directly usable as checklist-based evaluations), seldom cover more than 20% of the potential Web-applicable recommendations as found in traditional usability style recommendations. They (1998:129) correctly conclude that Web guidelines tend to emphasise the desire to overcome problems associated with technical constraints and the problems inherent in distributed, remote access, hypermedia systems (and also importantly an immature system), and neglect the other important general usability factors. It is noted (Mayhew, 1998:6) that Web authors and Web style guide authors are a varied group relative to traditional software developers. Many of them have not been exposed to the field of usability engineering and the available traditional style guides.

Mayhew (1998:7), continues and recommends that: '*Usability engineers must take traditional style guides, which have been rigorously developed using usability engineering techniques, and tailor them to be Web-specific – not by shortening them or even changing the basic principles, but by presenting new examples of the realization of these general principles on actual Web sites*'. This statement summarizes the approach that will be used here.

The previously discussed usability principles and properties can be presented as a number of *overlapping* guidelines sets for finding solutions in the surrounding usability problem space. Similarly the Web-specific guidelines as discussed here can be presented as another overlapping set cluster which superimposes differently on the first (general) set cluster. The term *superimpose* is used in its deliberate and penetrative sense – it is unfortunate that the interaction between usability properties in the Web application domain is often regarded in their *juxtapositional* sense.

**Figure 7.2 General and Web-specific usability guidelines in the usability problem space**





This view which is presented in figure 7.2, highlights another problem when choosing between general usability guidelines and Web-specific guidelines – they address usability issues from a different viewpoint. For example the important Web (and hypertext) aspect of *navigation* needs to be analysed in terms of visual clarity, user support/guidance, conceptual models, consistency, familiarity, predictability, synthesizability, control, observability and recoverability. (Note that the navigation set in figure 7.2 is intentionally drawn so as to indicate that the intersection of this set with learnability is larger than for the other two sets.) A similar observation can be made for most of the other Web-specific guidelines as presented here.

The solution chosen was to conduct the evaluation first on the basis of the general usability guidelines and then secondly and additionally, in terms of the more important Web-specific guidelines. This will ensure that those Web-specific aspects that were not covered by the general usability guidelines (unlikely!) are included. (These orphaned guidelines would be the complement of Navigation with the union of Flexibility, Robustness and Learnability in figure 7.2.)

In this regard Nielsen (URL: Useit2) refers to Dykstra (1993) and notes: *'In addition to the checklist of general heuristics to be considered for all dialogue elements, the evaluator obviously is also allowed to consider any additional usability principles or results that come to mind that may be relevant for any specific dialogue element. Furthermore, it is possible to develop category-specific heuristics that apply to a specific class of products as a supplement to the general heuristics. One way of building a supplementary list of category-specific heuristics is to perform competitive analysis and user testing of existing products in the given category and try to abstract principles to explain the usability problems that are found.'*

The previously mentioned important aspect of criteria *prioritisation* was handled by noting that during the evaluation those usability aspects that are important for the Web (such as navigation), tended to be *naturally* examined and discussed in more detail than the other usability aspects. Note that the relative size (area) of the guideline sets in figure 7.2 can be used to represent this aspect of prioritisation. The aspect of *Web-specificity* was handled by being careful to include all of the more important Web-specific heuristic aspects – as identified in this section – during the evaluation.

Finally table 7.9 then lists the categories of usability heuristics that had been used for Standard Bank's (and the other banks studied) Web site evaluation. The navigation properties presented in this table were extracted from Whitaker's (1998:69) ten principles for navigation design.

**Table 7.9 Experimental heuristics usability guidelines**

General Usability Principles and Properties		Web-specific Guidelines	
<b>Learnability</b>	Synthesizability	<b>Navigation</b>	Cues, prediction, landmarks, consistency, situational awareness, depth, user's purpose and proficiency, alternatives.
	Consistency		
<b>Flexibility</b>	Familiarity	<b>Aesthetic and minimalist design</b>	Dialogs, information relevance and visibility.
	Visual clarity		
	User support and Guidance		
	Predictability		
	Generalizability		
	Enjoyability		
	Interaction Metaphors		
	Conceptual Models		
<b>Robustness</b>	Dialog Initiative	<b>Match between system and the real world</b>	User's language, familiar concepts, natural and logical ordering of information
	Multi-threading		
	Task migratability		
	Substitutivity		
	Customizability		
	Control		
<b>Learnability</b>	Observability	<b>Consistency and standards</b>	Platform conventions, terminology.
	Recoverability		
	Responsiveness		
	Task conformance		
	WYSIWYG		

## 7.3 Checklist-based usability evaluations

This corresponds to the Bastien and Scapin (1995:184) category of expert judgement methods, as discussed previously. It is a subjective evaluation approach, and is closely related to the heuristic evaluation methods as examined in the preceding section. It is based on training, field experience, and an examination of human factors data. Once again the researcher acted as the sole evaluator – the pitfalls of which are discussed by Ravden and Johnson (1989:22,86) – but was careful to avoid rating bias – possibly introduced through aspects such as central tendency, leniency error, and cross-linking (Cooper et al., 1995). Some degree of bias will nonetheless unavoidably be present during the study.

### 7.3.1 Web-sourced checklist-based usability evaluations

It became clear during the study that existing online evaluations of Internet Web sites (including online banking sites as discussed in the next section), often employ criteria which could be fruitfully reshaped into a format suitable for use in a checklist. The more specific these criteria are the more suitable they were found to be for such a use. An example of this is the Borges and Morales (1998, URL: ACM1) usability list as presented earlier (table 7.6).

Another example is from Instone (1998, URL: W3J1), who lists under his 'Site Structure' criteria heading the following questions: (1) If users were taken directly to this page from an outside site, what would they figure out about the rest of the site from this one page? (2) Would it be clear how to navigate to other parts of the site? (3) Is it clear what part of the site they are in? (4) Is the site 'brand' clearly present? All four of these questions are appropriate for use in a checklist with only minor modification.

A similar comment applies to the Journal of Internet Banking and Commerce evaluation (as discussed in the next section): Questions such as 'rate the clarity/meaning of interactive messages on a scale from 1 (meaningless) to 10 (clear)', are eminently suitable for use in an evaluation checklist.

CNET's Webbuilder.com evaluation (1999, URL: CNET3) uses a rating from 1 to 5 for the factors **Usability, Navigation, Graphic Design, Content, Compatibility, Load Time, and Functionality**, in its checklist-based evaluation of Web sites. These are perhaps too general for use in an effective checklist-based evaluation but do have the advantage of being very quick and simple to use. The Pantos organization (1998, URL: Pantos1) emphasises the use of checklist Web site evaluations and provides a series of questions to be included on such a checklist (presented in table 7.10).

**Table 7.10 The Pantos organisation usability checklist for Web sites**

1	<b>Clarity of Communication</b> Does the site convey a clear sense of its intended audience? Does it use language in a way that is familiar to and comfortable for its readers? Is it conversational in its tone?
2	<b>Accessibility</b> Is load time appropriate to content, even on a slow dial-in connection? Is it accessible to readers with physical impairments? Is there an easily discoverable means of communicating with the author?
3	<b>Consistency</b> Does the site have a consistent, clearly recognizable 'look-&-feel'? Does it make effective use of repeating visual themes to unify the site? Is it visually consistent even <i>without</i> graphics?
4	<b>Navigation</b> Does the site use (approximately) standard link colors? Are the links obvious in their intent and destination? Is there a convenient, obvious way to maneuver among related pages, and between different sections?
5	<b>Design &amp; maintenance</b> Does the site make effective use of hyperlinks to tie related items together? Are there dead links? Broken CGI scripts? Functionless forms? Is page length appropriate to site content?
6	<b>Visual Presentation</b> Is the site moderate in its use of color? Does it avoid juxtaposing text and animations? Does it provide feedback whenever possible? (for example through the use of easily recognizable ALINK colors, or a 'reply' screen for forms-based pages)
Source: <a href="http://www.pantos.org/atw/35317.html">www.pantos.org/atw/35317.html</a>	

### 7.3.2 General Checklist-based Usability Evaluations

Two examples of empirically proven general usability checklist formats are those from Ravden et al. (1989) and Shneiderman (1992). (The Ravden checklist had been proven over a period of five years or more, on applications such as a transportation scheduling and control package, a project management information system, and the user interface to a large public-access database system.)

The aspect of question ‘ageing’ (Cooper and Emory, 1995:317), is however especially relevant in a rapidly evolving field such as a study of the human-computer interface – instruments designed specifically for a modern GUI would possibly have been more accurate.

**The Ravden and Johnson 1989 Usability Checklist:** Ravden et al. (1989) developed a comprehensive checklist and questionnaire for the qualitative evaluation of important usability aspects. (This checklist will be referred to hereafter as the Ravden checklist.) Their observation that practical and feasible evaluation methods are urgently needed to replace evaluations based on lists of design guidelines (1989:17) is important. They (1989:23) consider their checklist to be suitable for the comparison of existing systems and evaluating alternatives, as is the case here.

The usability aspects – which they refer to as usability criteria or goals (1989:19) – directly covered by their method are: (1) **Visual Clarity**, (2) **Consistency**, (3) **Compatibility**, (4) **Informative Feedback**, (5) **Explicitness**, (6) **Appropriate Functionality**, (7) **Flexibility and Control**, (8) **Error Prevention and Correction**, (9) **User Guidance and Support**, and (10) **General System Usability Problems**. These usability properties – which form a large subset of the general usability properties examined earlier – are evaluated in the form of a checklist consisting of about 160 questions divided into 10 sections.

Any modification of a well-designed checklist should be approached with caution – even if it only introduces page layout deviations rather than content changes. Scaling design and in general, research instrument design, is a specialist field and Ravden et al. (1989:23) warn against any nonresearched modification or augmentation of their instrument.

**The QUIS:** Shneiderman (1992:483-492) examines a related evaluation checklist (QUIS or ‘*Questionnaire for User Interface Satisfaction*’) – developed by the HCI Laboratory attached to the University of Maryland. This uses an extended – when compared to the Ravden checklist – 9 point rating scale, and it covers a partly different set of usability properties: learning, screen, terminology, system information, and system capabilities.

### 7.3.3 The experimental usability evaluation checklist

The Ravden and Johnson checklist had been designed some time ago for a general HCI usability evaluation. A study of the specific questions contained therein shows that perhaps unsurprisingly – it has been *proven* to be a well-constructed design – almost all of the questions are nonetheless applicable to a Web-based evaluation. This is especially true for a genuine goal-oriented application such as Internet banking. It was therefore decided – once again based on the author’s earlier experience (Van Dyk, 1996) in applying this list to a usability evaluation problem – to use the questionnaire with only minor modification. A single page 16 question section based on the Journal of Internet Banking and Commerce (URL: JIBC1) evaluation checklist (discussed in the next section), was added to the checklist evaluation. This additional checklist is specifically aimed at evaluating Internet banking applications and was the most comprehensive example found. The section guidelines for the Ravden checklist are presented in table 7.11. Because the checklist itself is a lengthy ten-page questionnaire it is presented as a (scored) appendix to chapter 9. The additional JIBC evaluation page is similar to table 7.12 as presented in the next section, and is also appended to chapter 9.

**Table 7.11 The Ravden usability questionnaire section guidelines.**

Section	Description
1	<b>VISUAL CLARITY:</b> Information displayed on the screen should be clear, well organized, unambiguous and easy to read.
2	<b>CONSISTENCY:</b> The way the system looks and works should be consistent at all times.
3	<b>COMPATIBILITY:</b> The way the system looks and works should be compatible with user conventions and expectations.
4	<b>INFORMATIVE FEEDBACK:</b> Users should be given clear, informative feedback on where they are in the system, what actions they have taken, whether these actions have been successful and what actions should be taken next.
5	<b>EXPLICITNESS:</b> The way the system works and is structured should be clear to the user
6	<b>APPROPRIATE FUNCTIONALITY:</b> The system should meet the needs and requirements of users when carrying out tasks.
7	<b>FLEXIBILITY AND CONTROL:</b> The interface should be sufficiently flexible in structure, in the way information is presented and in terms of what the user can do, to suit the needs and requirements of all users, and to allow them to feel in control of the system.
8	<b>ERROR PREVENTION AND CONTROL:</b> The system should be designed to minimize the possibility of user error, with inbuilt facilities for detecting and handling those which do occur; users should be able to check their inputs and to correct errors, or potential error situations before the input is processed.
9	<b>USER GUIDANCE AND SUPPORT:</b> Informative, easy-to-use and relevant guidance and support should be provided, both on the computer (via an online help facility), and in hard-copy document form, to help the user understand and use the system.
10	<b>SYSTEM USABILITY PROBLEMS:</b> When using the system, did you experience any problem i.t.o. the listed questions ?

Source: Ravden and Johnson (1989).

## 7.4 Existing Internet banking evaluation approaches

A number of Internet-based banking evaluation services are present on the Internet. The criteria used during these evaluations tend to be feature-oriented (i.e. what products, services and facilities are offered on the bank's Web-site), but it often includes other important usability properties. From these evaluations it is clear that factors such as Web site content, products and services, degree of interactivity (varying from informational/promotional to full customization approaches), customer relationship and support, and design (graphics and navigation), are usually regarded as being the most important categories.

### 7.4.1 The Journal of Internet banking and commerce checklist

The online-only (and fledgling) Canadian-based Journal of Internet Banking and Commerce (URL: JIBC1) offers an evaluation service from its Web site (URL: JIBC2). The criteria used during their evaluation (presented in table 7.12) are noteworthy because it is the most complete from a usability aspect of all the evaluation services studied. A graphical 10 point polar (categorical) ordinal scale rating design is used (Cooper and Emory, 1995:168).

**Table 7.12 JIBC Internet banking evaluation checklist**

No.	Criteria description	10 Point polar scale identifiers
1	Overall effort/commitment to create a functional online service	Low – High
2	Service speed	Slow – Fast
3	How often the system dies when online	Rarely – Often
4	Convenience of performing simple transactions through the Net	Not at all – Highly convenient
5	Visible evidence of service's independent security certification	Non existent – Displayed on the Homepage
6	Documentation/passwords send to prospective clients at the start of the service	Confusing – Clear
7	Simplicity/intuitive character of online forms and work processes	Confusing – Clear
8	Illogical interface deviations from the printed statements of the same bank	Substantial – Insignificant
9	Ease of help access through email	No access – Clear access points on every page
10	Clarity/meaning of interactive messages	Meaningless – Clear
11	On site explanations of banking terms	Inadequate – Adequate
12	Ease of help access through the voice call center	Incompetent, busy – Competent, easy to reach
13	Misbehaving/superfluous Java applets and scripts on the site	All-the-time, everywhere – Moderate
14	Graphics that are slow to download and are difficult to the eyes	Flashy, garish, moving, non-functional – Moderate
15	Advertising that is slow to download and is difficult to the eyes	Flashy, garish, moving, destructive – Reasonable
16	The likelihood that the bank's own CEO is personally using this service	Highly unlikely – Very likely

Source: [www.arraydev.com](http://www.arraydev.com).

A few problems in scaling design are evident from table 7.12. Items 12, 13, 14, and 15 are ambiguous and could present the rater with more than one conflicting choice – for example the help centre could be incompetent but yet easy to reach. Of more relevance is a comparison of the items with the usability criteria categories: Items 6, 9, 10, 11, and 12 are related to user support and guidance – item 10 also relates to visual clarity, generalizability, and predictability. Items 2, 14, and 15 relate to responsiveness and items 3 and 13 to robustness. Item 8 is related to familiarity; and items 4 and 7 relate to consistency and predictability. Usability properties that are not directly or sufficiently covered by this checklist include aspects such as customizability, control, dialog initiative, enjoyability, conceptual models, and synthesizability.

### 7.4.2 The Unisys Internet banking barometer

Europe-based Unisys (an information service company) offers a limited evaluation service from their extensive Internet-banking news, information and support Web site (URL: Unisys1). A number of European-only banks are evaluated on the site using criteria such as **Graphics** (light, intensive, high quality, animated etc.), **Technical Features** (Java applets (animation, security, service-support etc.), Java script, Cookies, Frames, Forms, PDF documents, CGI scripts, Shockwave etc.), **Content** (informational, customer contact, presentation, promotional, links etc.), **Products offered online**, as well as a general **Comment** section – the latter often contains the major part of the banking site’s interaction and usability information. Figure 7.3 is a screen dump of a typical evaluation from this site – in this case for the Royal Bank of Scotland (one of the other banks evaluated in this study).

Figure 7.3 Royal Bank of Scotland evaluation by Unisys: pages 1 and 2

Site Address	<a href="http://www.royalbankscot.co.uk">http://www.royalbankscot.co.uk</a> or <a href="http://www.rbos.co.uk">http://www.rbos.co.uk</a>
Content	Full Internet direct banking for PC's with Internet Explorer 3.0
Graphics	Elaborate animated Graphics
Technical Features	Extensive use of ActiveX, Online transactions possible, 24 hours a day, transfer to Mr Money available
Products Offered Online	Full Internet Banking
Links to Shopping Malls	None
Comments	This is the first real Internet banking service in the UK. However, it is only aimed at PC users. Mr Explorer 3.0 is a requirement for using this service

Additional information can be obtained from the Webmaster - Email: [webmaster@unisys-finance.com](mailto:webmaster@unisys-finance.com)

Copyright ©1996, Unisys Ltd. All Rights reserved.  
 Unisys Ltd. is a subsidiary of Unisys Corporation

Source: [www.internet-banking.com](http://www.internet-banking.com)

### 7.4.3 The CNET evaluation

CNET is an American-based news and technology Web site that offers an extensive guide to online banking in addition to conducting regular evaluations on USA-based banks. Their checklist (URL: CNET1) for choosing an online bank is however firmly feature-oriented – it includes functions such

as a provision to transfer funds, availability of (other) customer services, investment services, previewing and ordering cheque books, the use of custom security and banking software, and obtaining electronic statements. Their (unfortunately Web browser biased) evaluation (URL: CNET2), for the Bank of America (BOFA) – one of the additional banks studied here – reads as follows:

‘The Bank of America Web site currently offers economic forecasts, online credit card applications, and a popular mortgage finder feature that helps you determine how much house you can afford. In the late spring, B of A will roll out a full array of services: you’ll be able to transfer funds between accounts, pay bills, access accounts to check your balances, and see your bank statements online. These same services are offered to small business. Since more than 50,000 merchants have signed up for B of A’s check paying service, chances are good that almost everyone to whom you owe money will be able to drain your wallet electronically. B of A will deal with Net security by requiring that customers use the latest version of Netscape Navigator, which employs encryption to safeguard sensitive data.’

Source: [www.cnet.com/Contents/Features/Dlife/Banking/](http://www.cnet.com/Contents/Features/Dlife/Banking/)

#### 7.4.4 AOL and the bank rate monitor

AOL (America Online) in addition to being the largest Internet access and service provider in the world with about 15 million subscribers (based on an early 1999 figure), also physically hosts the online banking sections for two large American banks namely BOFA (Bank of America), and First Chicago. AOL’s guidelines for choosing an online bank include criteria such as: ‘*What fees are associated with an online account? How long does it take transactions to clear? What type of encryption do you use and how does it keep my money safe? What should I do in the event of a technical problem, or duplicated transaction? How difficult is it to close an online banking account?*’ (URL: AOL1). In addition they offer an extensive evaluation of a number of Internet banks through Bankrate.com’s Bank Rate Monitor service (URL: AOL2). The latter uses the following criteria categories for their evaluation of each bank’s Web site (presented in table 7.13). During the evaluation a rating out of 20 is given to each of the five groups for a maximum score of 100/100.

**Table 7.13 AOL’s Bankrate.com review criteria**

1	<b>Products and services</b>	Does the bank offer online banking – what services are available? Does the site do an adequate job of explaining security issues with online banking? Are descriptions of other products and services clear? Does the bank offer any online specials? Are fees for services listed?
2	<b>Interactivity</b>	Are there any calculators? Online applications? How long did it take the bank to respond to e-mail questions?
3	<b>Design</b>	Are the graphics attractive and understandable? Is the background attractive?
4	<b>Content</b>	Is the written content easy to understand? Does it address genuine banking needs? Is there information about the company?
5	<b>Navigation</b>	How user-friendly is the Web site? Are the links easy to find and access? How long does it take to move from one page to another?



Note that some of the more important usability issues are included, but arranged differently. Their evaluation for the Bank of America comments on the intuitiveness of Web site icons, relevant content, navigational clarity via sub-headings, and the effective use of colour. Their criteria groups are indirectly related to the set of usability properties: Task conformance, conceptual models, and user support via products and services. Enjoyability and user support through interactivity. Visual clarity, interaction metaphors, and user support via design. Familiarity, task conformance, user support, and generality through content, and lastly predictability, user support, consistency, and responsiveness via navigation.

### 7.4.5 MWeb-hosted Internet banking Survey

A large South African ISP (MWeb) carried an invitation to participate in an Internet banking survey in a prominent position (figure 7.4.1) on their home page during June 1998. The extensive (15 page) and wide range of questions covered in the survey are listed in table 7.14 and are grouped according to the page sequence. The MWeb survey invitation page and the first two pages of the survey (also shown in table 7.14), are presented as figures 7.4.1, 7.4.2, and 7.4.3 below.

Figure 7.4 Internet banking survey hosted by MWeb

Figure 7.4.1 Survey invitation

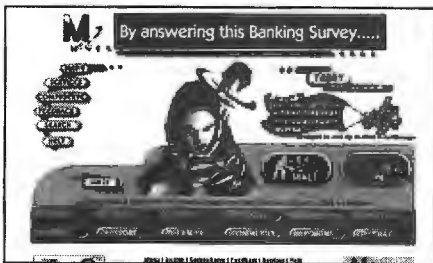


Figure 7.4.2 Survey page 1

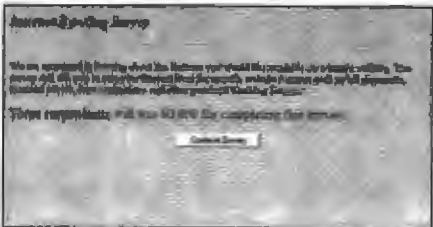
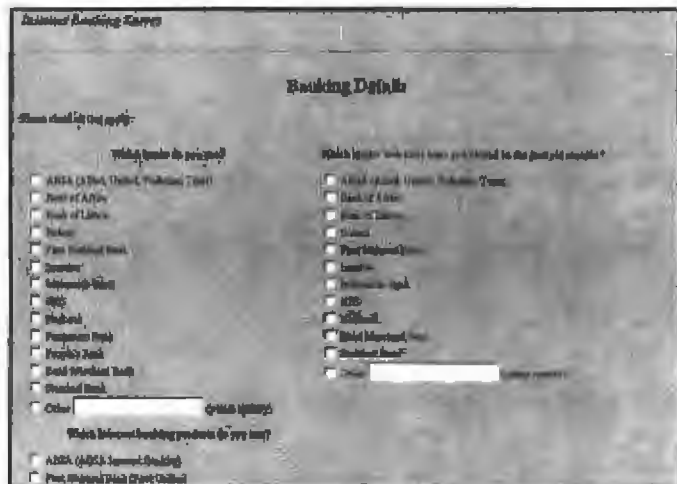


Figure 7.4.3 Survey page 2



Source: MWeb June 1998 (www.mweb.co.za).

Table 7.14 MWeb-hosted banking survey

Page	Survey Topic	Rating
1	1.1 Which banks do you use? 1.2 Which bank's Web site have you visited in the past six months? 1.3 Which Internet banking products do you use?	List of South African banks, other. List of South African banks, other. Absa, Mercantile, Nedbank, First-National, Standard Bank.
2	Rate each Web site visited.	1 Poor to 5 Excellent
3	Rate the Internet banking products used.	1 Poor to 5 Excellent
4	Indicate your interest in the following Web banking features. 4.1 Branch and ATM locator. 4.2 Stock market data (JSE only). 4.3 Loan and Account interest rates. 4.4 Links to related pages of interest. 4.5 Financial services search engine. 4.6 Site search engine.	5 Point scale:  1 Would use/access 2 3 Might use/access 4 5 Would definitely use/access
5	Rank the features you are interested in (from page 4).	1 Most important, 2 etc.
6	Rate your interest in the following interactive features. 6.1 Chat area. 6.2 E-mail comments and queries to bank representatives. 6.3 Audio link to personal banker. 6.4 Video link to personal banker. 6.5 Multimedia financial planning guide. 6.6 Financial market simulation game. 6.7 E-mail reminders, product information, news based on your personal profile. 6.8 On-line pager function to notify you of new e-mail, products etc. 6.9 Free e-mail account.	5 Point scale used:  1 Would use/access 2 3 Might use/access 4 5 Would definitely use/access
7	Rank the interactive features you are interested in (from page 6).	1 Most important, 2 etc.
8	Rate your interest in using the following bank product applications on your bank's Web site. 8.1 Current and Savings account applications. 8.2 Home loan applications. 8.3 Car loan application. 8.4 Credit card applications. 8.5 Insurance applications. 8.6 Cellular phone applications. 8.7 ATM applications.	5 Point scale:  1 Would use/access 2 3 Might use/access 4 5 Would definitely use/access
9	Rank the application features you are interested in (from page 8).	1 Most important, 2 etc.
10	Rate your interest in having access to the following statements on your bank's Web site. 10.1 Loan statements 10.2 Insurance account statements. 10.3 Cellular phone statements. 10.4 Home telephone bill (e.g. Telkom). 10.5 Home electricity bill (e.g. Eskom). 10.5 Home subscription TV bill (e.g. MNET).	5 Point scale:  1 Would use/access 2 3 Might use/access 4 5 Would definitely use/access
11	Rank the statement features you are interested in (from page 10).	1 Most important, 2 etc.
12	Rate your interest in the following transaction services on your bank's Web site. 12.01 Pay bills. 12.02 View/download transaction history. 12.03 Access frequent payee lists for quick payment of common bills (Eskom etc.). 12.04 Initiate debit orders. 12.05 Set up future dated payments. 12.06 Stop payments on cheques. 12.07 Purchase units trusts. 12.08 Trade JSE shares. 12.09 Order travelers cheques. 12.10 Book airline travel. 12.11 Shop online using credit card.	5 Point scale:  1 Would use/access 2 3 Might use/access 4 5 Would definitely use/access
13	Rank the transaction services you are interested in (from page 12).	
14	Are there any features or functions that have not been identified in the previous sections which you would like have available on your bank's Web site?	Comment box provided.
15	Personal details. 15.1 How old are you? 15.2 Are you 15.3 What is your home language? 15.4 What is your personal monthly income? 15.5 How much time do you spend on the Internet each week? 15.6 Where do you access the Internet? 15.7 How experienced at using the Internet do you consider yourself? 15.8 Would you be interested in home access to the Internet through your TV? 15.9 If your bank offered dial-up Internet access would you be willing to switch your Internet Service Provider?	Pull-down list age ranges. Select from: Male or Female Pull-down list Pull-down list income ranges. Pull-down list ranges. Select from: At home or/and At work. Pull-down list. Yes/No Yes/No

The survey is again mainly feature-orientated, but is nonetheless notable because of the following reasons. It shows a highly professional design, it demonstrates the South African financial institution's (and ISP's) commitment to, or at least interest in, Internet banking, and it is very thorough in its coverage of desirable online banking features. Other usability aspects take a back seat – the only questions that could conceivably impact on these more important usability properties are questions 2 and 3 which are aimed at a general rating of the Internet Web site product.

### 7.4.6 Standard Bank's survey

Standard Bank hosts a survey permanently from their own Internet banking site. The questions are all access and platform related, and are presented here as the four pages shown in figures 7.5.1 to 7.5.4. Standard Bank has perhaps missed an opportunity to collect usability information pertinent to their own Web site – it is a cause for concern that the questions indicate that the bank is probably more concerned with technology issues than with potential usability problems.

Figure 7.5 Standard Bank's online questionnaire: pages 1, 2, 3, and 4

Figure 7.5.1 Page 1

Standard Bank Internet Banking Survey	
Dear Internet Banker	
Standard Bank conducts ongoing research with its Internet Banker customers to ensure that a quality service is delivered and maintained. Please participate in the following survey so that we can better cater for your needs in the future. Please click on the appropriate block.	
Full name	<input type="text"/>
Service provider	<input type="text"/>
E-mail	<input type="text"/>
1. What PC do you currently use?	
a. 486 or less	<input type="radio"/>
b. Pentium	<input type="radio"/>
c. Pentium II	<input type="radio"/>
d. Apple Macintosh	<input type="radio"/>
e. Unix	<input type="radio"/>
2. If you are not currently running a Pentium or higher PC, are you planning to upgrade or switch?	
a. No	<input type="radio"/>
b. Yes, within 3 months	<input type="radio"/>
c. Yes, within 6 months	<input type="radio"/>
d. Yes, within 18 months	<input type="radio"/>

Figure 7.5.2 Page 2

3. How much RAM are you currently running on your system?	
a. Less than 6MB	<input type="radio"/>
b. 6MB	<input type="radio"/>
c. 16MB	<input type="radio"/>
d. 32MB and above	<input type="radio"/>
4. Are you accessing the Internet:	
a. Via dial-up modem	<input type="radio"/>
b. Via a network	<input type="radio"/>
c. Don't know	<input type="radio"/>
5. If accessing via modem, what is your modem speed?	
a. 11.4kbs	<input type="radio"/>
b. 28.8kbs	<input type="radio"/>
c. 33.6kbs	<input type="radio"/>
d. 56kbs	<input type="radio"/>
6. If accessing via a network, is your network protected by a firewall or similar security?	
a. Yes	<input type="radio"/>
b. No	<input type="radio"/>
c. Don't know	<input type="radio"/>

Figure 7.5.3 Page 3

7. What is your preferred Internet browser?		
a. Microsoft Internet Explorer	<input type="radio"/>	
b. Netscape	<input type="radio"/>	
c. Other	<input type="radio"/>	
8. If you are using Microsoft Internet Explorer version 3.x or Netscape Navigator version 3.x, will you be upgrading to version 4 that is currently available?		
a. No	<input type="radio"/>	
b. Yes, within 3 months	<input type="radio"/>	
c. Yes, within 6 months	<input type="radio"/>	
d. Yes, within 18 months	<input type="radio"/>	
9. Are you using Microsoft Windows 3.1 or Microsoft Windows for Workgroups 3.11?		Yes <input type="radio"/> No <input type="radio"/>
10. If yes, when will you upgrade to a newer version of Windows?		
a. Not at all	<input type="radio"/>	
b. Yes, within 3 months	<input type="radio"/>	
c. Yes, within 6 months	<input type="radio"/>	
d. Yes, within 18 months	<input type="radio"/>	

Figure 7.5.4 Page 4

11. What will you upgrade to?		
a. Windows 95	<input type="radio"/>	
b. Windows 98	<input type="radio"/>	
c. Windows NT	<input type="radio"/>	
12. If you are using Apple Macintosh, OS2 or any other non-Windows operating system, are you planning to switch to a Windows platform?		
a. Yes	<input type="radio"/>	
b. No	<input type="radio"/>	
13. Where do you use Standard Bank's Internet Banking service?		
a. At home	<input type="radio"/>	
b. At work	<input type="radio"/>	
c. Both	<input type="radio"/>	

Source: [www.sbic.co.za](http://www.sbic.co.za)

## 7.5 Conclusion

Both a heuristic evaluation and the related checklist-based usability evaluation are suitable instruments for a Web page usability evaluation. Existing (Internet-based) evaluation services tend to be feature (task-support) oriented and neglect other more important usability aspects. Although guidelines designed specifically for Web site evaluation do have merit – especially for the usability-naïve Web site designer – it is considered more rigorous to base the evaluations on the general usability standards, principles, and properties identified earlier. These should then be expanded by the carefully selected addition of Web-specific guidelines. This hopefully synergistic combination will be used as the framework for conducting the heuristic evaluation as described in chapter 8.

For similar reasons the general usability evaluation checklist developed by Ravden et al. (1989), in conjunction with the Journal of Internet Banking and Commerce (URL: JIBC1) Internet-banking evaluation checklist, will form the basis for the checklist-based evaluation conducted in chapter 9.

The post-evaluation measures as described by Levi and Conrad (1996) are impressive. Their assignment of severity ratings to usability violations appears to work well, and yields important additional qualitative data. It also partly solves a difficult problem during heuristic inspections namely the issue of comparison against an objective qualitative baseline or standard.

## Chapter 8

# Heuristic Usability Evaluation

**This chapter describes the application of a heuristic usability evaluation to a selection of local and international Internet-based banking Web sites.**

## 8.1 Introduction

This chapter presents the results of applying a heuristic evaluation to four of the local Internet-banking enabled banks (as in 1998), and also to a selection of mainly United States-based, Internet banking sites. Because Standard Bank (South Africa) was the only bank that was accessed as a full client as opposed to limited access through a banking demonstration facility, the main focus of the evaluation had been on this bank's Web site. The other banks were used to generate material supplementary to this main evaluation and are not extensively evaluated individually. The evaluation required capturing interaction behaviours through a large number of screendumps that are presented as chapter appendices A through to I.

The basis for the heuristic evaluation had been previously discussed and presented (chapter 7). The guidelines used are the three categories of general usability principles (*learnability*, *flexibility*, and *robustness*) and their properties (such as consistency, user guidance, and task conformance). These three categories are supplemented by four Web-specific guideline groups namely *navigation* issues, the use of *aesthetic* and *minimalist design*, the degree of *matching* between the *interactive system* and the *real world*, *consistent appearance* and *behaviour*, and the adherence to, and implementation of, accepted (formalised or de facto) Web-related *standards*.

## 8.2 Data collection methods for the heuristic evaluation

With the exception of Standard Bank, information for all the other banks had been collected through the use of their demonstration facilities. Where the latter facility was absent the system was examined up to the point where an actual banking client identification and password were required. The demonstration facilities encountered were different in their approach and sophistication. They varied between the presentation of a simple sequence of screendumps taken from the live online banking system to a full interaction-enabled simulation of the actual banking session. The Royal Bank of Scotland (1998), Absa Bank (1997-1998), Security First Network Bank (1998), Wells Fargo Bank (1998) and Standard Bank (1999) are examples of the first (passive) class of demonstration. Citibank (1998), Bank of America (1998), BankOne (1998), Standard Bank (1997-1998), Absa Bank

(1999), Nedbank (1997-1999), and First National Bank (1997-1999), are examples of the full interaction demonstration class.

Standard Bank reduced the learnability-usability of its Web site when it changed its fully interactive demonstration facility (available during 1997 and 1998), to an enhanced passive screendump-based presentation during early 1999. There is however an opinion that these demonstration facilities are not extensively used by existing clients to learn about the system but should rather be viewed as a promotional resource aimed at the incidental browser. It is likely that once a novice user has crossed the initial hurdle of making a transaction of one specific type at least once, he/she is unlikely to turn to the demonstration for further guidance – this guidance would be more effectively provided through onscreen help and information page options and links. In addition it was noticed that many of the interactive demonstration facilities were not synchronised with changes and improvements to the live system – they did not always accurately reflect updates made to the banking Web sites. Using a passive screen presentation system does have the advantage of less demanding maintenance and enables a quicker and simpler mirroring of the current live system.

**Table 8.1 List of heuristically evaluated subject banks**

Bank	HQ	Appendix	Demonstration facility type
Standard Bank South Africa	South Africa	8.E	Full interaction 1997-1998 Enhanced passive screendump 1999
First National Bank (FNB)	South Africa	8.F	Full interaction 1997-1999
Nedbank	South Africa	8.G	Full interaction 1997-1999
Amalgamated Banks of South Africa (ABSA)	South Africa	8.H	No demonstration facility 1997-1998 Full interaction 1999
Mercantile Bank (SA)	South Africa	8.I	No demonstration facility 1997-1998
Citibank	United States		Full interaction 1997-1998
Bank of America (BOFA)	United States		Full interaction 1997-1998
Bank One	United States		Full interaction 1997-1998
Security First Network Bank (SFNB)	United States		Enhanced passive screendump 1998
Royal Bank of Scotland (RBS)	Scotland		Simple passive screendump 1998
Wells Fargo Bank	United States		Enhanced passive screendump 1998

The sets of screen capture images that will be used during the evaluation are appended as five sections (appendices 8.E through to 8.I). Capture to local storage as discussed earlier (chapter 5) formed the sole basis for the evaluation and the appended screen images for the six international banks. For the four local banks image captures generated from local storage were supplemented with material from live sessions (as was the case for Standard Bank), and more recent material (up to April 1999) for Standard Bank, First National, Nedbank, and Absa. Mercantile Bank (SA) is only included in the study because it is a local bank (although a subsidiary of an international bank), that

also offers banking through the Internet. It requires custom software that is non-Web-browser-based, has no demonstration facility (1998), and added little of substance to the pool of evaluation data.

The range of transactions covered by the appended screen images are a banking session sign-on (or logon), a session logoff – (logically) seldom available from the demonstration facility, transaction option pages and their selection, account information (balances and statements), account transfers and payments, client services, and help facilities. The banks included in the evaluation are listed in table 8.1.

### 8.2.1 Standard Bank

Standard Bank offered two types (and three versions), of demonstration facilities from their banking site during the period 1997-1999. An initial (1997-1998) fully interactive demonstration facility was replaced by a simpler animation-enhanced passive screen image sequencing (slideshow) facility during 1999. The enhancement of the latter is primarily aimed at offering the user clues as what to press (or more correctly where to click), to continue to the next image presentation. Two versions of the earlier interactive demonstration existed, and were captured and presented here. These two versions (approximately) correspond to accompanying Web site live banking updates.

Screen images for both types of demonstration are presented in appendix 8.E. The first (1997) version of the fully interactive facility is represented by figures 8.E.5 through to 8.E.14. The second (1998) version of the interactive demonstration facility is represented by figures 8.E.16 to 8.E.24. Figures 8.B.9 to 8.B.11 (from appendix B) show screen captures from the (1999) passive demonstration facility. Standard Bank is the only bank that was accessed as a full user or banking-client, rather than a guest or interested browser. One advantage attached to this full access was that it was possible to observe differences between the demonstration and the actual live site. Examples of this difference are the noticeable difference in response times, the ‘double’ logoff present during the live banking session and the (expected) absence of transaction-induced error messages during the demonstration session. Figures 8.E.15, and 8.E.25 to 8.E.48 present screen captures from the live (actual) use of the Internet banking facility. The material as presented in appendices A (errors), B (user guidance and help), C (security), and D (page components), serves to augment these other live session captures.

Appendix E has five groups of screen dumps (table 8.2), which are summarized by the four simplified navigational sequence diagrams (appendix 8.E, figures 8.E.1 to 8.E.4):

The following is an overview of Standard Bank’s more important Internet banking Web pages, starting with navigation from the bank’s home page to the banking facility.

**Table 8.2 Standard Bank Internet banking Web site: versions A, B, and C**

(1)	<b>Web site version A. May to October 1997. Navigation diagram E.8.1. Screen dumps: figures 8.E.5 to 8.E-10.</b> This is a <b>demonstration run</b> using Microsoft Internet Explorer 3 and Windows 95 showing a few simple tasks namely signing-on (login), obtaining an account statement, transferring funds from one account to another, and signing-off (logoff). This section represents the initial (first) version of Standard Banks Internet banking Web site.
(2)	<b>Web site version A. September 1997. No navigation diagram. Screen dumps: Figure 8.E.15, 8.A.1 and 8.A.2.</b> These are screens dumps from the <b>actual use</b> of the banking facilities using Netscape Navigator 3 under Windows 3.11, and using Nanoteq's security software (*). The three screens show two error conditions and the logoff screen. This shows the same version of the Web site as in section (1).  <i>* Nanoteq – a Centurion-based networking security company in the then Persetel-QData (PQData) group of companies (now Compaex), developed Standard Bank's special security software.</i>
(3)	<b>Web site version B. November 1997 to October 1998. Navigation diagram 8.E.2. Screen dumps: Figure 8.E.16 to 8.E.20.</b> This is also a <b>demonstration run</b> using Microsoft Internet Explorer 3 and Windows 95 showing mainly the same tasks as in (1). This represents an updated Web site from the one shown in section (1) and (2).
(4)	<b>Web site version C. November 1998 to January 1999. Navigation diagram 8.E.3. Screen dumps: Figure 8.E.25 to 8.E.32.</b> This represents the site during actual use using Microsoft Internet Explorer 4.01 and Windows 98. This is the third version of the Web site.
(5)	<b>Web site version C.1. January 1999 to April 1999. Navigation diagram 8.E.4. Screen dumps: Figure 8.E.33 to 8.E.35.</b> This represents the site during actual use using Microsoft Internet Explorer 4.01 and Windows 98. This is a slightly modified (as shown by figures 8.B.8 to 8.B.10) third version C Web site as identified in section (4).

### 8.2.1.1 Standard Bank's home page

This is represented by figures 8.E.5, 8.E.25, and 8.E.33/34 in appendix 8.E. This is the bank's main welcoming page from where the user can choose to navigate to the Internet banking facility. Initially (version 1), this was done by clicking on the second tab in the left-hand frame (below the home shaped button representing the home page). Other elements on this early page include two advertising banners (used for in-house advertising and information only), and three buttons (Search, Index, and Contact) at the bottom of the screen. The left-hand frame contains buttons for the Home Page, Internet Banking, Personal, Business and International Services, What's New, and About Standard Bank. This early 1997 home page is shown in figure 8.1.

During November 1997 the left-hand information banner displayed a message (shown in figure 8.8), to the effect that *the 'users of Standard Bank's Internet facilities wanted less graphics, and they now have got it'*. This coincided with the launch of the bank's new (improved) and less graphics intensive site (October 1997). This seems to confirm that Standard Bank do receive and then also act, on feedback from their banking users.



Figure 8.1 Version 1: September 1997 Standard Bank home page

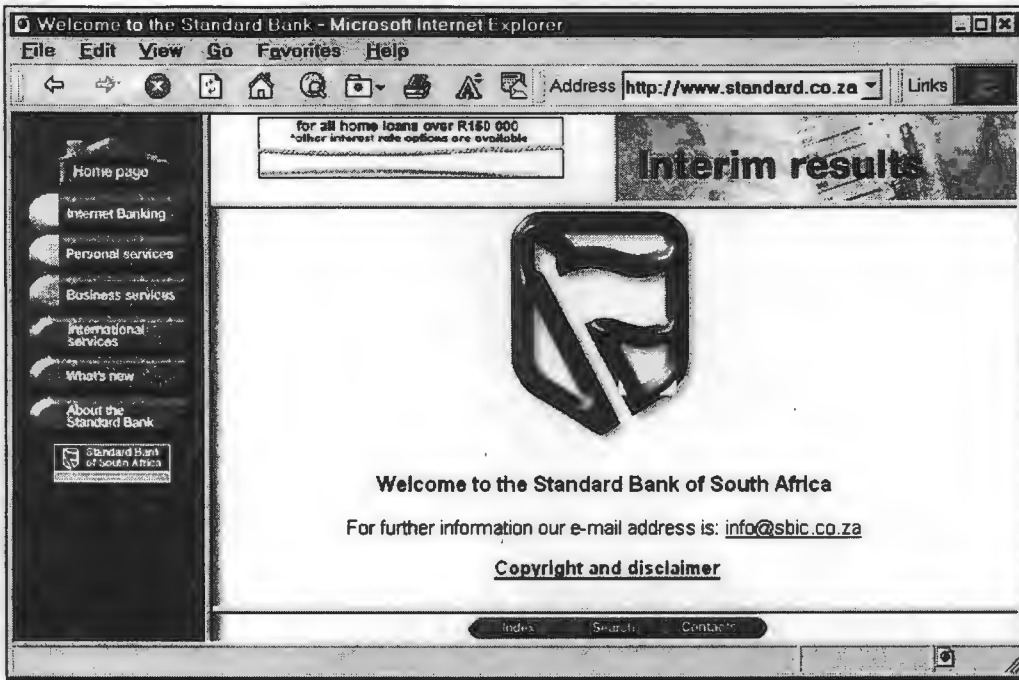


Figure 8.2 Version 2: May 1998 home page



Figure 8.3 Version 3: November 1998 home page



Figure 8.4 Version 4: January 1999 home page

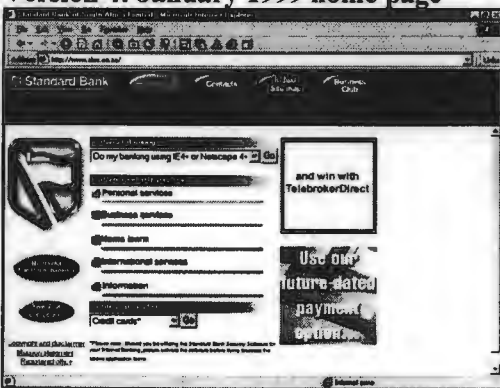
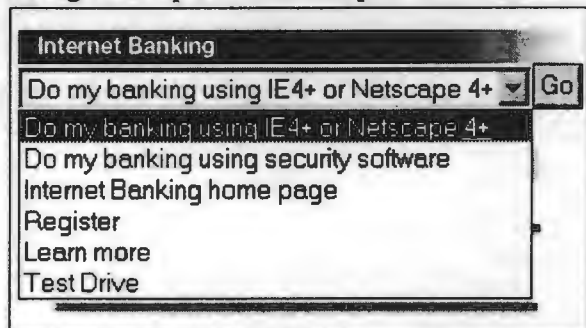


Figure 8.5 Navigation option from the pull-down menu



The buttons (figure 8.7) used on the main screen appear strange looking – they seem to be similar to the type of paddle switch (‘stops’) used in older music organs. Importantly they do not give visual feedback during focus (mouse-over), activation (press/click/selection) and post-activation. On the older site version these buttons were (at times inappropriately), present all the time during a secure transaction (as a separate left-hand frame) – i.e. users could have thought that they had an option to select functions from both secure and insecure pages. The use of the home-shaped button (figure 8.6) for navigation back to the home page is intuitively correct – but the icon itself appears fuzzy and unclear in its appearance. The button for Internet banking could perhaps have been made more prominent, to attract attention to the (then) new service. Its placement should have been above the home page icon.

**Figure 8.6**  
Home page button

**Figure 8.7**  
Internet banking button

**Figure 8.8**  
User feedback acknowledgment



Source: www.sbic.co.za (September 1997).

Between October 1997 and January 1999 Standard Bank’s home page went through some changes but these were mainly concerned with the addition of links. In January 1999 a significant navigational change was made when a number of direct links to the Internet banking facilities were made available by opening a pull-down menu – this is also called a (form) selection menu (Stanck, 1996:378) – as shown in figure 8.5. The default option shown links directly to the Internet banking ‘terms and conditions’ page (i.e. the pre-sign-on page). This option bypasses the Internet banking home page and it therefore is a one-page navigation saving. It would be more efficient to link directly to the sign-on page from the home page (i.e. saving another one page navigation). Figure 8.4 presents the current (April 1999) page that may be compared to the two versions of the 1997/1998 period home pages as shown in figures 8.2 and 8.3.

**8.2.1.2 Standard Bank’s Internet banking home page**

Figures 8.15 to 8.18 are screen dumps for the four versions of Standard Bank’s Internet-banking home page during the period 1997 to 1999. (Note that the previously discussed updates and changes to the bank’s home pages do not necessarily correspond to changes in the Internet banking home page.) There are quite significant differences between these four versions – especially when comparing the first (September 1997) and last (January 1999) versions.

The first and third versions used frames – a right-hand vertical frame contained the various options in the September 1997 version, and the third (May 1998) version made use of a horizontal upper frame. The current version does away with the use of frames (table 8.3). It should be remembered

that the current site has as its default option a link from Standard Bank’s home page that bypasses this page. This implies that the page discussed in the next section – the ‘Terms and Conditions’ page (i.e. the pre-sign-on page) is now more important (it has increased access).

The various options are presented at the top as two buttons for doing the banking – one for direct access banking using version 4 browsers and the other using security software combined with version 3 browsers, and various help and support options on the left-hand side. The Internet banking related options available from these pages have increased from 7 in September 1997 and 6 in November 1997, to 12 in May 1998 and 10 in January 1999. These options are listed in table 8.3.

**Table 8.3**  
**Options and frames for the four versions of Standard Bank’s Internet banking home page**

Version 1	Version 2	Version 3	Version 4				
<table border="1"> <tr> <td>Frame 1</td> <td>Frame 2</td> </tr> </table>	Frame 1	Frame 2	No Frames	<table border="1"> <tr> <td>Frame 1</td> </tr> <tr> <td>Frame 2</td> </tr> </table>	Frame 1	Frame 2	No Frames
Frame 1	Frame 2						
Frame 1							
Frame 2							
<ul style="list-style-type: none"> <li>Getting started</li> <li><b>Do banking</b></li> <li>Terms and Conditions</li> <li>FAQ</li> <li>Security software download</li> <li>Troubleshooting</li> <li><b>Demo</b></li> </ul>	<ul style="list-style-type: none"> <li>Terms-Conditions</li> <li><b>Demo</b></li> <li>Help</li> </ul>	<ul style="list-style-type: none"> <li>Getting started</li> <li><b>Do banking</b></li> <li>FAQ</li> </ul>	<ul style="list-style-type: none"> <li>Security SW</li> <li><b>IE4 banking</b></li> <li><b>Banking the new way</b></li> <li>Terms-Conditions</li> <li><b>Demo</b></li> <li>Getting started</li> <li>System requirements</li> <li>FAQ</li> <li>Charges</li> <li>Help with security software</li> <li>Error messages</li> <li>Internet package access</li> </ul>	<ul style="list-style-type: none"> <li>Security SW</li> <li><b>IE4 banking</b></li> <li>Learn more</li> <li>FAQ</li> <li>Charges</li> <li>Terms-Conditions</li> <li>Internet package access</li> <li><b>Test drive</b></li> </ul>			
<p><b>Notes:</b> There are navigation options displayed which are not shown here. For example all four versions have links back to Standard Bank’s home page. Versions 3 and 4 have a link to a registration page. Version 4 also has a number of information links displayed lower down the page such as ‘Common questions’.</p>							
<p>Source: www.sbic.co.za (1997, 1998, and 1999).</p>							

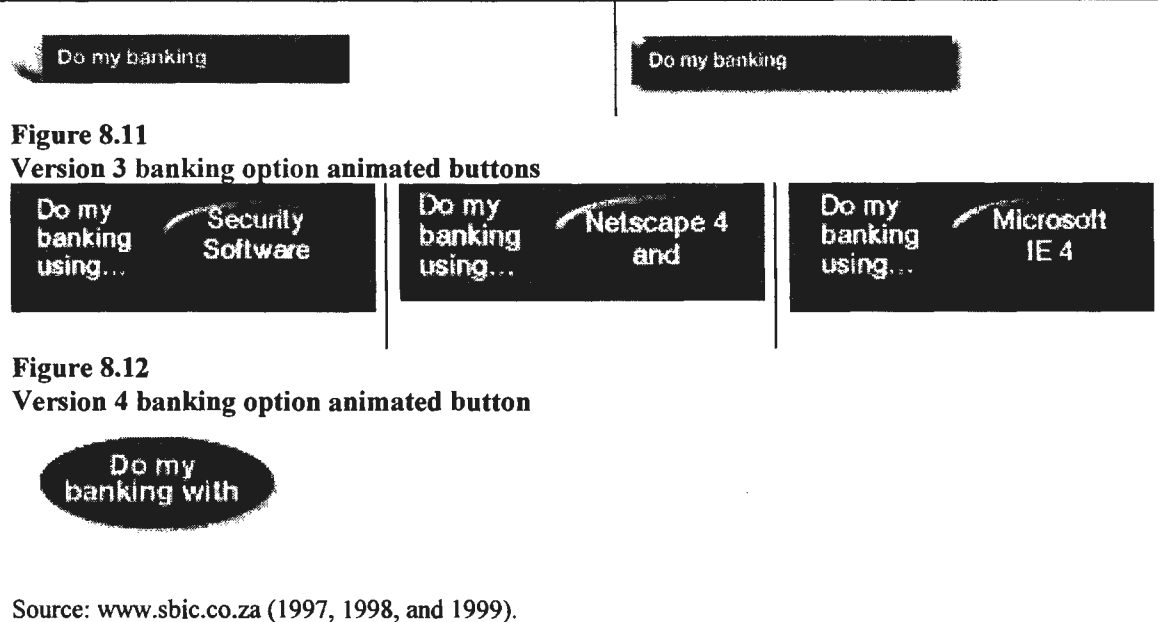
As expected – because the use of version 4 browsers became more common – the options related to security software (download and help), have steadily decreased in prominence and are absent from the 1999 version. The link to the demonstration facility is present in all four versions but the labeling has changed from ‘Demo’ in the first three versions to ‘Test drive’ in the current version which shows an American influence – all the American banks examined use the latter label.

Significantly the various information options from the previous three versions were replaced with a link called ‘Learn more’ in the fourth version. The third and current versions included direct links to a banking-charges information page that were absent from the first two versions. The link to Standard Bank’s home page is present in all four versions with the second version showing it in the

most prominent position. The static graphics buttons in the first two versions linking to the next (normal) navigational step – doing the banking – have been replaced by animated oval buttons placed in a more prominent position at the top of the page in the third and fourth versions – this attracts the user attention to these options and also minimises the mouse distance traveled to select it. The four button types are shown in figures 8.9 to 8.12.

**Figure 8.9**  
Version 1 banking option static button

**Figure 8.10**  
Version 2 banking option static button



The home page button in the previous version is replaced with a very prominent button in the second version (figure 8.13). The third and fourth versions increasingly reduced the prominence of the home page button – in the current version it is an option in the lower left-hand side (figure 8.14).

**Figure 8.13** Version 2 prominent home page navigation button

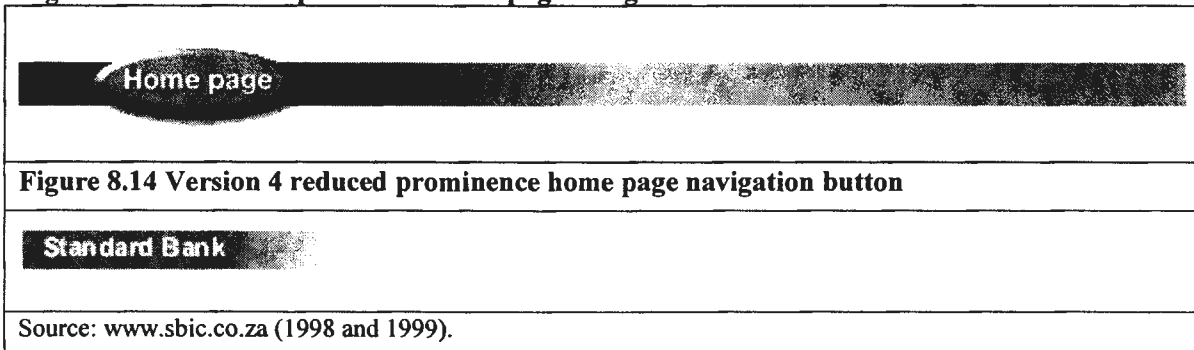


Figure 8.15 Version 1: September 1997 Standard Bank Internet banking home page

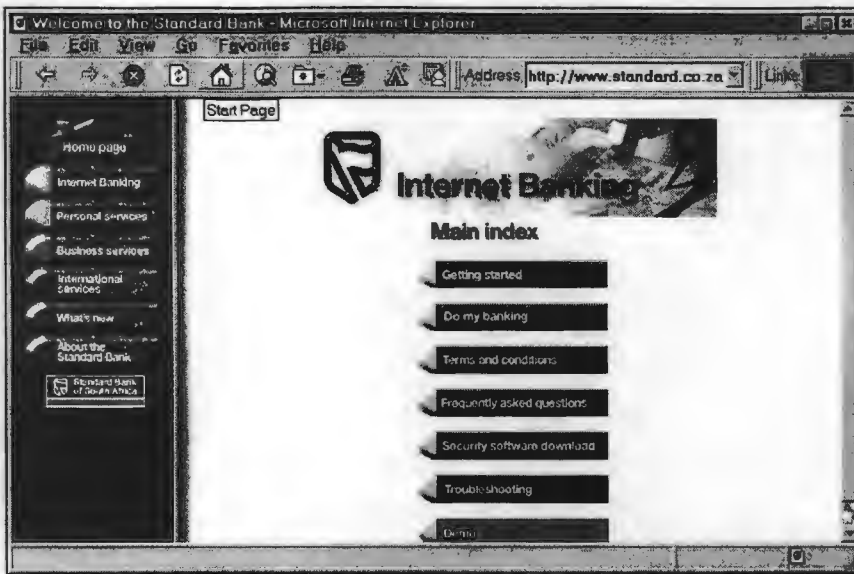


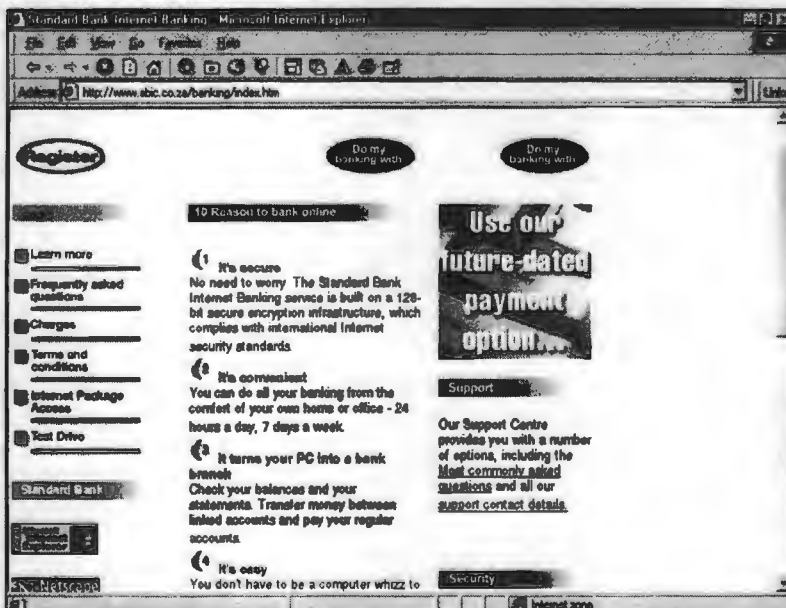
Figure 8.16 Version 2: November 1997 Internet bank home page



Figure 8.17 Version 3: May 1998 Internet bank home page



Figure 8.18 Version 4: November 1998 – April 1999 Internet banking home page



Source: www.sbc.co.za (1997, 1998, and 1999).

### 8.2.1.3 Standard Bank's pre-sign-on page: the 'Terms and Conditions' page

Standard Bank themselves refer to this page as the 'sign-on' page but this name is better suited to the next page in the navigational sequence – i.e. the page where the user enters his banking number (client ID) and password (PIN number). Figures 8.E.7 (September 1997), 8.E.17 (October 1997 to October 1998), and 8.E.27 (November 1998 to April 1999) represent the three versions of this page for the period 1997 to 1999.

This used to be an intermediate and low importance page in the first two versions, but the (default) direct link from the bank's home page to this page means it is now the first Internet banking related page that the user encounters after opening the bank's home page. In previous versions its only purpose seemed to be to remind the user about the terms and conditions that are attached to the use of Standard Bank's online facilities – hence the title used above. Somewhat surprisingly, little was changed on the page between November 1998 and January 1999 when the direct link from the home page was introduced. The current 1999 page has five prominent links. Three are help and information related namely links to the terms and conditions, FAQ, and Learn more, and there is a link back to the bank's home page, and a prominent animated button which leads to the sign-on page. Removing the strange-looking left-hand options frame present in the first version improves the visual layout and clarity for this page.

### 8.2.1.4 Standard Bank's sign-on page

There have been three versions of this page as shown by figures 8.E.8, 8.E.18 and 8.E.28. This page has form-based text input fields for entering the user's banking number (usually a card number), a user selected PIN, two radio buttons for changing the PIN, and two buttons labeled **Reset** and **Submit** in the first version, and **Clear** and **Submit** in the two more recent versions. The reset and clear buttons have the same function – they remove entries in the edit boxes. The two recent versions are frameless, and display only a home page navigation button instead of the many options present in the first versions' left-hand frame. A brief explanation on how to use the page is displayed at the top for the old version and on the left-hand side separated by a vertical divider for the new versions. Although very similar, the current third version improves on the second version by grouping the set of controls and input fields closer together – this removes the requirement in the previous version to scroll the page down to reach the submit button. In the first version the clear and submit buttons were graphics objects but these are currently form generated buttons which reduce file download sizes. It also replaces the overly large and prominent home page button with a less prominently sized but more prominently placed home page link – it is now positioned in the top left hand of the page.

### 8.2.1.5 Standard Bank's banking services options page

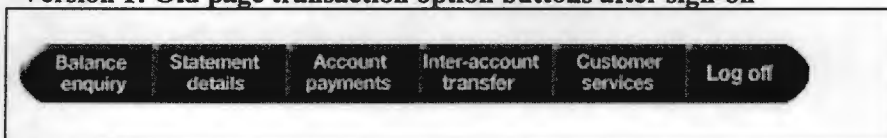
There have been essentially two versions of this page as shown when comparing figure 8.E.9 and figures 8.E.19/8.E.29/8.E.36. Small changes related to the available transaction options were made to the second version between May 1998 and November 1998. There is however a significant

difference between the initial version and the second group of pages. The old site displays a vertically arranged list of five service options labeled **Balance Enquiry**, **Statement details**, **Account Payments**, **Inter-account Transfers**, and **Customer Services**. The early 1998 version had seven options arranged horizontally in a text menu bar format – for **Balance Enquiry**, **Statement Details**, **Account Payment**, **Inter-account Transfer**, **Customer Services**, **De-registration**, and **Logoff**. The current version also has seven options, but replaces the de-registration option with a late 1998 addition of managing third party payments – it is labeled **Beneficiaries Maintenance** (figure 8.20). In the old site all the pages that followed from this page showed a group of six buttons (figure 8.19) at the bottom of the page for the five service options as listed above, plus a sixth option labeled **Logoff** – as shown in figures 8.E.12 to 8.E.14. The two new versions are more consistent and use the same menu bar styled text links.

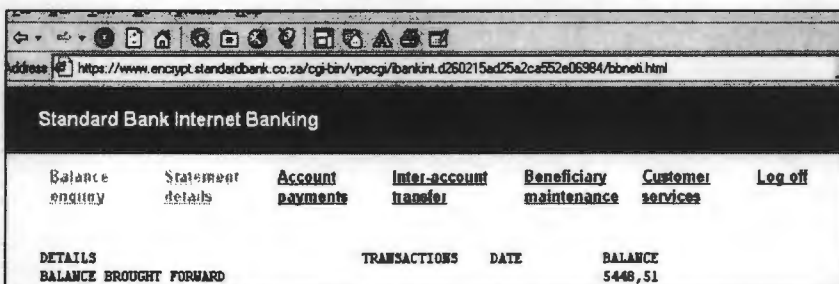
This page also displays the date and time when the user last made use of these services. It would have been better to personalize this by displaying the user’s name and/or at least add the word ‘Welcome’ to the message displayed. The new site is improved in that feedback is given as to which option is selected – through the normal HTML text link colour change – which was not the case for the old site. There is a problem with this simple feedback mechanism in that all previously activated links are shown as selected (figure 8.20) which gives the user less unambiguous feedback on his/her position in the banking session.

Appropriately Standard Bank has decided that the user should not have the option to return to the bank’s home page from here – the button for this is present in the first version but absent in the second. The reasoning is that the user is now signed on to the secure site, and should logoff (and exit gracefully) before accessing insecure pages.

**Figure 8.19**  
Version 1: Old page transaction option buttons after sign-on



**Figure 8.20**  
Version 2: New transaction option text links after sign-on



Source: www.sbic.co.za (1997-1999).

### 8.2.1.6 Standard Bank's transaction pages

The Internet banking transaction pages are briefly examined below.

#### (1) Account statements.

**Statement selection page.** This allows the user to choose an account from a pull-down menu, and has a single submit button – the position of this button has inconveniently moved from being centered directly underneath the account number and name, to the bottom of the page. Apart from feedback on which service has been selected in the new version, other changes between previous versions (figures 8.E.11 and 8.E.20) and the current (8.E.40) pages are the previously discussed group of five service buttons at the bottom of the page for the initial version compared to the seven menu options at the top of the page for the newer versions, and options for selecting the time period for the statement option in the new version.

**Statement details page.** This displays details about the previously selected account, and lists all the transactions for that account as the transaction name, amount, date and the balance (figures 8.E.12 and 8.E.21). Previous versions contained more graphics elements, but the format of the statement itself has changed little between the page versions – it seems to be a direct translation of the printed hardcopy format onto the screen page. There are options for downloading the statement (in comma delimited format), but printing has to be done by using the browser's print option from its file menu.

#### (2) Account transfers.

**Accounts transfer selection.** This transaction is represented by figures 8.E.8, 8.E.13, 8.E.22, and 8.E.37. Changes between the early and current versions include the display of the transaction options as already discussed, and the removal of almost all graphics elements from the current page. This page has pull-down menu lists for the source ('From') and destination ('To') accounts, and has two text input fields for entering the amount in Rand and Cent. Two form-generated buttons labeled Reset (clear), and Submit (proceed), are also shown.

**Accounts transfer proceed.** This is a confirmation to proceed from the previous screen and shows the balance of the two accounts (before the transfer), and the transfer amount. A single submit button serves as an intermediate confirmation of the transaction. Very little difference exists between the old and new versions (figures 8.E.14, 8.E.23, and 8.E.38).

**Accounts transfer confirmation.** This is a confirmation of the transfer and shows the balance of the two accounts after the transfer. Apart from the removal of graphic elements little has changed from previous versions (figure E.8.39).

#### (3) Other pages.

**Accounts balance page.** (Figures 8.E.10 and 8.E.30/31) This is a shortened version of the statement details page and gives balances for all the user's linked accounts on the same page – i.e. no account



selection is required. Apart from the service option buttons and graphics removal, there is nothing significantly different between the old and new sites.

**Beneficiaries payment page.** (Figures 8.E.43 and 8.E.44) These pages enable the user to set up options for transferring money to third parties (i.e. external to the bank) such as Telkom and Escom. This is a new option introduced in late 1998.

**De-registration page.** This used to be a menu option on previous versions (figure 8.E.24) of the banking site but is now accessible from the customer services page (figure 8.E.45). It allows the user to de-register from having access to the site's banking facilities. As this would be a seldom-used option its removal from the main menu bar is appropriate.

**Logoff page.** (Figures 8.E.15 and 8.E.32) This bare-looking page is displayed after the user selected the logoff option, and is present in the same format and layout on both the old and new sites. A single button labeled Logoff is generated – even though this option had been executed already. It provides the bank an opportunity for sending the user a ‘thank-you’ message.

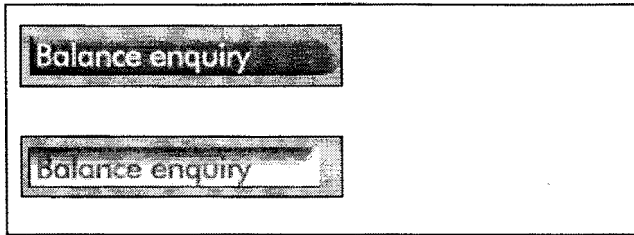
## 8.2.2 Other local (South African) banks

### 8.2.2.1 First National Bank (FNB)

A detailed usability examination of FNB's banking site is not included here, but over a two year period a series of simple banking tasks were executed, similar to those for Standard Bank, using FNB's comprehensive demonstration facility. The pages generated during these exercises are shown in appendix F as figures 8.F.1 to 8.F.19. Similar to Standard Bank their Web site has also undergone a number of major changes, as represented by the three groups in appendix 8.F. Interestingly Shapshak (1997:86), shows an even earlier screen image of FNB's banking page which looks very different from the early pages as captured in appendix F.

Usability issues originating from the use of their banking facility will be used in the heuristic evaluation that follows. One example that will be given here is the excellent feedback given by their transaction options buttons when compared to the text links used by Standard Bank (figure 8.21). They improve on Standard Bank in some aspects of their use of terminology such as preferring ‘Go’ instead of ‘Submit’ for their account selection and sign-on buttons. Compared to Standard Bank, FNB offers a wider range of services such as both mini and full statements, and it added third party transfers at a much earlier stage than Standard Bank. Their statements use screen attributes effectively – it is not a simple print-page transferred to the screen. They also make effective use of graphics elements. The overall impression is that their site seems professionally designed and it appears pleasing, uncluttered, functional, and with good aesthetics.

**Figure 8.21**  
**Feedback via two button states**



### 8.2.2.2 Nedbank

Similar to FNB, a detailed usability examination of Nedbank's Internet banking site (NetBank), is not included here, and a similar series of simple banking tasks were executed – but Nedbank's demonstration facility is much more limited in its functionality. The pages generated during this exercise are shown in appendix G as figures 8.G.1 to 8.G.12. Nedbank is the only bank to offer a low graphics version – which downloads faster – of its Internet bank as shown in figure 8.G.9.

### 8.2.2.3 Amalgamated Banks of South Africa (ABSA)

ABSA only introduced a demonstration facility on their Internet banking site early in 1998. Appendix H (figures 8.H.1 to 8.H.16) shows the evolution of their site for the period 1997 to April 1999. The change from a somber, solemn and bleak looking conservative bank to a more modern and approachable banking service is immediately noticeable – compare figures 8.H.3 to 8.H.6 and 8.H.16 (their Internet banking home page), as well as figures 8.H.1 to 8.H.15 (the bank's home page).

### 8.2.2.4 Mercantile Bank

Mercantile Bank only offers a banking service through the Internet using their own custom and non-browser-based software. They do not have an online demonstration facility. Figures 8.I.1 and 8.I.2 show the bank's home page and their Internet banking page.

## 8.2.3 Overseas banks

Six non-South African banks were also examined – five from the United States and one from the United Kingdom. Captured Web page images for these six banks are shown in appendix I, figures 8.I.3 to 8.I.51. At the stage when their banking Web sites were captured to local storage (the middle of 1998), Internet banking were still poorly supported in Europe – as reflected by the then limited facilities offered by the Royal Bank of Scotland. This is not the case for the US – all six banks examined exhibit feature-laden and professionally designed and developed banking Web sites. The five American banks represented in appendix I are Security First Network Bank – this is an Internet-only bank with no physical banking facilities, Wells Fargo Bank, Citibank, the Bank of America (BOFA), and BankOne. McCoy (1996:182-186) comments favourably on Web page design aspects for previous versions of BOFA's and Wells Fargo's Internet banking sites.

## 8.3 A heuristic evaluation of Standard Bank's Internet-banking Web site

This is a usability inspection that is based on a set of usability guidelines or heuristics. The Internet banking interactive interface is examined for cases where these 'rules-of-thumb' are violated or adhered to. The usability criteria that will be used have been identified (chapter 7), examined (chapters 2 and 3), and will be discussed using the following categories: **Learnability**, **Flexibility**, **Robustness** and **Web-specific** guidelines. The latter category includes aspects such as navigation, aesthetic and minimalist design, the match between the system and the real world, and consistency and standards. These are not mutually exclusive categories. There are interaction, interdependence, and a large degree of overlapping between these categories. This will be reflected in the discussion. In addition the page overview presented in the previous section already touched upon some of these usability aspects, as did chapter 5 (Data collection), and chapter 6 (Task analysis).

### 8.3.1 Learnability

Ten usability properties are included in this category which deals with the ease with which maximal performance and effective interaction is achieved.

#### 8.3.1.1 User guidance and support

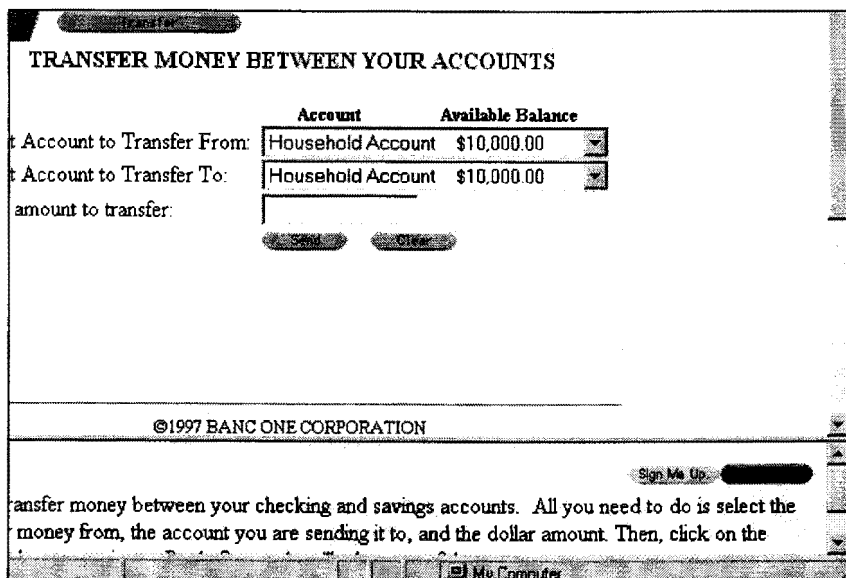
Extensive **user support and user guidance** are provided on the banking pages, either through facilities such as the demonstration section, or other less interactive options which includes the Getting started, Troubleshooting, Help and FAQ (Frequently-Asked-Questions), buttons and/or links on the Internet home page. The instructions and messages that are displayed on the screen during the interactive banking session such as during the sign-on procedure, and the display of error dialog boxes and messages generated as the result of both system and user errors, are also examined here. Appendix A (errors), B (user guidance and help), and C (security) presents page image examples of these facilities and messages. In addition chapter 5 (data collection), previously extensively examined the page structure and layout of the demonstration and help facilities. The navigational diagram shown in figure 8.29 presents a summary of the large number of user guidance options available from the bank's pre-transaction pages.

**Online help and information pages.** Standard Bank use a simple no-frames page structure to display their online help (figures 8.B.1 to 8.B.6) – which is an improvement when compared to the earlier (1998) poorly designed frames-based FAQ help pages (figure 8.B.7). However, there is no option for obtaining help during the actual transaction session, and access to the help/information pages is only provided before and after the secure transaction session.

This is in contrast with other banks such as FNB which included a general help option (FAQ) during their transaction session (figure 8.F.7) in the earlier version of their Web site – but removed this option from their current Web page (figure 8.F.16). The (probable) reason for this is because their help pages are placed on a non-secure section of their server – this is similar to Standard Bank's site construction. ABSA (figure 8.H.11) provides access to their help section through a prominent help button – this is available at all times and for all transactions. Nedbank does the same with help available from a button on a horizontal bar during the transaction (figure 8.G.10).

All four of the local banks can benefit from studying the way that *context sensitive help* is implemented by some of the American banks. BOFA (Bank of America) implements a sophisticated context sensitive help function as shown in figure 5.4. BankOne makes effective use of a horizontal context sensitive help window displayed at the bottom of the transaction page which is visible at all times (figures 8.22, 5.13 and 8.I.33).

**Figure 8.22 BankOne's context sensitive help frame**



Source: [www.bankone.com](http://www.bankone.com)

Citibank does not have a matching facility, but impress with the way in which its help pages are designed – as shown in figure 5.11. BankOne's pop-up help (in a separate window) seems to work effectively, but is surpassed by the usable design of BOFA's detailed indexed help as shown in figure 5.4. It was earlier noted that the use of frames could lead to problems. All of the American banks do make use of frames, but this is well managed and do not lead to the problems experienced with Standard Bank's frames implementation such as the font size problem as shown in figure 8.A.5 in appendix A.

**Demonstration facilities.** These important functions have been examined in chapter 5 focusing on their page structure – they invariably use a frame-based page structure acting as containers to separately hold the transaction options and transaction details. BankOne added an effectively implemented windowed container at the bottom of the page that displays context sensitive help – this is visible at all times during the transaction.

Standard Bank removed its full interaction demonstration facility during 1999 and replaced it with a less interactive and relatively unsophisticated screen image sequence facility (figures 8.B.8 to 8.B.11). The earlier comment (section 8.2.1) on the usefulness of these facilities to the bank client after an initial learning period could be one reason for this change. If the demonstration facility is viewed as a promotional rather than learning resource it is likely that its development and maintenance will decrease in importance. Keeping the live banking site and the demonstration site synchronized places a significant additional strain on the bank's maintenance personnel. On the other hand it should be noted that a badly designed demonstration facility with poor usability could damage the bank's client recruitment rate. Enforcing a synchronized simulated live banking facility would ensure that the demonstration function benefits from the same usability improvements made by the bank for their live facility, and which could be the result of client feedback when using their facility. A fully interactive demonstration facility will promote experimentation and trial-and-error learning, and as a result promote user 'learning-through-using' (De Wet, 1994:81).

**Error messages:** The clarity, meaning, and guidance offered by the error messages as shown in appendix A varies between very good and very bad. System errors tend to generate sometimes completely incomprehensible error dialogs – as shown in figures 8.A.1, 8.A.2, 8.A.10, and 8.A.14. On the other hand user (or transaction-induced) error messages are far more helpful and meaningful – examples of this are shown in figures 8.A.7, 8.A.8, 8.A.9, 8.A.11, 8.A.12, and 8.A.13. The reason for this could be due to the bank having more control over transaction/user errors. But system errors are usually a reflection of the immature Internet technology, its unreliability, and its distributed nature. An unfortunate effect of assessing usability only through a bank's demonstration facility is that many of these error-handlers are not simulated and cannot be induced – hence the comment made earlier that it would not be representative to conduct a usability evaluation on a demonstration facility alone.

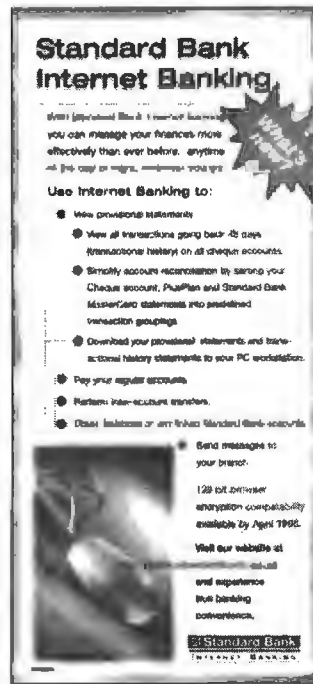
**Offline help.** All four local banks distinguish themselves by the paucity of their printed (promotional, user guides or instructional) material related to their Internet banking facilities. Standard Bank at a stage late in 1998 (i.e. more than a year after it introduced its banking facility), could only offer a photocopy of a single page promotional pamphlet for this service. They currently do have an eight-page information brochure. Figures 8.23.1 to 8.23.3 show the three variants of their Internet banking information and promotional material for the period 1997 to 1998. Nedbank excels through the production of its lavish and expensive looking information brochure that is accompanied

by a quick-access floppy disk. Standard Bank is currently in the process of offering a limited Internet access service – restricted to e-mail facilities and access to its banking facility only – in conjunction with the Internet service provider Icon. None of the banks have access software available on CDROM. The production of a small printed user guide should be a priority for all four of the local banks.

**Figure 8.23.1 to 8.23.3 April 1998 to January 1999**  
**Three stages in Standard Bank's Internet banking information brochure**



**Figure 8.23.1 (above) April 1998. Single page photocopied brochure.**



**Figure 8.23.2 (right) November 1998. Single page brochure.**



**Figure 8.23.3 (far right) January 1998. Multi-page brochure.**

Source: Standard Bank (SA).

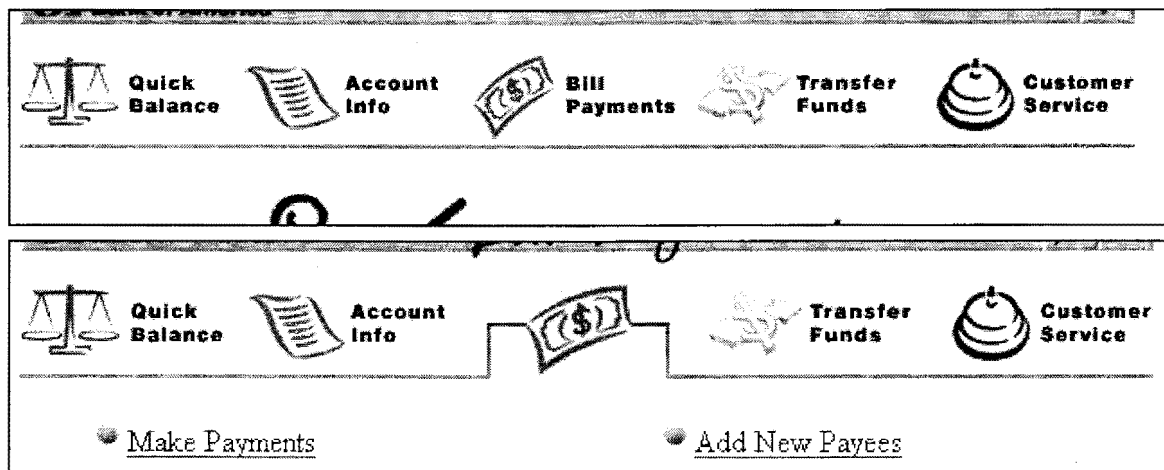
**Other help and guidance techniques.** Features such as tool-tips (used effectively by Citibank as in figure 8.26), fly-by-hints, wizards, and use of the status and title bars, are all available techniques and could be employed to provide the user with improved guidance and help within the context of this Web-based interactive application. Use of the ALT image tag displays a tool-tip that could be used to further explain the meaning of a graphics element – this is not used to any significant extent by Standard Bank.

An examination of the captured images in appendix E (figures 8.E.1 to 8.E.48), reveals that the title bar header is limited to general titles such as 'Welcome to Standard Bank' or 'Internet Banking'. The only exception to this is figures 8.E.28 and 8.E.32 where the title does give an indication of

what the page purpose is – these pages are titled ‘Logoff’ and Sign-on’. The message area in the browser’s status bar is not used by the bank.

A lock or key metaphor – both Internet Explorer and Netscape Navigator use this reasonably effectively as shown in figures 8.C.1 to 8.C.6 – gives feedback on the security context of the connection. Feedback on the transaction option selected is poorly implemented by Standard Bank. They use a standard text link, which uses a colour change to show if it had been previously selected. This could result in more than one option or even all seven options displaying the same as shown in figure 8.A.5 and 8.20. FNB uses a much more effective technique (figure 8.21), and the change of state of the graphic button element leaves no doubt about which option is currently selected. The most effective feedback technique for transaction selection is used by BOFA (figures 8.24 and 8.I.25), whilst BankOne (figure 8.I.37) used a prominent graphic bar to identify the current transaction.

**Figure 8.24 BOFA’s transaction selection feedback mechanism**



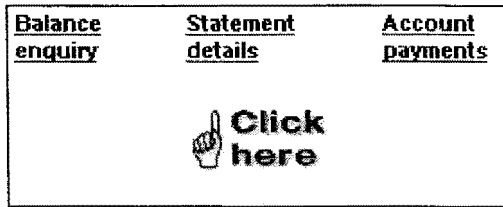
Source: www.bofa.com.

The feedback implementation techniques as observed for all the banks are perhaps modally restricted – sound could be used in an effective and engaging manner to alert the user – for example when a transaction is completed. (Sound generating hardware on a the client computer, even in a business environment, should be considered to be a standard item.)

Animation is used during Standard Bank’s (new) passive demonstration facility to guide the user to the next step in the sequence (shown as the hand icon in figure 8.25 which is taken from figure 8.B.11), and also to attract his/her attention to the link to banking sign-on page (figures 8.11 and 8.12). Advertising banners and animations on other sections of their Web site are fairly widespread, and should be reduced as its use causes larger page downloads and it is considered distracting.

**Figure 8.25 Standard Bank's new demonstration facility.**

**The animated hand guides the user to the next stage in the demonstration**



Source: www.sbic.co.za (1999).

### 8.3.1.2 Predictability

**Surprise avoidance** and predictability are promoted by the close similarity between what the user-client is used to do during (physical) bank transactions and/or autobank transactions, and what he/she is required to do here – i.e. these pages support the typical user's mind model of how such a banking site should behave, as his/her perception of the workings of such a computer-accessed banking system will be shaped by his/her experience with the first two familiar systems. This application of the user's real-world knowledge to the system also aids in improving the site's familiarity and guessability properties.

Operation visibility is another example of predictability – i.e. the display of the next set of available operations. For this application this type of visibility is purposefully and severely restricted to button pressing. As the user completes a banking task he/she is guided to the next stage by buttons labeled 'Submit', and directed back (albeit an imperfect undo), by the use of 'Clear' or 'Reset' buttons. Perhaps re-labeling the button to 'Proceed', 'Next' or 'Continue' would strengthen this characteristic.

Improving the system transparency also improves its predictability characteristic (De Wet, 1994:93). The pointing action inherent in WIMP (Windows, Icons, Menus and Pointers) interfaces acts as an aid in this respect. Consistent behaviour strengthens its predictability – this argues against the use of customized software banking solutions as used by Mercantile Bank. It also argues against the use of security software in conjunction with the browser – this tends to add non-standard behaviours to the interaction as shown in figures 8.C.9 and 8.C.10. The typical user will already be familiar with both the Windows environment and the Web browser operating within it – any change in the access software behaviour would lead to uncertainty on the user's part. Related to this is the use of the old (1997) site's house-shaped button for navigation back to the home page. This had been discussed and is an illustration of the effective (although obvious), use of familiar and predictable metaphors on the site. Thimbleby (1990:188) also considers predictability to be related to a predictable standard of software quality. Again the typical user will be more familiar with an abnormal behaviour of his/her browser than when using unfamiliar security or customized software.



### 8.3.1.3 Generalizability

This as a usability property is promoted by the use of standard Web browser environments. The user is likely to be familiar with the operation and behaviour of either Netscape or Explorer, and is therefore almost certainly able to extend acquired knowledge from previously encountered behaviours to understanding and interpreting the new unfamiliar situation. A specific example of this could be the masked display of the PIN number which would be similar to entering passwords in other Web and networking environments (and the Windows environment in general). The Web browser itself – which implements the common associations of ‘file’ with the menu options ‘save’, ‘save-as’, and ‘print’, is another example (De Wet 1994:94).

### 8.3.1.4 Familiarity

Because this property is related to the compatibility of the system with user conventions and expectations – its guessability and its ease in initiating an interaction, it is closely related to the aspects of consistency, generalizability, and synthesizability – the comments made in those sections also apply here. The banking system makes use of familiar analogies such as graphical soft-buttons where its appearance suggests its use – as illustrated by BOFA’s exemplary use of icons (figure 8.24) – and is also cognitively compatible by mimicking the physical banking process.

### 8.3.1.5 Consistency

The broad property of **consistency** is (also) enhanced by the use of these standard access environments. It is also improved through the correspondence that exists in the naming conventions between real-world banking terminology and the terms as used on the banking site. It is helped by the use of standard objects on the screen such as form-based buttons and editboxes, but is definitely harmed by the technical labeling of task-oriented components such as the ‘Logoff’ and ‘Submit’ buttons. To repeat, ‘Bye’, ‘Finish(ed)’, ‘Proceed’, ‘Done’ or ‘OK’ would be more appropriate, less data-processing focussed, and more customer-friendly.

Because consistency is such a broadly impacting concept, it is interdependent on many of the other properties such as familiarity and generalizability, and comments made for those could be applied here. Access through the familiar Windows environment probably contributes more than any other single factor to ensure a consistent behavioural likeness across the many possible similar tasks and situations during the banking session. It ensures consistent and known action-response pairing across a multitude of aspects such a mouse movement, use of the mouse buttons (for example to select sections from a displayed statement to save or print), and through the use of menus and the window controls.

It could be argued that the likeness of the transaction options ‘menubar’ that looks like a menu bar but behaves differently – it is in fact a horizontal list of text links, is inconsistent. If something looks

like a menu then it should behave like the normal windowed menus. Similarly, Standard Bank's use of graphics buttons that give no feedback when clicked is also inconsistent with the Windows environment. On a more intuitive level the lack of an undo for the transactions is inconsistent considering that many Windows applications – and the Web browsers themselves – offer such an option.

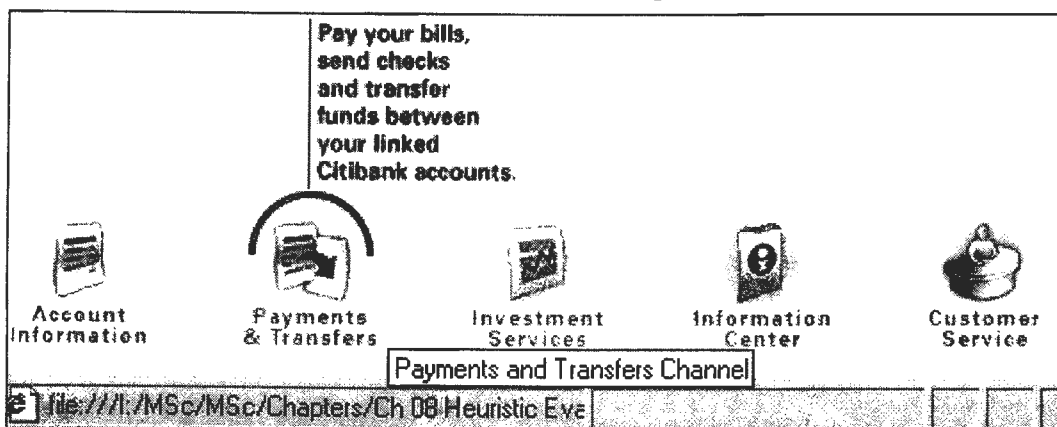
### 8.3.1.6 Synthesizability

This shows moderate improvement in recent site versions through improved feedback to the user on what transaction he/she is busy with, and what transactions had been executed before – as discussed previously. As this property deals with an association of past (historic) actions with the current system state, the previously discussed aspect of user feedback of the current selection through graphics page elements is also applicable here.

Mayhew's (1992:320) comment on auto-selection when the mouse cursor enters an active region is commonly implemented in the Web environment as JavaScript mouse-over effects. JavaScript is used by Standard Bank on their Web pages – although they do not use Java applets – probably because of the uncertainty that still surrounds certain security aspects of this language. Introducing mouse-over effects on their Web pages would help to make it more of an engaging experience for the user and will also improve its feedback and synthesizability properties. Citibank (figure 8.26) used these JavaScripted mouse-over functions to good effect.

Drag and drop operations tend to support synthesizability through an immediacy effect – but the banking tasks as designed tend to be mainly keyboard and button operated. This is however a limitation of the Internet and browser technology rather than the banking application.

Figure 8.26 Citibank's use of mouse-over and tool-tip effects



Source: [www.citibank.com](http://www.citibank.com).

### 8.3.1.7 Visual clarity

Visual clarity could be improved – even though animated buttons such as ‘Do my banking’ are used they should have a more distinctive and recognizable design. They are sufficiently prominently positioned in the new site (but were not positioned adequately in the 1997 version) – and they support the notion that this is the user’s most frequent navigation task. The change from the bottom-positioned services option button in the old site to the top – and with the addition of a user feedback sub-option – is commendable.

The use of colour on the site is appropriately subdued, and supports Standard Bank's corporate colour scheme of white and dark-blue. The removal of (almost) all graphic elements from the bank’s transaction pages in late 1998 did improve its appearance – the page seems less cluttered and as a result the important screen components and options are easier to find. This aspect is examined in more detail during the section on minimalist and aesthetic page design.

### 8.3.1.8 Positive engagement

This is perhaps a more appropriate term than **enjoyability** for these online banking services. The site encourages some degree of experimentation and browsing outside of its secure use (through its colourful animated advertising and information banners and ‘what's new’ buttons and links), but such browsing could compromise the security if done during the actual banking transaction. The removal of the options for the home page during banking transactions on the new site has been discussed, and is considered to be a required change.

In general the (artistic) visual appearance of the site has improved in its newer incarnations. A similar comment applies to the bank’s (main) home page – but all Standard Bank’s Web pages could use the professional graphics design and layout input that is evident for three of the American banks studied – namely Citibank, BOFA and BankOne.

### 8.3.1.9 Conceptual models

This relates to the user’s mind picture of how he/she perceives the system to function (internally). A correct mental model could aid in enhancing its learnability characteristics, whilst an incorrect mental picture of the system workings would act against ease-of-learning.

When there is close match between the actual system and the user’s picture of it, the intuitiveness of the system is enhanced (Sutcliffe, 1988:167), which aids in its learnability. The banking application is accessed through a GUI system that is object-oriented. Because the user already carries an object-oriented mental model of the surrounding support system, it is likely that this intuitive (i.e. effortless learning) property will be enhanced if the banking application itself also behaves in an object-oriented manner.

This is however not the case – the banking paradigm is very firmly functionally oriented. The transaction is the central theme and the primary (trans)-actions are applied to the secondary objects (such as accounts and third parties). A better approach would be to first select the account, and the user would be then be presented with a range of possible transactions on this account-object – i.e. an increased degree of page customization would be required. FNB hints at the implementation of this object paradigm by centrally presenting the account object (at the top middle of the page), but then adds little else to further support this approach. The same comment applies to all the other local and overseas banks studied. Although the range of operations on the possible banking objects are not as yet sufficiently varied (they are mostly account type objects), this should nonetheless considered to be a mismatch between the user's models of the supporting system and the banking system.

#### **8.3.1.10 Interaction metaphors**

Iconic metaphors should preferably represent the tool (object) rather than the action (method) (Mayhew, 1992:323). All the banks examined use a mixture of the two approaches. The home page icon link (figure 8.6) is an example of the first. The transaction icons used by BOFA (figure 8.24) contain examples of both. The banknote refers to the account object, whilst the arrowed dollar sign refers to the transfer action. The well chosen bell icon for customer services could be an example of combining both approaches. Note that Citibank (figure 8.26) uses a 'person-behind-a-counter' metaphor for this purpose – which is effective but at smaller screen sizes it is less easily recognizable. In the case of BOFA the operation-oriented icons are appropriate and visually clear – to refer to the object (account) could be confusing in this case.

The 'banking-hall' metaphor (i.e. similar to the desktop metaphor) finds some expression on the South African banking sites. ABSA used a tiled floor background on their older Web page that reminds one of the appearance of older banking halls (figure 8.H.3). The use of lock and key metaphors by the browsers to indicate a secure/unsecure connection has already been discussed and is presented in appendix C. Smilowitz (1997) examines the use of metaphors within the context of Web browsing, and validates their effectiveness during interaction – their use leads to significantly reduced error rates.

### **8.3.2 Flexibility**

This refers to the multiplicity of information exchange options present in the system. Although flexibility is a desired usability property, it should be balanced against its impact on system security and reliability characteristics – the latter two aspects are especially important for security-sensitive banking applications.

### 8.3.2.1 Dialog initiative

**User pre-emption** enhances flexibility as opposed to **system pre-emption**, which has the opposite effect – but perhaps the issue of **dialog initiative** should be limited in an application such as banking – as commented on by De Wet (1994:83). It may result in the user losing track of the tasks initiated, or may lead to compromised data integrity when a mixture of secure (banking) and insecure (help/information) pages are accessed. Practically, this would imply that only the set of often-used banking tasks that the user would want to perform would be catered for. An example of this is the many modal dialog boxes that are generated during the typical banking session such as the various security connection warnings (figures 8.C.11 to 8.C.12). In this case it would be undesirable to have the user continue with the task without first acting on the dialog box request. As Windows 95/98 is a pre-emptive multi-tasking system, the user could however continue with another perhaps also banking related task such as retrieving documentation from a second separate browser window.

### 8.3.2.2 Task migratability

A comment similar to the above would apply to the related properties of task migratability, user rather than system control, and substitutivity. There are instances where the control should be exclusively embedded in the system and also situations where the user should have exclusive control – such as when either party detects a condition when security may be compromised, or when data corruption had been detected or is suspected.

### 8.3.2.3 Multi-threading

Multi-threading – i.e. the ability to support the user in more than one task at the same time – would similarly be unsafe for this type of application. Standard Bank has discouraged this by restricting the options available during the secure transaction phases – especially on its more recent Web sites. It is also considered unlikely that a user would want to do anything else that is unrelated to banking whilst using these services – most of the tasks are fairly simple and short in duration.

### 8.3.2.4 Substitutivity

The user's input in many of the banking tasks is reflected as system input – such as during the three-phased account transfer task. Here the original choices of accounts and amounts during step 1 (chapter 6) are used by the system during steps 2 and 3.

The user also has some choice in selecting how the system state is displayed, although this is usually browser-related rather than application-controlled. An example is the choice to configure the system to display security dialog boxes. The ability to do this could lead to a better task-match (Dix et al., 1998:170), but could compromise the security aspects of the system. Another example is the availability and use of keyboard shortcuts (De Wet, 1994:107). This is implemented through browser and operating system accelerators ('hot-keys'), such as Control+A for a 'select-all.' operation.

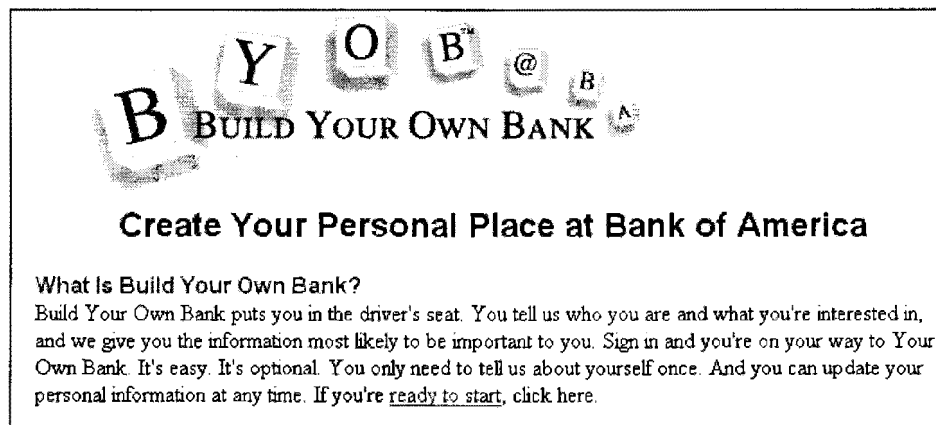
### 8.3.2.5 Control

This is closely related to customizability – i.e. the user should be able to organize their work environment according to their own preferences (De Wet, 1994:109). Because the application runs as a windowed system the user has all the normal window-controls (resize, minimize, move etc.) available for control. This aspect is reinforced by the requirement that the user *physically* interacts with the system (Mayhew, 1992:322). He/she is continually reminded through switch (keyboard and mouse) feedback, that he/she has control.

### 8.3.2.6 Customizability

A large effort should be directed towards both client and server-side **customizability** of the site, such as examined by Thimbleby (1990:162) when he speaks about a programmable user interface where the designer can delegate design choices to the user. Apart from the customization offered by the operating system and the browser – which is considerable, the banks offering this type of service could differentiate their Internet banking products quite considerably (and therefore perhaps gain a competitive advantage), by implementing effective page customization for each user. Each user will gain access to a site that suits his/her own needs, which will appear differently, and even behave differently when compared to another client.

Figure 8.27 The Bank of America's banking customization facility



Source: www.bofa.com.

BOFA hosts a mature and feature-rich implementation of this idea through the customization and user preference facilities offered by their 'Build-your-own-bank' section (figure 8.27). Standard Bank offers some degree of customizability through the options available in their statement options. But as this is a volatile configuration and choice (i.e. it is not retained from session to session), it is flexibility rather than customizability. Another obvious example of how they could improve on this aspect is the welcoming message given after signing-on. Instead of just displaying the date and time last logged in, they should display a personalized message which contains the client's name as part

of the welcome displayed – Amazon.com ([www.amazon.com](http://www.amazon.com)) makes use of this in a particularly effective and engaging manner.

### 8.3.3 Robustness

If robustness is viewed as the extent to which the system can continue to perform despite a violation of its specification assumptions (De Wet, 1994:90), then these (and all other), Web-based applications would have problems in satisfying this usability principle. In addition, for secure transactions, it is desirable not to continue the operation beyond a point of compromised security.

#### 8.3.3.1 Errors

Although discussed earlier (section 8.3.1), the focus here is on the generation and causes of errors rather than on their informative aspects. Appendix A presents some of the errors that were encountered, and there are two types – system and user (transaction) induced errors. The first category is represented by figures 8.A.1, 8.A.2, 8.A.5, 8.A.6, and 8.A.10. User and transaction errors are shown in figures 8.A.7 to 8.A.10 and 8.A.11 to 8.A.14. Note that figure 8.A.13 could also be viewed as a system error – a closed account could not be displayed, and similarly figure 8.A.10 could be classed as a user navigational error.

Figures 8.A.1 and 8.A.2 are interesting in that they illustrate some of the system errors that were encountered when accessing the very earliest version of the banking site (using Windows 3.11, Netscape Navigator 3.0, and Standard Bank's special security software). The first screen appears to be due to a loss of authentication – trying the two suggestions shown on the screen had no effect in restoring the secure connection and it was necessary to sign-on again. The second screen is the result of a time-out, also because of a lost connection, and similar to the previous problem a sign-on was required (which is the option suggested on the screen). The two small floating windows on the two screens are generated by Nanoteq's security software. In the first screen the lock icon is shown open – which is appropriate to the error condition, but the crossed-out lock icon as shown in the second screen would have been more appropriate for the first screen. The security software reveals its state by displaying a key-bunch (before a secure connection is established), or an open lock, or a closed lock with the number '128' – for the 128 bit encryption length used. The timing of the appearance of the open and closed lock symbol was found to be confusing – for example an open lock is shown whilst the browser's status bar shows data to be transferred from the bank.

Both earlier browsers used (Netscape and Explorer 3.0), show a security icon on the bottom left-hand of the status bar. In the case of Netscape 3.0 it is a broken key if it is a non-secure connection and a complete key if it is a secure connection (figures 8.C.3 and 8.C.4). The version 4 browsers both use a lock metaphor (figures 8.C.1, 8.C.2, 8.C.5, and 8.C.6), but place the icons differently – Netscape has a more prominent position on the taskbar.

The error messages shown when a connection is lost – either partially or completely – are unsatisfactory – the technical detail about sockets, digital id's etc. shown in the first screen of appendix A is bound to be a source of confusion to an unsophisticated user. He/she probably had some reservations about the security before this event occurred.

The user-induced errors shown in appendix A were caused by incorrect logins, incorrect amounts entered, and when the system could not display information on a closed account. The error messages are as expected for the situation, match the error condition, and are sufficiently informative and to the point. They are however not very noticeable – they should be displayed in a colour other than the page default colour (black) – red would be a good choice.

#### 8.3.3.2 Observability

Both **browsability** and **reachability** as aspects of observability should be limited during the transactions session (but encouraged outside of it) – note the comment made earlier about Standard Bank's removal of help/information options during transactions. As reachability can also be referred to as **navigational ease**, it will be discussed in more detail during the Web-specific section that follows.

**Persistence** is another aspect of observability – this should be encouraged within the application, but it is perhaps more a function of the browser environment than the site design – the use of the browser status bar, address bar, and animated logo all enhance the implementation of this property.

**Defaults** are another aspect of observability – the banking application itself is poor in its implementation of dynamic defaults – i.e. remembering the system state between sessions. The browser system does implement defaults more extensively as it is a normal part of the Windows environment. The comments made previously on site customization are also of relevance here.

#### 8.3.3.2 Recoverability

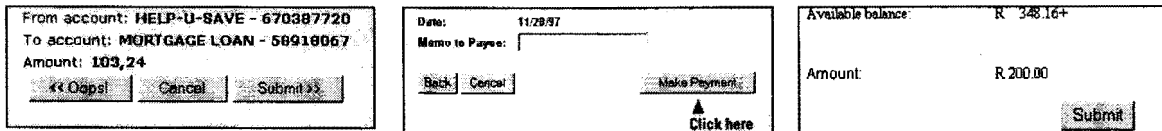
Recoverability is a serious problem with these remote accessed Web applications. As previously discussed for the two communication error pages, often the only method of recoverability would have been to abandon/abort the session and restart. Recoverability on the server-side will probably follow good fault-tolerance and rollback principles and is outside of the scope of this evaluation. During transaction errors the user is able to take corrective action on most error conditions – as discussed the messages displayed are reasonably helpful and enable the user to correct him/herself.

Because recoverability is also related to system reliability and dependability it will suffer from the lack of reliability which is inherent in the distributed nature of the banking application – this is reflected by the high frequency of communications errors experienced during use of the facility.



The **undo** function finds expression during the submission stage of account transfers. Standard Bank does not give the user an option to move back to the first step in this transaction (figures 8.E.37 to 8.E.39), but ABSA does – note the presence of the ‘Oops’ button in figure 8.H.12. Standard Bank only has a single submit button at this stage. Similar to ABSA Citibank offers the user an undo through a ‘Back’ button as shown in figure 8.I.11. Enlarged views of these images are presented in figure 8.28.

**Figure 8.28 Undo during an account transfer for ABSA, Citibank and Standard Bank**



Source: www.absa.co.za, www.citibank.com, and www.sbic.co.za.

**8.3.3.3 Responsiveness** is a major problem in these types of applications – the Internet is notorious for its slow response. This property also impacts on user guidance through its delaying effect on informative feedback (De Wet, 1994:88). System feedback is given through mechanisms such as the hourglass icon displayed in the browser window while the page is loading, and the indication in the browser’s status bar of the progress made in downloading page elements and in connecting to the Web site.

Standard Bank has steadily reduced the graphics content of its banking pages over the period 1997 to 1999. The transaction-option buttons were replaced by text links, and the size of other graphics object reduced on the new site – as a result the new site is more responsive. As a comparison appendix D (figures 8.D.1 to 8.D.3) lists the graphics and text components for three different sign-on pages of the banking site. The early 1997 site had a total of 14 objects (13 graphics components) in two frames with a total size of 57 kilobyte. The newer May 1998 site had a total of 7 objects (6 graphics) in a single frame for a total size of 19 kilobyte – i.e. it loads at less than half the size of the previous site. The most recent 1999 version has only one object (the HTML source file itself) and no graphics objects – its total (uncompressed) size is less than 3 kilobytes.

The reason for the sudden and large reduction of graphical page components on Standard Banks Web site is interesting. Figures 8.A.3 and 8.A.4 (appendix A), show two messages – regarding an Internet Explorer 4 security problem – which appeared on the bank’s Web site during June 1998. Page graphics components were blamed for the compromised security. As a consequence all graphics were removed, and remained absent from the site’s transaction pages for a period of about five months. Graphics did reappear on these pages during 1999 – note the Standard Bank home page banner (figure 8.E.37), situated in the top frame.

The approach taken by Mercantile Bank (figure 8.I.1) where most of the processing is executed off-line on the client machine, and only the result uploaded to the bank, could improve the perceived response for this type of application.

#### 8.3.3.4 Task conformance

This is the degree to which the user is supported in the execution of his/her task for the achievement of his/her goals. It can be described in terms of **task coverage** and **task adequacy**. The services offered by Standard Bank on their site were initially (1997/1998) limited in their variety and flexibility. The service-options were limited in the types of services offered – for example only transfers to owner accounts were allowed. These were later (late 1998), extended to third-party payments. They were also initially limited in the flexibility of the service itself – for example during 1997 a statement could only be obtained for a fixed period in a fixed format – compare figures 8.E.20 and 8.E.40 for the additional options which are now available during a statement selection. The bank therefore showed an improvement in both task coverage (sufficient width and depth), and in task adequacy – the manner in which the tasks are executed are similar to the way it is done in a bank.

It is informative to compare the services offered by the banks at an early developmental stage. Services available during this period (late 1997), for three of the South African banks Web pages are shown in tables 8.4 to 8.6 at various points in their task sequence. At this stage Standard Bank lagged behind the services offered by the other two banks for all three task-stages. Since then they have extended the facilities on their transaction page by including third-party payments. They have also added an extensive Business Electronic Banking (BEB) service, which is linked to from the bank's home page (figures 8.E.46 to 8.E.48). Design aspects as shown on the login page for the BEB should be applied to the bank's Internet banking services. The user is issued with an ID that is easier to remember and enter than the cumbersome 16 digit account number currently used..

**Table 8.4 Services offered by the banks from their home page**

Standard Bank	First National Bank	Nedbank
Internet Banking	Talk to Us	NedCredit
Personal Services	Value added Services	NetBank
Business Services	First Online	NedCor Results
International Services	WesBank	NedTel Cellular
What's New	FirstWorld	International
About Standard Bank	First National Equities	Business Banking
Search	Online Investment Services	Personal Banking
Contact	First News	Your Views
Index	Financial Reports	Contents

**Table 8.5 Services offered by the banks from their Internet page before sign-on**

Standard Bank	First National Bank	Nedbank
Getting Started	Get online	Demo
Do my Banking	Start online demo	NetBank
Terms and Conditions	Start online banking	About Security
Demo	Access requirements	Overview
Help	New Features	FAQ
Home Page	Terms and Conditions	Apply
FAQ	Online Security	Cancel NedTex
	Home	Tips and Tricks
		What's New
		Your Views

**Table 8.6 Services offered by the banks from their Internet pages after sign-on**

Standard Bank	First National Bank	Nedbank
Balance Enquiry	Balance Enquiry	Logoff
Statement Details	Mini Statement	Balance Enquiries
Account Payments	Interim Statement	Statement Enquiries
Inter-account Transfer	Account Payment	Download Statement
Customer Services	Account Transfer	Inter-account Transfers
De-register	Linked Accounts	Account Payments
Logoff	Logoff	Change Nedbank PIN
Home Page (demo)	Home	Cheque book Orders
	FAQ	Forex Rates
	Client Support	Apply
	Talk To Us	Overview
		About Security
		Help
		Disconnect

### 8.3.3.5 WYSIWYG

The design of screens such as those showing balances and statement details appears to be a screen-copy of the printed copy – more use should be made of the additional attributes that are available for screen displays such as using colour and designing for a resolution-independent layout. All of Standard Bank's pages are designed for an 800 by 600 screen resolution, which is acceptable as a compromise – this represents the majority of their users.

### 8.3.4 Web-specific Usability Aspects

These are usability aspects that have been covered in the previous section, but are now viewed from a different perspective and priority.

### 8.3.4.1 Navigation

Navigational usability issues are interwoven with almost all of the previously discussed properties. As such it had been examined during the discussions presented in many of the previous sections. Cockburn and Jones (1996) present a broad overview of the navigational problems associated with the World Wide Web, and Whitaker (1998:69) lists ten Web navigation design principles. Importantly Whitaker (1998) notes that there are two types of navigation behaviours – navigation to achieve a specific goal and navigation for exploration. The Internet banking application evaluated incorporates both navigational classes.

The more important questions that should be answered when evaluating navigational usability are listed in table 8.7 – these factors were compiled from material as presented in chapter 7.

**Table 8.7 Navigational usability issues**

(1)	Does the site use (approximately) standard link colors?
(2)	Are the links obvious in their intent and destination? Names of links should be concise and provide a hint on the content of the page that they link to. (Avoid using technical words such as servers, links, etc.) Avoid adding explanatory comments to textual links. Avoid 'linking-mania' (making a link every time a keyword of a page is mentioned in a text) Linking icons should have a distinctive feature of the page they are linking to. Links should be to primary aspects or characteristics of the institution. Textual information should be left for secondary pages.
(3)	Is there a convenient, obvious way to maneuver among related pages, and between different sections?
(4)	If a user were taken directly to this page from an outside site, what would they figure out about the rest of the site from this one page?
(5)	Would it be clear how to navigate to other parts of the site? Pages should not be overcrowded with links.
(7)	Is it clear what part of the site they are in?
(8)	Is the site 'brand' present?
(9)	Does the site make effective use of hyperlinks to tie related items together? Organize links as primary and secondary topics. Links to resources or other repositories on the Internet should be placed on a secondary page. (This page should be reached with a link on the primary page.)
(10)	Are there dead links? Verify that links connect to existing pages

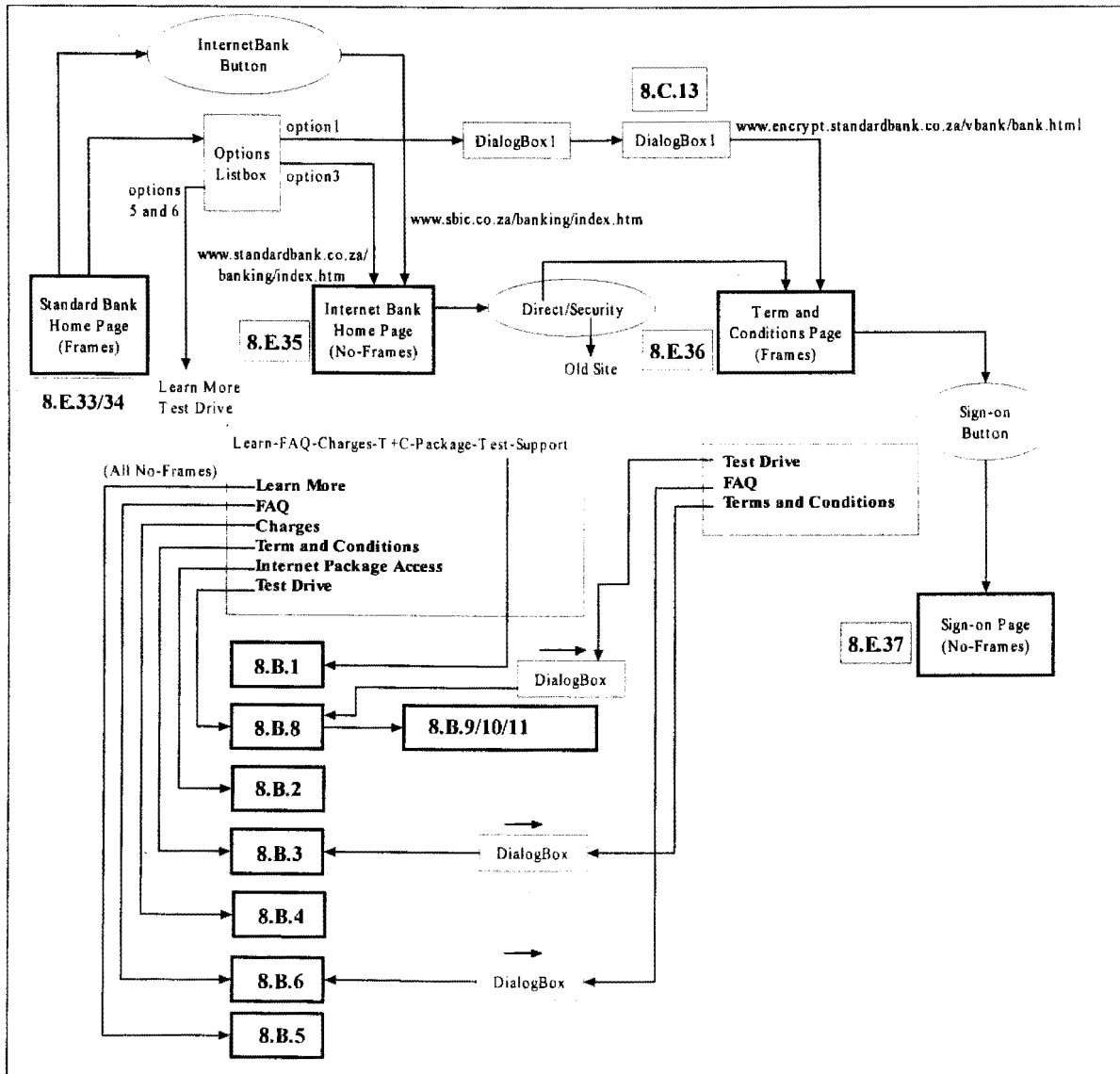
As an indication of the (hidden) navigational complexity that the user faces when accessing Standard Bank's Web site, the navigational diagram shown in figure 8.29 may be examined. This shows the number of navigational paths that can be followed, when starting from the bank's home page, to reach the Internet banking sign-on page. The recent change in the home page with direct links to the 'terms and conditions' page reduces the length of the required navigation – but it could be improved even further by an option to link directly to the sign-on page.

Figure 8.A.10 shows the result of a navigational freedom gone wrong – the error shown 'No transaction possible due to a technical error. Please try again later.' was displayed during a banking session conducted in January 1999. The error occurred after the logoff option was chosen from the transaction menu-bar. The browser's back-navigation button was pressed, the login

verification page was then inappropriately displayed. The error page shown immediately followed this action. Pressing the back arrow on the display of this error page showed the login page – however entering login details here resulted in the display of the same error message. The situation could only be resolved by leaving the secure site and returning to the bank’s Internet banking home page from where a correct login could be effected.

Figure 8.29

Navigational and user guidance diagram for Standard Bank’s home page to the sign-on page



Notes:

- (1) 8.B.x refer to the screen images as presented in appendix B.
- (2) Dialog box refers to the secure connection change dialog box displayed.
- (3) The oval shape refers to a graphical button link.
- (4) The options list box is the pull-down menu present on the bank’s home page.

The cause of this is due to the browser having a navigational sequence independent from the transaction's intended sequence. This is a problem that can be resolved by running the browser in a full-screen mode with its toolbar hidden. However this could then impact on the familiarity aspects of the interaction. Tauscher and Greenberg (1997) examine, and offer solution for, some of the problems associated with Web browser's navigation mechanisms.

Some of the issues as presented in table 8.7 are of less relevance during the inspection of the transaction pages. For example it would not be possible to link to any of pages past the login page – the issue of page identification in this part of the site is therefore less important (but not unimportant). Page identification issues had been discussed previously. These secure pages are an example of a condition where navigation should be limited – but this should not be at the cost of diminished user guidance and help. The use of technical terms had also been discussed – the bank should avoid the use of data processing terms such as 'submit'.

Standard Bank makes a reasonable effort to promote the visibility of their brand-name throughout the site by either displaying their bank's logo prominently (figures 8.E.39 and 8.E.40), or having the bank's name present in a prominent position on the page. The consistent use of the bank's colours (shades of dark-blue) also serves to establish a strong corporate identity throughout the site. They tend to be careful in grouping relevant links together – the best example of this is the pull-down menu on their home page linking to the Internet bank (figure 8.5), and the grouping of links found underneath the customer services option from their transaction options 'menu' bar (figure 8.E.45).

The use of descriptive graphical links had already been examined – they should however replace their (poorly designed and visually meaningless) animated link to their sign-on page (figure 8.12) with something more intuitive. The problem associated with multiple selected text links on the transaction option bar had also been discussed and suggestions for improvement offered. During the two years use of the Web site no dead links were ever found.

#### **8.3.4.2 Aesthetic and minimalist design**

Some important 'design-for-simplicity-and-clarity' issues are related to the previously discussed navigational properties of the site and will not be repeated. Nielsen (1996(2):92) examined the issue of good aesthetics – 'The importance of being beautiful'. He concludes that a major consideration with the multitude of Web pages is the dramatically increased need for Web designs to stand out in the crowd – this would be a more important consideration for the bank's home page than the pages lower down in the hierarchy. In a comment on building Internet banking brand awareness on the Web, McCoy (1996:180) notes that '*One of the most glaring deficiencies is the lack of attention paid to the graphical look of the sites. By investing in the up-front design, a bank makes the statement that their Web site is a serious business vehicle that has the full faith and backing of the institution, and is not simply an experiment or project.*'

Other relevant issues and questions that should be addressed include aspects such as: Dialogs should not contain information that is irrelevant or rarely needed – every extra unit of information in a dialog competes with the relevant units of information and diminishes their relative visibility. Is the site moderate in its use of color? Does it avoid juxtaposing text and animations? Does it provide feedback whenever possible? Some of these issues have been addressed previously – such as user guidance and error messages, and consistent use of the bank's corporate colour scheme.

Garzotto and Mainetti's (1995), criteria for a usability evaluation of hypermedia design include aspects such as self-evidence or guessability, predictability or anticipation, reachability, clarity, regularity and similarity, which have been examined already. Their other criteria include Web aspects such as **informational** and **navigational richness**, which are related to aspects of enjoyability, user support and guidance, browsability, and reachability. **Reuse of objects and operations** will enhance the response of the application and the learnability of the system – Standard Bank's site supports these criteria through their use of consistent button and graphics objects throughout the set of page sequences. The different design of the home page option buttons compared to those on the Internet banking site, and those that link to service options, is unnecessary, and could confuse the user.

Bevan's (1997, URL: NPL1), usability guidelines raises questions about the planning that the bank used for the design of the site. It seems as though very little was done in terms of task analysis (as examined in chapter 6). This would have hinted at the removal of redundant screens such as the second logoff screen, and would have perhaps suggested an auto-display of balances as a first screen after a sign-on – instead only the date and time of the last access is currently displayed. The sometimes inconsistent grouping and placement of buttons also suggest that the bank should use the techniques employed by Sun (as described by Nielsen, URL: Sun1), such as card sorting and paper prototypes. This would have suggested that some of the Internet options such as 'Do my banking' are more important than the 'Demo' option in the earlier versions of the Web site. The label-change from 'Troubleshooting' to 'Help' is an improvement – cognitively this is also a change that will improve the user's emotional response to the site. The prominent and enthusiastic placement of an 'FAQ' hotspot reinforces the suspicion that the technical developer rather than interface and/or human experts executed the design of the site.

Standard Bank does seem to follow some of Bevan's suggestions for **site maintenance** – they have used user feedback to reduce the amount of graphics on their page, and seem to plan for growth in the use of their site. Concerning the use of more sophisticated (but as yet less reliable!) programming enhancements – such as **Java** applets – none of this was evident. This is perhaps appropriate as its inclusion may tarnish the security of the session, and place restrictions on the browser technology.

The use of frames on a Web page is also related to non-minimalist design. The main purpose of framesets is to subdivide the browser window. This way, only part of the window scrolls or gets replaced as a user navigates through a Web site's pages. Web builders often use framesets for toolbars – where one frame holds a static menu page, and users click items in the menu to change the content in another frame. This approach can reduce overall file size because the menu page doesn't have to be included as part of each content page.

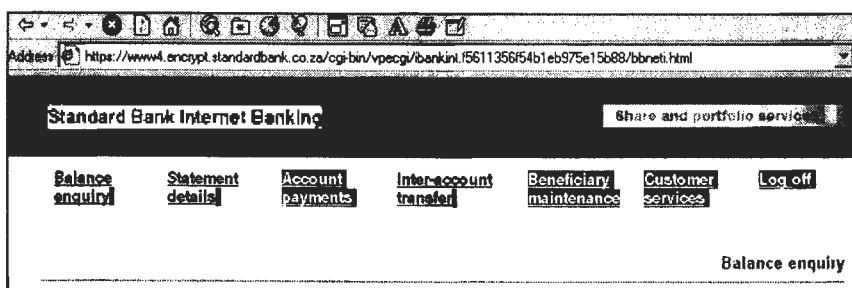
The trade-off for smaller files is increased complexity. Frames add another level of site management, since hyperlinks must not only point to the appropriate pages but also load into the appropriate frames. Another problem with frames is that most browsers can bookmark only the initial contents of a frameset. No matter how deep into the frameset users are when they set the bookmark, using the bookmark returns the browser to the initial pages of the frameset. That makes it very difficult for the user to return to specific content placeholders.

But if the site's information is well organized, with navigation only a few levels deep, frames may serve the user well. They may appreciate the easier navigation, even if they have to click through the opening page each time they visit the site.

Navigation isn't the only reason to use frames. Frame-use enhances the use of interactive tools and interactive pages that will work with most of the variety of browser versions currently in use. (Dynamic HTML works even better for creating dynamic pages, but DHTML is not supported by all browsers.) The multi-document structure of frames also makes them uniquely suited to certain JavaScript applications.

The problem with Standard Bank's frames implementation and the font size problem has already been mentioned. Frame-use causes another problem when text (such as a bank statement) is selected. Unless the bottom frame section is first given 'focus' by clicking on it a 'select all' will select the transaction menu bar rather than the statement. This problem is illustrated in figures 8.30a and b. Frames may also lead to navigational and linking inconsistencies such as displaying a new page in the frame instead of the whole window – this is however less of a problem with the secure pages as used by the banks.

**Figure 8.30a** Frames selection problem 1





**Figure 8.30b Frames selection problem 2**

Balance enquiry	Statement details	Account payments	Inter-account transfer	Beneficiary maintenance	Customer services	Log off
						Balance enquiry
<hr/>						
ACHIEVEPLAN		01- -0 2-5				
Available balance:		R 525.67+				
Latest balance :		R 525.67+				
<hr/>						
HOME LOAN		01- 51- 49-1				
Available balance:		R 4 445.15+				
Latest balance :		R 2 568.41-				

Source: [www.sbic.co.za](http://www.sbic.co.za) (1999).

### 8.3.4.3 Match between system and the real world

This implies that the system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Real-world conventions should be followed, and information should appear in a natural and logical order. All of these aspects had been addressed in previous discussions – such as the button terminology used and the aspect of conceptual model matching. It is important however that this real-world and system alignment should nonetheless allow for the opportunity to take advantage of the screen-based interaction – it was commented on previously that the blind duplication of paper statements onto the computer screen are wasteful of the opportunities offered by the different technology.

### 8.3.4.4 Consistency and standards

This means that users should not have to wonder whether different words, situations, or actions mean the same thing. Platform conventions should be followed. It also implies that the use of additional security software and the special software as used by Mercantile Bank act against this property. Consistency had been discussed in detail previously, and the bank succeeds reasonably well in maintaining a common look and feel across its Internet banking Website. However when comparing this site with its Electronic Business Banking (EBB) Web site the two sites appear (and probably behave) significantly differently. Standards outside of the application-configurable characteristics are ensured by the required access through a uniform browser environment – at least the popular Netscape and Microsoft browsers share in more commonalities than differences!

Schutzer (1997:169) comments on the aspect of electronic commerce standards – the Internet has shifted the standard setting process towards open platforms which will ensure that future banking clients will access the services in a familiar, convenient, and secure manner.

## 8.4 Conclusion

Standard Bank requires online help to be available during the transaction sequence and this should preferably be context sensitive. Although the user can open two separate browser windows – one which is used to conduct the transaction and the other in the unsecured Web section from where the help options can be accessed – this would require a degree of pre-planning by the user that is unlikely to be realized in practice. It also presumes that the user will expect the need for help during the transaction session, which may not always be true. Their help pages could be better structured and should be indexed – this is one of the few cases where a frame structure works well and is acceptable. An interactive demonstration facility should be reintroduced as this aids the learning experience through trial and error. Standard Bank should develop a small printed user guide for their Internet banking service. They should make increased use of other feedback techniques such as an improved use of the status and title bars, and tool-tips. The transaction indication feedback is poorly implemented and ambiguous.

Predictability, familiarity, generalizability and consistency properties argue against the use of security software or customized software solutions, and argue for access from a standard Web browser environment.

The suspicion exists that for the local banks technical developers rather than human computer interface experts were responsible for the technical, layout, and interface aspects of their banking sites. In contrast contributions from such human-oriented designers, and graphical presentation experts, are evident during a study of the American bank's Web sites.

The heuristic usability evaluation of Standard Bank's Web pages indicates emphatically that general usability principles and properties can be applied with good effect to the hypermedia-hypertext designs as found on the World Wide Web. But for this type of specialized application, i.e. online banking, a restricted or qualified implementation of some of these usability properties is argued for. Usability properties such as flexibility, browsability, reachability, system versus user control, multithreading, user and system pre-emption and dialog initiative, can not be promoted to the same extent for these secure, high risk systems when compared to other (insecure or less risky) applications. All the properties that constitute the principle of learnability such as consistency, positive engagement, mental model support, metaphors, clarity, and predictability, do apply meaningfully and strongly to this type of application.

Many of the Web-specific usability guidelines are perhaps only renamed or re-labeled general usability properties – but even this could serve the useful purpose of drawing attention to a specific aspect of a given property. What is different is that the importance of some of the general criteria had been amplified for these hypermedia applications – an aspect such as navigation which is related to usability properties such as reachability, user guidance, and browsability is one example of this.

For many Web entertainment sites and applications the properties of enjoyability and responsiveness are disproportionately important, whilst other sites that value their information content should be evaluated more thoroughly in terms of properties such as visual clarity, user guidance, and task conformance. The importance of navigational usability is also promoted by Instone (URL: Webreview2). He added Web-specific comments to Nielsen's (1993) list of ten usability heuristics, and in his introduction he pertinently states that: *'the overriding theme for applying these (Nielsen's ten) heuristics to the Web is to use links effectively'*.

The examination of the Internet-banking facilities offered by the four banks was illuminating. Standard Bank improved the usability of their site considerably between May 1997 and April 1999 – especially for aspects such as reduced navigational complexity, enhanced task support, improved visual clarity, and faster response. Their site must rank with First National Bank as the most usable and functional of the local sites evaluated. On a more subjectively intuitive note, both the FNB and Standard Bank sites visually appear light and cheerful, and those from ABSA and Nedbank darker and gloomier.

This (heuristic) approach to Web site evaluation should be considered exploratory in nature – a more structured evaluation, that for example includes a proper task analysis of the various (usually simple), tasks executed during online banking, can yield other valuable usability insights related to task conformance and user conceptual support. The research should also be expanded to include a more detailed study of the now large number of overseas online banking organizations. What was found of particular significance and help during this evaluation was the observed and documented change in Standard Bank's site during the evaluation period – the comparison of 'old' versus 'new' site structure yielded usability information that would have been impossible to obtain otherwise. The approach taken by Levi and Conrad (1996) where severity ratings are assigned to usability problems and as commented on in chapter 7 also has merit, and should be developed further.

Given the extremely rapid rate of development, use, adoption and acceptance of these Web-based commercial facilities, a formalized and structured in-depth research study would form an ideal and very topical subject for advanced level research in human computer interaction as well as in applied commerce.

## 8.5 Appendices: Chapter 8 heuristic evaluation

Appendix	Title	Figure
8.A	Error generating conditions: Standard Bank Web pages	8.A.1 – 8.A.14
8.B	User guidance and help: Standard Bank Web pages	8.B.1 – 8.B.11
8.C	Security: Standard Bank Web pages	8.C.1 – 8.C.16
8.D	Page components: Standard Bank sign-on page	8.D.1 – 8.D.3
8.E	Bank transactions: Standard Bank (South Africa)	8.E.1 – 8.E.48
8.F	Bank transactions: First National Bank (FNB)	8.F.1 – 8.F.18
8.G	Bank transactions: Nedbank (South Africa)	8.G.1 – 8.G.12
8.H	Bank transactions: Amalgamated Banks of South Africa (ABSA)	8.H.1 – 8.H.16
8.I	Bank transactions: Other banks May 1998	8.I.1 – 8.I.51

## APPENDIX 8.A

## Error generating conditions: Standard Bank Web pages

Environment and Date	Figure	Title
<b>September 1997</b> <b>Actual Use</b> Netscape Navigator 3.0 MS Windows 3.11	8.A.1	Connection authentication error
	8.A.2	Connection broken error
<b>November 1998–April 1999</b> <b>Actual use</b> Netscape Navigator 4.05 or MS Internet Explorer 4.01 MS Windows 98	8.A.3	Internet Explorer picture/Security problem notice
	8.A.4	Internet Explorer picture/Security problem solved
	8.A.5	Menubar font size problem
	8.A.6	Menubar multiple selection problem
	8.A.7	Log-in error 1
	8.A.8	Log-in error 2
	8.A.9	Log-in error 3
	8.A.10	General error - navigation
	8.A.11	Transaction error - beneficiaries
	8.A.12	Transaction error – minimum amount
	8.A.13	Transaction error – balance unavailable
	8.A.14	JavaScript error during registration for online banking

Figure 8.A.1

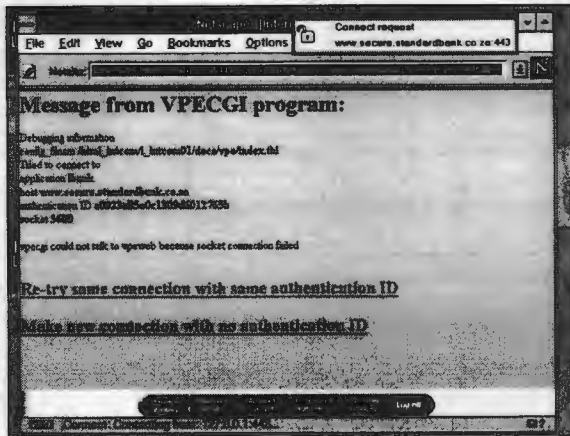


Figure 8.A.2



Figure 8.A.3

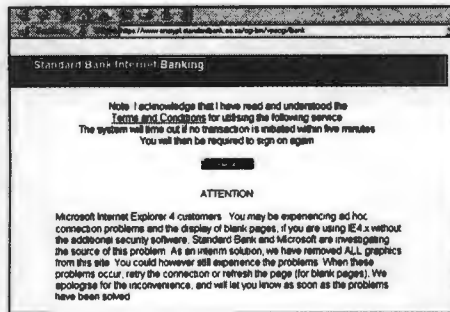


Figure 8.A.4

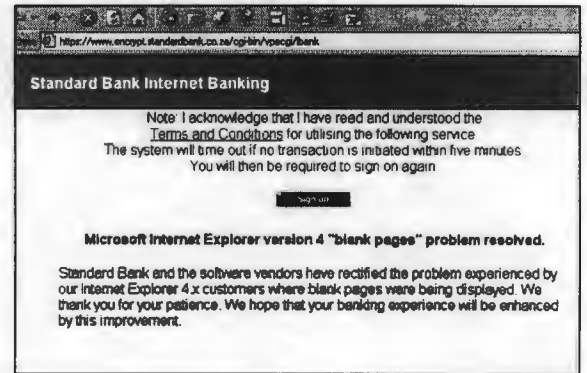


Figure 8.A.5

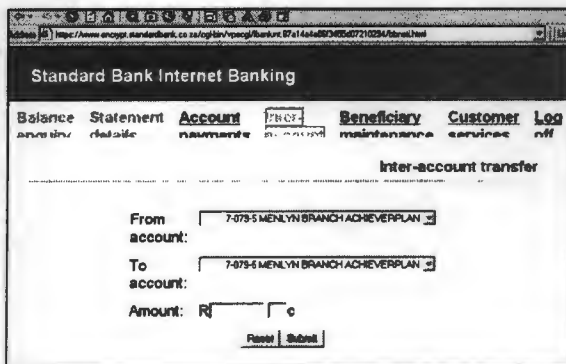


Figure 8.A.6



Figure 8.A.7

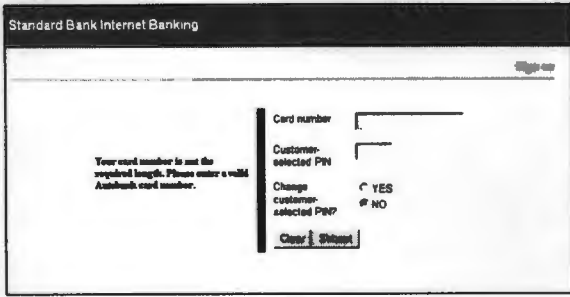


Figure 8.A.8

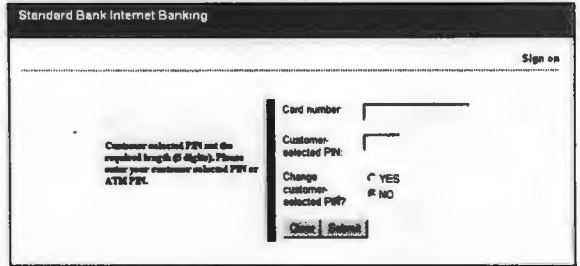


Figure 8.A.9

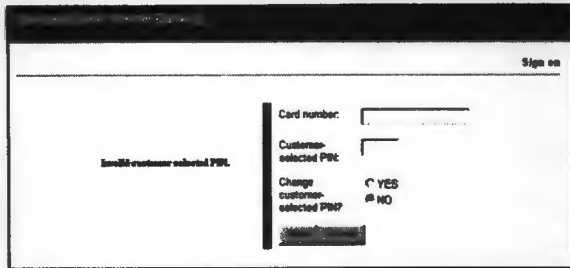


Figure 8.A.10



Figure 8.A.11

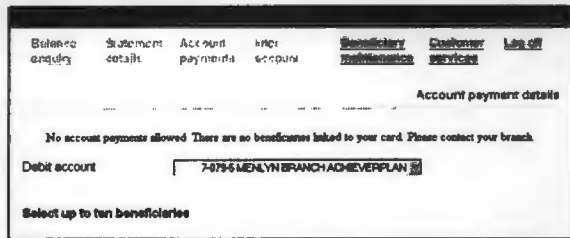


Figure 8.A.12

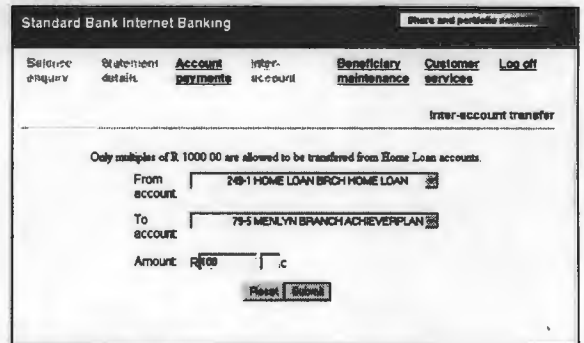
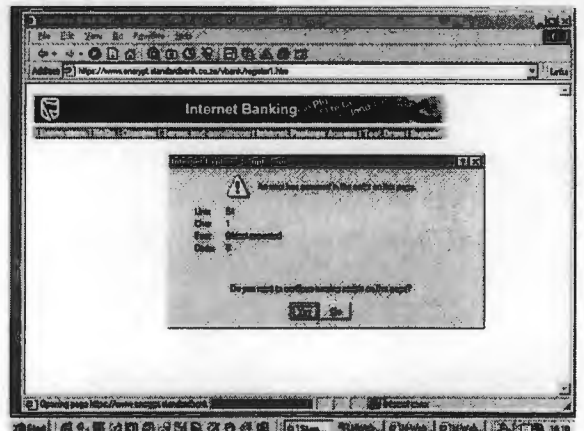


Figure 8.A.13

Balance enquiry	Statement details	Account payments	Inter-account transfer	Beneficiary maintenance	Customer services	Log off
ACHIEVERPLAN		7-	9-5			
Available balance:			R 111.25+			
Latest balance :			R 111.25+			
HOME LOAN		1-2	-1			
Available balance:			R 617.13+			
Latest balance :			R 492.91-			
STUDENT LOAN		1-9	4-5			
Balance on account unavailable. Please contact the branch staff.						

Figure 8.A.14



**APPENDIX 8.B****User guidance and help: Standard Bank Web pages**

<b>Category and Date</b>	<b>Figure</b>	<b>Title</b>
<b>Help page options</b> January 1999 - May 1999	8.B.1 8.B.2 8.B.3 8.B.4 8.B.5	Support option Internet access questions Terms and Conditions Charges Learn more
<b>FAQ pages</b> January 1999 and July 1998	8.B.6 8.B.7	FAQ New page 1999 FAQ Old page 1998
<b>Demonstration pages</b> January 1999	8.B.8 8.B.9 8.B.10 8.B.11	Demonstration introduction page Demonstration sign-on page Demonstration transaction 1 page Demonstration transaction 2 page



Figure 8.B.1

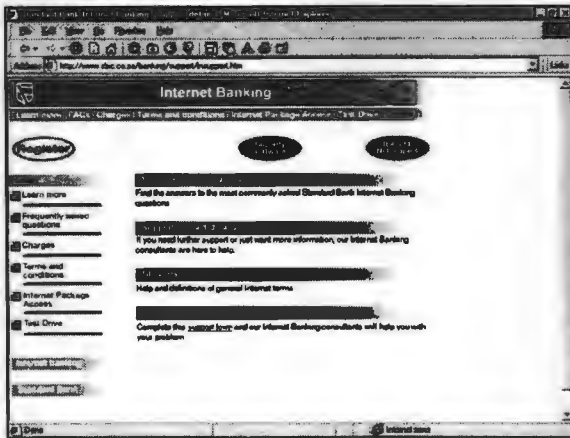


Figure 8.B.2

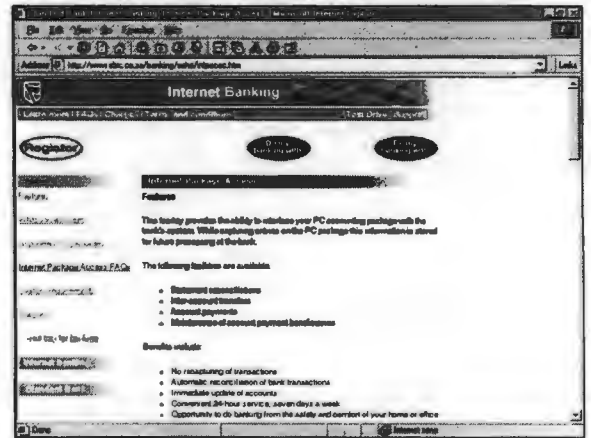


Figure 8.B.3

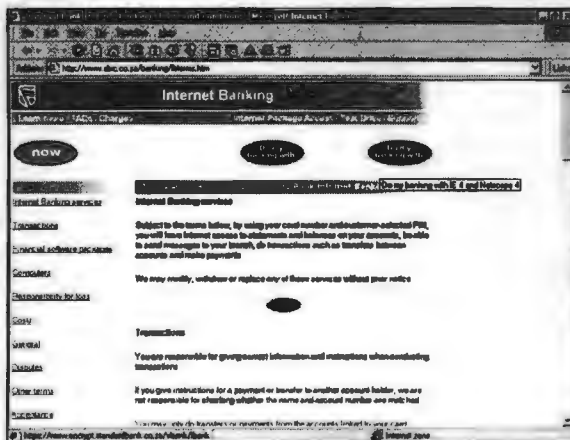


Figure 8.B.4

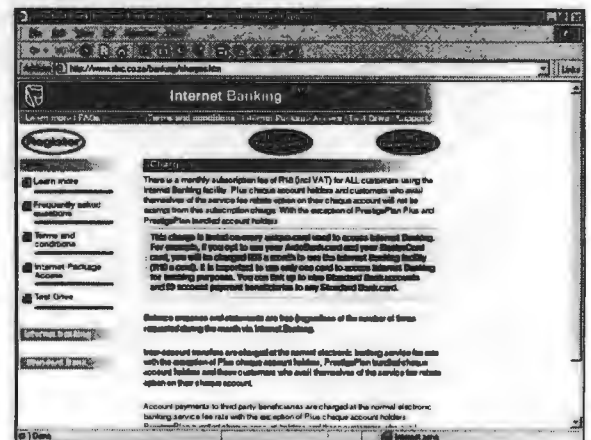


Figure 8.B.5

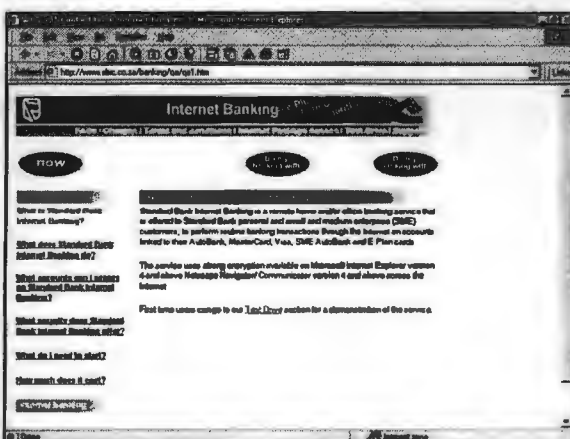


Figure 8.B.6

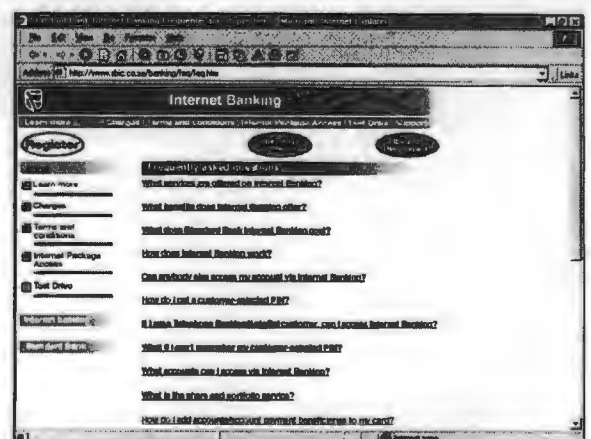


Figure 8.B.7

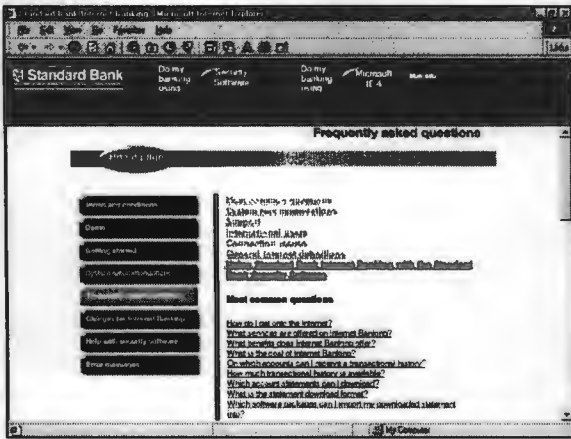


Figure 8.B.8

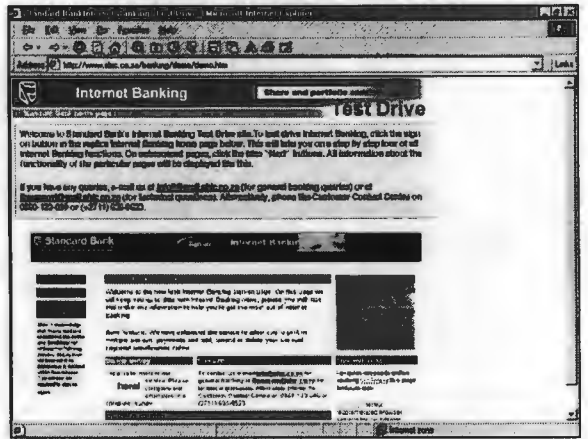


Figure 8.B.9

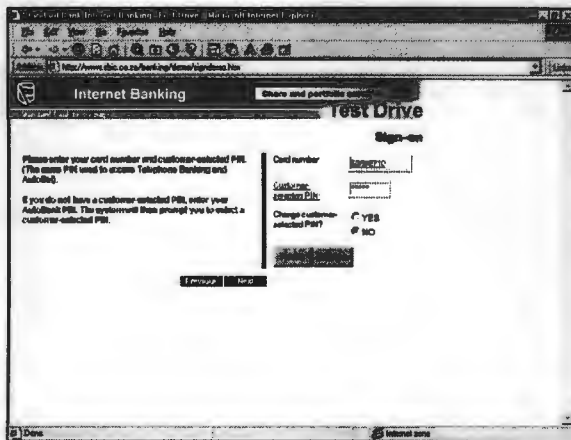


Figure 8.B.10

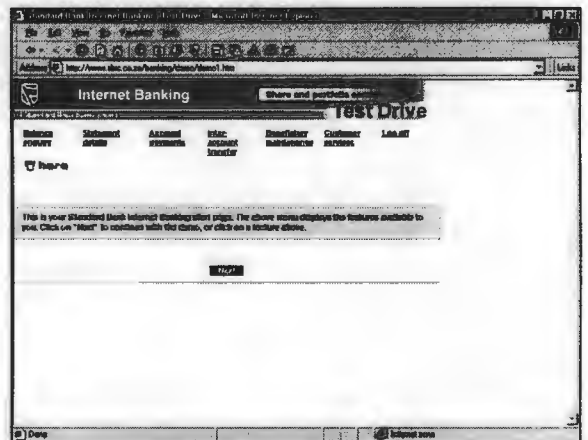
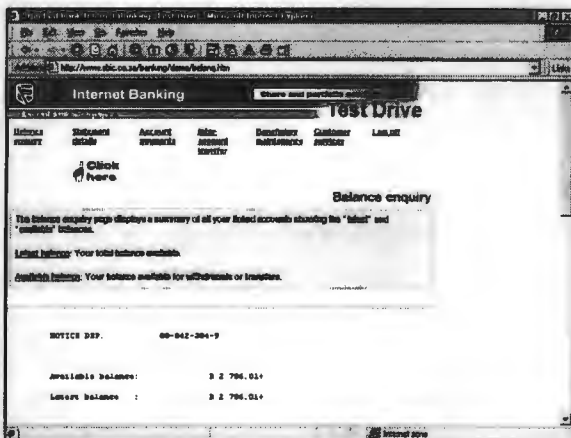


Figure 8.B.11



## APPENDIX 8.C

### Security: Standard Bank Web pages

<b>Environment and Date</b>	<b>Figure</b>	<b>Title</b>
Windows 95/98 1998/1999	8.C.1	MS Internet Explorer 4.01 secure connection
	8.C.2	MS Internet Explorer 4.01 non-secure connection
Windows 3.1 1997 Windows 3.1 1997 Windows 95/98 1998/1999	8.C.3	Netscape Navigator 3.0 secure connection
	8.C.4	Netscape Navigator 3.0 non-secure connection
	8.C.5	Netscape Navigator 4.5 secure connection
	8.C.6	Netscape Navigator 4.5 non-secure connection
	8.C.7	Netscape 4.5 about box security details
	8.C.8	MS Internet Explorer 4.01 about box security details
	8.C.9	Standard Bank security software failure
	8.C.10	Standard Bank security software non-secure
	8.C.11	Secure connection information dialog box
	8.C.12	Non-secure connection information dialog box
	8.C.13	Secure connection dialog box placement
	8.C.14	Secure connection document no details available
	8.C.15	Secure connection document details dialog box
	8.C.16	Standard Bank security information help page

Figure 8.C.1

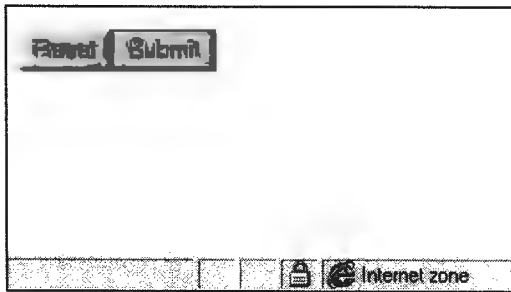


Figure 8.C.2

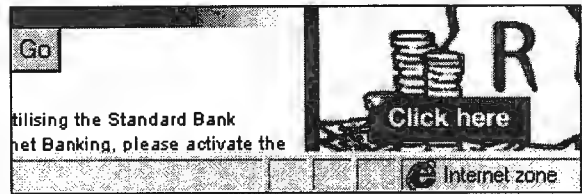


Figure 8.C.3



Figure 8.C.4



Figure 8.C.5

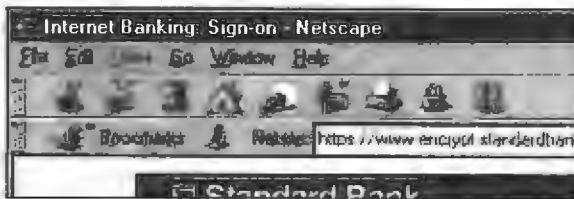


Figure 8.C.6



Figure 8.C.7

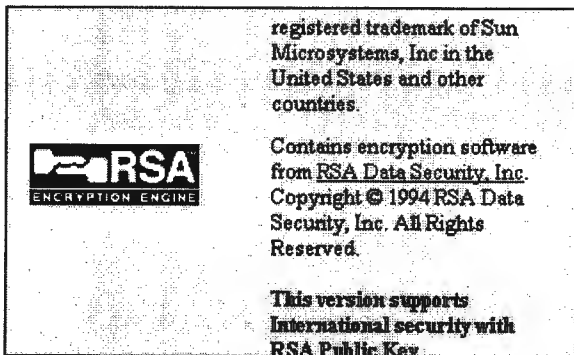


Figure 8.C.8

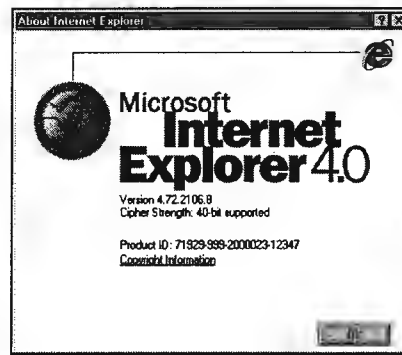


Figure 8.C.9

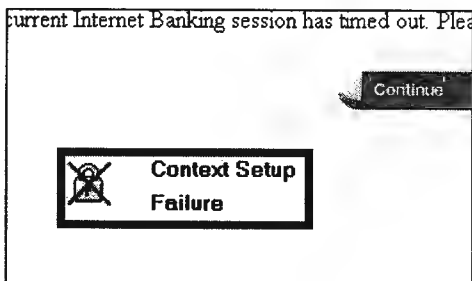


Figure 8.C.10



Figure 8.C.11

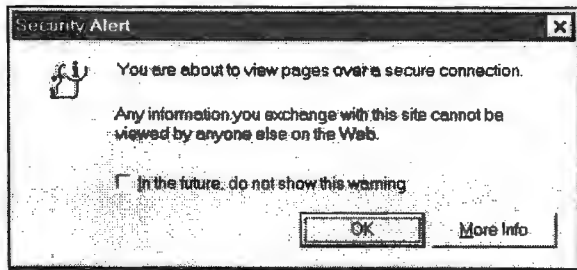


Figure 8.C.12

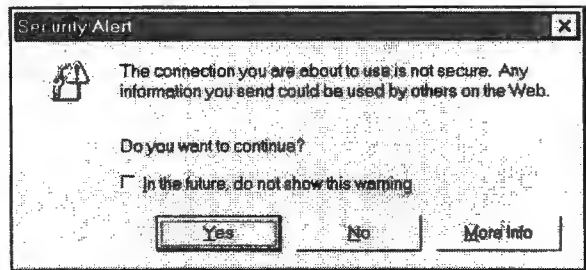


Figure 8.C.13



Figure 8.C.14

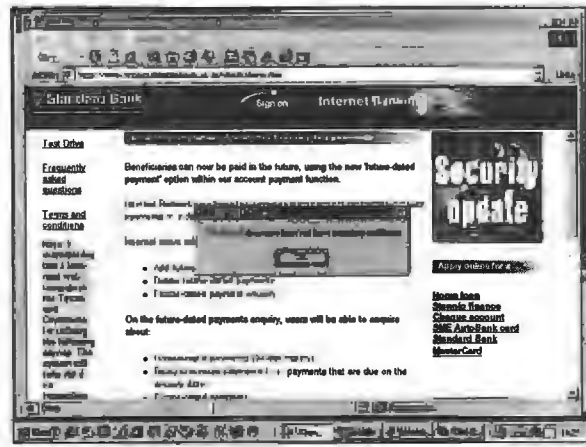


Figure 8.C.15

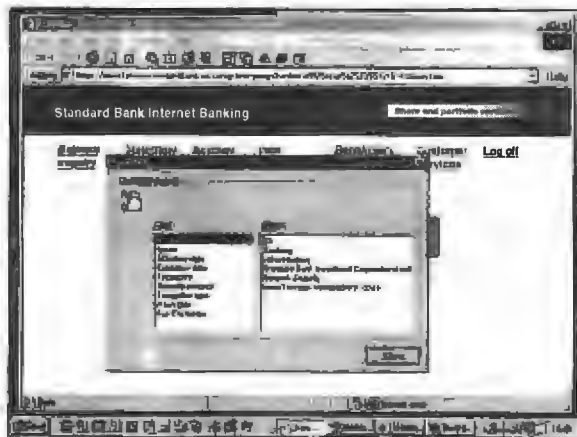
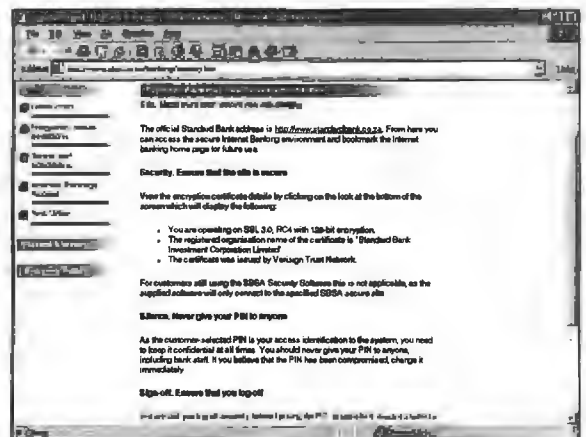


Figure 8.C.16



APPENDIX 8.D

Page components: Standard Bank sign-on page

Figure 8.D.1, 2, 3 Standard Bank Sign-on page 1997 - 1999					
Figure 8.D.1 August 1997		Figure 8.D.2 November 1997		Figure 8.D.3 November 1998 - current	
Right-hand Frame: 5 Objects	Size	Single Frame: 7 objects	Size	Single Frame: 1 object	Size
	24		4		0
	1		10		
	1		1	Note: These two buttons are not graphics objects but are form-generated.	
	1		1		
			1		
			4		
HTML Source File	2	HTML Source File	2	HTML Source File	2.5
<b>Total Size RH Frame (kb):</b>	<b>27.4</b>	<b>Total Size (kb):</b>	<b>19.4</b>	<b>Total Size (kb):</b>	<b>2.5</b>
Left-hand Frame: 9 Objects					
	1				
	5				
	5				
	4				
	4				
	5				
	5				
	4				
	2				
<b>Total Size LH Frame (kb):</b>	<b>29.7</b>				
<b>Total Size (kb):</b>	<b>57</b>	<b>Total Size (kb):</b>	<b>19</b>	<b>Total Size (kb):</b>	<b>3</b>

Source: www.sbic.co.za

Note: The sizes listed do not take compression during download into account. The HTML source files will be significantly compressed, whilst the graphics object will benefit very little from any compression.

## APPENDIX 8.E

## Bank transactions: Standard Bank (South Africa)

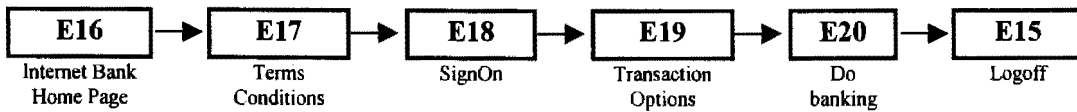
Environment and Date	Figure	Title
September 1997 to current	8.E.1 to 8.E.4	Web page navigation sequences
September 1997 Online Demo MS Explorer 3.02 MS Windows 95	8.E.5 8.E.6 8.E.7 8.E.8 8.E.9 8.E.10 8.E.11 8.E.12 8.E.13 8.E.14	Standard Bank Home Page Standard Bank Internet Banking Home Page Terms and Conditions Sign-on Banking Services Options Balance Enquiry Account Selection Account Statement Detail Inter-account Transfer Transfer Confirmation
September 1997 Actual Use Netscape 3.0, MS Win 3.11	8.E.15	Log-off Screen
November 1997 On-line Demo MS Explorer 3.02 MS Windows 95	8.E.16 8.E.17 8.E.18 8.E.19 8.E.20 8.E.21 8.E.22 8.E.23 8.E.24	Standard Bank Internet Banking Home Page Terms and Conditions Sign-on Welcome and Banking Services Options Account Selection Account Statement Detail Inter-account Transfer Transfer Confirmation De-registration Screen
November 1998 Actual use MS Explorer 4.01 and MS Windows 98	8.E.25 8.E.26 8.E.27 8.E.28 8.E.29 8.E.30 8.E.31 8.E.32	Standard Bank Home Page Standard Bank Internet Banking Home Page Terms and conditions Sign-on Welcome and Banking services options page Balance option Balance display Log-off
January-May 1999 Actual use MS Explorer 4.01 and MS Windows 98	8.E.33 8.E.34 8.E.35 8.E.36 8.E.37 8.E.38 8.E.39 8.E.40 8.E.41 8.E.42 8.E.43 8.E.44 8.E.45 8.E.46 8.E.47 8.E.48 8.E.49 8.E.50 8.E.51 8.E.52	Standard Bank Home Page – default option Standard Bank Home Page – all options Standard Bank Internet Banking Home Page Terms and conditions Inter-account transfer – select accounts Inter-account transfer – confirm to proceed Inter-account transfer – completed confirmation Statement selection Account payments Beneficiaries maintenance Beneficiaries selection Beneficiaries selection - choices Customer services Customer details E-mail Change PIN Standard Bank Business Electronic Banking (BEB) BEB sign-on page BEB Internet technology pledge BEB transaction

**Figures 8.E.1 to 8.E.4 Standard Bank's Web page navigation sequence**

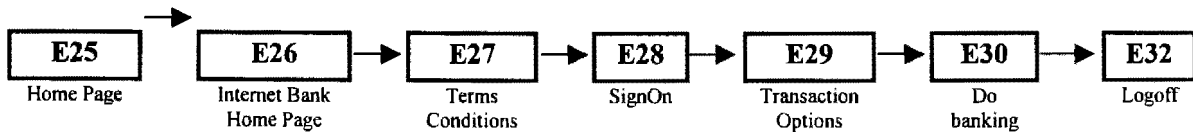
**Figure 8.E.1 May - October 1997:**



**Figure 8.E.2 November 1997 - October 1998:**



**Figure 8.E.3 November 1998 - January 1999:**



**Figure 8.E.4 January 1999 - current:**

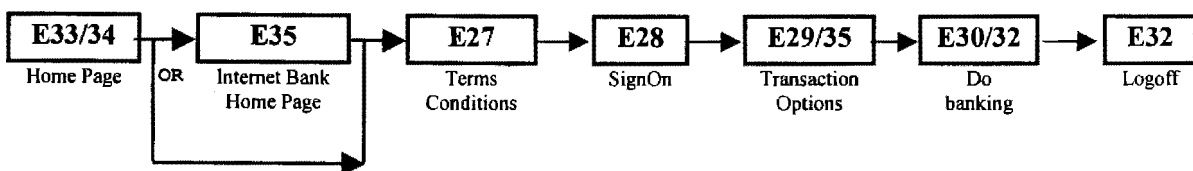




Figure 8.E.5

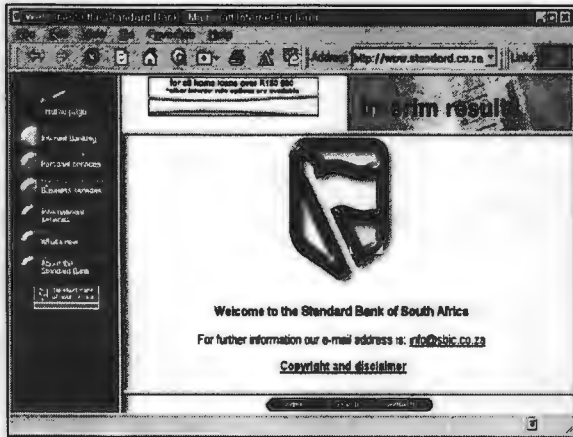


Figure 8.E.6

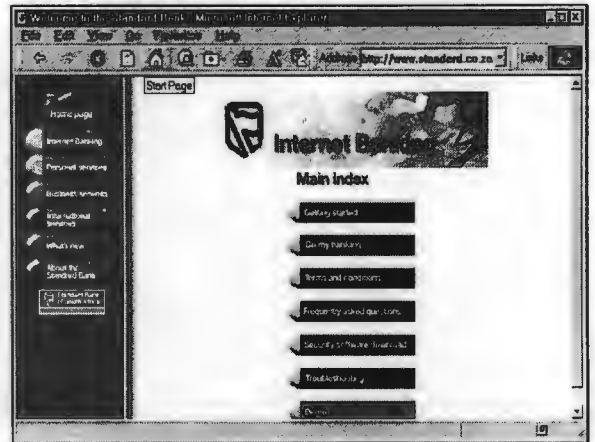


Figure 8.E.7



Figure 8.E.8

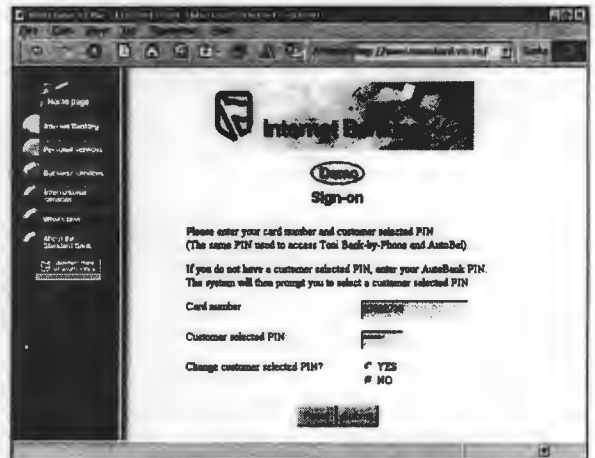


Figure 8.E.9

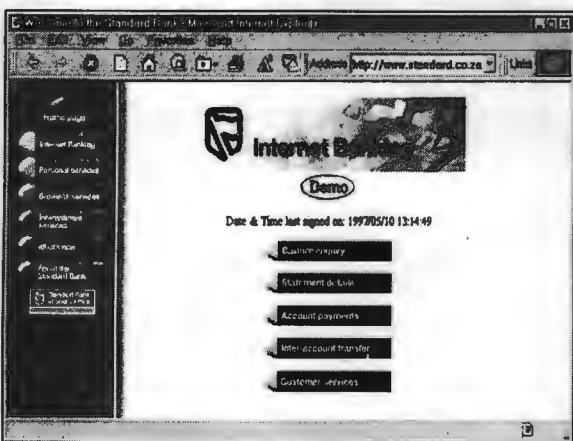


Figure 8.E.10

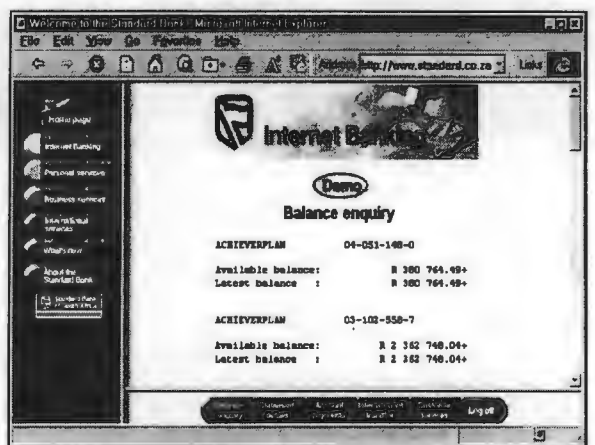


Figure 8.E.11



Figure 8.E.12



Figure 8.E.13

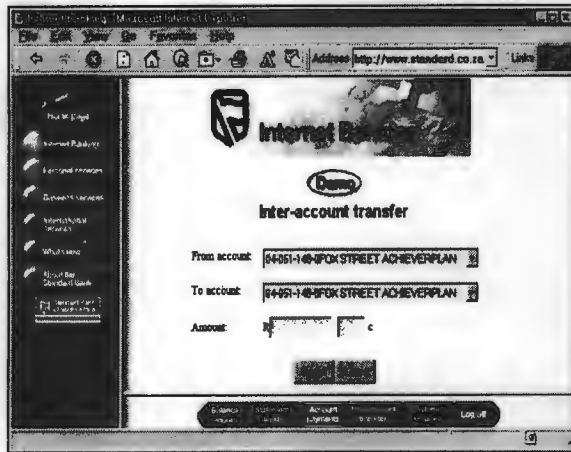


Figure 8.E.14

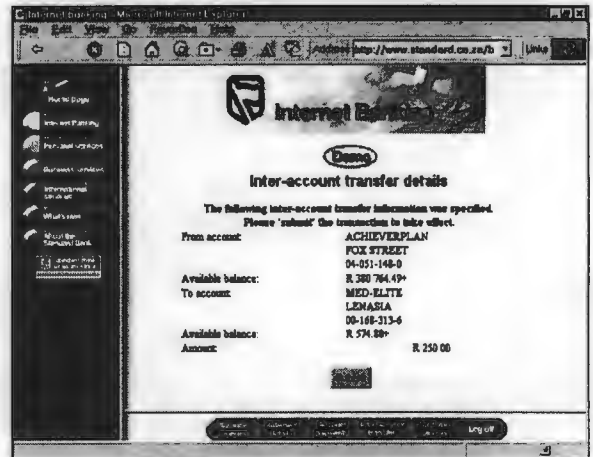


Figure 8.E.15



Figure 8.E.16



Figure 8.E.17

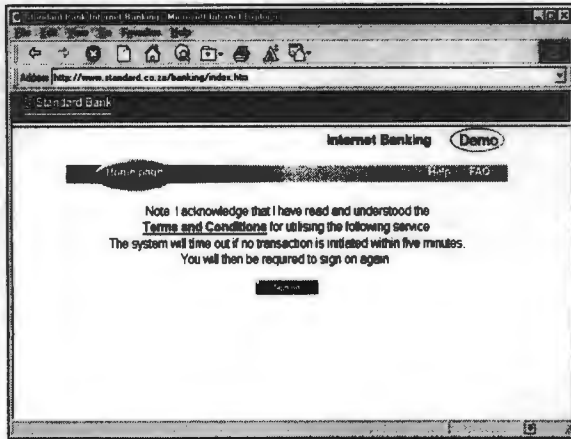


Figure 8.E.18

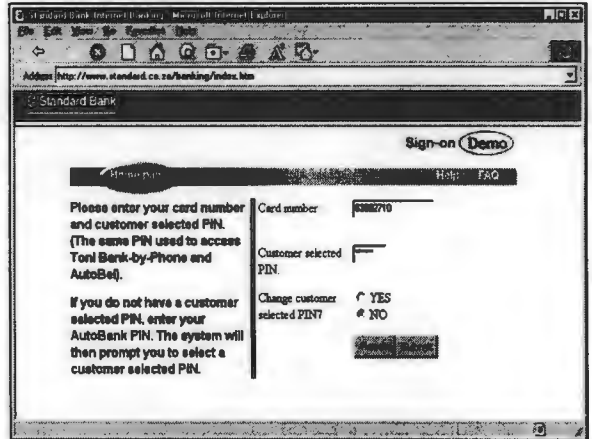


Figure 8.E.19

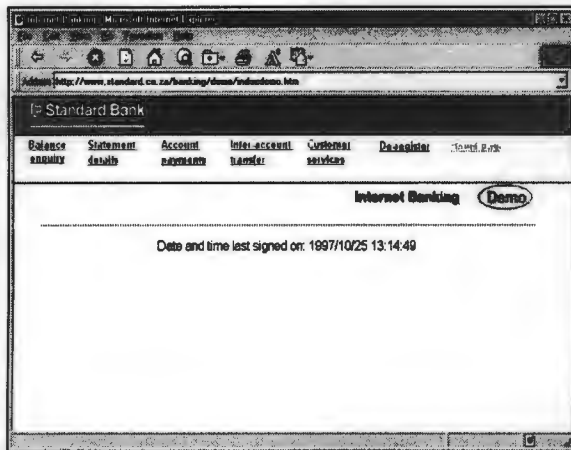


Figure 8.E.20

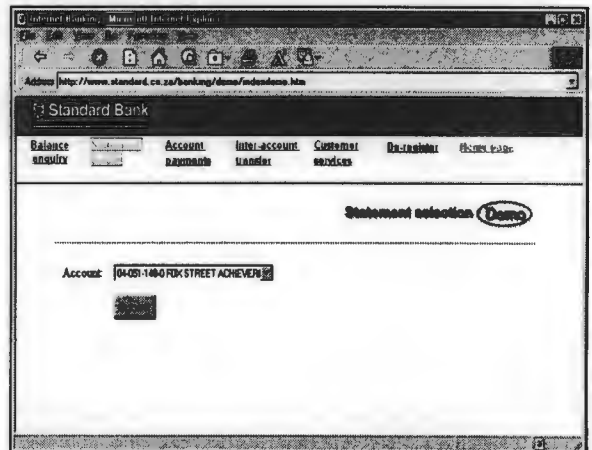


Figure 8.E.21

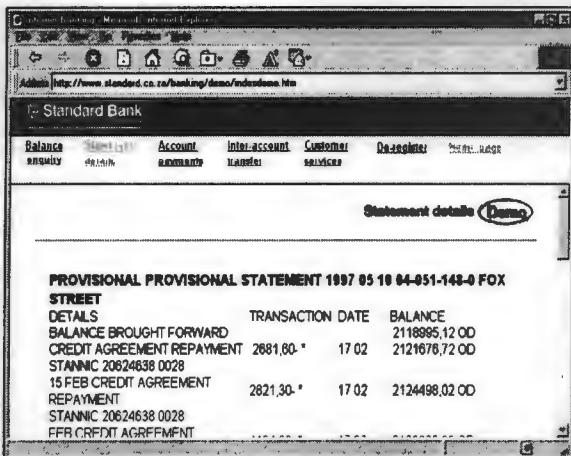


Figure 8.E.22

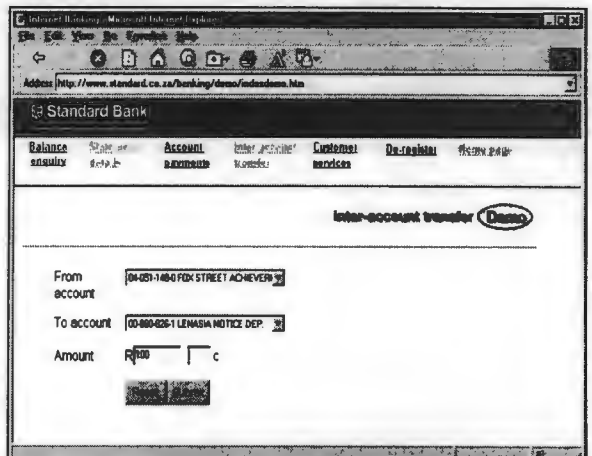


Figure 8.E.23

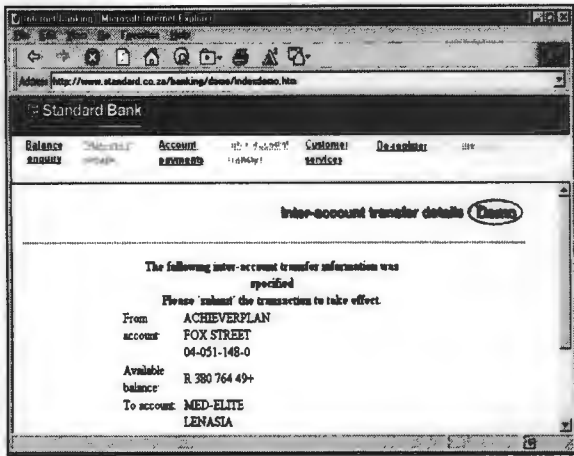


Figure 8.E.24

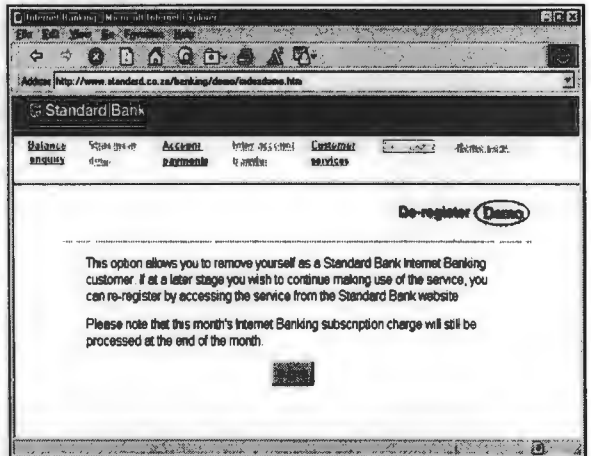


Figure 8.E.25



Figure 8.E.26



Figure 8.E.27

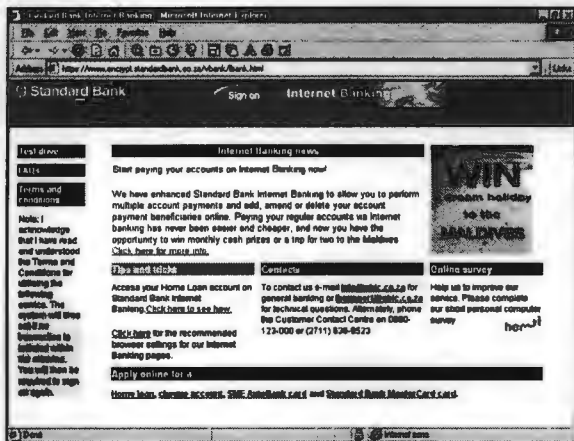


Figure 8.E.28

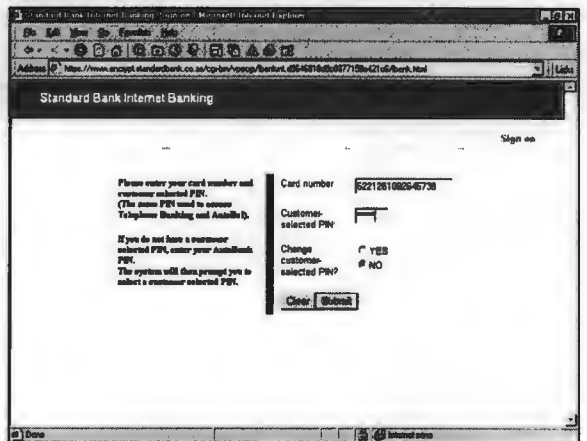


Figure 8.E.29



Figure 8.E.30

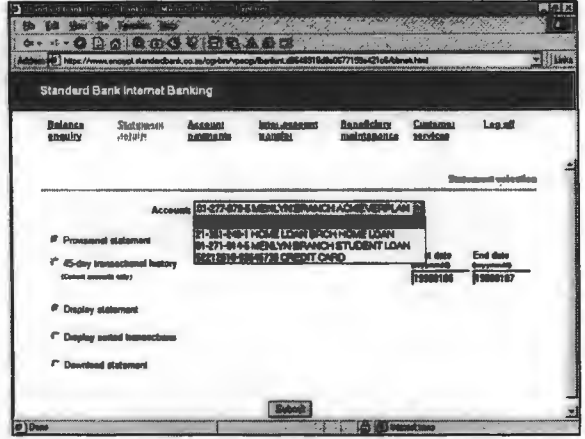


Figure 8.E.31

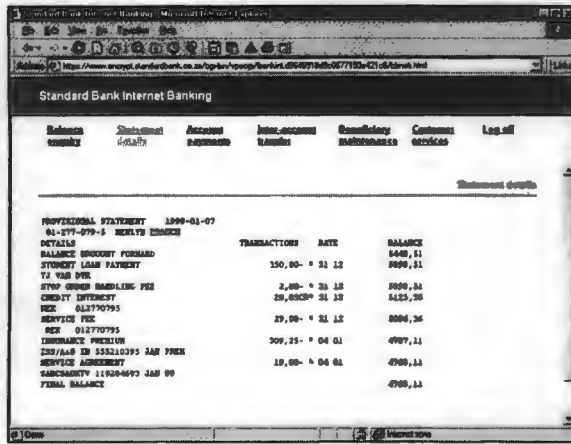


Figure 8.E.32

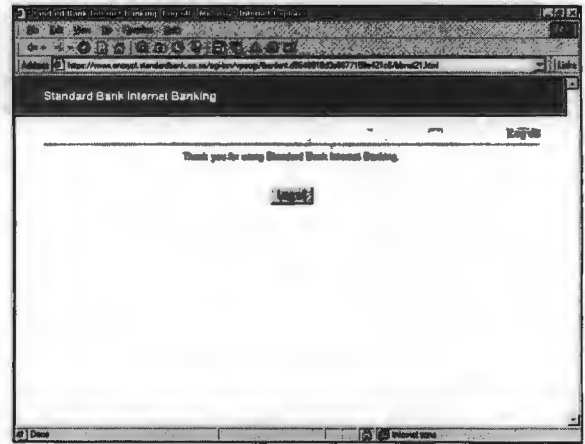


Figure 8.E.33

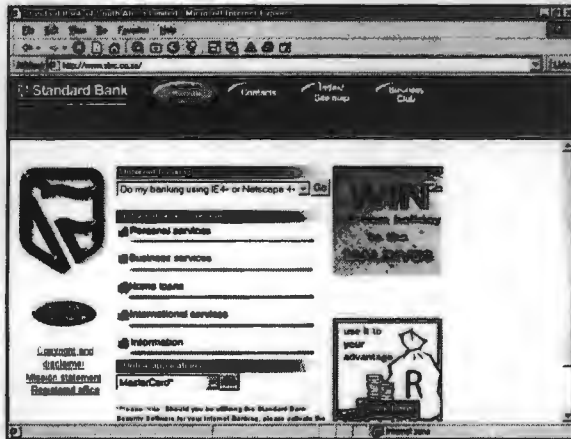


Figure 8.E.34

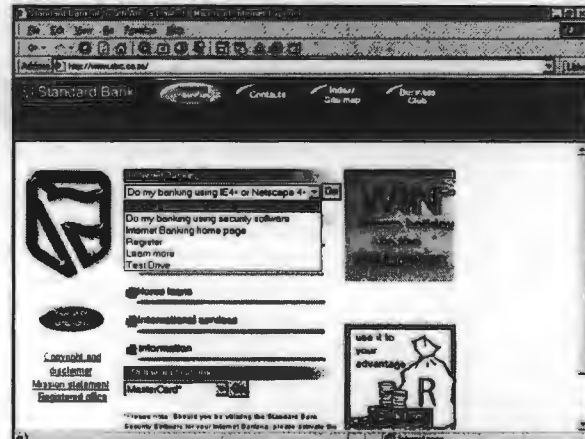


Figure 8.E.35

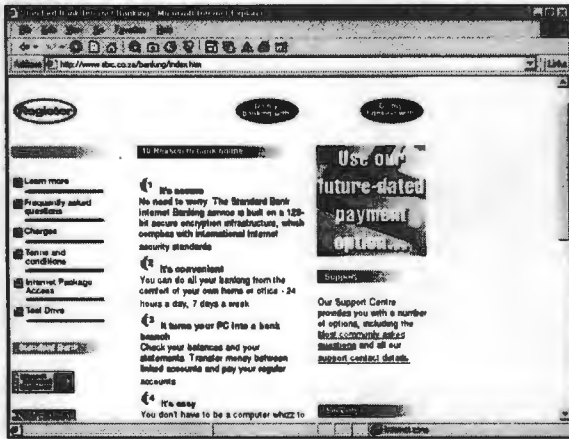


Figure 8.E.36

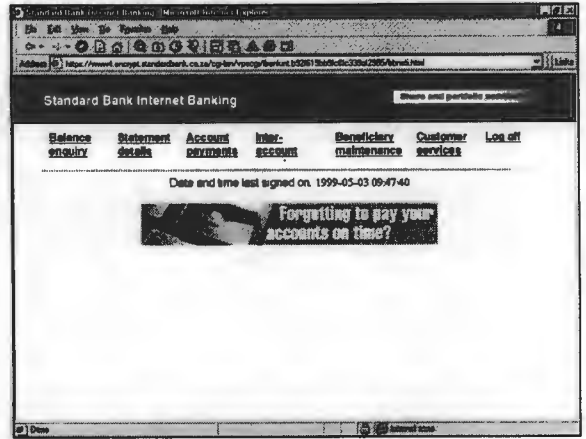


Figure 8.E.37

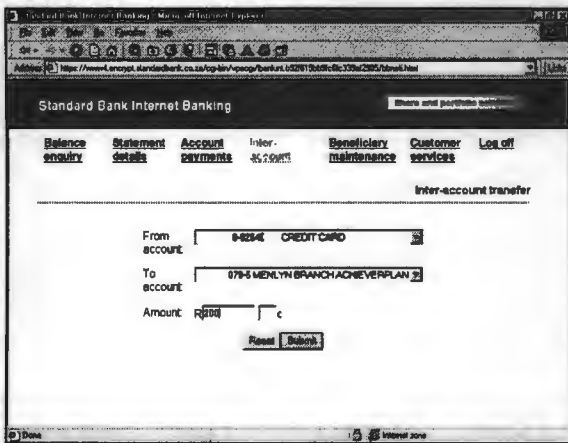


Figure 8.E.38

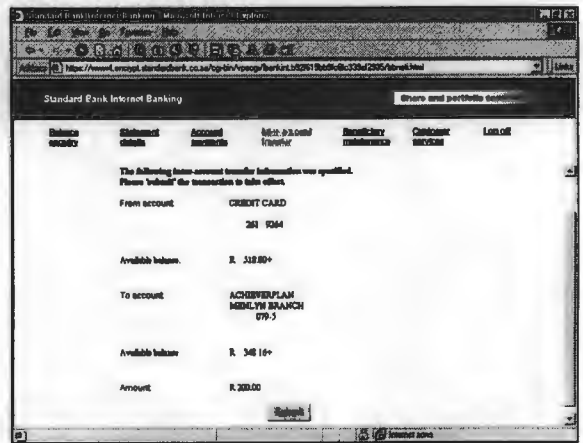


Figure 8.E.39

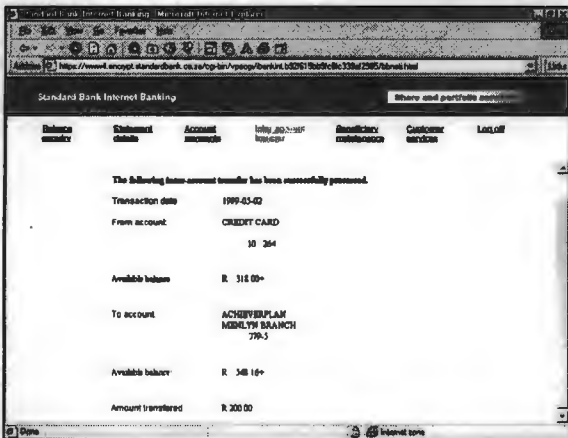


Figure 8.E.40

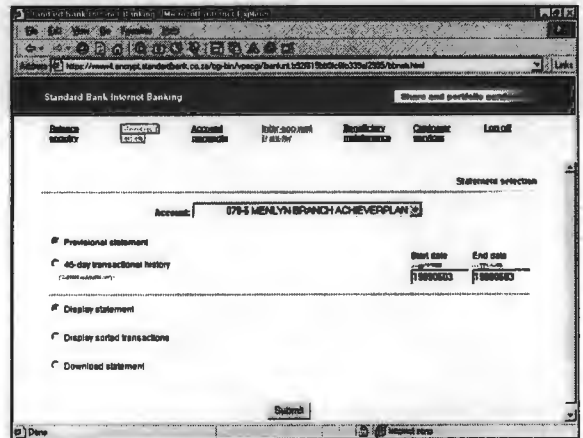




Figure 8.E.41

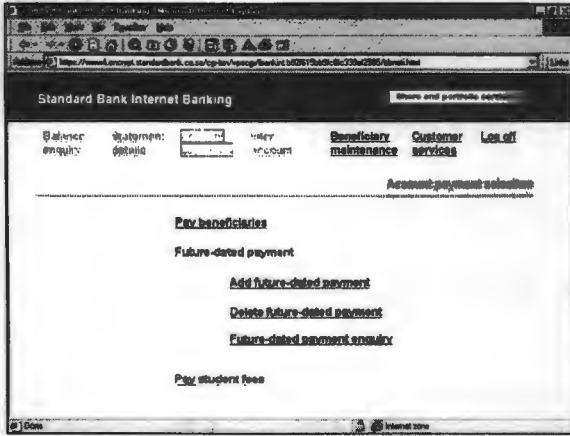


Figure 8.E.42

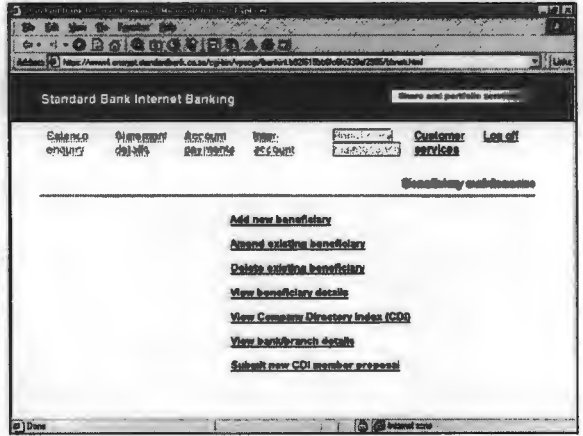


Figure 8.E.43

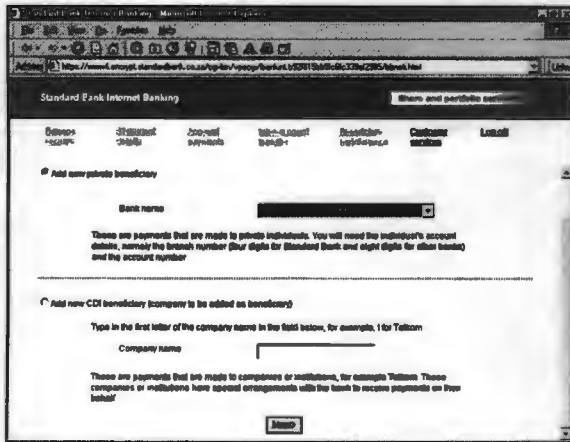


Figure 8.E.44

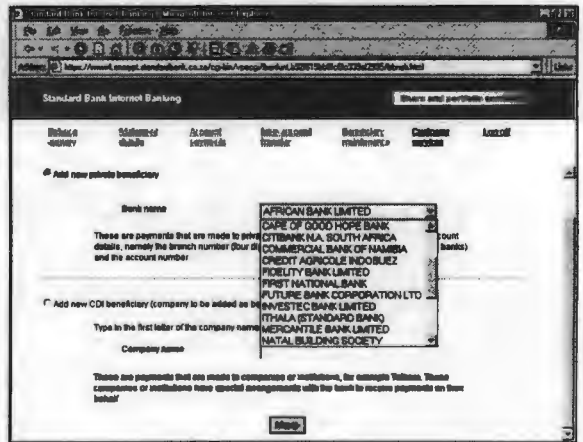


Figure 8.E.45

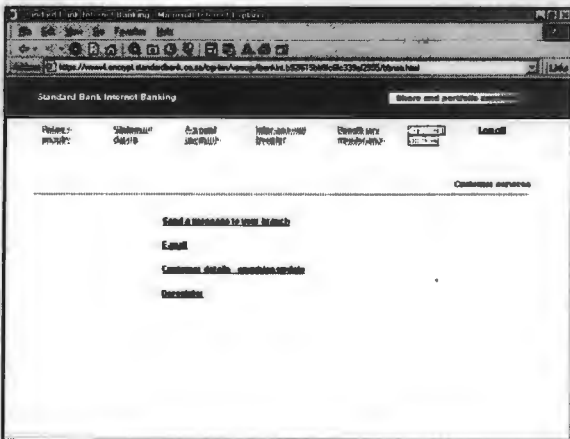


Figure 8.E.46

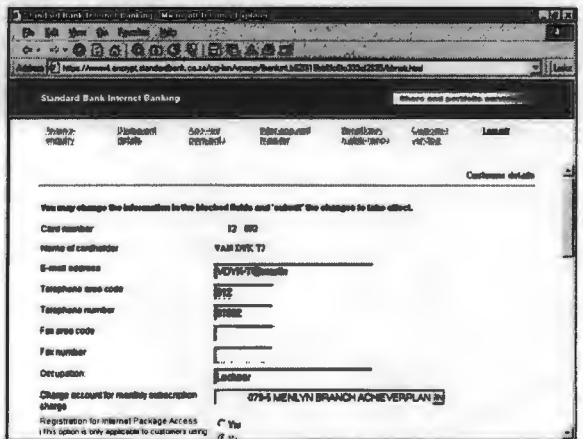


Figure 8.E.47

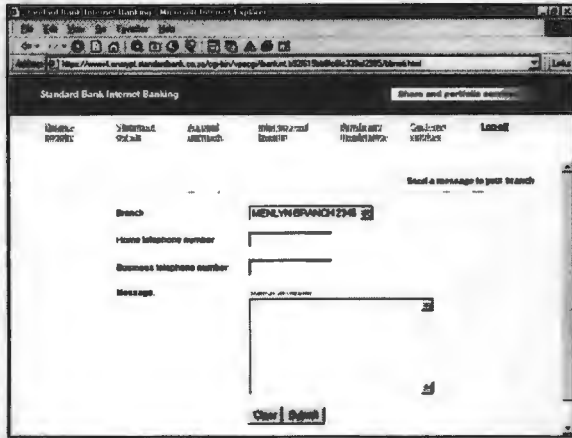


Figure 8.E.48

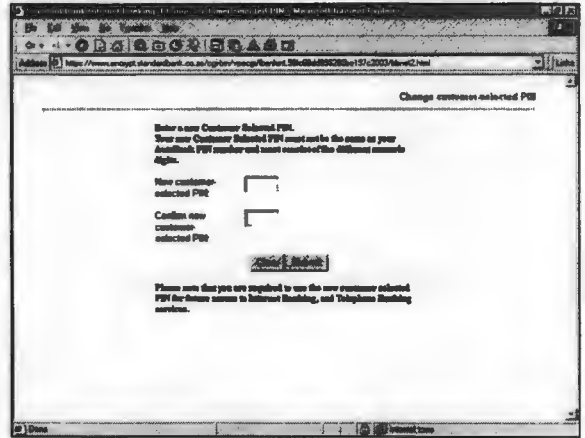


Figure 8.E.49

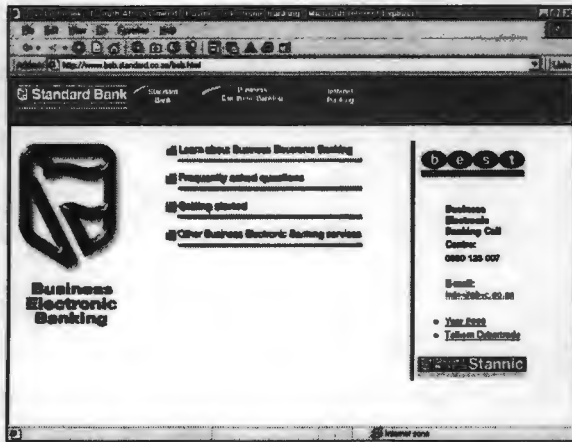


Figure 8.E.50

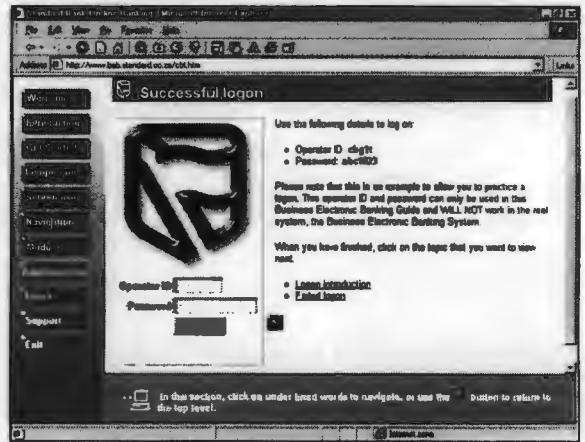


Figure 8.E.51

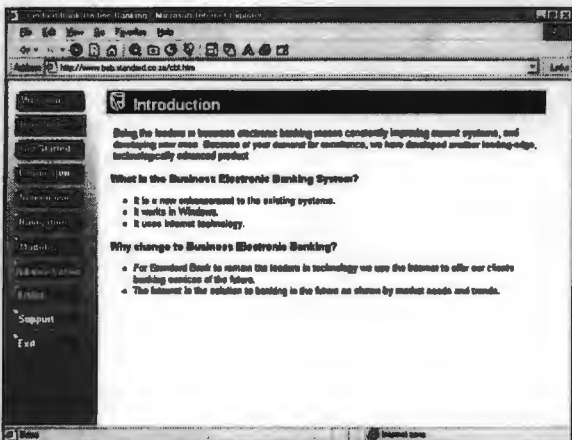
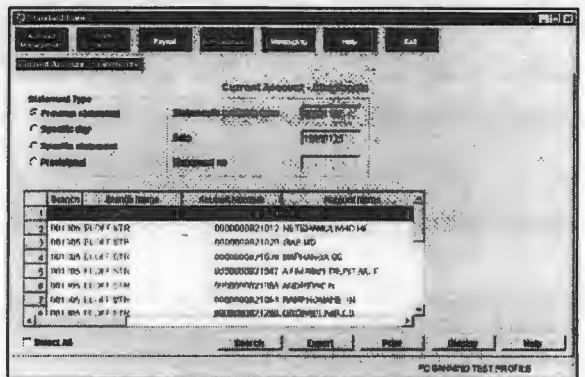


Figure 8.E.52





## APPENDIX 8.F

## Bank transactions: First National Bank (FNB)

Environment and Date	Figure	Title
<b>September 1997</b> <b>Demonstration</b> MS Internet Explorer 3.02 MS Windows 95	8.F.1	FNB home page
	8.F.2	FNB Internet bank home page: First Online
	8.F.3	Demonstration page: sign-on
	8.F.4	Demonstration page: transaction options
	8.F.5	Demonstration page: statement selection
	8.F.6	Demonstration page: statement display
	8.F.7	Demonstration page: account transfer select
	8.F.8	Demonstration page: account transfer confirmed
<b>May 1998</b> MS Internet Explorer 4.01 MS Windows 98	8.F.9	FNB home page
	8.F.10	FNB Internet bank home page: First Online
	8.F.11	FNB Sitemap
<b>April 1999</b> MS Internet Explorer 4.01 MS Windows 98	8.F.12	FNB home page
	8.F.13	FNB Internet bank home page: First Online
	8.F.14	Demonstration page: sign-on
	8.F.15	Demonstration page: transaction options
	8.F.16	Demonstration page: statement display
	8.F.17	Demonstration page: account general select
	8.F.18	Demonstration page: multiple account payment
8.F.19	Demonstration page: account transfer	
Note: Demonstration pages during 1998 were the same as in 1997 – changes to the demonstration facility were only made during 1999.		

Figure 8.F.1

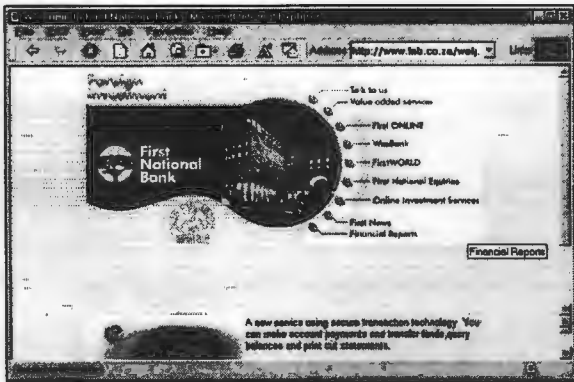


Figure 8.F.2

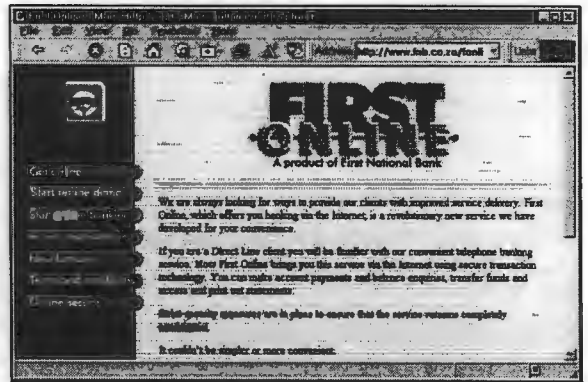


Figure 8.F.3

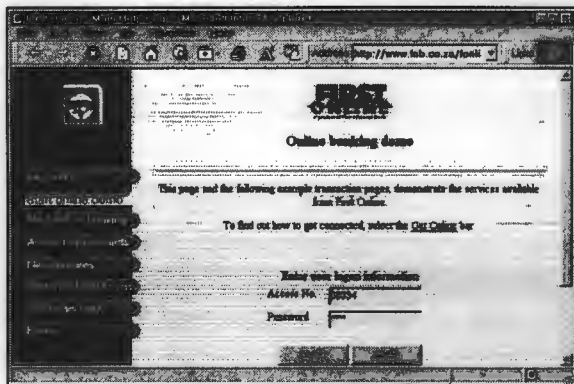


Figure 8.F.4

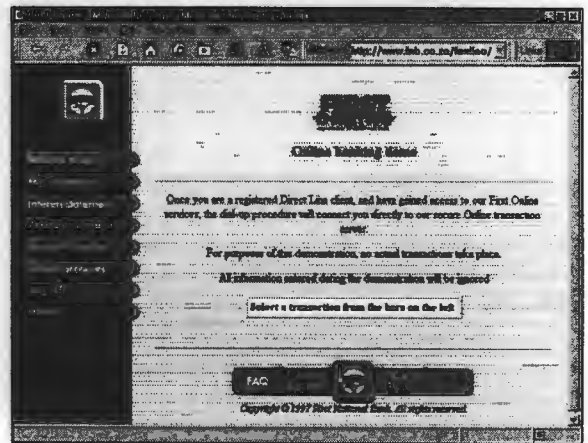


Figure 8.F.5



Figure 8.F.6

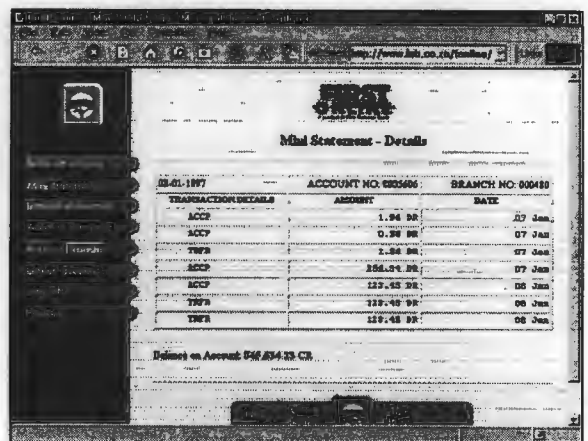


Figure 8.F.7



Figure 8.F.8

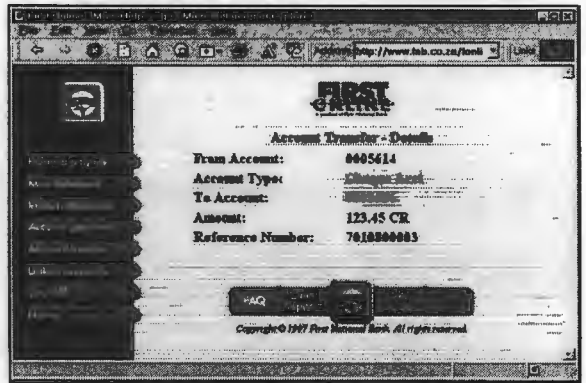


Figure 8.F.9

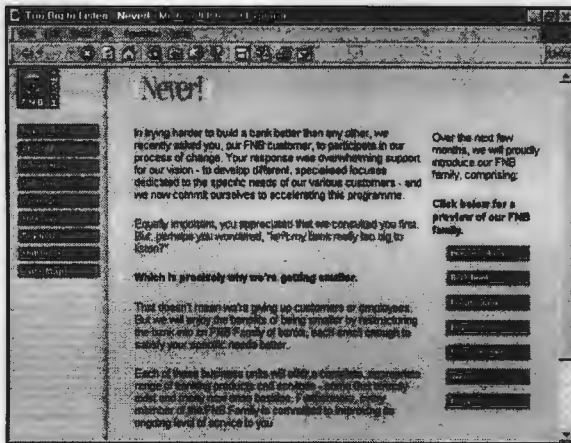


Figure 8.F.10

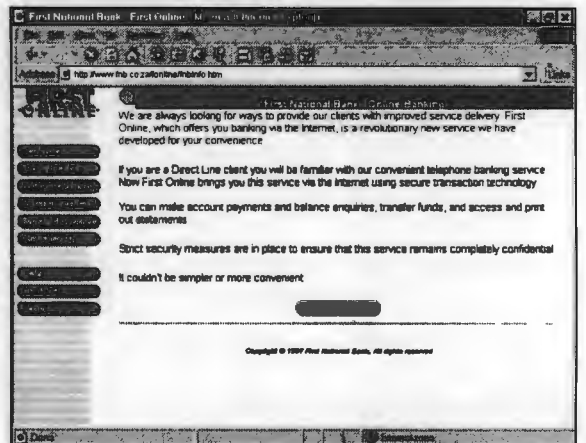


Figure 8.F.11

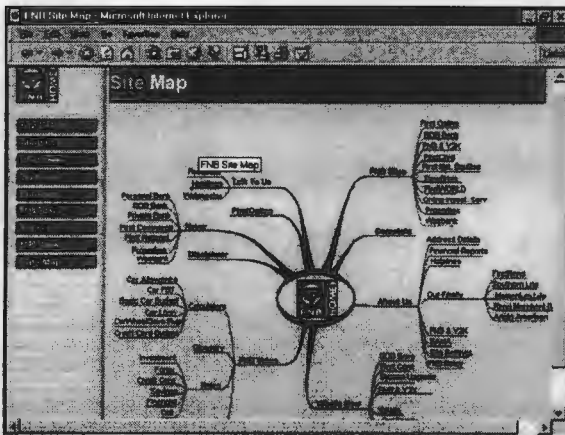


Figure 8.F.12



Figure 8.F.13

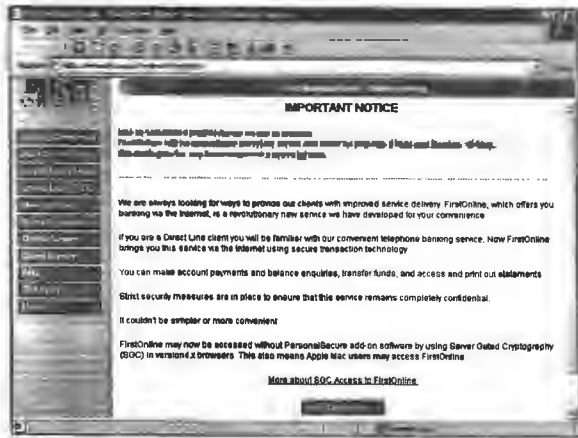


Figure 8.F.14

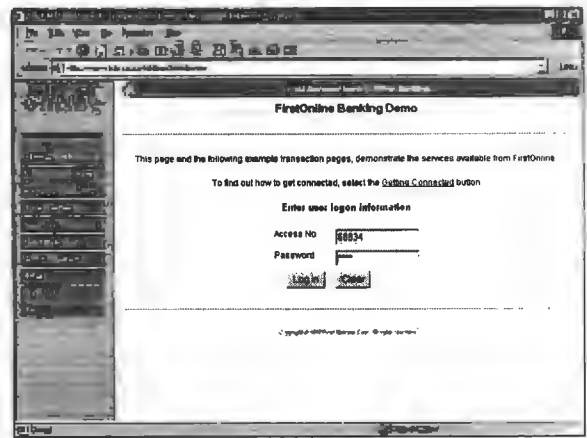


Figure 8.F.15

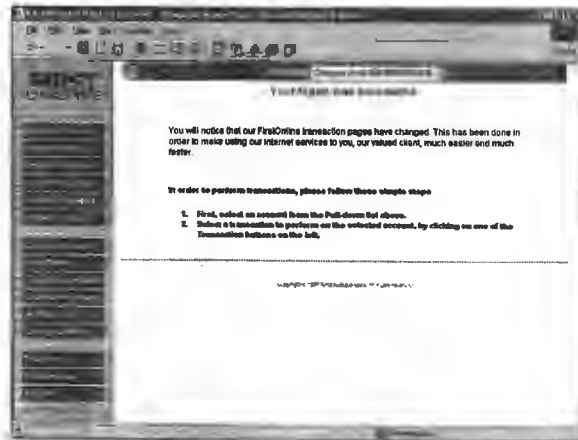


Figure 8.F.16

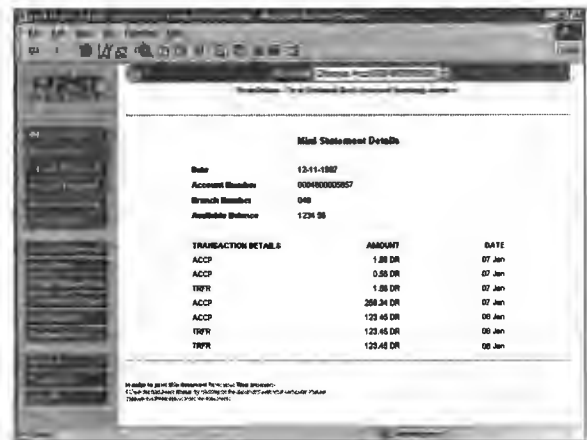


Figure 8.F.17

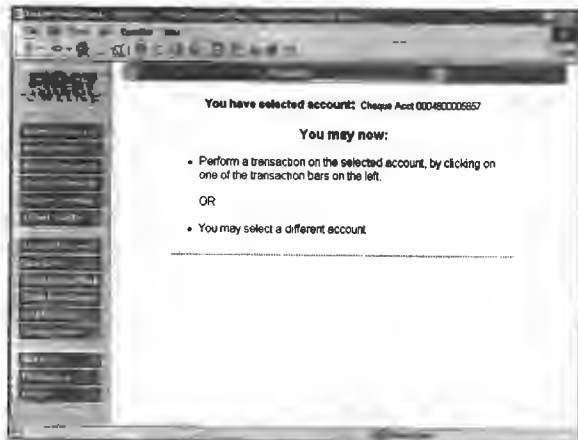


Figure 8.F.18

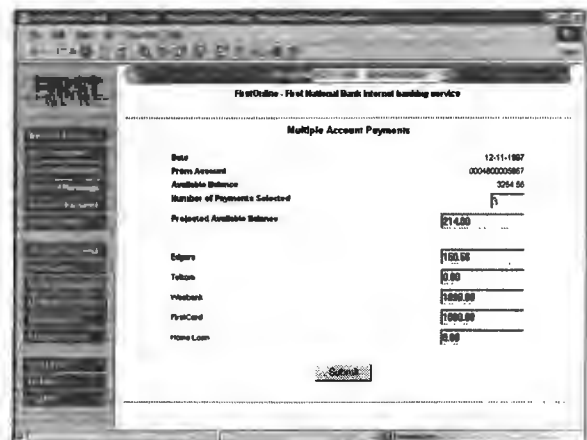
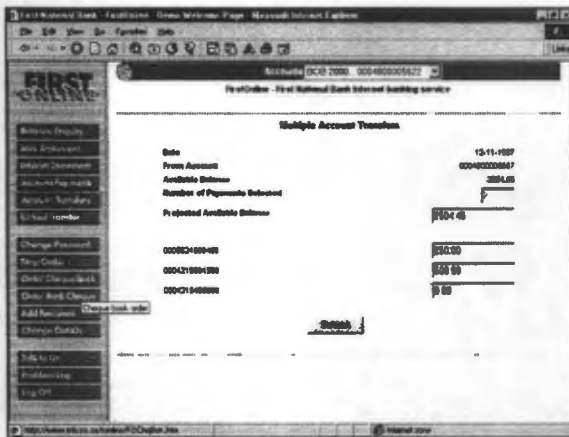


Figure 8.F.19



## APPENDIX 8.G

### Bank transactions: Nedbank (South Africa)

Environment and Date	Figure	Title
<b>September 1997</b> <b>Demonstration</b> MS Internet Explorer 3.02 MS Windows 95	8.G.1	Nedbank home page
	8.G.2	Nedbank Internet bank home page: Netbank
	8.G.3	Demonstration page: sign-on
	8.G.4	Demonstration page: balance and transaction options
	8.G.5	Demonstration page: account transfer
	8.G.6	Demonstration page: statement summary
	8.G.7	Demonstration page: log-off
<b>May 1998</b> <b>Demonstration</b> MS Internet Explorer 4.01 or Netscape 4.0, MS Windows 98	8.G.8	Nedbank Internet bank home page: Netbank
	8.G.9	Netbank low-graphics version
	8.G.10	Demonstration page: help pages
<b>April 1999</b> <b>Demonstration</b> MS Internet Explorer 4.01 MS Windows 98	8.G.11	Nedbank home page
	8.G.12	Nedbank Internet bank home page: Netbank
Note: Demonstration pages during 1998 and 1999 were effectively the same as in 1997.		

Figure 8.G.1

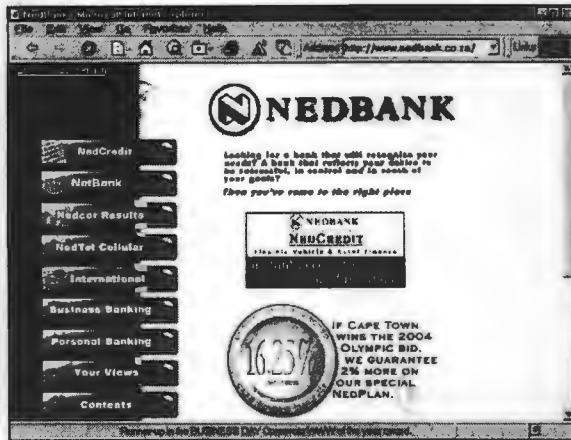


Figure 8.G.2

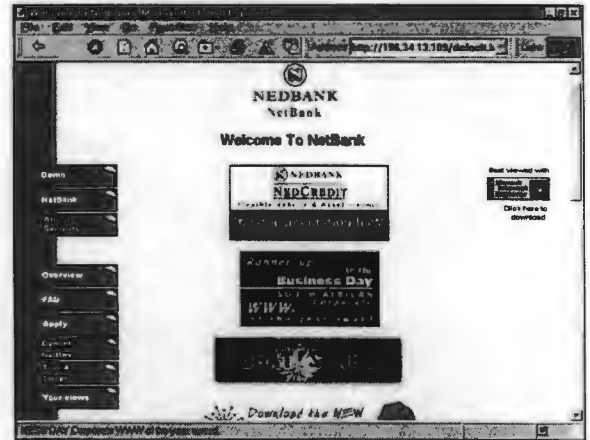


Figure 8.G.3



Figure 8.G.4

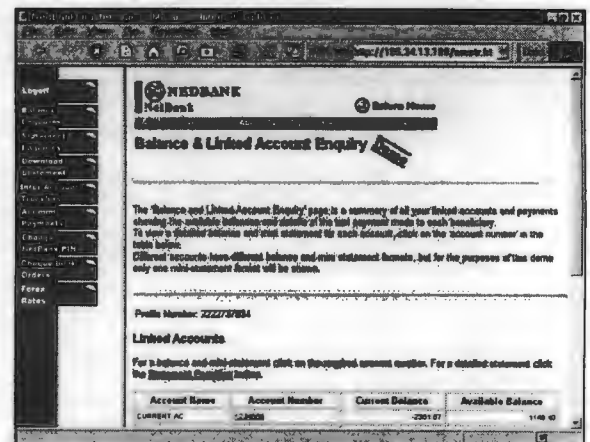


Figure 8.G.5

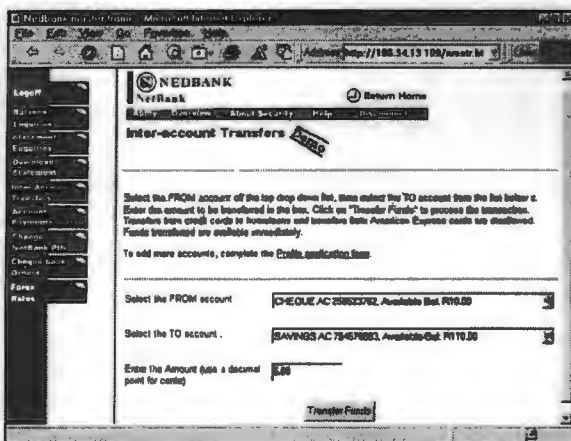


Figure 8.G.6

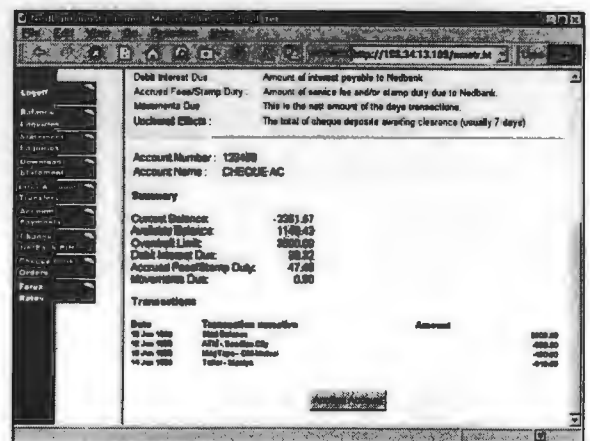


Figure 8.G.7

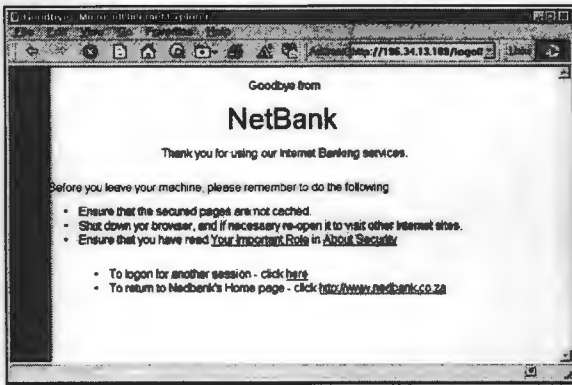


Figure 8.G.8

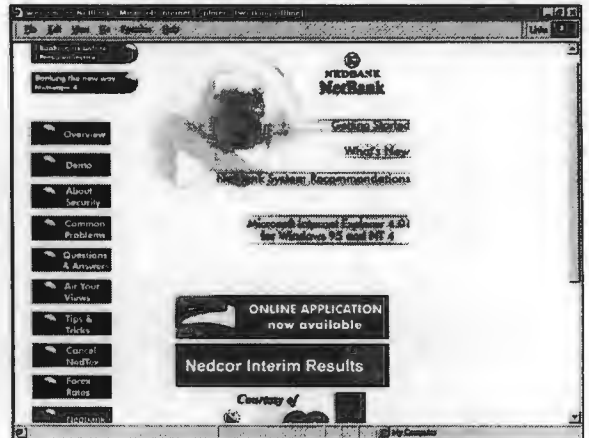


Figure 8.G.9

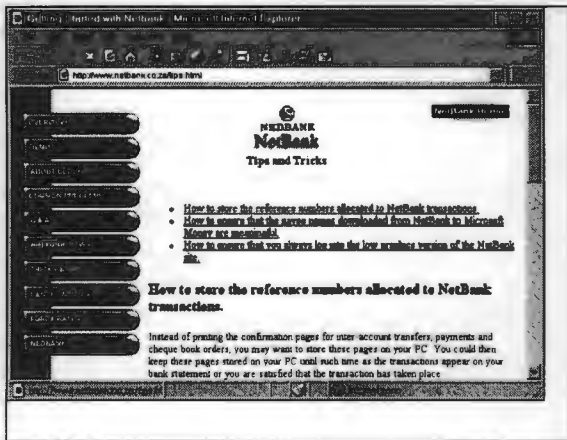


Figure 8.G.10

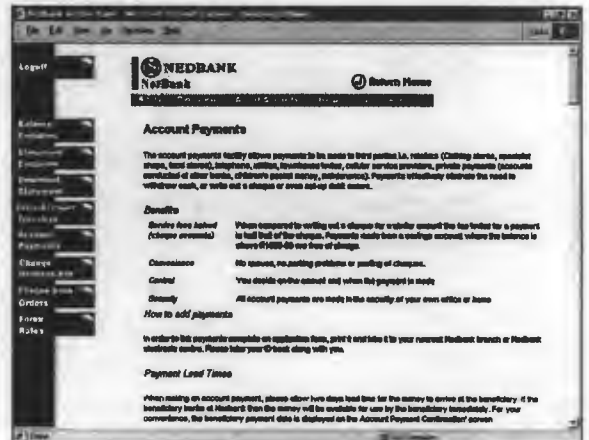
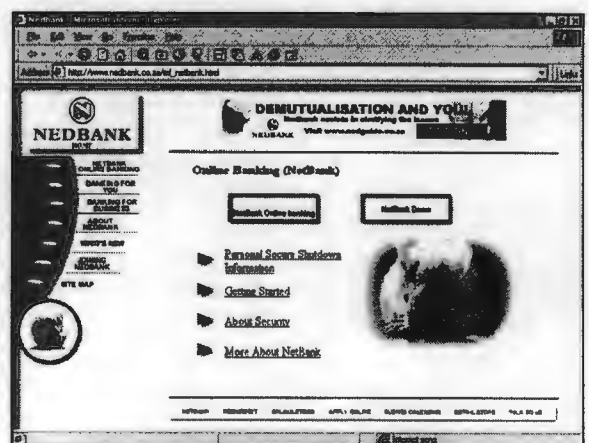


Figure 8.G.11



Figure 8.G.12





## APPENDIX 8.H

### Bank transactions: Amalgamated Banks of South Africa (ABSA)

Environment and Date	Figure	Title
<b>September 1997</b> No demonstration available MS Internet Explorer 3.02 MS Windows 95	8.H.1	Absa home page
	8.H.2	Absa Internet bank home page 1: Homebank
	8.H.3	Absa Internet bank home page 2: Homebank
	8.H.4	Absa Internet bank home page 3: Homebank
	8.H.5	Demonstration page: sign-on
<b>May 1998</b> Demonstration MS Internet Explorer 4.01 MS Windows 98	8.H.6	Absa Internet bank home page: Homebank
	8.H.7	Demonstration page: sign-on
	8.H.8	Demonstration page: transaction options
	8.H.9	Demonstration page: statement select
	8.H.10	Demonstration page: statement display
	8.H.11	Demonstration page: account transfer select
	8.H.12	Demonstration page: account transfer proceed
	8.H.13	Demonstration page: account transfer confirmed
8.H.14	Demonstration page: beneficiaries payment	
<b>April 1999</b> Demonstration MS Internet Explorer 4.01 MS Windows 98	8.H.15	Absa home page
	8.H.16	Absa Internet bank home page
Note: Demonstration pages during 1998 and 1999 were essentially the same. There was no demonstration facility during 1997.		

Figure 8.H.1

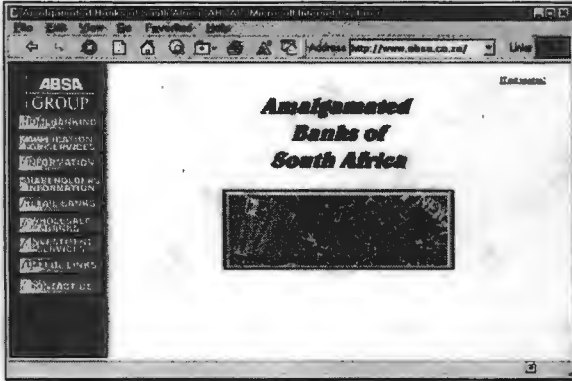


Figure 8.H.2

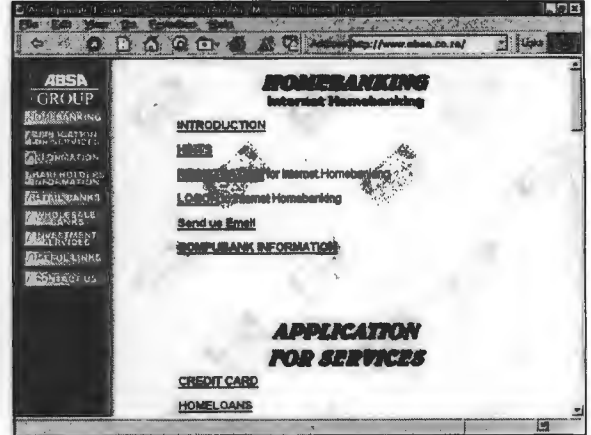


Figure 8.H.3

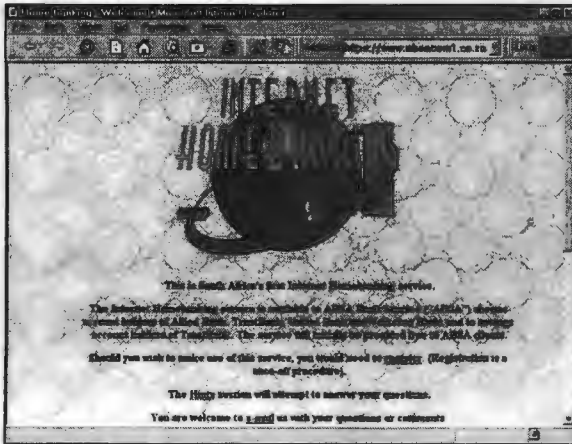


Figure 8.H.4

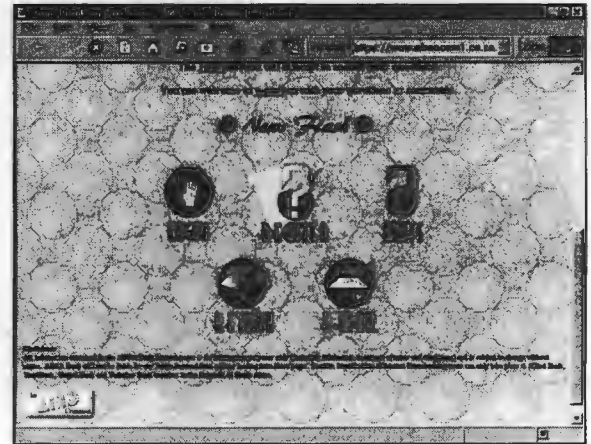


Figure 8.H.5

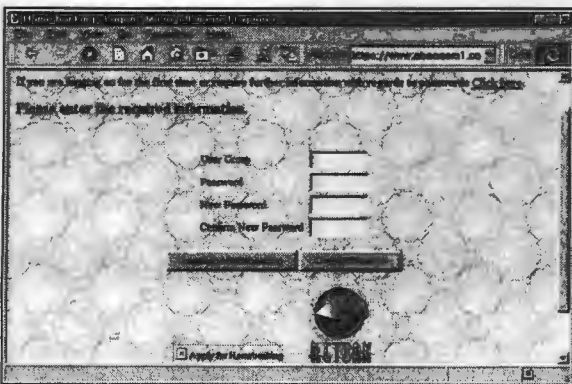


Figure 8.H.6

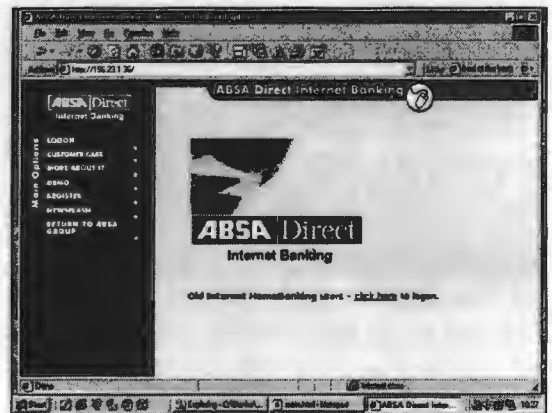


Figure 8.H.7

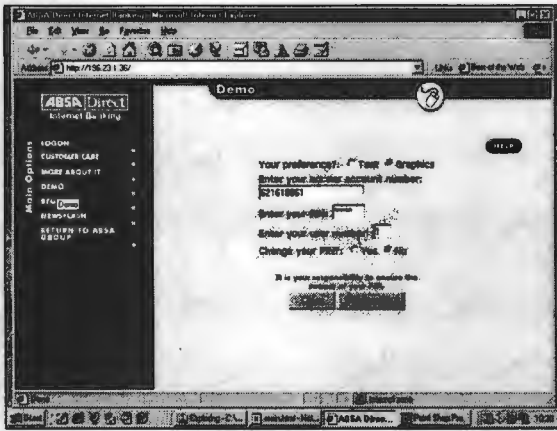


Figure 8.H.8

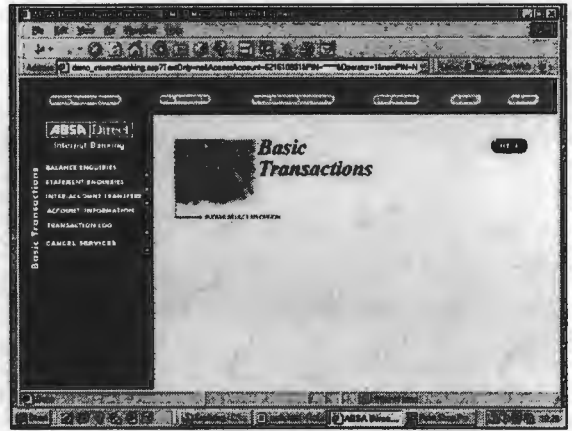


Figure 8.H.9

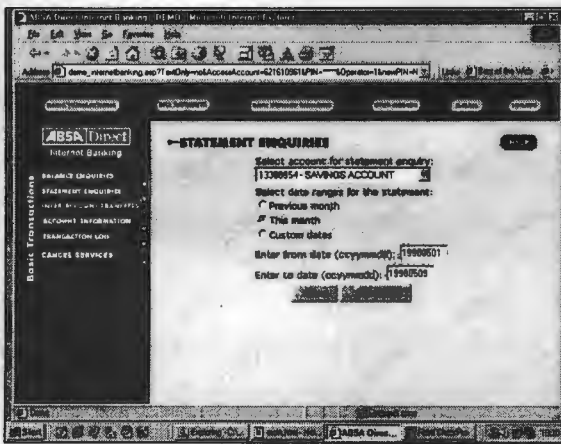


Figure 8.H.10

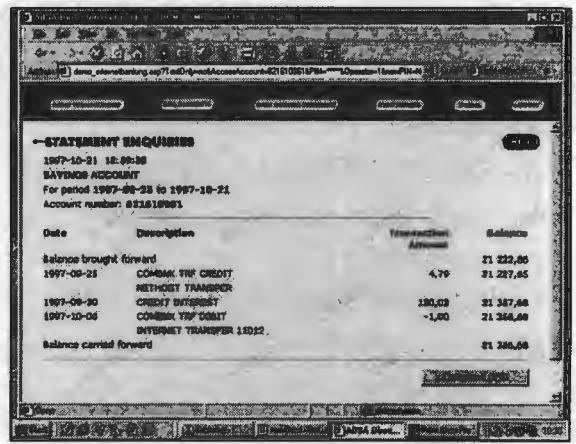


Figure 8.H.11

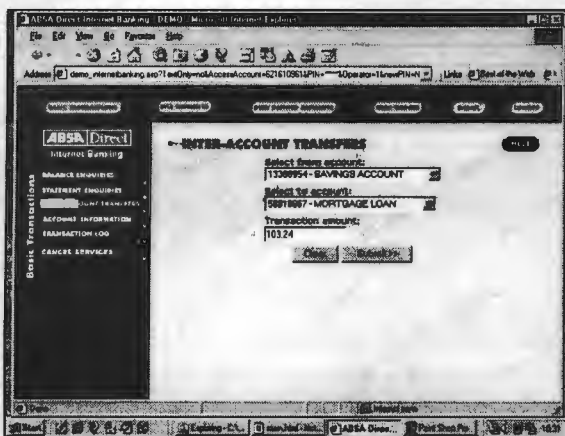


Figure 8.H.12

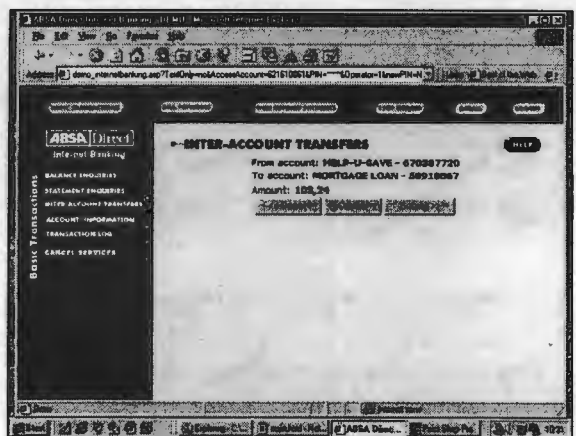


Figure 8.H.13

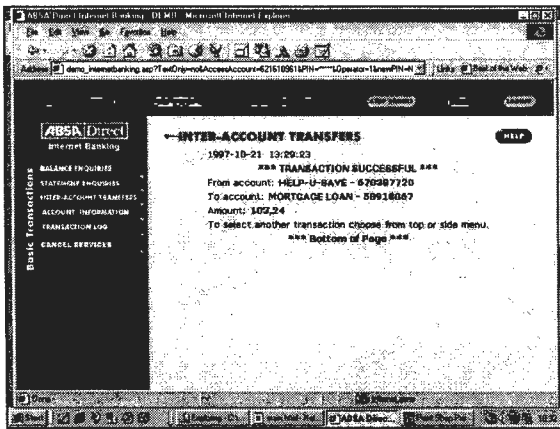


Figure 8.H.14

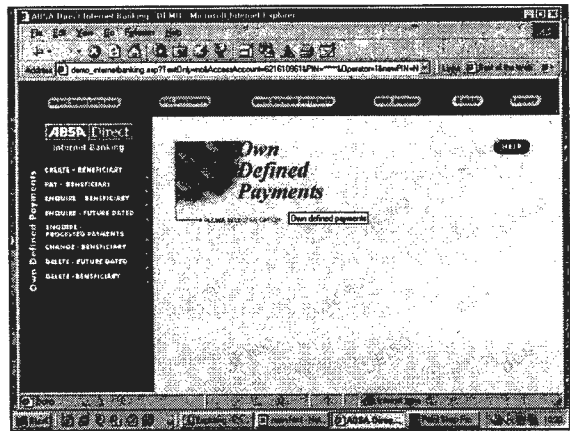


Figure 8.H.15

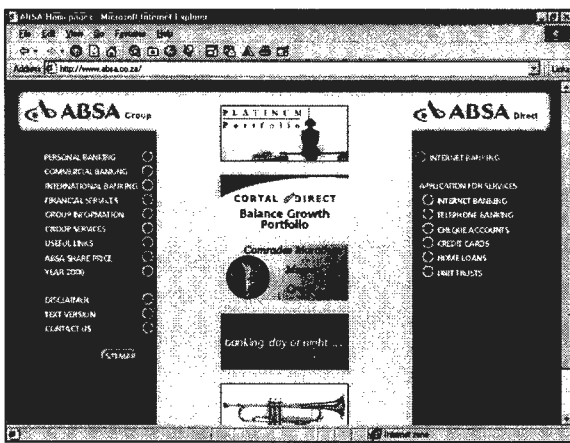
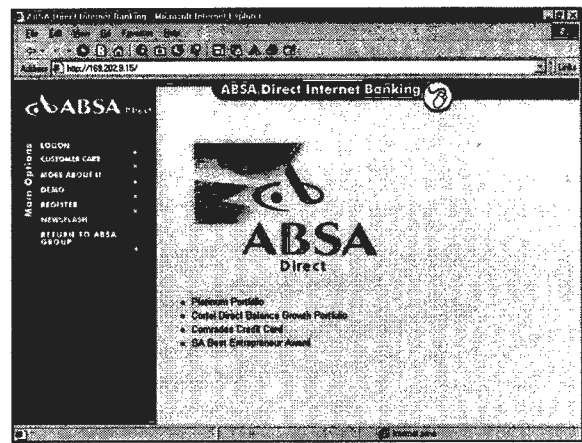


Figure 8.H.16



## APPENDIX 8.I

## Bank transactions: Other banks May 1998

Bank	Figure	Title	
<b>Mercantile Bank (SA)</b> (no demonstration)	8.I.1	Mercantile Bank home page	
	8.I.2	Mercantile Internet banking home page	
<b>Citibank (USA)</b> (full interactive demonstration)	8.I.3	Citibank home page	
	8.I.4	Citibank Internet banking select region	
	8.I.5	Citibank Internet banking home page: Direct Access	
	8.I.6	Sign-on page	
	8.I.7	Transactions options page	
	8.I.8	Payments and transfers	
	8.I.9	Bill payment	
	8.I.10	Payment select destination	
	8.I.11	Payment information	
	8.I.12	Payment confirmation	
	8.I.13	Account information option	
	8.I.14	Account selection	
	8.I.15	Account information display	
	8.I.16	Customer service options	
	8.I.17	Information center options	
	<b>Bank of America (BOFA)</b> (USA) (full interactive demonstration)	8.I.18	BOFA home page
		8.I.19	BOFA Internet banking home page
8.I.20		BOFA sign-on page	
8.I.21		Transactions options page	
8.I.22		Transfer funds options	
8.I.23		Transfer funds selection	
8.I.24		Transfer funds proceed	
8.I.25		Transfer funds confirmation	
8.I.26		Account balance details	
8.I.27		Help pages	
8.I.28		BOFA Build-your-own-bank home page	
<b>Bank One (USA)</b> (full interactive demonstration)	8.I.29	BankOne home page	
	8.I.30	BankOne Internet banking browser check	
	8.I.31	BankOne Internet banking browser check help pages	
	8.I.32	BankOne Internet banking home page	
	8.I.33	Sign-on	
	8.I.34	Transaction options	
	8.I.35	Transfer funds selection	
	8.I.36	Transfer funds confirmation	
	8.I.37	Accounts details	
	8.I.38	Account payments	
	8.I.39	Customer services	
<b>Security First Network Bank (SFNB) (USA)</b> (full feature passive demonstration)	8.I.40	Security First Network Bank (SFNB) home page	
	8.I.41	SFNB demonstration account summary	
	8.I.42	Cleared check display	
	8.I.43	Transfer funds	
	8.I.44	Statement options page	
<b>Royal Bank of Scotland (RBS) (UK)</b> (screenshot demonstration)	8.I.45	RBS home page	
	8.I.46	RBS Internet banking demonstration page	
	8.I.47	Screenshots demonstration	
<b>Wells Fargo Bank (USA)</b> (full passive demonstration)	8.I.48	Wells Fargo home page	
	8.I.49	Demonstration sign-on	
	8.I.50	Account information	

Figure 8.I.1

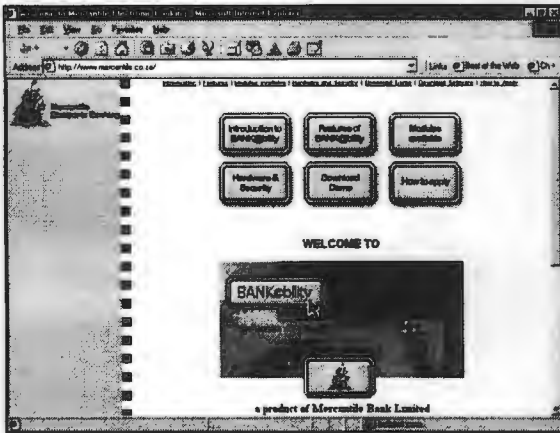


Figure 8.I.2

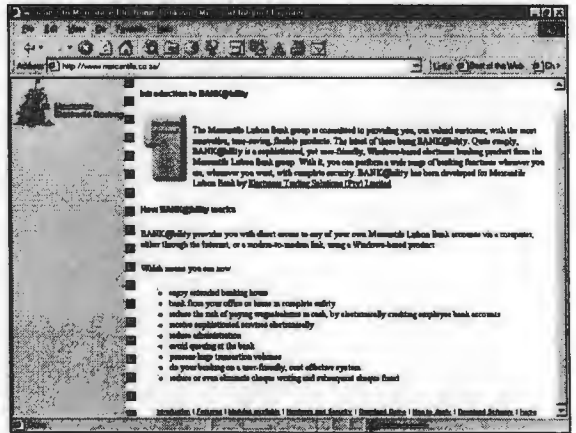


Figure 8.I.3

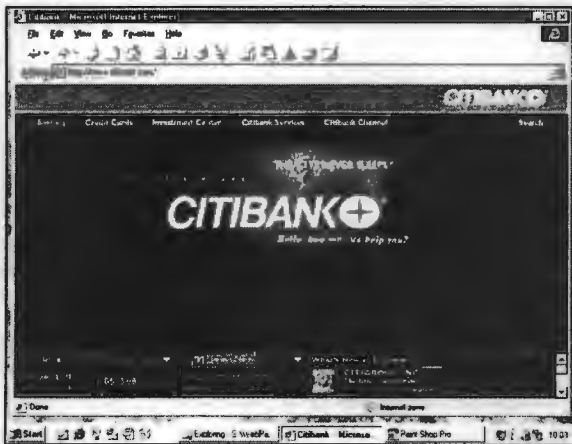


Figure 8.I.4



Figure 8.I.5

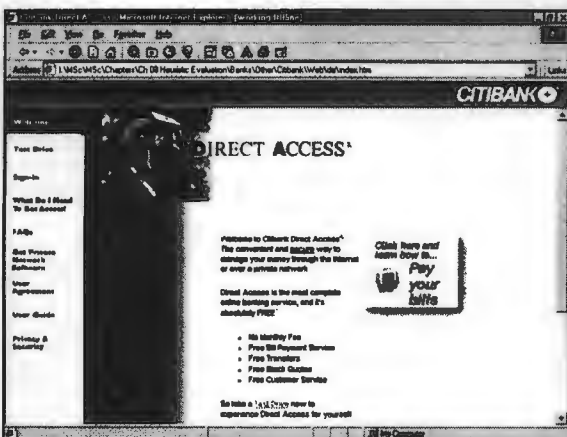


Figure 8.I.6

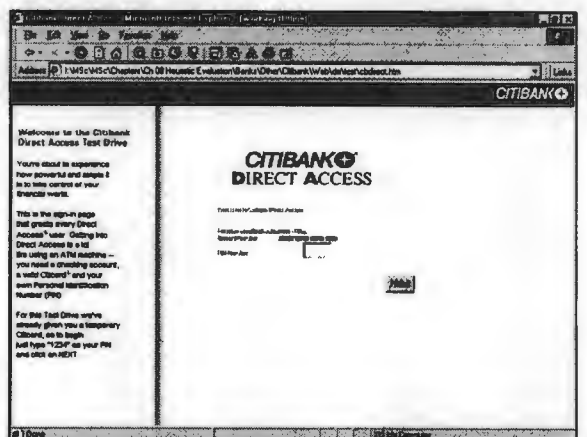


Figure 8.I.7

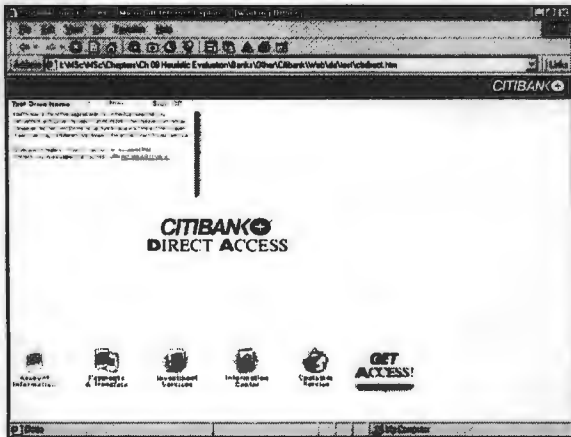


Figure 8.I.8

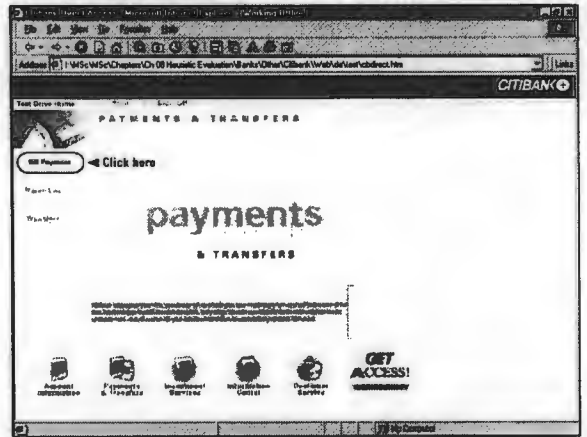


Figure 8.I.9



Figure 8.I.10

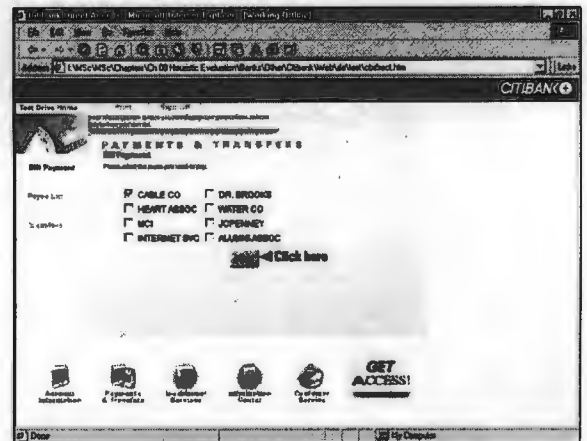


Figure 8.I.11

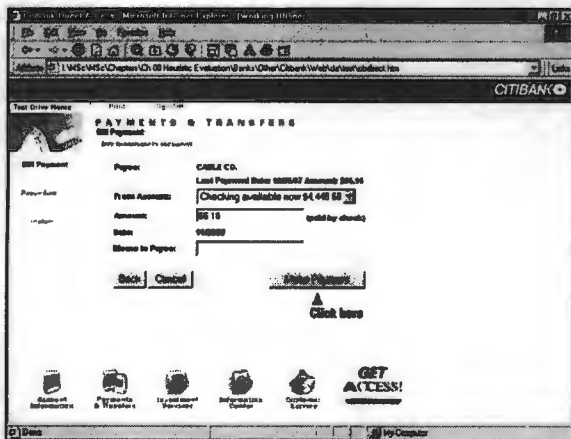


Figure 8.I.12

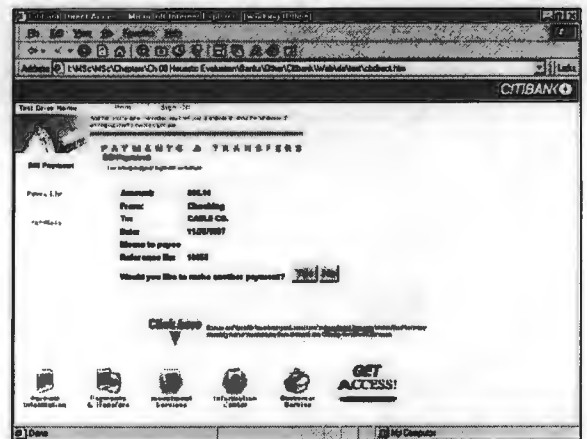


Figure 8.I.13

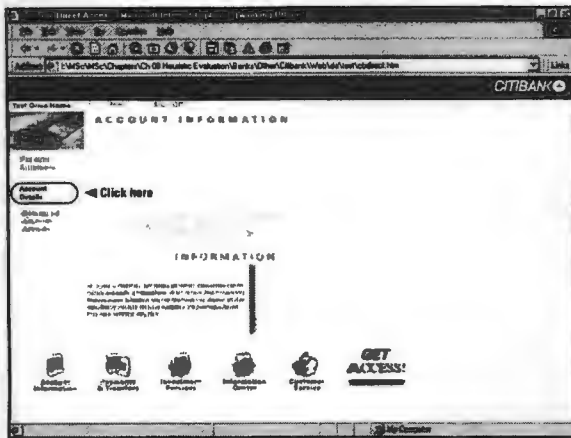


Figure 8.I.14

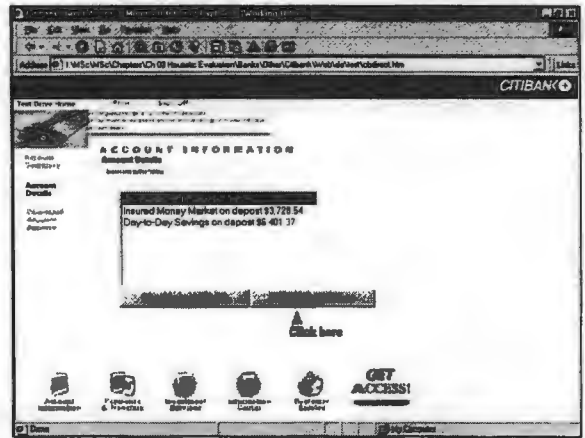


Figure 8.I.15

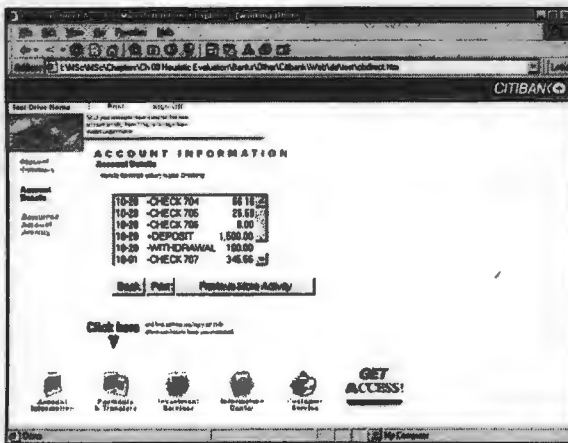


Figure 8.I.16

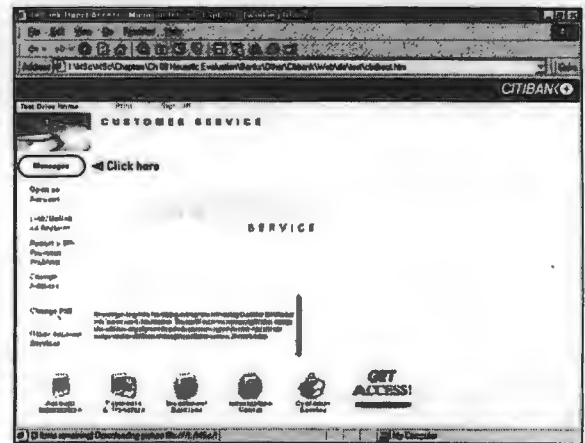


Figure 8.I.17



Figure 8.I.18

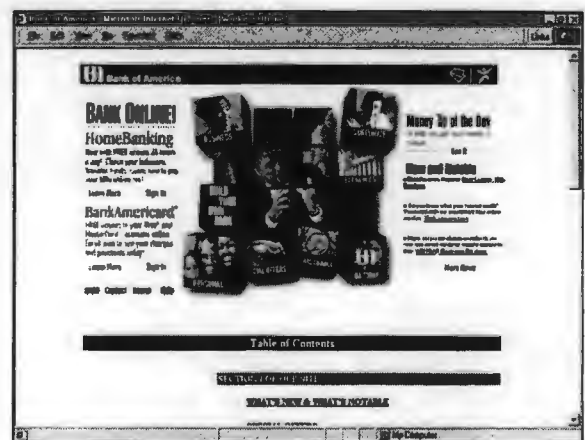




Figure 8.I.19

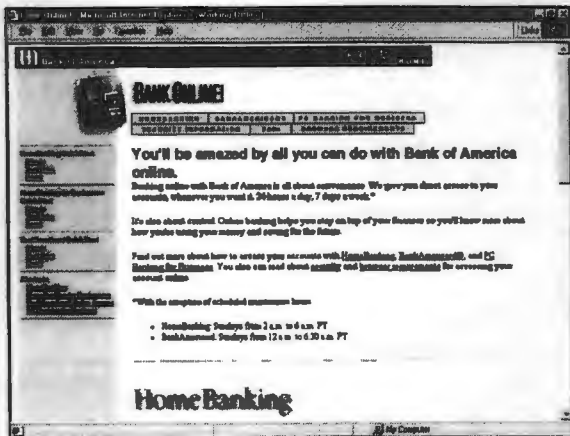


Figure 8.I.20

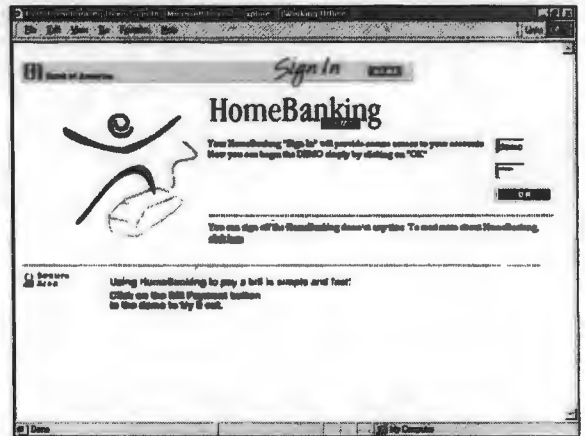


Figure 8.I.21

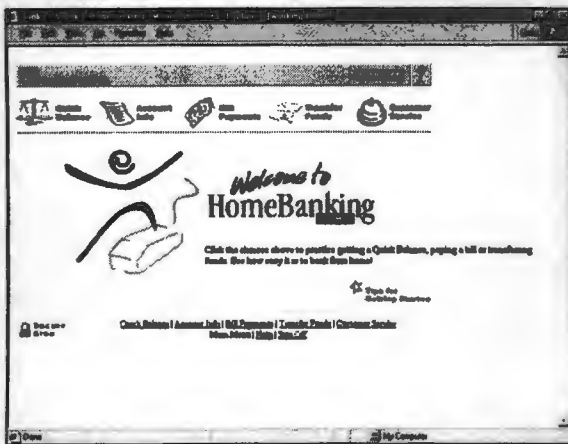


Figure 8.I.22

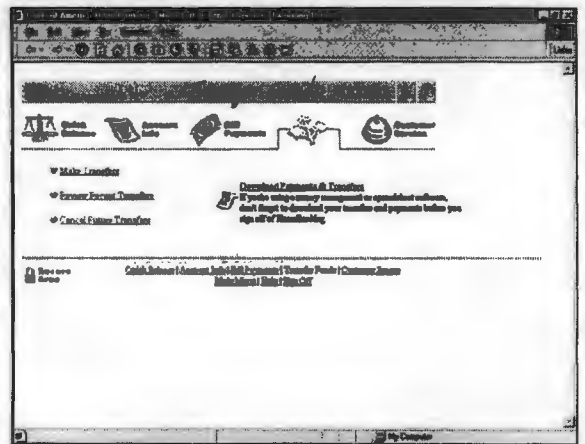


Figure 8.I.23

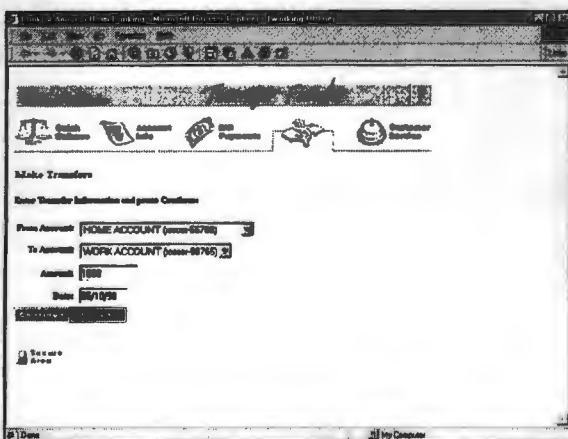


Figure 8.I.24

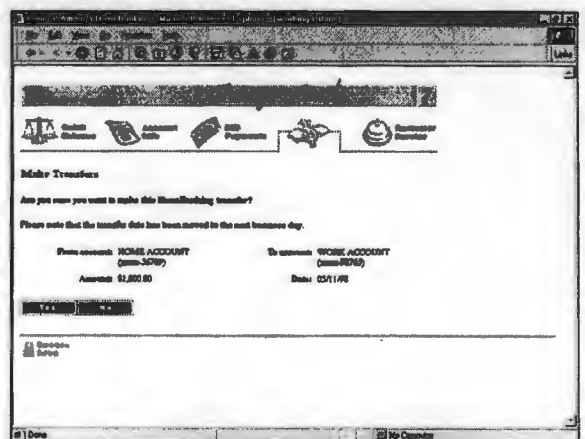


Figure 8.I.25

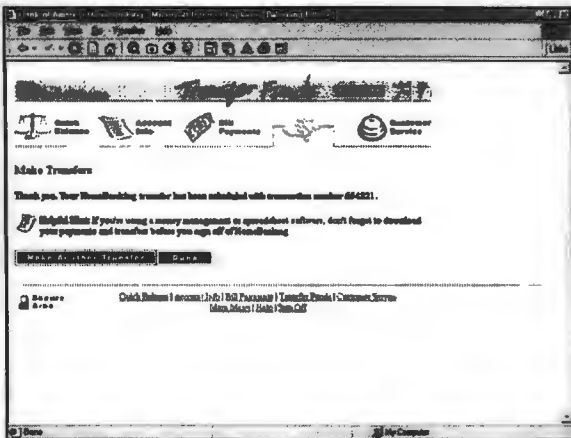


Figure 8.I.26

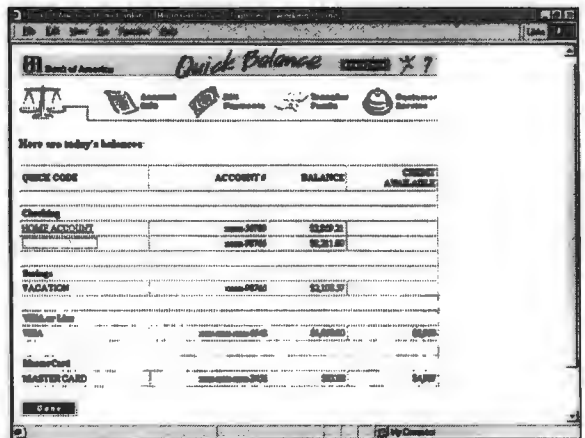


Figure 8.I.27

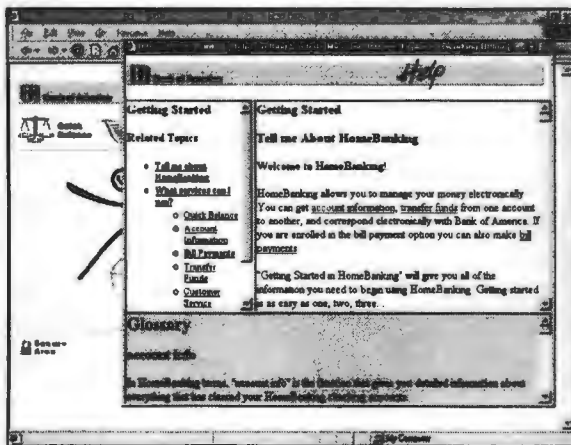


Figure 8.I.28

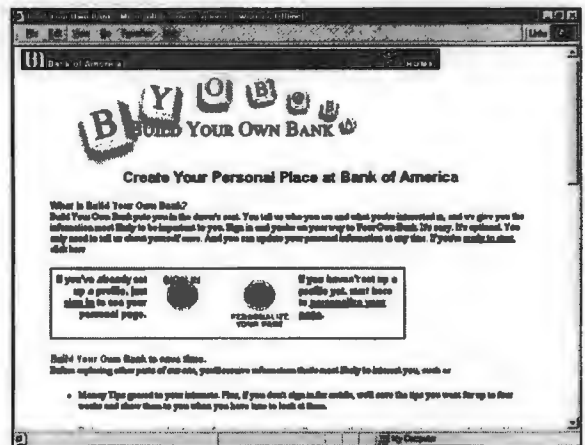


Figure 8.I.29

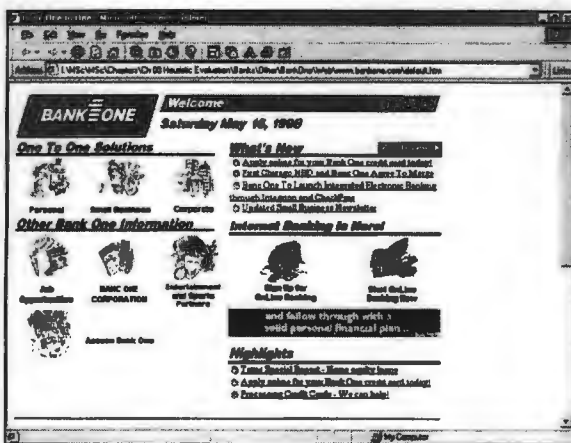


Figure 8.I.30

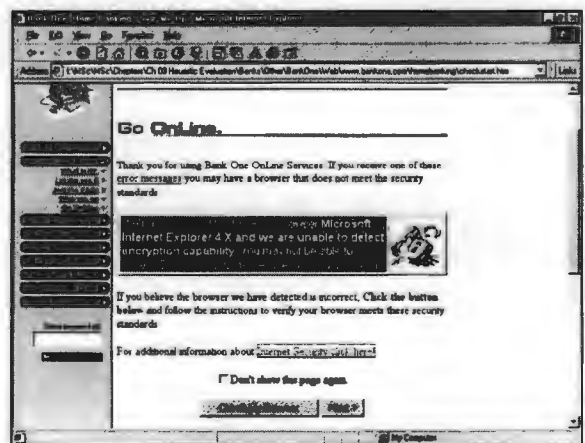


Figure 8.I.31

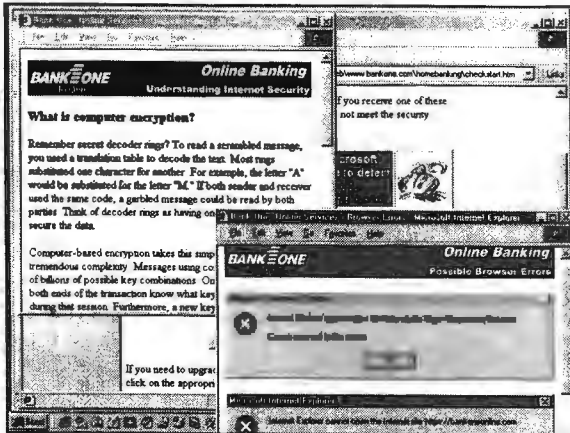


Figure 8.I.32

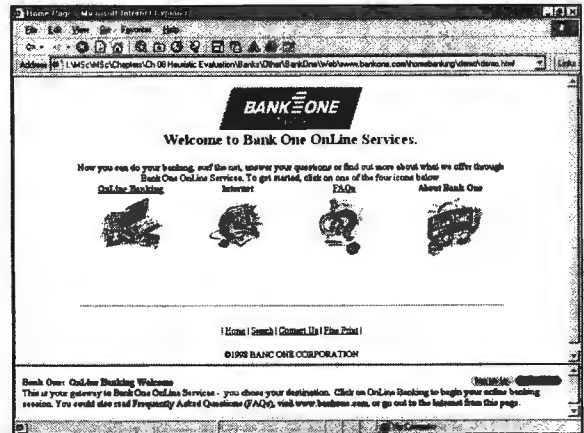


Figure 8.I.33

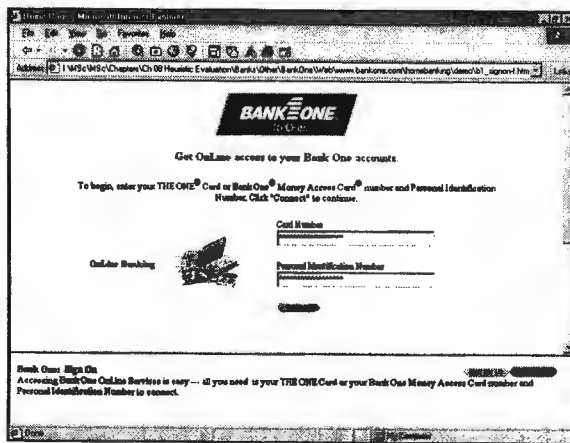


Figure 8.I.34

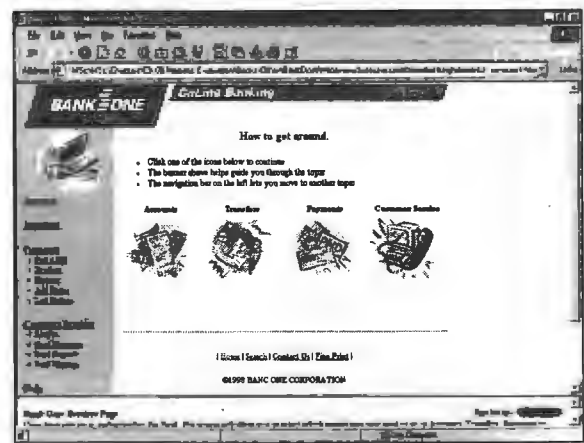


Figure 8.I.35

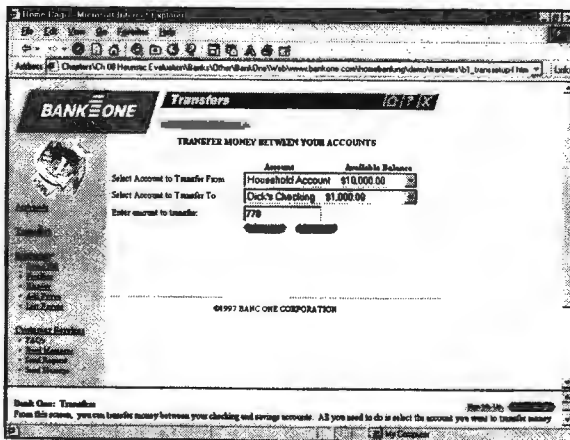


Figure 8.I.36



Figure 8.I.37

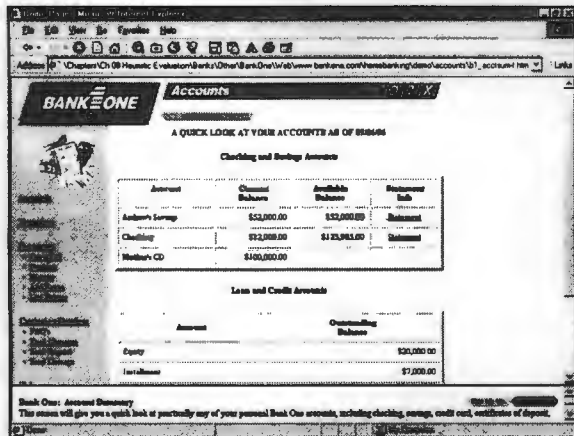


Figure 8.I.38

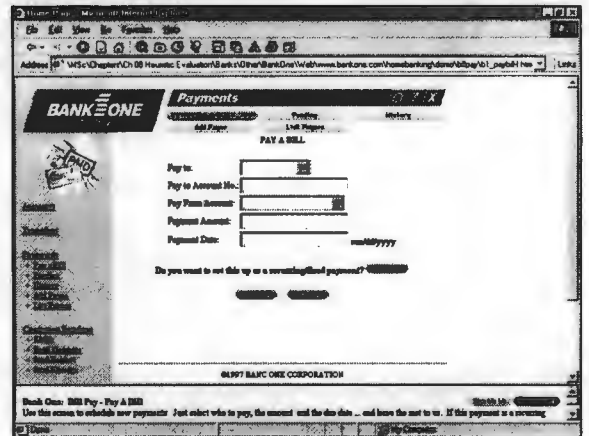


Figure 8.I.39

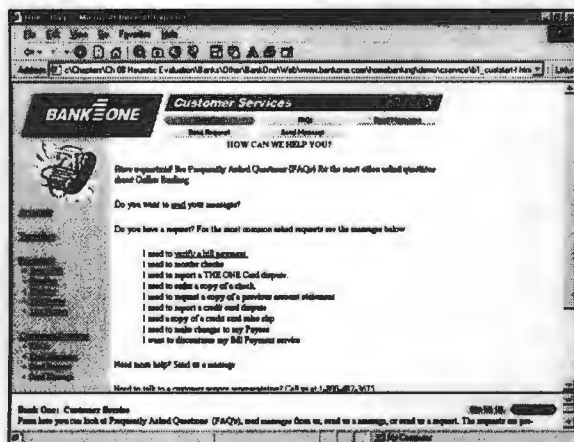


Figure 8.I.40



Figure 8.I.41

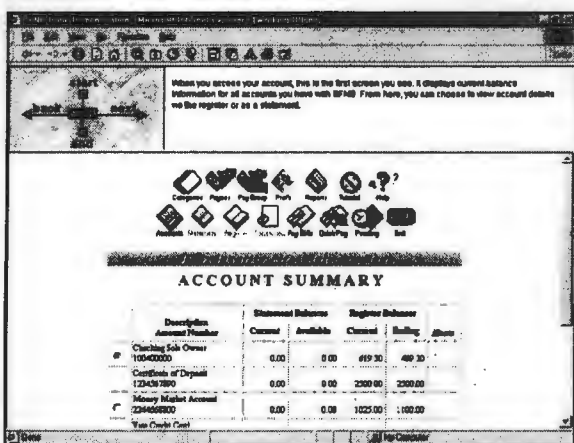


Figure 8.I.42

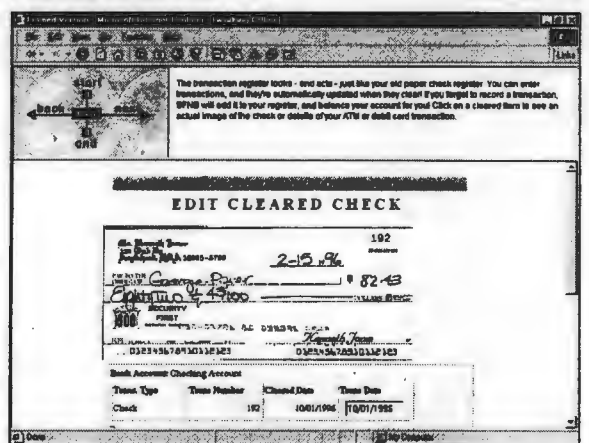


Figure 8.I.43

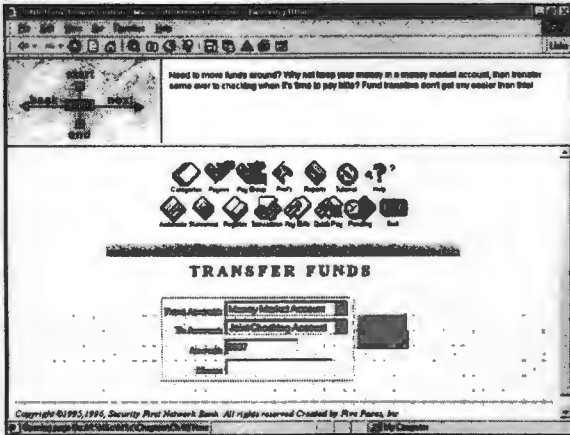


Figure 8.I.44

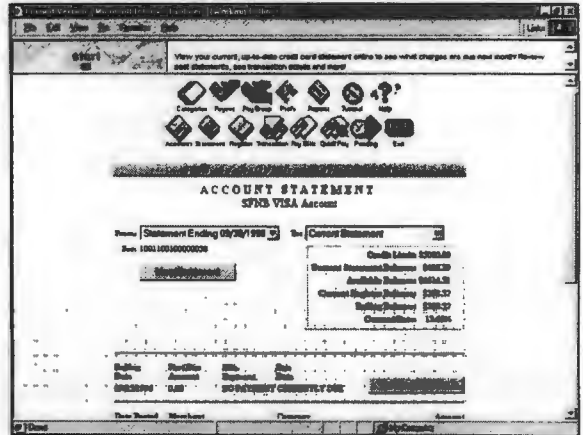


Figure 8.I.45

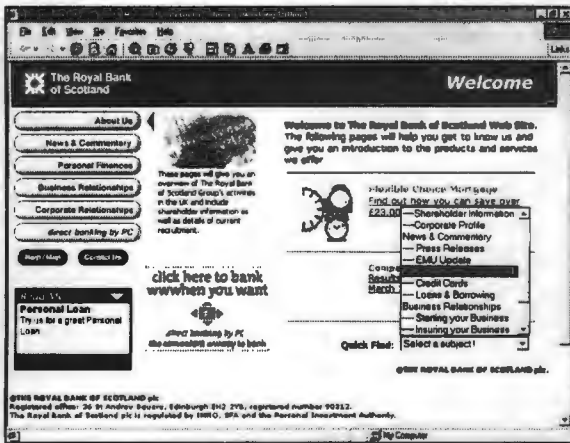


Figure 8.I.46



Figure 8.I.47

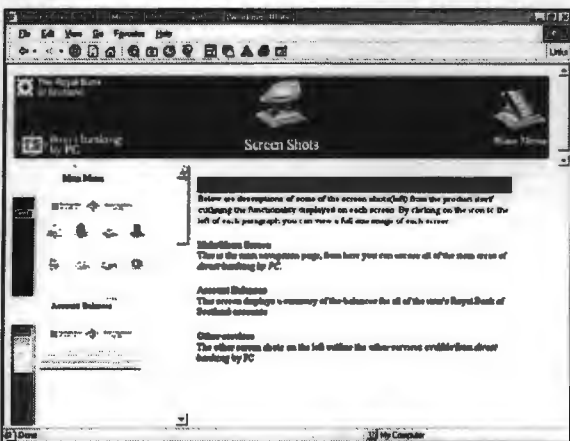


Figure 8.I.48

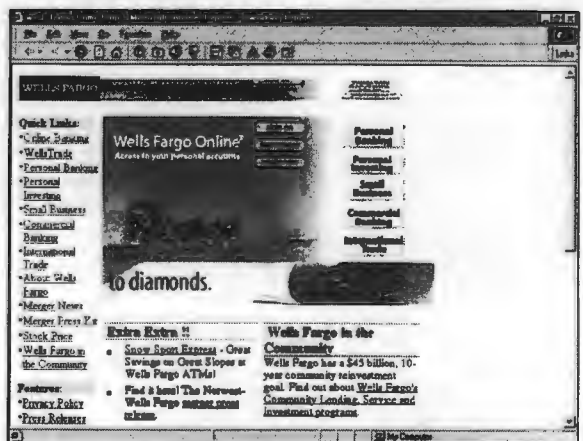


Figure 8.I.49

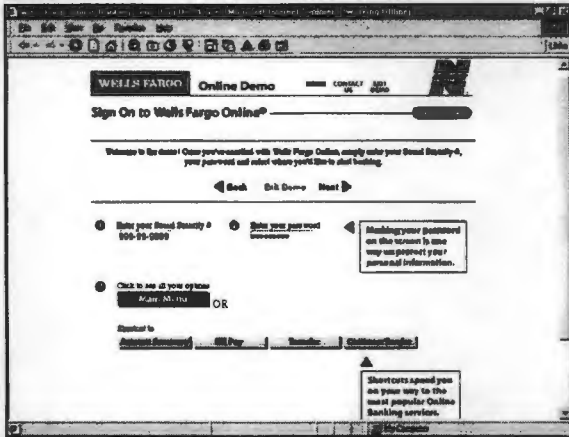
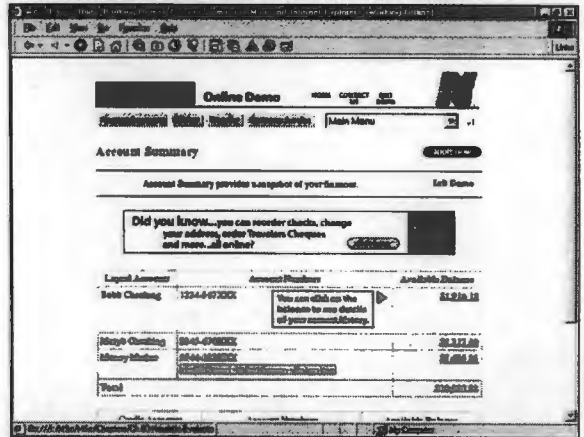


Figure 8.I.50



## Chapter 9

# Checklist-based Evaluation

This chapter describes the application of a checklist-based usability evaluation to a banking task executed within the Internet banking facility of a local bank.

## 9.1 Introduction

Two usability checklists were scored for a banking task – using Standard Bank’s facility – as identified and examined in chapter 6 (Task analysis). The first checklist is the Ravden and Johnson (1989) questionnaire, which in its original format consisted of eleven sections and 179 questions. Although it had been constructed ten years ago and is designed for a general usability evaluation, it is considered to be (still) eminently suitable for this type of goal-directed task-oriented Web application. In addition it is very detailed and empirically proven. The second checklist is a single section questionnaire containing 16 questions. This is more specific – it is designed for an Internet-based banking evaluation – and had been used by the JIBC – Journal of Internet Banking and Commerce (URL: JIBC1). The two scored checklists are presented as appendices 9.A and 9.B in section 9.5.

## 9.2 The Ravden and Johnson checklist

Ravden et al. (1989) developed a comprehensive checklist and questionnaire for the (qualitative) evaluation of important usability aspects. This checklist will be referred to as the Ravden checklist. They (1989:23) consider their checklist to be suitable for the comparison of existing systems and evaluating alternatives, as is the case here. The usability guidelines – which they refer to as usability criteria or goals (1989:19) – covered by their method have been discussed previously and were presented in chapter 7 (table 7.10).

### 9.2.1 Data collection and analysis methods

Ravden et al. (1989:78), recommends that an evaluation using their checklist should be preceded by a task analysis and definition, which should be aimed at testing as much of the system behaviour and functionality as possible. This requirement is met by applying the evaluation on the same ‘obtain-statement’ task as discussed in chapter 6 (Task analysis).

In true Web spirit the 1989 Ravden checklist was simplified for this evaluation. Its original section 11 presents a series of general questions on system usability, all of which had been covered by the discussions in the previous chapter, and this section is therefore not included. For the same reason the general comment question included for each of sections 1-9 was omitted. Their 1989 page format was also slightly modified. Nielsen (1993:212) notes that questionnaires tend to avoid the use of these 'open' questions because of the danger that these natural language entries may be difficult to interpret correctly.

The four point categorical rating scale used by Ravden et al. (1989) for sections 1-9 of the checklist, do not have a midpoint rating ('average'), and is probably not an equal interval scale – this implies that the use of a simple nonresistant statistic should be sufficiently accurate. This takes the form of a single evaluation indicator: A number 0,1,2 and 3 had been assigned to each of the categories never, sometimes, mostly and always, and a mean calculated for the 9 sections. This scoring approach seems to be intuitively correct – a section with all scores equal to 'never' will score 0%, and with all scores as 'always' the score is 100%. For all questions scored as 'mostly' the score is 67% and with all scores equal to 'sometimes' it is 33%. The last question in sections 1-9 – which constitutes an overall rating for the category – could be used to check the score obtained for each section. Section 10 uses a three-point scale, and is evaluated as the score obtained for each question.

This method was preferred to the less specific 'ideal' to 'unfavourable' analysis recommended by Ravden et al. (1989:95). The cross-referencing guide provided by them (1989:100) for examining relationships between the different checklist sections is useful, but a complete implementation would be very research intensive. Note that the researcher acted as the sole evaluator (the pitfalls of which are discussed by Ravden et al. (1989:22,86)) - but was careful to avoid rating bias – this could possibly be introduced through aspects such as central tendency, leniency errors, and cross-linking (Cooper et al., 1995).

## 9.2.2 Discussion of the results

The scores obtained for sections 1 to 9 are listed in table 9.1. This also presents a single percentage score calculated from the numerical transformation of the sectional scores, which is 46% for the Web site's usability. Calculating an overall percentage value on an equal-weighted category basis yields a similar figure.

It is unfortunate that this score can not be compared to values obtained for other banks – a comparative score would have been useful. It would be unrepresentative to use a demonstration facility for this evaluation as many of the questions test live behaviours such as errors and responses. Nielsen (1993:37) comments on this aspect and observes that for scale calibration, anchoring ('baseling') should be done through comparison with other systems. It can however be compared to three other GUI usability scores calculated in the exact same manner. Two Windows-based file



manager components scored 76% and 63% (overall) for a file-move task (Van Dyk, 1997:76). This could indicate that the banking task lacks in certain usability aspects when compared to these earlier (but non-distributed) applications – the most notable differences were observed in the scores obtained for explicitness, flexibility, and control. The latter two categories scored at 40% of the file-manager values whilst the others categories varied between 60% and 90% of the earlier scores. The first of these is concerned with understanding the way that the system works and its structure, and the second with aspects such as customizability. Both aspects have been identified as specific problem areas during previous discussions.

**Table 9.1 Ravden and Johnson checklist scores for Standard Bank**

Section		Scores		
		Raw (maximum)	Raw (actual)	%
1	Visual Clarity	48	23	48
2	Consistency	42	27	64
3	Compatibility	51	28	55
4	Explicitness	39	11	28
5	Appropriate Functionality	33	16	49
6	Informative Feedback	54	23	43
7	Flexibility and Control	45	16	36
8	Error Prevention and Correction	42	25	60
9	User Guidance and Support	33	10	33
Total score:		387	179	46.3
Average score:				46.2

**Notes Table 9.1**

- (1) Maximum Raw: The maximum score if all questions are checked as a 3 (always).
- (2) %: The system score as a percentage of the maximum score.
- (3) Total Score =  $100 * (\text{sum of section raw scores}) / (\text{sum of maximum raw scores})$ .
- (4) Average Score =  $100 * (\text{sum of the section percentage scores}) / 9$ . This average score weighs each criterion equally (unlike the total score).

The three lowest scores in table 9.1 are for explicitness, user guidance, and flexibility – this result confirms the opinion expressed above. User guidance suffered in this evaluation because of the absence of help during transaction stages, poor page layout, and the absence of hard-copy documentation. The highest scores are shown for error prevention and consistency. The first is (in part) due to the security-conscious design of the application, and the second because the application runs and behaves similar to a standard Windows application.

The results from section 10 of the checklist show that the bank task had 28% ‘major’ problems and 8% ‘no-problem’ questions. This is a believable result – the bulk of the usability problems could be classed as ‘minor’. The ‘major-problem’ scores were shown for aspects such as system documentation, inflexible user help, response times, and error recovery. The latter shows the danger

associated with a single question covering a wide range of situations – although error correction is satisfactory for transaction errors there are serious problems with system error recovery. Both the frequency and impact of problems were taken into account when section 11 was scored – i.e. a problem may have a low impact (such as sluggish response), but as it is frequent (essentially it is present *all the time*), it warrants being viewed as a serious problem.

### 9.3 The JIBC checklist

The Journal of Internet Banking (JIBC, URL: JIBC1) started to publish evaluations of the service quality for financial institutions (FI) that conduct transactional delivery of banking services through Internet during late 1998. Using a checklist-based questionnaire it evaluates only those FI's that actively use the Internet to provide banking services. JIBC readers are actively encouraged to submit their own evaluations, add to the existing evaluation criteria, and suggest corrections to the evaluations that have already been posted.

The results of the evaluation using their 16-question evaluation (presented in appendix 9.B) show Standard Bank to have obtained an average of 59%. The highest scores were obtained for the questions relating to avoiding large graphics and advertising downloads, the absence of misbehaving and/or superfluous Java applets, the convenience offered by the service, and the effort and commitment shown by the FI in developing and maintaining its Web site. These results are in alignment with those sections of the Ravden checklist where a comparison can be made. Qualities such as effort and commitment can be derived from the usability scores, whilst convenience is a broad concept that could encompass many of the traditional usability properties – but it also includes viewing the user-client within the context of his/her total task environment.

The lowest scores are shown for the questions that deal with the clarity and meaning of interactive messages, and the various help facilities. The problem with attaching a single score to the first of these has been discussed previously – the information quality of transactional messages is good but bad for system messages. Similarly user guidance and help are good in some aspects (the pre-transactional facilities), but bad in other aspects such as help during the transaction and the lack of hardcopy help/manuals. Chapter 7 examined other aspects related to poor scaling design for this questionnaire.

A small number of FI's have been evaluated by the JIBC – these are however mainly Web brokerage services and do not (as yet) include Internet banks. The average scores shown by these evaluations vary between 40 and 50%, with similar problems and areas of excellence as observed for Standard Bank. Considering that these evaluations were applied to a different class of FI than an Internet bank, it would be inaccurate to compare the evaluations too closely. It is nonetheless of interest that their evaluations show the same weaknesses such as service speed (response) and help facilities, and

similar strengths such as displaying a commitment to developing and improving the facility, and also avoiding the use of large graphics downloads. The latter characteristic puts even more pressure on these institutions to improve their service response – this is rated as poor despite their effort to avoid incorporating large graphics page components.

## 9.4 Conclusion

Even though the JIBC checklist is custom-designed for an evaluation of Financial Institutions such as Internet banks it exhibits several weaknesses. Firstly it is not a good reflection of the site usability – restricting the list to 16 questions unavoidably required discarding many important usability test factors. Secondly it has not been applied to a sufficiently large group of financial institutions to enable a valid conclusion to be made regarding the meaning of the usability scores. In addition it has not been applied to Internet banks and there seems to have been little response to the invitation to help to develop the questionnaire – it has remained unchanged over a period of six months. In its defense it should be noted that this was by far the most detailed usability-related checklist found on the Internet, and that it is preferred above the simple four or five point, six category rating methods as employed by other sites (as discussed in chapter 7).

In contrast the Ravden checklist is considered to have once again proved it worth. It is possible to use it with no change to evaluate hypermedia applications such as Internet banking. It could be argued that questions such as Q11 in section 1 that deals with monochrome displays are outdated. But this specific question also caters for disabled (colour-blind) user access – if the checklist is to be criticized then it should be on the moderate neglect shown by the checklist for this type of user.

The Web community's professed quest for quick and simple solutions to complex usability problems will not succeed, and the solution does not lie in abbreviating the measurement instrument to a point beyond all usefulness and accuracy. Nielsen (1993:17, 175) is correct in noting that usability heuristic evaluation methods (of which checklists are but one derivative), should be regarded as **discount approaches** – i.e. a comparatively large return is shown for a relatively small investment in time, resources, and expertise. This does not mean that reaching the point where such a return will be shown will be easy, and achieving this will require at least a reasonable degree of effort, time, and knowledge.

## 9.5.1 Appendix 9.A

### The Ravden questionnaire applied to the Standard Bank Web site

SECTION 1: VISUAL CLARITY		N	S	M	A
1	Is each screen clearly identified with an informative title or description?		✓		
2	Is important information highlighted on the screen? (e.g. cursor position, instructions, errors.)		✓		
3	When the user enters information on the screen is it clear:				✓
	(a) Where the information should be entered?			✓	
	(b) In what format it should be entered?			✓	
4	Where the user overtype information on the screen, does the system clear the previous information, so that it does not get confused with the updated input?	✓			
5	Does information appear to be organized logically on screen? (e.g. menus organized by probable sequence selection, or alphabetically.)		✓		
6	Are different types of information clearly separated from each other on the screen? (e.g. instructions, control options, data displays.)			✓	
7	Where a large amount of information is displayed on the screen, is it clearly separated into section on the screen?			✓	
8	Are columns of information clearly aligned on the screen? (e.g. columns of alphanumerics left-justified, columns of integers right-justified.)		✓		
9	Are bright or light colours displayed on a dark background, and vice versa?			✓	
10	Does the use of colour help to make the displays clear?		✓		
11	Where colour is used, will all aspects of the display be easy to see if used on a monochrome or low-resolution screen, or if the user is colour-blind?	✓			
12	Is the information on the screen easy to see and read?			✓	
13	Do the screens appear uncluttered?			✓	
14	Are schematic and pictorial displays (e.g. figures and diagrams) clearly drawn and annotated?			✓	
15	Is it easy to find the required information on a screen?		✓		
16	Overall, how would you rate the system in terms of visual clarity?				
	Very satisfactory			✓	Moderately unsatisfactory
					Very unsatisfactory
<b>KEY: N: Never S: Sometimes M: Mostly A: Always</b>					
Score: 2(0) + 6(1) + 7(2) + 1(3) = 23/48 or 48%					

SECTION 2: CONSISTENCY		N	S	M	A
1	Are different colours used consistently throughout the system? (e.g. errors always highlighted in the same colour.)		✓		
2	Are abbreviations, acronyms, codes and other alphanumeric information used consistently throughout the system?			✓	
3	Are icons, symbols, graphical representations and other pictorial information used consistently throughout the system?		✓		
4	Is the same type of information (e.g. instructions, menus, messages, titles) displayed:			✓	
	(a) In the same location on the screen?			✓	
	(b) In the same layout?			✓	
5	Does the cursor appear in the same initial position on displays of a similar type?		✓		
6	Is the same item of information displayed in the same format, wherever it appears?			✓	
7	Is the format in which the user should enter particular types of information on the screen consistent throughout the system?			✓	
8	Is the method of selecting options (e.g. from a menu) consistent throughout the system?		✓		
9	Is the action required to move the cursor around the screen consistent throughout the system?				✓
10	Is the method of selecting options (e.g. from a menu) consistent throughout the system?				✓
11	Where a keyboard is used, are the same keys used for the same functions throughout the system?				✓
12	Are there standard procedures for carrying out similar, related operations? e.g. updating-deleting information, starting and finishing transactions.)			✓	
13	Is the way the system responds to particular user action consistent at all times ?			✓	
14	Overall, how would you rate the system in terms of consistency?				
	Very satisfactory		✓	Neutral	Moderately unsatisfactory
<b>KEY: N: Never S: Sometimes M: Mostly A: Always</b>					
Score: 0(0) + 4(1) + 7(2) + 3(3) = 27/42 or 64%					

SECTION 3: COMPATIBILITY		N	S	M	A				
1	Are colours assigned according to conventional associations where these are important? (e.g. red = alarm/stop.)		✓						
2	Where abbreviations, acronyms, codes and other alphanumeric information are displayed:		✓						
	(a) Are they easy to recognize and understand?			✓					
3	Where icons-symbols, graphical representations and other pictorial information are displayed:		✓						
	(a) Are they easy to recognize and understand?		✓						
4	Where jargon and terminology is used within the system, is it familiar to the user?		✓						
5	Are established conventions followed for the format in which particular types of information are displayed? (e.g. layout of dates, and telephone numbers.)			✓					
6	Is information presented and analysed in the units with which users normally work? (e.g. batches, kilos, dollars.)			✓					
7	Is the format of displayed information compatible with the form in which it is entered into the system?				✓				
8	Is the format and sequence in which information is printed compatible with the way it is displayed on the screen?			✓					
9	Where the user makes an input movement in a particular direction (e.g. using a direction key, mouse or joystick), is the corresponding movement on the screen in the same direction?				✓				
10	Are control actions compatible with those used in other systems with which the user may need to interact?			✓					
11	Is information presented in a way that fits the user's view of the task?		✓						
12	Are graphics displays compatible with the user's view of what they represent?		✓						
13	Does the organization and structure of the system fit the user's perception of the task?		✓						
14	Does the sequence of activities required to complete a task follow what the user would expect?			✓					
15	Does the system work the way the user thinks it should work?			✓					
16	Overall, how would you rate the system in terms of compatibility?								
	Very satisfactory		Moderately satisfactory	✓	Neutral		Moderately unsatisfactory		Very unsatisfactory
<b>KEY: N: Never S: Sometimes M: Mostly A: Always</b>									
Score: 0(0) + 8(1) + 7(2) + 2(3) = 28/51 or 55%									

SECTION 4: EXPLICITNESS		N	S	M	A				
1	Is it clear what stage the system has reached in a task?		✓						
2	Is it clear what the user needs to do in order to complete a task?			✓					
3	Where the user is presented with a list of options (e.g. in a menu), is it clear what each option means?		✓						
4	Is it clear what part of the system the user is in?		✓						
5	Is it clear what the different parts of the system do?		✓						
6	Is it clear how, where and why changes in one part of the system affect other parts of the system?	✓							
7	Is it clear why the system is organized and structured as it is?	✓							
8	Is it clear why a series of screens are sequenced as they are?		✓						
9	Is the structure of the system obvious to the user?	✓							
10	Is the system well organized from the user's point of view?		✓						
11	Where an interface metaphor is used (e.g. the desktop metaphor in office applications), is this made explicit?		✓						
12	Where a metaphor is employed, and is only applicable to certain parts of the system, is this made explicit?		✓						
13	In general, is it clear what the system is doing?		✓						
14	Overall, how would you rate the system in terms of explicitness?								
	Very satisfactory		Moderately satisfactory		Neutral		Moderately unsatisfactory	✓	Very unsatisfactory
<b>KEY: N: Never S: Sometimes M: Mostly A: Always</b>									
Score: 3(0) + 9(1) + 1(2) + 0(3) = 11/39 or 28%									

SECTION 5: APPROPRIATE FUNCTIONALITY				N	S	M	A
1	Is the input device available to the user (e.g. pointing device, keyboard, and joystick), appropriate for the tasks to be carried out?					✓	
2	Is the way in which information is presented appropriate for the tasks?					✓	
3	Does each screen contain all the information which the user feels is relevant to the task?				✓		
4	Are users provided with all the options which they feel are necessary at any particular stage in a task?				✓		
5	Can users access all the information that they feel they need for their current task?				✓		
6	Does the system allow users to do what they feel is necessary in order to carry out a task?					✓	
7	Is system feedback appropriate for the task?					✓	
8	Do the contents of help and tutorial facilities make use of realistic task data and problems?				✓		
9	Is task-specific jargon and terminology defined at an early stage in the task?				✓		
10	Where interface metaphors are used, are they relevant to the tasks carried out using the system?				✓		
11	Where task sequences are particularly long, are they broken into appropriate subsequences? (e.g. separating a lengthy editing procedure into its constituent parts.)					✓	
12	Overall, how would you rate the system in terms of appropriate functionality?						
	Very satisfactory	Moderately satisfactory	✓ Neutral	Moderately unsatisfactory	Very unsatisfactory		
<b>KEY: N: Never S: Sometimes M: Mostly A: Always</b>							
Score: 0(0) + 6(1) + 5(2) + 0(3) = 16/33 or 49%							

SECTION 6: INFORMATIVE FEEDBACK				N	S	M	A
1	Are instructions and messages displayed by the system concise and positive?					✓	
2	Are messages displayed by the system relevant?				✓		
3	Do instructions and prompts clearly indicate what to do?				✓		
4	Is it clear what actions the user can take at any stage?				✓		
5	Is it clear what the user needs in order to take a particular action? (e.g. which options to select, which keys to press.)				✓		
6	When the user enters information on the screen, is it made clear what this information should be?					✓	
7	Is it made clear what shortcuts, if any are possible? (e.g. abbreviations, hidden commands, type ahead.)				✓		
8	Is it made clear what changes occur on the screen as a result of a user input or action?					✓	
9	Is there always an appropriate system response to a user input or action?				✓		
10	Are status messages (e.g. indicating what the system is doing, or has just done):				✓		
	(a) Informative?				✓		
	(b) Accurate?				✓		
11	Does the system clearly inform the user when it completes a requested action successfully or unsuccessfully?					✓	
12	Does the system promptly inform the user of any delay, making it clear that the user's input or request is being processed?					✓	
13	Do error messages explain clearly:				✓		
	(a) Where the errors are?				✓		
	(b) What the errors are?				✓		
	(c) Why they have occurred?				✓		
14	Is it clear to the user what should be done to correct an error?				✓		
15	Where there are several modes of operation, does the system clearly indicate which mode the user is currently in? (e.g. update, enquiry, simulation.)				✓		
16	Overall, how would you rate the system in terms of informative feedback?						
	Very satisfactory	Moderately satisfactory	✓ Neutral	Moderately unsatisfactory	Very unsatisfactory		
<b>KEY: N: Never S: Sometimes M: Mostly A: Always</b>							
Score: 0(0) + 13(1) + 5(2) + 0(3) = 23/54 or 43%							

SECTION 7: FLEXIBILITY AND CONTROL					N	S	M	A
1	Is there an easy way for the user to 'undo' an action, and step back to a previous stage or screen? (e.g. if the user makes a wrong choice, or does something unintended.)					✓		
2	Where the user can 'undo', is it possible to 'redo' (i.e. to reverse the action)?				✓			
3	Are shortcuts available when required? (e.g. to bypass a sequence of activities or screens.)					✓		
4	Do users have control over the order in which they request information, or carry out a series of activities?					✓		
5	Can the user look through a sequence of screens in either direction?						✓	
6	Can the user access a particular screen in a sequence of screens directly? (e.g. where a list or table covers several screens.)						✓	
7	In menu-based systems, is it easy to return to the main menu from any part of the system?						✓	
8	Can the user move to different parts of the system as required?						✓	
9	Is the user able to finish entering information (e.g. when typing in a list or table of information), before the system responds? (e.g. by updating the system.)					✓		
10	Does the system prefill repeated information on the screen, where possible? (e.g. to save the user having to enter the same information several times.)				✓			
11	Can the user choose whether to enter information manually or to let the computer generate information automatically? (e.g. where there defaults.)				✓			
12	Can the user override computer-generated (e.g. default) information, if appropriate?					✓		
13	Can the user choose the rate at which information is presented?					✓		
14	Can the user choose how to name and organize information that may be needed to be recalled at a later stage? (e.g. files, directories.)					✓		
15	Can users tailor certain aspects of the interface for their own preferences or needs? (e.g. colours, parameters.)					✓		
16	Overall, how would you rate the system in terms of flexibility and control?							
	Very satisfactory	Moderately satisfactory	Neutral	✓	Moderately unsatisfactory	Very unsatisfactory		
<b>KEY: N: Never S: Sometimes M: Mostly A: Always</b>								
Score: 3(0) + 8(1) + 4(2) + 0(3) = 16/45 or 36%								

SECTION 8: ERROR PREVENTION AND CORRECTION					N	S	M	A
1	Does the system validate user inputs before processing, wherever possible?							✓
2	Does the system clearly and promptly inform the user when it detects an error?						✓	
3	Does the system inform the user when the amount of information entered exceeds the available space? (e.g. trying to key five digits into a four digit field.)							✓
4	Are users able to check what they have entered before it is processed?						✓	
5	Is there some form of cancel (or 'undo') key for the user to reverse an error situation?					✓		
6	Is it easy for the user to correct errors?					✓		
7	Does the system ensure that the user corrects all detected errors before the input is processed?						✓	
8	Can the user try out various possible actions (e.g. using a simulation facility), without the system processing the input and causing problems?						✓	
9	Is the system protected against common trivial errors?					✓		
10	Does the system ensure that the user double-checks any requested actions that may be catastrophic if requested unintentionally? (e.g. large-scale deletion.)						✓	
11	Is the system protected against possible knock-on effects of changes in one part of the system?					✓		
12	Does the system prevent users from taking actions which they are not authorized to take? (e.g. by requiring passwords.)							✓
13	In general, is the system free from errors and malfunctions?					✓		
14	When system errors occur, can the user access all necessary diagnostic information to resolve the problem (e.g. where and what the fault is, what is required to resolve it.)					✓		
15	Overall, how would you rate the system in terms of error prevention and correction?							
	Very satisfactory	Moderately satisfactory	✓	Neutral	Moderately unsatisfactory	Very unsatisfactory		
<b>KEY: N: Never S: Sometimes M: Mostly A: Always</b>								
Score: 0(0) + 6(1) + 5(2) + 3(3) = 25/42 or 60%								

SECTION 9: USER GUIDANCE AND SUPPORT						N	S	M	A
1	If there is some form of help facility/guidance on the computer to aid the user when using the system:						✓		
	(a) Can the user request this easily from any point in the system?							✓	
	(b) Is it clear how to get in and out of the help facility?						✓		
	(c) Is the help information presented clearly, without interfering with the user's current activity?					✓			
	(d) When the user requests help, does the system clearly explain the possible actions that can be taken, in the context of what the user is currently doing?						✓		
	(e) When using the help facility, can the user find relevant information directly, without having to look through unnecessary information?						✓		
	(f) Does the help facility allow the user to browse through information about other parts of the system?						✓		
2	If there is some form of hard-copy guide to the system? (e.g. user guide or manual), then:					✓			
	(a) Does this provide an in-depth, comprehensive description, covering all aspects of the system?					✓			
	(b) Is it easy to find the required section in the hard-copy documentation?						✓		
3	Is the organization of all forms of user guidance and support related to the tasks that the user can carry out?						✓		
4	Do user guidance and support facilities adequately explain both user and system errors, and how these should be corrected?						✓		
5	Are all forms of user guidance and support maintained up-to-date?							✓	
6	Overall, how would you rate the system in terms of user guidance and support?								
	Very satisfactory	Moderately satisfactory	Neutral	Moderately unsatisfactory	Very unsatisfactory	✓			
<b>KEY: N: Never S: Sometimes M: Mostly A: Always</b>									
Score: 3(0) + 6(1) + 2(2) + 0(3) = 10/33 or 33%									

SECTION 10: SYSTEM USABILITY PRBLEMS		None	Minor	Major
When using the system, did you experience problems with any of the following:				
1	Working out how to use the system.		✓	
2	Lack of guidance on how to use the system.		✓	
3	Poor system documentation.			✓
4	Understanding how to carry out the tasks.		✓	
5	Knowing what to do next.		✓	
6	Understanding how the on-screen information, relates to what you are doing.		✓	
7	Finding the information you want.			✓
8	Information which is difficult to read clearly.		✓	
9	Too many colours on the screen.		✓	
10	Colours which are difficult to look at for any length of time.		✓	
11	An inflexible, rigid system structure.		✓	
12	An inflexible Help (guidance) facility.			✓
13	Loosing track of where you are in the system or of what you are doing or have done.		✓	
14	Having to remember too much information while carrying out a task.			✓
15	System response times that are too quick for you to understand what is going on.	✓		
16	System response times that are too slow.			✓
17	Unexpected actions by the system.			✓
18	Information that does not stay on the screen long enough for you to see.		✓	
19	An input device which is difficult or awkward to use.	✓		
20	Knowing where or how to input information.		✓	
21	Having to spend too much time inputting information.		✓	
22	Having to be very careful in order to avoid errors.		✓	
23	Working out how to correct errors.			✓
24	Having to spend too much time correcting errors.		✓	
25	Having to carry out the same type of activity in different ways.		✓	
<b>KEY: None: No Problems Minor: Minor problems Major: Major problems</b>				
Score: 2(None) + 16(Minor) + 7(Major)				



## 9.5.2 Appendix 9.B

### The JIBC Internet banking evaluation checklist – scored for the Standard Bank Web site

No.	Criteria description	10 Point polar scale identifiers
1	Overall effort/commitment to create a functional online service?	Low 1 2 3 4 5 6 7 8 9 10 High ✓
2	Service speed?	Slow 1 2 3 4 5 6 7 8 9 10 Fast ✓
3	How often does the system die when online?	Rarely 1 2 3 4 5 6 7 8 9 10 Often ✓
4	Convenience of performing simple transactions through the Net?	Not at all 1 2 3 4 5 6 7 8 9 10 Highly convenient ✓
5	Visible evidence of service's independent security certification?	Non existent 1 2 3 4 5 6 7 8 9 10 Displayed on the Homepage ✓
6	Documentation/passwords send to prospective clients at the start of the service?	Confusing 1 2 3 4 5 6 7 8 9 10 Clear ✓
7	Simplicity/intuitive character of online forms and work processes?	Confusing 1 2 3 4 5 6 7 8 9 10 Clear ✓
8	Illogical interface deviations from the printed statements of the same bank?	Substantial 1 2 3 4 5 6 7 8 9 10 Insignificant ✓
9	Ease of help access through email?	No access 1 2 3 4 5 6 7 8 9 10 Clear access points on every page ✓
10	Clarity/meaning of interactive messages?	Meaningless 1 2 3 4 5 6 7 8 9 10 Clear ✓
11	On site explanations of banking terms?	Inadequate 1 2 3 4 5 6 7 8 9 10 Adequate ✓
12	Ease of help access through the voice call center?	Incompetent 1 2 3 4 5 6 7 8 9 10 Competent Busy ✓ Easy to reach
13	Misbehaving/superfluous Java applets and scripts on the site?	All the time 1 2 3 4 5 6 7 8 9 10 Moderate Everywhere ✓
14	Graphics that are slow to download and are difficult to the eyes?	Flashy Garish Moving 1 2 3 4 5 6 7 8 9 10 Moderate Non-functional ✓
15	Advertising that is slow to download and is difficult to the eyes?	Flashy Garish 1 2 3 4 5 6 7 8 9 10 Reasonable Moving Destructive ✓
16	The likelihood that the bank's own CEO is personally using this service?	Highly unlikely 1 2 3 4 5 6 7 8 9 10 Very likely ✓
Source: www.arraydev.com.		TOTAL SCORE: (94/160) or 59%

## Chapter 10

# Conclusion: Usability and Internet-based banking

The research problem and questions as presented in chapter 1 are revisited. An overview of the conclusions reached in each chapter are presented and integrated. Directions for future research are offered and the research project is concluded.

## 10.1 Introduction

Chapter 1 formulated three research questions based on the research problem:

- (1) *How to improve the quality of interaction (usability) during an Internet-based banking task.*
- (2) *Identifying the most suitable approach and method for measuring this usability.*
- (3) *Creating an awareness that Web usability is important.*

The conclusions reached during the research project suggest the following solutions for these questions.

(1) **Improving the quality of the interaction.** Chapter 6 (Task analysis), chapter 7 (Usability evaluation), chapter 8 (Heuristic evaluation), and chapter 9 (Checklist-based evaluation) all yielded information that may be fruitfully applied to enhance the interaction-usability during an Internet-based banking task. Chapter 6 noted that the simple banking task analysed consisted in the main of a series of simple actions repeated many times, and that these actions were often related to navigation goals. The results from chapters 7 and 8 suggest that general and Web-specific usability guidelines are both suitable frameworks for a heuristic evaluation. Chapters 7 and 9 indicate that a checklist-based evaluation technique is also suitable, but less so. Care should be taken in interpreting the results in absolute terms and it is best used comparatively across a range of similar purpose Web sites or Web pages on the same site.

(2) **Choosing an evaluation approach.** Chapters 8 and 9 applied low-cost usability evaluation methods to Internet banking. These are discount methods and are exemplary examples of this high-yield low-resource-cost approach. The two methods suffer from one important deficiency – it is difficult to obtain reliable quantitative data through their use. As implemented in this study they are more suitable for yielding comparative and qualitative information. The approach taken by Levi and Conrad (1996) where severity and frequency ratings are assigned to usability problems and classes

of problems, can ameliorate this and should be further investigated. Although a checklist may seem to generate statistically derived absolute values they are nonetheless largely subjective. A heuristic evaluation can be used to provide qualitative data if use is made of frequency counts of problems detected for each guideline section – problems could then be categorized in terms of their impact on usability and corrective maintenance.

### **(3) Creating an awareness of the importance of usability for the Web.**

Chapter 3 (Web usability) refers to comments from Buckingham-Shum and McKnight (1997), and Smith, Newman and Parks (1997). They emphasise that most of the Web community are ignoring usability and hypermedia research, that there is a missing link between the large amount of hypermedia and human-computer interaction research and its application to the Web, and that research into the usability of hypertext and hypermedia systems can be directly applied to World Wide Web pages.

The reason for this phenomenon is in part due to the easy access of Web developers to the application domain – few significant barriers of entry exist. This implies that many usability naive and software development inexperienced developers operate in this domain. In this context the various discount engineering methods as proposed by Nielsen (1993), Instone (1998, URL: W3J1), and others (chapter 7) offer a solution. These are low resource (time, money, and skills) – high yield (usability improvement) methods, which are suitable for use by the inexperienced Web developer. Two of these discount methods were used in the research project namely a heuristic and a checklist-based evaluation (chapters 7, 8, and 9).

It is important to note that usability problems identified from the local Internet banking sites are only in part because of the reasons examined above. The bulk of their problems originate from using technology-wise developers for building the interaction interface and the page layout and design, in addition to the backend technical systems and functions. Local banks should follow the example of overseas banks and assemble and make use of multi-disciplinary teams with a broad range of skills. Human, psychology, graphical design, and document design skills should be added to their (in-house and outsourced) development and maintenance teams, in addition to the normal technical expertise present on these teams.

Allocating development and maintenance resources for this purpose is not a problem for the local banks – they are all financially sound. The problem is therefore probably one of (management) attitude and perception, and the apparent low importance that they attach to these less tangible aspects of their critical business support systems. Financial people head these banking institutions – and they often show a lack of understanding of the strategic importance attached to the service of people through technology. To the best of the author's knowledge Jarvis from FNB is the only South African bank board member with an extensive IT background.

## 10.2 A chapter overview

An overview of the conclusions reached for each of the chapters is presented below.

**Chapter 1. Introduction: Internet-based banking and Usability.** This contained a description of the important research dimensions such as the research problem, its aims, the research instrument and the importance of the research and its potential benefits. It was suggested that improving usability for the Internet be of paramount importance given the probable large impact of poor interaction quality. Internet banking is an example of the rapidly expanding Internet e-commerce application domain, and is considered to have many of the characteristics of typical GUI software products. It was noted that accurate, repeatable, and cost-and-time efficient usability measurement approaches and techniques are a pre-condition for such an improvement – else any usability changes may go unnoticed.

**Chapter 2. HCI and Usability.** Important issues related to human-computer interaction were examined. A discussion of usability and its underlying principles and properties, an examination of the modern GUI, and a study of the different classes of interaction styles were presented. It was shown that these aspects of usability and interaction are the basis for all subsequent chapters. The Internet banking task includes elements from many of the interaction styles identified, and operates in a typical GUI and WIMP environment.

**Chapter 3. Usability for the Web.** It was shown that the general usability principles and properties examined earlier also apply to the design of Web-based software products. Web-based applications have an increased focus on certain usability dimensions, and are characterised by a very wide range of client characteristics. It will be advantageous to include the large pool of hypermedia and hypertext research into studies of Web usability.

**Chapter 4. Internet-based banking.** The current state of Internet banking in South Africa, Europe and the USA is sound and adoption rates amongst both banks and banking clients are rapidly increasing. A classification for banks with a Web presence shows that most banks are rapidly approaching the level as described for full Internet banking. The advantages associated with Internet-based banking are manifold and indisputable, and the disadvantages are mainly related to rigid management attitudes, and start-up, maintenance and skilled personnel costs and availability.

**Chapter 5. Data collection.** It was proved necessary to capture Internet banking Web sites to local storage in a browsable and interaction-enabled form, because of the frequent changes exhibited by these facilities. This exercise was made difficult because many banks use dynamically generated Web pages. Page architectures, and specifically help pages, were analysed. These show that the use of frames can be used to good effect in the construction of indexed and context sensitive help pages.

**Chapter 6. Task analysis.** Task analysis concepts, approaches, and benefits were examined. One specific approach – TAKD or task analysis for knowledge description – was applied to an Internet banking task. The advantages associated with the use of such a structured approach in uncovering usability problems were demonstrated and discussed. It was shown that most of the interaction associated with Internet-based banking is related to navigation, and that Internet applications operate within a technology restricted environment. The interaction is characterized by a small number of simple actions that are repeated many times.

**Chapter 7. Usability evaluation.** Web-specific guidelines have merit but it is considered more rigorous to base the evaluations on general usability standards, principles, and properties (chapter 2). Two suitable evaluation methods were proposed and discussed, a heuristic evaluation on the overall Internet banking site and a closely related checklist-based technique on a specific task (chapter 6) within the site. Both methods combined elements from general and Web-specific approaches in their applied format. Implementation details for the actual evaluations are included in chapter 8 and chapter 9.

**Chapter 8. Heuristic evaluation.** The results of applying a heuristic evaluation to South African and international Internet banking sites were presented. The guidelines used were the three categories of general usability principles – learnability, flexibility, and robustness, and were supplemented by four Web-specific guidelines navigation, aesthetic and minimalist design, matching the system and the real world, and consistency and standards. With the exception of Standard Bank which was accessed as a full client, information for all the other banks had been collected through the use of their demonstration facilities. Interaction behaviours were captured as a series of screendumps. A number of usability problems were identified, and one of the reasons suggested for this is the use of the technical developer to design the interaction interface. It was concluded that for security-sensitive applications it is not desirable to promote all classes of usability properties in an unconstrained and unqualified manner. The local Internet banks compare poorly in terms of their usability characteristics to the USA-based banks.

**Chapter 9. Checklist-based evaluation.** The application of two checklist-based usability evaluations to a banking task was presented. The two scored usability checklists were as identified and examined in chapter 7. They are the general usability and detailed Ravden and Johnson (1989) questionnaire, and a single page JIBC checklist designed specifically for an Internet-based banking evaluation. It was shown that the Ravden checklist is suitable for Web evaluation, and that the JIBC checklist is too abbreviated and fails to cover most of the important usability aspects. Both these checklists should be interpreted relatively rather than absolutely – this implies that the results from the study would have been more useful if a number of other Internet banks were also included. As

Standard Bank was the only bank that could be accessed as an actual banking client this was not possible.

### **10.3 Conclusion and directions for future research**

Web usability and Internet banking usability in particular, can benefit from the application of general usability principles and hypermedia and hypertext specific design guidelines. An evaluation of Internet banking yielded a wide range of usability problems across the usability spectrum. A usability discount approach to inspection, which includes techniques such as the heuristic and checklist-based evaluations applied during the research, are suitable techniques but yield mainly qualitative data, and should therefore be interpreted comparatively rather than absolutely. It was emphasized that usability measurement is a precondition for usability improvement – ‘one cannot manage which can’t be measured’.

To put the importance of the study into commercial perspective, Spira (1999:2) prophesizes that ‘no other industry will be changed more by the Internet than banking and insurance’. He also concludes that banking in the new millenium will be dominated by only a handful of global banks, and that this second wave of banking consolidation, which will be cross-border in nature, is imminent.

One area of possible future research could be related to adapting the heuristic evaluation technique to provide reliable, objective, quantitative data in addition to qualitative pointers to usability problems. The work as described by Levi and Conrad (1996) would be useful in this regard. Given the rapid rate of development, use, adoption and acceptance of these Web-based e-commerce facilities, an expanded – both in width (i.e. number of banks and evaluation approaches) and depth – research study would form an ideal and very topical subject for advanced level research in human computer interaction or applied commerce. What is certain is that research in this field has to be kept current – conclusions reached a year before may be inappropriate today.

## List of references

- Accone T. 1999. **This Week on the Internet Column**. Business Times. 1 May 1999.
- Atkinson M and Kydd C. 1997. **Individual Characteristics Associated with World Wide Web Use: An Empirical Study of Playfulness and Motivation**. The DATA BASE for Advances in Information Systems. 28(2).
- Baecker R M and Buxton W A S (Editors) 1987. **Readings in Human-Computer Interaction: A Multidisciplinary Approach**. Morgan Kaufmann Publishers.
- Baguley R. 1998. **Who to Trust with your Money Online**. Internet Magazine. Nov. 1998.
- Bastien J M C and Scapin DL. 1992. **A Validation of Ergonomic Criteria for the Evaluation the Evaluation of Human-Computer Interfaces**. International Journal of Human-Computer Interaction 4(2).
- Bastien J M C and Scapin DL. 1995. **Evaluating a User Interface with Ergonomic Criteria**. International Journal of Human-Computer Interaction 7(2).
- Bevan N. 1997. **Usability Issues in web site design**. In: Proceedings of HCI International '97, San Francisco. 24-30 August 1997, Elsevier.
- Bieber M and Isakowitz T. 1995. **Designing Hypermedia Applications**. Communications of the ACM. 38(8).
- Bieber M, Vitali F, Ashman H, Balasubramanian V and Oinas-Kukkonen H. 1997. **Fourth Generation Hypermedia: Some Missing Links for the World Wide Web**. International Journal of Human-Computer Studies. 47(1).  
URL: <http://ijhcs.open.ac.uk/bieber/bieber-01.html>
- Borges J A, Morales I and Rodriguez N J. 1998. **Page Design Guidelines Developed through Usability Testing**. In Forsythe C, Grose E and Ratner J. (Ed). 1998. **Human Factors and Web Development**. Lawrence Erlbaum Associates.
- Brown D, Totterdell P and Norman M. (Editors). 1990. **Adaptive User Interfaces**. Academic Press.
- Buckingham-Shum S and McKnight C. 1997. **World Wide Web Usability: Introduction to this Special Issue**. International Journal of Human-Computer Studies. 47(1).  
URL: <http://ijhcs.open.ac.uk/>
- Business Times. 1997. **No more Queues as Banks go On-line**. Sunday Times. 14 September 1997.
- Card S K, Moran T P and Newell A 1983. **The Psychology of Human-Computer Interaction**. Lawrence Erlbaum Assoc. (London).
- Cockburn A and Jones S. 1996. **Which way now? Analyzing and Easing Inadequacies in WWW Navigation**. International Journal of Human-Computer Studies. 45.
- Computer Week. 1997. **Another Player Enters E-Banking Realm**. Computer Week (South Africa). 6 October 1996.
- Computer Week. 1997. **Don't Bank on Electronic Commerce Just Yet**. Computer Week (South Africa). 18 November 1996.

- Computer Week. 1997. **Persel Steps Into Electronic Commerce**. Computer Week (South Africa). 26 August 1996.
- Cooper D R and Emory C.W. 1995. **Business Research Methods**. 5th Ed. RD Irwin.
- Crede A. 1997. **International Banking and the Internet**. In: Cronin M J (Editor). *Banking and Finance on the Internet*. Van Nostrand Reinhold.
- Cronin M J (Editor). 1997. **Banking and Finance on the Internet**. Van Nostrand Reinhold.
- Cusumano M A. and Selby R W. 1995. **Microsoft Secrets**. Harper Collins.
- Cusumano M A. and Selby R W. 1997. **How Microsoft Builds Software**. *Communications of the ACM*. 40(5).
- Davis S and Bostrom R. 1992. **An Experimental Investigation of Roles of the Computer Interface and Individual Characteristics in the Learning of Computer Systems**. *International Journal of Human-Computer Interaction* 4(2):143-172.
- De Wet L 1994. **A Comparison of the Usability Properties of Character-based and Graphical-based User Interfaces**. M.Sc Dissertation. University of South Africa.
- Diaper D. 1989. **Task Analysis for Human-Computer Interaction**. Ellis Horwood Ltd.
- Diaper D and Addison M. 1991. **User Modeling: The Task Oriented Modeling (TOM) Approach to the Designer's Model**. In *People and Computers VI*. (Ed.) Diaper D and Hammond N. Cambridge University Press.
- Diaper D and Addison M. 1992. **Task Analysis and Systems Analysis for Software Development**. *Interacting with Computers*. 4(1).
- Diaper D. 1997. **Integrating HCI and Software Engineering Requirements Analysis: A demonstration of Task Analysis Supporting Entity Modeling**. *SIGCHI Bulletin*. 29(1).
- Dieli M, Dye K, McClintock M, and Simpson M. 1994. **The Microsoft Corporation Usability Group**. In Wiklund M.E. (Ed). **Usability in Practice: How Companies Develop User-friendly Products**. AP Professional.
- Diniz E. 1998. **Web Banking in USA**. URL: [www.arraydev.com/commerce/JIBC/](http://www.arraydev.com/commerce/JIBC/)
- Dix A, Finlay J, Abowd G and Beale R. 1993. **Human-Computer Interaction**. Hemel Hempstead: Prentice-Hall International (UK).
- Dix A, Finlay J, Abowd G and Beale R. 1998 (2<sup>nd</sup> Ed.). **Human-Computer Interaction**. Hemel Hempstead: Prentice-Hall International (UK).
- Dykstra, D J. 1993. **A Comparison of Heuristic Evaluation and Usability Testing: The Efficacy of a Domain-Specific Heuristic Checklist**. Ph.D. Department of Industrial Engineering, Texas A&M University, College Station, TX. Op.cit.
- Financial Mail. 1997. **Red Carpets for the Rich**. 10 October 1997.
- Frascina T and Steele R A. 1993. **Task Analysis in the Design of a Human-Computer Interface for a Ward-based System**. *Proceedings of the Fifth International Conference on Human-Computer Interaction '93*. Vol. 2.
- Garzotto F, Mainetti L, and Paolini P. 1995. **Hypermedia Design, Analysis, and Evaluation Issues**. *Communications of the ACM*. 38(8).
- Getz M. 1997. **FNB pledges to deliver more on the Internet**. *The Star*. 19 May 1997.



- Getz M. 1997. **FNB lends weight to Banking's Customer Care Revolution.** The Star. 22 September 1997.
- Gibson, J L, Ivancevich J M and Donnelly J H. 1991. **Organizations: Behavior, Structure, Processes.** 7<sup>th</sup> Ed. Irwin.
- Gordon G. 1999. **Users click in to the benefits of Internet Banking.** Sunday Times, Business Times. 25 April 1999.
- Greenberg R M. 1989. **Faces of Unix.** PC Magazine (USA), September 12, 1989.
- Grose E, Forsythe C and Ratner J. **Using Web and Traditional Style Guides to Design Web Interfaces.** In Forsythe C, Grose E and Ratner J. (Ed). 1998. **Human Factors and Web Development.** Lawrence Erlbaum Associates.
- Grudin J. 1993. **Interface: An Evolving Concept.** Communications of the ACM. 36(1).
- Gulliksen J and Sandblad G. 1995. **Domain-specific Design of User Interfaces.** International Journal of Human-Computer Interaction 7(2):135-151.
- Harrison M.D. 1996. **Techniques for the Design and Analysis of Human Error Resilient Systems.** Department of Computer Science, University of York.
- Hennigan M and Gourvenec Y. 1998. **Internet Banking in Europe.**  
URL: [www.unisys.com](http://www.unisys.com).
- Hoffman D L, Kalsbeek W D and Novak T P. 1996. **Internet and Web Use in the U.S.** Communications of the ACM. 38(12).
- Humphreys K. 1997. **Banking on the Web: Security First Network Bank and the Development of Virtual Financial Institutions.** In: Cronin M J (Editor). Banking and Finance on the Internet. Van Nostrand Reinhold.
- Intelligence. 1998. **The new Generation of Bankers.** September 1998.
- Johnson C and Gray P. 1998. **Temporal Aspects of Usability: Assessing the Impact of Time on User Interface Design.**  
URL: [www.acm.org/sigchi/bulletin/1996.2/Cghris-Johnson.html](http://www.acm.org/sigchi/bulletin/1996.2/Cghris-Johnson.html).
- Johnson P, Diaper D and Long J. 1985. **Tasks, Skills and Knowledge: Task Analysis for Knowledge Based Descriptions.** Human Computer Interaction – INTERACT '84. Shackel B. (Ed.) Elsevier-Science Publishers.
- Johnson P. 1989. **HCI models in Software Design: Task Oriented Models of Interactive Software Systems.** In Software Engineering Environments: Research and Practice. Bennett K H (Ed). Ellis Horwood.
- Katz J and Aspden P. 1997. **Motives, Hurdles, and Dropouts: Who is on and off the Internet and why.** Communications of the ACM. 40(4).
- Kim J and Moon J Y. 1998. **Emotional Usability of Customer Interfaces – Focusing on Cyber Banking System Interfaces.**  
URL: [www.acm.org/sigchi/chi97/proceedings/short-talk/jki.htm](http://www.acm.org/sigchi/chi97/proceedings/short-talk/jki.htm).
- Kirwan B. and Ainsworth L K. (Ed.) 1992. **A Guide to Task Analysis.** Taylor and Francis.
- Lagoutte V. 1996. **The Direct Banking Challenge.** Research project. B.A.(Hons) (European Business Administration). Middlesex University.

- Levi M D and Conrad F G. 1996. **A Heuristic Evaluation of A World Wide Web Prototype**. Interactions. July-August 1996.
- Maguire M. 1997. **RESPECT User Requirements Framework Handbook**. HUSAT Research Institute, The Elms, Elms Grove, Loughborough, Leicester, UK.
- Marcus A. 1993. **Human Communications Issues in Advanced UI's**. Communications of the ACM, 36(4).
- Marsden G. 1997. **Electronic Banking on the Internet**. PC World (Southern Africa). February 1997.
- Mashey J. 1996. **An Interview with John Mashey of SG**. PC Pro (UK). 18.
- Mayhew D J. 1992. **Principles and Guidelines in Software User Interface Design**. Eaglewood Cliffs, New Jersey: Prentice Hall.
- Mayhew D. 1998. **Introduction to Human Factors and the Web**. In Forsythe C, Grose E and Ratner J. (Ed). 1998. **Human Factors and Web Development**. Lawrence Erlbaum Associates.
- McCoy J. 1996. **Mastering Web Design**. SYBEX.
- Melliard J. 1998. **Online Banking**. Smart Office Computing. September/October 1998.
- Moran T P. 1981. **The Command Language Grammar: A representation for the user interface of interactive computer Systems**. Int. Journal of Man Machine Studies. 15(1981). Op.cit.
- Morville P. 1997. **Web Design and Development '97**. SIGCHI Bulletin. 29(3).
- Newman W M and Lamming M G. 1995. **Interactive System Design**. Wokingham: Addison Wesley.
- Nielsen J. 1990. **A Meta-Model for Interacting with Computers**. Interacting with Computers. (2).
- Nielsen, J. 1992. **Finding usability problems through heuristic evaluation**. Proceedings ACM CHI'92 Conference (Monterey, CA, May 3-7).
- Nielsen J. 1993. **Usability Engineering**. AP Professional Academic Press.
- Nielsen J. 1995(1). **A Home-Page Overhaul using other Web Sites**. IEEE Software. May 1995.
- Nielsen J. 1995(2). **Using Paper Prototypes in Home-Page Design**. IEEE Software. July 1995.
- Nielsen J. 1995(3). **Applying Discount Usability Engineering**. IEEE Software. January 1995.
- Nielsen J. 1996(1). **Usability Metrics: Tracking Interface Improvements**. IEEE Software. November 1996.
- Nielsen J. 1996(2). **The Importance of being Beautiful**. IEEE Software. January 1996.
- Nielsen J. 1998. **The Difference between Web design and GUI Design**.  
URL: [www.useit.com/alertbox/9705a.html](http://www.useit.com/alertbox/9705a.html).
- Nielsen J. and Mack R. (Editors). 1994. **Usability Inspection Methods**. John Wiley and Sons.
- Nielsen J and Molich R. 1989. **Teaching User Interface Design based on Usability Engineering**. ACM SIGCHI Bulletin 21(1). Op.cit.

- Pecenik M. 1998. **The Experience of the Nuova Banco Di Credito Di Trieste with Electronic Commerce.** URL: [www.arraydev.com/commerce/JIBC/9801-12.htm](http://www.arraydev.com/commerce/JIBC/9801-12.htm).
- Pederson S. 1998. **Barclays Bank criticised over its online service charges.** Personal Computer World (UK). July 1998.
- Preece J and Rombach H D. 1994. **A Taxonomy for Combining Software Engineering and Human-Computer Interaction Measurement Approaches: Towards a Common Framework.** International Journal of Human Computer Studies (1994) 41.
- Ravden S J and Johnson G I. 1989. **Evaluating Usability of Human-Compute Interfaces: A Practical Method.** John Wiley and Sons.
- Rice J, Farquhar A, Piernot P, and Gruber T. 1998. **Using the Web Instead of a Window System.** URL: [www.acm.org/sigchi/proceedings/papers/Rice/jpr\\_txt.htm](http://www.acm.org/sigchi/proceedings/papers/Rice/jpr_txt.htm).
- Robbins S P. 1991. **Organizational Behavior: Concepts, Controversies, and Applications.** 5<sup>th</sup> Ed. Prentice Hall International.
- Rumbauch J, Blaha M, Premerlani W, Eddy F. and Lorenzen W. 1991. **Object-Oriented Modeling and Design.** Prentice-Hall International.
- Shneiderman B. 1992. **Designing the User Interface: Strategies for Effective Human Computer Interaction.** 2nd Edition. Reading MA: Addison Wesley.
- Shneiderman B. 1997. **Designing Information-abundant Web Sites: Issues and Recommendations.**  
URL: [www.hbuk.co.uk/ap/ijhcs/webusability/shneiderman/shneiderman.html](http://www.hbuk.co.uk/ap/ijhcs/webusability/shneiderman/shneiderman.html)
- Schutzer D. 1997. **Foundations for Electronic Commerce: Standards and Strategic Alliances.** In: Cronin M J (Editor). Banking and Finance on the Internet. Van Nostrand Reinhold.
- Seymour J. 1989. **The GUI: An Interface You won't Outgrow.** PC Magazine September 12, 1989.
- Shapshak D. 1997. **FAQ.NET: On-line Banking.** .NET (South Africa). September 1997.
- Smilowitz E. 1997. **Do Metaphors Make Web Browsers Easier To Use?**  
URL: [www.baddesigns.com/mswebcnf.htm](http://www.baddesigns.com/mswebcnf.htm).
- Smith P A, Newman I A and Parks L M. 1997. **Virtual Hierarchies and Virtual Networks: Some Lessons From Hypermedia Usability Research Applied to the World Wide Web.** International Journal of Human-Computer Studies. 47(1).  
URL: <http://ijhcs.open.ac.uk/smith/smith-01.html>
- Smith S L and Mosier J N. 1986. **Guidelines for Designing User Interface Software.** Mitre Corporation Report MTR-9420. Mitre Corp. Op.cit.
- Spira J. 1999. **New Millenium a Second Coming for IT and Financial Services.** The Star, Business Report. 2 June 1999.
- Stanck W R. 1996. **HTML, CGI, SGML, VRML, JAVA Web Publishing Unleashed.** Sams.net Publishing.
- Sullivan J W and Tyler S W. (Editors). 1991. **Intelligent User Interfaces.** Addison Wesley.
- Sunday Times. 1999. **Bank-fee Logic is Hard to Understand.** 14 March 1999.
- Sutcliffe A. 1988. **Human-Computer Interface Design.** MacMillan Education.

- Tauscher L and Greenberg S. 1997. **How People Revisit Web pages: Empirical Findings and Implications for the Design of History Systems.**  
URL: [ijhcs.open.ac.uk/tauscher/tauscher.html](http://ijhcs.open.ac.uk/tauscher/tauscher.html).
- Tebbut D. 1997. **Future Interfaces.** PC Pro. (36). October 1997.
- The Star. 1997. **It Makes Sense to Bank on the Internet.** Star. 4 November 1997.
- The Star. 1997. **Pick and Pay Financial Services: Banking on the Internet.** Star. 27 November 1997.
- Thimbleby H. 1990. **User Interface Design.** Wokingham: Addison-Wesley.
- Thimbleby H. 1997. **Gentler: A Tool for Systematic Web Authoring.** International Journal of Human-Computer Studies. 47(1).  
URL: <http://ijhcs.open.ac.uk/thimbleby/thimbleby-01.html>
- Treu S. 1994. **User Interface Design: A Structured Approach.** Plenum Press.
- Thüring M, Hannemann J and Haake J M. 1995. **Hypermedia and Cognition: Designing For Comprehension.** Communications of the ACM. 38(8).
- Uddin Q Z. 1999. **LUTAKD Tool for the Representation of JSD/Entity Life History.**  
URL: <http://www.soc.staffs.ac.uk/~cmrzuq/abstakd.html>.
- Van Dyk T J. 1996. **A Usability Evaluation of the File and Resource Browser-Manager Components of three Common Desktop Graphical User Interfaces.** Unpublished research report. Department of Computer Science and Information Systems. University of South-Africa.
- Van Dyk T J. 1998. **Laboratory-based Usability Testing.** Unpublished research report. Department of Computer Science and Information Systems, University of South Africa.
- Vora P. 1998. **Human Factors Methodology for Designing Web Sites.** In Forsythe C, Grose E and Ratner J. (Ed). 1998. **Human Factors and Web Development.** Lawrence Erlbaum Associates.
- Wallace C P. 1998. **Why are Europeans Still so Cool to E-Commerce?** Fortune Magazine. 21 December 1998.
- Webster C. 1997. **For Whom the Beltel Tolls.** Computer Week (South Africa). 27 August 1997.
- Weir G R S and Alty J L (Editors). 1991. **Human-Computer Interaction and Complex Systems.** Academic Press (London).
- Weller H G and Hartson H R. 1992. **Metaphors for the Nature of Human-Computer Interaction in an Empowering Environment: Interaction style influences the Manner of Human Accomplishment.** Computers in Human Behaviour 8:313-333.
- Whitaker L A. 1998. **Human Navigation.** In Forsythe C, Grose E and Ratner J. (Ed). 1998. **Human Factors and Web Development.** Lawrence Erlbaum Associates.
- Winograd T. 1995. **From Programming Environments to Environments for Designing.** Communications of the ACM, 38(6).
- Woodson W E, Tillman B and Tillman P. 1992. **Human Factors Design Handbook.** (2nd Ed.). McGraw Hill.

## Internet URL's

- ACM1.** 1998. [www.acm.org/sigchi/chi96/proceedings/shortpap/Rodriguez/rn\\_txt.htm](http://www.acm.org/sigchi/chi96/proceedings/shortpap/Rodriguez/rn_txt.htm)
- Borges A and Morales I. 1998. **Guidelines for Designing Usable World Wide Web Pages.**
- Andiron1.** 1997. [www.andiron.com](http://www.andiron.com)
- AOL1.** 1998. [www.aol.com](http://www.aol.com)
- AOL2.** 1998. [www.bankrate.com/brm/](http://www.bankrate.com/brm/)
- Branaghan1.** 1997. [www.branaghan.com/volume21.htm](http://www.branaghan.com/volume21.htm)
- CNET1.** 1998. [www.cnet.com/Contents/Features/Dlife/Banking/index.html](http://www.cnet.com/Contents/Features/Dlife/Banking/index.html)
- CNET2.** 1998. [www.cnet.com/Contents/Features/Dlife/Banking/ss03.html](http://www.cnet.com/Contents/Features/Dlife/Banking/ss03.html)
- CNET3.** 1999. [www.builder.com](http://www.builder.com)
- IBM1.** 1997. [www.ibm.com/ibm/hci/guidelines/design/principles.html](http://www.ibm.com/ibm/hci/guidelines/design/principles.html)
- IBM2.** 1999. [www.ibm.com/ibm/hci/guidelines/design/3d/toc.html](http://www.ibm.com/ibm/hci/guidelines/design/3d/toc.html)
- IBM3.** 1997. [www.ibm.com/guidelines/web/intro.html](http://www.ibm.com/guidelines/web/intro.html)
- JIBC1.** 1998. [www.arraydev.com/commerce/JIBC/](http://www.arraydev.com/commerce/JIBC/)
- JIBC2.** 1998. [www.arraydev.com/commerce/JIBC/grnline.htm](http://www.arraydev.com/commerce/JIBC/grnline.htm)
- MAST1.** 1997. [www.mast.co.za/mit/mit\\_intro\\_uec.htm](http://www.mast.co.za/mit/mit_intro_uec.htm)
- Microsoft1.** 1997. [www.microsoft.com/win32dev/uiguide/uigui006.htm](http://www.microsoft.com/win32dev/uiguide/uigui006.htm)
- Microsoft2.** 1997. [www.microsoft.com/usability/faq.htm](http://www.microsoft.com/usability/faq.htm)
- Netbanker1.** 1998. [www.netbanker.com](http://www.netbanker.com)
- NPL1.** 1997. [www.npl.co.uk/npl/sections/us/frames/fweb.html](http://www.npl.co.uk/npl/sections/us/frames/fweb.html)
- Orc1.** 1997. [www.orcc.com/orc-fis.htm](http://www.orcc.com/orc-fis.htm)
- Pantos1.** 1998. [www.pantos.org/atw/35317.html](http://www.pantos.org/atw/35317.html)
- Sun1.** 1997. [www.sun.com](http://www.sun.com)
- Unisys1.** 1998. [www.internet-banking.com/barom.html](http://www.internet-banking.com/barom.html)
- Useit1.** 1998. [www.useit.com/papers/heuristic/heuristic\\_list.html](http://www.useit.com/papers/heuristic/heuristic_list.html)
- Useit2.** 1998. [www.useit.com/papers/heuristic/heuristic\\_evaluation.html](http://www.useit.com/papers/heuristic/heuristic_evaluation.html)
- Nielsen J. 1998. **How to Conduct a Heuristic Evaluation.**
- Uswest1.** 1998. [www.uswest.com/Web-conference/proceedings/scholtz.html](http://www.uswest.com/Web-conference/proceedings/scholtz.html)
- Scholtz J. 1998. **Web Usability: The Search for a Yardstick.**
- W3J1.** 1998. [www.w3j.com/5/s3.instone.html](http://www.w3j.com/5/s3.instone.html)
- Instone K. 1998. **Usability Engineering for the Web.**
- Webreview1.** 1998. [webreview.com/wr/pub/97/10/10/usability/index.html](http://webreview.com/wr/pub/97/10/10/usability/index.html)
- Webreview2.** 1998. [webreview.com/97/04/25/usability/index.html](http://webreview.com/97/04/25/usability/index.html)