Table of contents

Chapter 1

Introduction

1.1. Orientation

In recent times, people have become increasingly reliant on computers, and using them on a daily basis. This is evident everywhere; from businesses and schools to private homes. As a result, the need has arisen to optimise the task-related experience in terms of time-efficiency, which demands effective training in software skills. This study focuses mainly on the methods used to assess a user's software skills. It also considers aspects of human-computer interaction (HCI) in developing the optimum software skills e-assessment system. Here, human-computer interaction refers to the "study, planning, and design of what happens when you and a computer work together" (Danino, 2001, p1). The field also encompasses the process of comparing, optimising, and implementing different user interfaces in ways that enable users to best interact with computers (Hewett, Baecker, Card, Carey, Gasen, Mantei, Perlman, Strong and Verplank, 1992, 1996).

Having to adapt to new technologies and software scenarios, the modern computer user is always involved in the steady process of mastering skills, a process referred to as the acquisition of software skills (Smith, 2004). These particular skills are taught and learned at most modern educational institutions, and are seen as a crucial part of any graduate's knowledge, especially where basic computer literacy is concerned. In today's terms, computer literacy refers to how proficiently individuals use computer operating systems and their software programs (Harvard Glossary, 2007). In terms of software skills, the focus in this research project is on word processing.

The most popular software package is the Microsoft Office suite of software applications (Escobedo II, 2007). Microsoft Office Word (MS Word) is one of the programs included in the Microsoft Office package. In essence, it is a multi-functional word processing system capable of reading, editing and redistributing documents, websites and many other types of files. The first release of the software dates back to 1983, when it was known as Multi-Tool Word (Pollson, 2006). At the time, the application was released to work on XENIX systems. Adaptations of Word were

subsequently released to run on many other systems, including the likes of UNIX, DOS, Apple Mac and eventually the Windows operating system environment (Allen, 2001).

When involved in determining a user's knowledge with regard to word processing (or other software-related functions), the term "assessment" comes into play. Munduca, Savina and Merritts (2007, p1) state that, "in an educational context, assessment refers to the process of observing learning; describing, collecting, recording, scoring, and interpreting information about a student's or one's own learning". It is important to note that this definition not only focuses on accumulating observed objectively verifiable data (adhering to what is known as the positivist paradigm), but also on interpreting results (in agreement with, amongst others, the adherents of subsequent non-positivist paradigms, such as constructivism) (Guba, E.G., Ed 1990). This means that assessment progressively recognises the influence of psychological, normative, cognitive and motivational issues (see chapter 2) on the results of assessment. Assessment has advanced from an emphasis on objectivity (obtaining so-called pure data) to subjectivity (acknowledging the complexity of the user's interaction with the computer); from a paradigm of theory and value neutrality to one of theory and value ladenness (Guba, E.G., Ed 1990).

1.2. Motivation

1.2.1. Wide-spread use of MS Word

The main reason for choosing MS Word to aid the fundamental hypothesis (see section 1.5) of this research study, is its frequency of use: practically everyone with the Microsoft Windows operating system on their personal computer (PC) will also use MS Word at some point. The scope of PC users ranges from children and teenagers who need the application for school homework and reports, to people in every field of work, who assist their companies in all forms of business-related or other means of communication, not to mention the vast world of tertiary education and the even more encompassing domain of book publishing. Because MS Word is so widely used, proficiency in performing essential word processing functions quickly and effectively is essential for job seekers in today's competitive business world.

1.2.2. The need for an optimal e-assessment tool

One of the problems that spurred the development of a new software skills assessment tool is the reported dissatisfaction of users (students at the University of the Free State - UFS) with the **virtual, simulated** MS Word software environment used to electronically assess their word processing skills.

As a result of such shortcomings, and in trying to improve some of the assessment paradigms employed by the existing test system, Microsoft's Office Development tools were researched for the purpose of developing a new automated software skills assessment application (see chapter 5 and appendix A). The new application can be used to investigate the effect of using assessment methods to potentially enhance the software skills assessment process. It does not emulate the MS Word environment, but uses the real environment while monitoring software objects (i.e. monitoring MS Word objects and their properties with the use of Microsoft Office Tools for Visual Studio 2005) in a new way.

It must be noted that assessing MS Word skills is only sensible if done in an automated manner. Other methods place great strain on lecturers, especially if student numbers range in the hundreds or even thousands. At the UFS, there are several hundred students enrolled in computer literacy courses.

For the purpose of assessing a user's word processing skills within MS Word, the existing test system used at the UFS employs a virtual, Flash-driven software environment (this is addressed in more detail in section 6.3.3.2) where users are presented with a similar environment as that within MS Word. In essence, the application looks and "feels" like a version of MS Word, but with very limited functionality. These limitations include restricting the user to fixed methods for many tasks, as well as eliminating the ability to experiment by means of trial-and-error.

The problem is that such limitations can potentially lead to an unreliable representation of a student's true knowledge of the software program and the skills required to operate it, which can also hamper student learning and creativity. As a

result, students may feel that they have not been assessed in a comprehensive or reliable way with regard to their acquired skills.

As an example, the existing test system model does not allow users to see the result of a task performed in the simulated MS Word environment. After users have performed the steps involved in completing a certain task as part of a software skills test, they would be presented with the option of submitting the answer (for the existing test system program to assess) or repeating the question.

Another example: in some instances, the existing test system forces users to use a fixed method to perform a task, even though there are many different ways to accomplish certain actions within MS Word. This might confuse and irritate users who are accustomed to certain shortcut keys and other methods. Users might not want to or know how to use the proposed methods.

The grievances reported by UFS students with regard to these assessment methods, motivated an investigation of such methods to determine their reliability in assessing a user's word processing skills. In this sense, it was vital to not only determine which methods would assess word processing skills most reliably, but also to find a way that would be least frustrating to the students being assessed.

Chapter 2 will detail how frustrating and limited user interfaces affect the mood of the end-user. From personal experience observed by administering MS Word skills tests on hundreds of students with the existing test system in September 2006, the author has encountered the following problem: Numerous situations occurred where, if students are irritated with the MS Word skills assessment program, then they submit test answers even if they aren't sure that the answer would be correct. More details regarding the development of WordAssessor and its usage with the purpose to optimise the software skills learning and assessment process, are discussed and explained in chapter 5 and appendix A.

1.3 Aim / Main objective

The aim of this study is to determine the optimum e-assessment paradigm for assessing word processing skills. Different assessment methods are compared (see chapter 7) to determine which yields the most reliable representation of a user's true software skills knowledge.

Another purpose of this project is to determine if a more realistic software skills assessment (computer-based) environment would allow users a greater degree of certainty with regard to the correctness of the task performed. Very few users of any software application can instantly perform all the key operations within a software environment. The reason for this is that a certain amount of trial-and-error operations are sometimes involved in learning and operating the software (Edwards, 2004).

The aim is furthermore to find an assessment method that can yield positive experiences for users. A study of user behaviour in this sense could provide guidelines for future interactive learning projects.

Also, the study aims to investigate certain aspects of formative assessment in a software skills assessment environment. Harlen (2007) emphasises that formative assessment involves the process of assessing student knowledge in a certain area, simultaneously aiding the learning process itself. Boyle (2007) states that further research is needed in the area of feedback for e-tests (i.e. explaining to students how or where they went wrong in following a certain form of electronic/automated assessment).

He explains that the design principles for e-test feedback methods need to be established in order to optimise a student's learning experience. Consequently, this research will implement a practical approach to formative e-assessment (see section 5.3.5), with the aim to determine student preference with regard to which of the different types of post-assessment feedback seems to best stimulate learning.

1.4. Research methodology

The interaction, performance, and assessment methods utilised by two different software skills assessment programs are investigated and compared. The first program is referred to as "the existing test system", and is commercially available. The second program is referred to as "WordAssessor" – a system that the author developed for the purpose of this research study (see chapter 5 and appendix A).

The newly proposed e-assessment "methods" utilised by the "WordAssessor" program builds upon the foundations of methods used by the existing test system, but attempts to improve the entire software skills assessment experience by removing possible limitations, while broadening functionality (see chapter 5).

Furthermore, a personalised test is conducted (a list of tasks/questions is distributed to users, who are asked to perform the tasks within the MS Word environment while being supervised and assessed by an observing evaluator (person)). This personalised test will be used as a benchmark (see section 7.3.1) for determining users' true software skills knowledge and plays a vital role in this investigation.

A practical approach is used to analyse and compare test results obtained from a personalised software skills test with results from the computerised software skills tests (see chapter 6 for details regarding data collection methods, user groups and test setting). This comparison is used to determine which e-assessment methods assess users' knowledge most reliably. Chapter 6 details how "the existing test system" and "WordAssessor" differ from each other.

A case study is conducted to determine student preference with regard to the assessment methods employed by the automated systems. Student preference pertaining to different forms of e-assessment feedback is also examined. Quantitative research is conducted by means of a questionnaire that students will receive directly following their word processing skills assessment.

This research study is based on the constructivist teaching paradigm (see section 4.3), whereby students construct knowledge for themselves instead of simply reproducing certain facts received from teachers (Guba, E.G., Ed 1990).

To further enhance learning, the e-assessment paradigms mentioned above attempt to focus on the optimal usability of an e-assessment program for positive user-computer interaction. The document *ISO 9241-11 (1998),* issued by the International Standards Organisation, defines usability as: "The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use". The document also defines effectiveness, efficiency and satisfaction as follows: Effectiveness - "Accuracy and completeness with which users achieve specified goals". Efficiency - "Resources expended in relation to the accuracy and completeness with which users achieve goals". Satisfaction – "Freedom from discomfort, and positive attitudes towards the use of the product".

In this way, the current method of assessing users in this area can be compared to the proposed new methods and provide aid in the optimisation of assessment paradigms and testing strategies.

To provide a clearer picture of what this comparative study will attempt, hypotheses must be constructed.

1.4.1. Hypotheses

"A hypothesis is a tentative assumption or explanation for an observation, phenomenon or scientific problem that can be tested by further investigation" (Leach, 2004, p58). Basically, hypothesis testing is used in evaluating the probable validity or invalidity (using collected data) of a postulated theory (QMSS, 2007). The following null hypotheses - statements that are accepted to hold unless proved otherwise - were formulated for this study:

$H_{0,1a}$: There is no difference in the assessment outcome of a computerised assessment tool that evaluates the path/method followed, and a personalised software skills test.

7

$H_{0,1b}$: There is no difference in the assessment outcome of a computerised assessment tool that evaluates the task outcome, and a personalised software skills test.

$H_{0,2}$: When referring to an end-user computer task, there is no difference in the assessment outcome when assessing the path/method followed, as opposed to the task outcome.

$H_{0,3}$: Allowing a user to see the end result of a performed action does not result in a more reliable indication of his/her skills.

$H_{0,4}$: There is no difference in the assessment outcome of a computerised assessment tool that restricts users to use certain methods to perform a task, as opposed to allowing the users to use any method to perform a task.

$H_{0,5}$: There is no difference in the preference of students to work in a simulated, scaled-down software environment, and the actual (possibly complex) software environment.

$H_{0,6}$: There is no difference in the preference of students when receiving plain text-based feedback after a software skills test, as opposed to receiving feedback via video tutorials.

$H_{0,7}$: Directly following a software skills test, students do not feel that they have learned more effectively if the solutions of incorrectly answered test questions are presented to them via video tutorials.

With the purpose of this research study clearly defined, a brief summary of the contents is presented.

1.5. Outline of the dissertation

In the following chapter, human-computer interaction in skills assessment is analysed with regard to the cognitive and emotional aspects of the user, as well as which aspects of usability need to be focused on in the development of software skills training computerised tools.

The current practice with regard to methodology and techniques used in software skills assessment is detailed in chapter 3 to provide a foundation for aiding the main comparative investigation in chapter 7. In chapter 4, assessment paradigms, various forms of feedback, and the role of assessment in higher education and software skills assessment are discussed.

The development process, as well as the technical structure of the newly developed computerised skills assessment tool, WordAssessor, are covered in chapter 5 and appendix A. In addition, this new system and its assessment methods are compared to the assessment methods employed by the existing test system.

In chapter 6, the details and practical testing methodology used in software skills assessment for a group of UFS students are discussed. In this experiment, students' software skills in MS Word were assessed in three ways. Firstly, one group (25 students) was assessed without the aid of a computerised assessment tool; a second group of students (160 students) was assessed using the existing system package, with a third group of students (160 students) assessed with the WordAssessor system. The two groups of students assessed with computerised assessment tools were taken from a population of about 1000 students.

An interpretation and analysis of the aforementioned test results and experience are presented in chapter 7. From the results, conclusions are drawn in chapter 8 and recommendations provided to assist the future development of software skills assessment programs, as well as detailing the assessment paradigms that can allow for the most comprehensive assessment process with regard to reliably determining a student's word processing skills knowledge.

Chapter 2

Human-computer interaction in software skills assessment

This chapter focuses on the following aspects:

- **The growth of human-computer interaction**
- **User information processing**
- **Emotion**
- **Visual design principles**
- **Focus points in software skills assessment**

2.1. Introduction

In the previous chapter, a brief introduction explains the aim of the study. In this chapter, the various aspects of human-computer interaction (HCI) involved in software skills assessment, as well as the guidelines necessary to develop the most usable software skills assessment tool, are discussed. These guidelines are necessary to aid the development of the software skills assessment tool named "WordAssessor". It is vital that this tool builds upon the most recent and successful methods of e-assessment in order to determine the optimum e-assessment paradigm for assessing users' word processing skills.

2.1.1. The role of HCI in software skills assessment

To highlight the importance of effective software training and skills assessment, it is vital to understand how widespread the need for adequately trained software users has become. This necessity began to enjoy greater recognition during the 1980's, when the term HCI replaced the previously accepted man-machine interface or interaction (MMI) (Faulkner, 1998). The main understanding implied is that human beings interact with computers in order to complete work. In this study, "work" refers to document editing, restructuring, and general word processing.

To gain more insight into the nature of HCI, one has to consider a broad definition of this concept. HCI is "a discipline devoted to helping people meet their needs and goals by making computing technology accessible, meaningful, and satisfying" (Carrol, 2002, p191).

As this research project involves the development of a computer program for the purpose of assessing word processing skills (as mentioned in section 1.3), the above-mentioned definition of HCI can serve as a guideline for a program that can most reliably assess software skills, and therefore optimally benefit the user.

As the complexity and capability of technology increases, new software and hardware are creating new and exciting opportunities for HCI (Preece, Rogers, Sharp, Benyon, Holland, Tom, 1994). Progress in HCI can most aptly be characterised by persistent improvements in hardware capability (Grudin, 2005). This allows the inspired visions of developers to become reality, serving a wide audience of users (Grudin, 2005).

An example can be seen in chapter 5 and appendix A (the creation of a word processing skills assessment programme by means of recently released office programming tools for MS Visual Studio 2005). However, as is the case with any development process, a clear set of goals needs to be established before progress can take place. The main goals of HCI are to "produce usable and safe systems, as well as functional systems" (Preece *et al.*, 1994, p14). In this instance, usability ("usable") refers to a vital concept in human-computer interaction that is involved in trying to make software skills simple to learn and utilise (Preece *et al*, 1994).

To improve the understanding of which HCI aspects need to be focused on during the preparation and execution of the software skills assessment comparative study in chapter 7, it is helpful to briefly consider how HCI has evolved during the past five or so decades.

2.2. HCI growth and development

Fogg (2002) states that there have been five primary "waves" of focus that are representative of how computing has matured.

The *first* **of these "waves"** is the time period (which began over half a century ago) where the primary focal point of computing was on the **function** of devices. The primary function of the tool to aid this research study, is the automated assessment of students' word processing skills.

The *second* computing "wave" (**the entertainment "wave"**) that Fogg (2002) describes was inspired by the inception of digital gaming. Today, the number of leisure activities for which a personal computer can be used can be so addictive that it can hamper productivity.

According to Fogg, computing experienced a **third** wave of development in the 1980's where "**ease of use"** became a priority, since computers were now targeting ordinary people (*as opposed to engineers, scientists etc.*) as their key demographic group. This is an important inspirational "wave" for one of the main purposes of this research study (*developing a highly usable, efficient software skills assessment tool in order to determine optimal software skills assessment paradigms*).

The next "wave" (*fourth*) that Fogg describes involves the growth and integration of **networks** on a massive scale. He refers to the birth of the World Wide Web in the 1990s. As will be seen in chapter 5, the WordAssessor program briefly described in chapter 1, heavily relies on the modern use of computer networks for data collection purposes onto a central data storage server (see section 5.3.1).

Fogg sees the last (*fifth*) "wave" as a period where computers use methods of persuasion to motivate users to use certain programmes or return to certain websites. The WordAssessor e-assessment tool used for this study attempts to persuade students to choose its methods of e-assessment over others by allowing them to use trial-and-error to find the solutions to problems (see section 2.5.6).

Methods of achieving a positive influence on users are discussed in the remainder of the chapter.

2.3. Human information processing in computer interaction

When a human being interacts with a computer, the focus is essentially an information processing task (Proctor & Vu, 2002). Users usually have a number of **objectives** to accomplish while interacting with the computer; launching specific software and giving the computer commands whereby, for example, the reformatting of documents can occur within a word processor environment (Proctor & Vu, 2002).

Users must adapt to changes taking place on the screen as a result of their actions, and adjust/respond to those changes by recalling which (for *example*, word processing) **functions** need to be activated in order to reach the pre-set (e.g. document reformatting) goals, and remember how to activate them (Proctor & Vu, 2002).

From the above-mentioned example (*visible interface changes resulting from user interaction with a word processor*), it can be seen that the information processing procedure is of importance in analysing the way users act within a word-processing environment. For the purpose of this research study, refer to a standard word processing environment. If one knows more about how users feel and think whilst operating in this environment, one can begin to understand which ways of skills assessment will be representative of their actual knowledge (*see chapter 7 for further discussion of user interaction analyses and their interaction preferences in a word-processing environment*).

Users' thought and behavior process can be observed for example by how they format a certain selection of text bold (making the text appear thicker e.g. B (not bold) -> **B (bold)**). In MS Word 2003, there are at least four different methods of bolding a selection of text. Knowing which method will be the most convenient may be a matter of personal preference. However, such preference cannot be assumed. Finding the best way of assessing a user's knowledge in this regard can be investigated by means of a comparative empirical study (see chapter 7). To clarify, the above-mentioned empirical comparison involves the process of determining whether users perform better in a software skills test if they have a choice of methods to answer a question, instead of being told which method to use.

2.3.1. Emotion and memory of users during skills assessment

Emotion is an "affective state of consciousness in which joy, sorrow, fear, hate, or the like is experienced…" (Flexner, 1993, p637). **Emotion** plays an important role in the way a user interacts with a computer or particular software. Emotional reaction to certain situations can affect performance. For example, as human beings, we struggle to solve complex problems when we are stressed or irritated, yet we find it easier to solve the same problems under relaxed conditions, allowing us to be creative (Dix, Finlay, Abowd, Beale, 2004).

Brave and Nass (2002) note that, in essence, an effective user interface needs to be able to regulate emotions or the state of a user's mood in a way that directs whatever the user currently feels into a productive direction. They provide two examples from both ends of the emotional spectrum:

The first is how the **regulation** of too much positive emotion (e.g. comical, hysterical behaviour) in HCI can enhance productivity, e.g. minimise unsuitable laughter caused by an excess of positive stimulation in a working environment. The second example concerns user frustration: An effective interface should be able to "sense" when the user is unhappy with a current task and allow for the possibility of pursuing some other task (Brave & Nass, 2002).

In the field of **software skills assessment,** this would translate to the ability of a skills assessment tool to limit a linear questioning strategy as much as possible in favour of a more open-ended approach. If users are asked to perform a certain task (as a test question), they should not be forced to submit the answer without having the option of skipping ahead to another question and returning to the original one at a later stage. This approach is implemented in the design of WordAssessor (see section 5.3.3).

Faulkner (1998) points out that, after each task is executed through the user interface, there should always be some indication (or cue) that progress is taking place. Examples include a message box, some aural confirmation signal, or a slight run-time user interface modification (*such as icon shading when selecting a toolbar option in MS Word*). This minimises the amount of information that a user has to remember,

preventing over-taxation of the user's working **memory.** This type of progress indication is also implemented in the WordAssessor interface (see section 5.3.3)

2.4. Guidelines for UI design in software skills assessment

The following sections discuss which concepts need to be focussed on to develop the most usable user interface for a software skills assessment tool.

2.4.1. Developing the user interface

The tool that has been most common in "invading" office environments is the word processor (Preece *et al*, 1994). The widespread requirement for this type of software is that *what you see is what you get*, usually referred to as "Whizzeewig!" - from the acronym WYSIWYG (Dix *et al*, 2004). See Figure 2.1 for an example of WYSIWIG and Non-WYSIWIG editor environments.

Figure 2.1. An example of WYSIWYG and Non-WYSIWYG editor environments.

It is the WYSIWYG type of environment that provides a solid foundation for trying to determine how users process and reshape information. In order to assess a user's skills

in a software environment in the best and most comprehensive way, knowledge of the underlying methods to solve simple software operational tasks will be of value.

2.4.2. Visual design principles

In order to develop a tool that can assess a user's skills in a way that is almost **transparent** to the user (*meaning that the interaction between the **user** and the **tool** must operate in a way that the user is never distracted from the actual software skills assessment process*), the most important aspects of visual interface design must be defined.

Watszman (2002) contends that good design does not need to be noticed as such; the requirement is merely that it (the application in conjunction with the user interface) should work. She states that too much visible functionality in a user interface can confuse users and prevent a quick and easy learning process in its use. Software creators are required to do everything they can to ensure that the user experience is as uncomplicated and practical as possible, and that underlying technologies and processes remain hidden from users (Watszman, 2002).

In assessing software skills, it should be taken into account that a "visually deafening" user interface will waste valuable time on becoming acquainted with the interface, which could have been more effectively spent on performing the tasks for the test.

To establish which design principles are best for the user interface of a skills assessment system, it is helpful to consider Watzman's (2002) argument that there are **three related design concepts** that are paramount to all others:

- She speaks of the **harmony** when all the elements of a certain design complement each other, whilst enhancing the fundamental basis (*of the design*) and concealing the strategies used for achieving that harmony from the user.
- In addition, she mentions **balance,** which is primarily concerned with the visual weight of design components and how comfortable they feel to the user. In the same light, Watszman describes symmetrical (*centred design components, e.g. images and text*) and asymmetrical (*dramatic use of colours*

and interface component positioning so as to stimulate the user in a visual way) design as the two primary methods to achieve this comfort level.

- **Simplicity** is the final important design concept in Watzman's discussion. She refers to an effective and simple interface as one that is "*effortlessly devoid of unnecessary decoration*" (Watzman, 2002, p266).

When considering the above principles in designing the interface of a software skills assessment tool, it is important to realise that the primary aim of the design is to allow the user to work effortlessly "past" or 'through' the interface. To clarify, the terms 'past' and 'through' refer to the fact that the user will be working mostly in a word-processing environment in performing certain tasks. It is only when the user has finished a task and wants to submit an answer (or proceed to another question), that the interaction with the assessment tool interface occurs directly. The terms "harmony", "balance" and "simplicity" also suggest that the user should find the visual design aesthetically pleasing.

The effective use of the design principles above can prevent the interface from affecting **emotions** in a negative way and allow the user to carry out test tasks in an unperturbed manner (T*he design process of the assessment tool is described in chapter 5 and appendix A*).

2.4.3. Usability for guidance

In terms of usability, the groundwork should be laid early on to provide the reader with a clear view of the goals of the study in terms of the end product (WordAssessor). The above-mentioned visual design principles are vital in the sense of co-conditioning the usability of the skills assessment system (see s*ection 1.4 for the definition of usability*). The most significant **aspects** of usability in developing the WordAssessor tool are detailed below.

In order to reveal some of these **aspects**, Galitz (2002) raises certain questions to test the usability of a system by speaking to people who use the interface in question.

"Are people asking a lot of questions or often reaching for the manual?" (Galitz, 2002, p58). Galitz emphasises that if the answer is affirmative, then the system is not optimally usable. As users interact with the new WordAssessor skills assessment system, it is important to note how many users ask questions. In addition, the frequency and volume of these questions should be observed during user interaction. The content of the questions must also be noted to provide a collective view of the potential shortcomings that the system. Also in this regard, Nielsen (2003) asserts that observing which actions users perform can be more effective than simply listening to what they say. In aiming to optimise the WordAssessor tool, questions that users ask during the pilot test of the program, in addition to what actions users perform while these questions are asked, are to be noted (see section 7.2).

"Are frequent exasperation responses heard?" (Galitz, 2002, p59)

Galitz notes that if users are being particularly vocal in a negative way, then the reason behind such anger should be investigated immediately. Galitz (2002) refers to phrases like "Damn it!" or "Come on!" etc. to be particularly noteworthy in the process of observing how users interact with the system. He also advises that some users do not display their emotions in such an open manner, and that silence should not be regarded as acceptance. It might also be helpful to observe body language. (see section 7.2), as this could indicate a negative, irritated, or even exasperated interface experience, e.g. frowning, sweating, pursing lips, etc.

"Are there many things to ignore?" (Galitz, 2002, p59)

When the primary aim is to keep the user's attention on relevant interface elements, Galitz (2002) suggests that we investigate the possibility that some interface components might distract the user from most efficiently attending to the task at hand (see section 7.2 for an example of this in the WordAssessor scenario). Since the proposed software skills assessment tool interface is merely a "managing guide" that controls the software tasks/skills assessed, it must be ensured that users spend most of their time in the environment where their skills are assessed (*MS Word*) and not "fiddling needlessly" with the interface of the assessment tool.

"Do a number of people want to use the product?" (Galitz, 2002, p59)

Galitz argues that people most often want to use a product that makes their lives easier. In addition, he states that a high usability rating may be deduced from the fact that many users would like to use the system. In the case of the WordAssessor environment, a simple questionnaire can be used to ask a few basic questions about the user experience with the system. They can be asked whether they prefer the e-assessment method of the existing test system, or the new method implemented by WordAssessor. Through these means, the success of the testing tool can be determined quickly (see section 7.7 for questionnaire results).

2.4.4. Usability: Comparing interfaces

A study by Tohidi, Buxton, Baecker, and Sellen, (2006) investigated the usability of a specific user interface and compared the results with the usability of three other interfaces that offered the same functional abilities, while being stylistically distinct.

The authors established that users obtained much better usability scores when they were presented with only one design. In addition, the users were less critical of using one design than three different ones in succession (Tohidi *et al.*, 2006). It appears that a variety of design options allows users to identify design flaws and strengths more clearly. Obviously, the design that is intuitively the easiest to work with will automatically allow the user to notice and appreciate its unique strengths. In turn, a less favoured interface design will more readily reveal is flaws than its strong points.

Another valuable finding by Tohidi *et al.* (2006) was that usability testing, either by presenting users with one or many design options, does not yield a useful means of improving interface design as based solely on the recommendations of said users. It is "a means to identify [design] problems, not provide [design] solutions" (Tohidi *et al.*, 2006, p1243).

In terms of analysing the assessment methods and usability problems presented by the existing test system and WordAssessor, it is valuable to note that, according to the findings of Tohidi *et al.* (2006), the two programs should be compared with each other directly in order to identify problems in a targeted and simple way.

This will simplify the process of determining and comparing the suitability of the user interfaces presented by each program in reliably assessing software skills.

2.4.5. HCI: Simplifying the process

Fogg (2002) notes that, if technology can assist users by simplifying any given process, it can prevent or diminish obstacles that affect user behaviour negatively. He refers to online shopping sites, where users choose products, place these into a virtual shopping cart, and then go through a number of steps such as filling in credit card details and entering the shipping address. All of these steps have to be completed on the website by the user before the purchase can be finalised and the products shipped.

For pointing out the ways to overcome the tedious nature of this procedure, Fogg (2002) describes the process that popular online sites such as Amazon.com have employed to minimise repetition. Large online e-commerce sites for example now allow the online storage of user information so that when a returning customer makes a purchase, the amount of clicks needed to complete a transaction is greatly reduced (Fogg, 2002).

An important lesson is that, even an interface that appears very straightforward can still be simplified in some ways. Many programs use techniques such as "Tip-of-the-day" (see Figure 2.2), whereby useful hints about program usage are provided via a dialog box at the start of the programme.

Figure 2.2. An example of the "Tip-of-the-day" hint system.

Another popular method is the "pop-up balloon tooltip" (see Figure 2.3) whereby the user is offered hints by a subtle pop-up balloon in the MS Windows system tray (bottom right corner of the Windows desktop screen).

Figure 2.3. An example of the "Balloon-tip" programme feature.

The above-mentioned methods of simplifying and/or guiding user interaction with software programs can also be applied (see section 5.3.3) to the optimisation of the user interface of the newly developed WordAssessor system (*section 1.3)*.

2.4.6. Cause and effect

Cause and effect (also known as trial-and-error – as mentioned in section 2.2) in this context refers to computer simulations that "allow users to vary the inputs and observe the effects" (Fogg, 2002, p363). Fogg (2002) also refers to the fact that cause-and-effect simulators can effectively and credibly reveal the results of actions without delay. He contends that, if people are allowed to investigate the causes as well as effects of certain situations, then this may positively influence their attitude and behaviour.

This particular cause-and-effect strategy will be employed in the development of the WordAssessor software skills assessment tool. An example to demonstrate why this is important is detailed as follows:

A user is asked to do a certain task in MS Word. Utilising the existing test system tool, the following happens: If a user interacts with a part of the interface not directly related to that particular question/task and its primary answering strategy, the test is paused. The user is then presented with a screen asking her/him to submit the answer or retry. Instead of following the approach of the existing test system, WordAssessor allows users to explore the entire interface, only asking them to submit their answer when they feel comfortable with what they see on the screen. This allows a process of learning even while assessment takes place.

2.4.7. Software credibility

A popular definition of credibility is "a perceived quality made up of multiple dimensions" (Fogg, 2002, p365). Fogg (2002) argues that, if a certain computer product is both trustworthy and requires a high level of expertise to use, then it can be considered highly credible. He states that a change in attitude and persuasion can be brought about by products that exhibit a high amount of credibility.

This aspect of HCI is important when developing software, as users are positive and relaxed, and work most effectively when they trust the application. To establish a solid framework of knowledge about the usability of a software skills assessment tool, its credibility should be assessed. In this regard, Fogg (2002) presents some guidelines: Firstly, he argues that credibility is at stake when computer programs provide users with certain **knowledge or data**.

In addition, he states that credibility is at risk when software **instructs or guides** users in operating the software. A further key point in Fogg's (2002) work is the menu layout or default button configuration of a user interface. If this is poorly designed, the credibility of the product could be hampered. He explains that the reason for this is the subtle guidance that interface components provide to users even without them consciously realizing it (Fogg, 2002). The implication is that the user's experience should be facilitated by the strategic design of the user interface in a way that would make interaction choices more obvious (e.g. default button highlights, logical tab order for interface components etc.).

Another important scenario mentioned by Fogg (2002) is the **computer's reporting ability**. This means that inaccuracies in a **software program's** reports on work completed, could cause the programme to lose credibility. Fogg provides the example of a spell-check program that finds no misspelled words in a given document. If the user performs a manual search of the document and finds a word that has indeed been misspelled, the software application's credibility will definitely decrease (Fogg, 2002).

In the case of this study, the above-mentioned scenario is applied to the assessment of a user's software skills, with the programme reporting the results of the exercise. A valuable means of affirming the credibility of the reported results would be to **explain to the user** why his/her answer is wrong, as well as (*equally important*) how to rectify this mistake in the future (see section 5.3.5). In this way, the user is not confused or wondering if the program has provided a reliable and fair adjudication.

2.4.8. User interruption

The attention of a computer user is a valuable resource, and is easily disturbed by interruptions (Adamczyk & Bailey, 2004). Many applications do not take into account the impact of interruptions on users, and as such they can end up causing what Adamczyk & Bailey (2004) call "interruption overload".

It has been found that when interrupted, users tend to make more task-related errors, are indecisive, have less effective memory, and generally tend to be less effective overall (Gievska & Sibert, 2005). To be more specific, it has been shown that if tasks are interrupted at random intervals, users may need up to 30% more time to properly resume their work (Iqbal & Bailey, 2006). In addition, the same users may make up to twice the normal amount of mistakes in addition to feeling up to twice the amount of negativity when interruptions are not properly timed (Iqbal & Bailey, 2006).

The following is noted from the results of a study conducted by Adamczyk & Bailey (2004): Properly timed interruptions produce "less annoyance, frustration, and time pressure, require less mental effort, and were deemed by the user more respectful of their primary task" (Adamczyk & Bailey, 2004).

The reason for including this aspect of usability is mainly the fact that the existing test system incorporates a poorly timed interruption strategy. As explained in section 2.4.5, when users do not immediately perform the instructed task and follow all the correct steps, they are interrupted by a screen requesting them to submit the answer or retry the question.

According to Adamczyk and Bailey (2004), some studies indicate that the best times for interruptions are either at the beginning or in the middle of a task, or after the task

has been completed. They contend that a user should be interrupted whilst using "few cognitive resources", allowing the temporary period of rest available to attend to the interruption (Adamczyk & Bailey, 2004, p2).

In this regard, Gievska & Sibert (2005) argue that there is a direct relationship between a system's appropriate behaviour and the comfort level of the user. One of the main focus points in the development of WordAssessor has been to allow the user a large degree of freedom. The user has the freedom to explore the word processor's interface and functionality completely during assessment. The interruption strategy here is to let the user submit or retry any given question when they feel ready, thereby eliminating potential frustration due to valuable time being lost as a result of untimely interruptions.

In this research study, another aim of the new WordAssessor tool is to point out the benefit of only interrupting the user at opportune times. To clarify, opportune times refer to periods during which the user is not actively involved in the execution of a certain important task.

2.5. Chapter summary

The term HCI refers to Human-Computer Interaction. In the simplest terms, human beings interact with computers in order to complete work. In utilising new application development technologies, it has become possible to monitor and assess users' software skills in an automated manner within a certain application environment. The role that emotion plays in HCI has been acknowledged. Findings in this regard suggest that human beings who are calm and comfortable perform better and can solve complex problems more quickly than when stress and other irritation factors are involved.

With regard to regulating users' emotions, visual design principles are suggested in this chapter to prevent frustration during a software skills test. In addition, recommendations are made to accommodate the user's memory.

Finally, guidelines for the development of an e-assessment tool are discussed, including usability focus points during the development of the e-assessment tool. In

addition, suggestions that could help to identify potential user interface problems during the pilot test of the e-assessment tool's development are outlined.

In the following chapter, the methodologies used in software skills training are discussed in detail. Key training issues are considered with regard to pedagogical strategies, delivery methods and the different types of settings in training end-users.

Chapter 3

Skills training and assessment methodology

This chapter focuses on the following topics:

- **Software skills training**
- **Users and their learning environment**

3.1. Introduction

According to Marshall (2004), students understand software by searching for solutions to problems. In this study, these problems are mainly word processor based. In all areas of computing, technology is constantly changing, and software skills need to be constantly maintained and updated accordingly. The core capabilities of the numerous iterations of MS Word have remained fairly constant over the past decade. As a result, the main focal points of word processing tasks in terms of document formatting have allowed for easy migration to new iterations of MS Word. For the comparative investigation in chapter 7, MS Word 2002 and MS Word 2003 are used.

Computer software should no longer be seen as merely a bridge between tasks and goals. Its purpose has expanded from necessity to a desire of sorts. It is now commonplace for students to study a course based on their enjoyment of using the software. As an example, Marshall (2004) refers to a recent survey. Apparently, 65% of interviewed students proclaimed software/computers as the reason they chose visual communication as their field of study.

Consequently, to further enhance the enjoyment of a certain software program, developers need to focus on also enhancing the satisfaction derived from the skills assessment process. This can enhance the performance of students being assessed, as emotion plays a vital role in HCI (see section 2.3.1). The previous chapter explains that negative emotion can cause difficulty in problem-solving. In order to minimise assessment difficulties, certain issues regarding software and learning need to be

addressed. Wiedenbeck, Zila and McConnell, (1995) find that a multitude of problems are encountered during the initial phases of learning a new software package. The authors attribute these problems to issues such as massive amounts of training materials; users that are not properly focussed on their real tasks; improper analogies due to a lack of general software experience, etc.

In order to minimise the above-mentioned problems, meaningful learning may be promoted by setting goals that go further than the normal actions in the training guide/manual (Wiedenbeck *et al*., 1995). The importance of properly trained software users is highlighted in the following section.

3.2. Software skills training in general

Well-trained software users are crucial to the survival of any modern organisation. Their expertise with regard to the use of software directly affects productivity levels, in addition to influencing the profitability and economic strength of an organisation. There is always a lingering concern about keeping software skills up to date with the latest technological trends. This usually leads to software training programs within organisations. These same training programs are offered at most tertiary learning institutions. Such a training program is utilised as a basis of software skills assessment in this study.

A good example of why effective end user training is so important is the "I love you" virus mentioned by Mahapatra & Lai (2005), which infected e-mails and consequently millions of computers worldwide. If employees had been properly trained in virus avoidance techniques, then billions of dollars could have been saved. Two examples of these techniques are:
- Only opening e-mails and attachments from trusted sources.
- Maintaining up-to-date firewall and anti-virus software at all times.

Assessment effectiveness is of vital importance in constantly improving the quality of training programs. This is vital in this study as it will provide valuable insight into the optimisation of the assessment portion of the training program.

3.2.1. Evaluating training and assessment effectiveness

In order to test effectiveness, Mahapatra & Lai (2005) developed a comprehensive framework that can be used to evaluate the success of user training. This framework has been fruitfully applied in the past and is particularly suited for "teaching basic skills involved in mainstream business applications" (Mahapatra & Lai, 2005, p70). This framework will be used to evaluate the comparative effectiveness of the assessment strategies investigated (WordAssessor versus the existing test system, chapter 7). In particular, it will be used to evaluate the overall effectiveness of the software skills assessment process. The framework consists of two dimensions: one that points out who is to carry out the evaluation and one to propose what will be evaluated. The different levels of the two dimensions are shown in Figure 3.1..
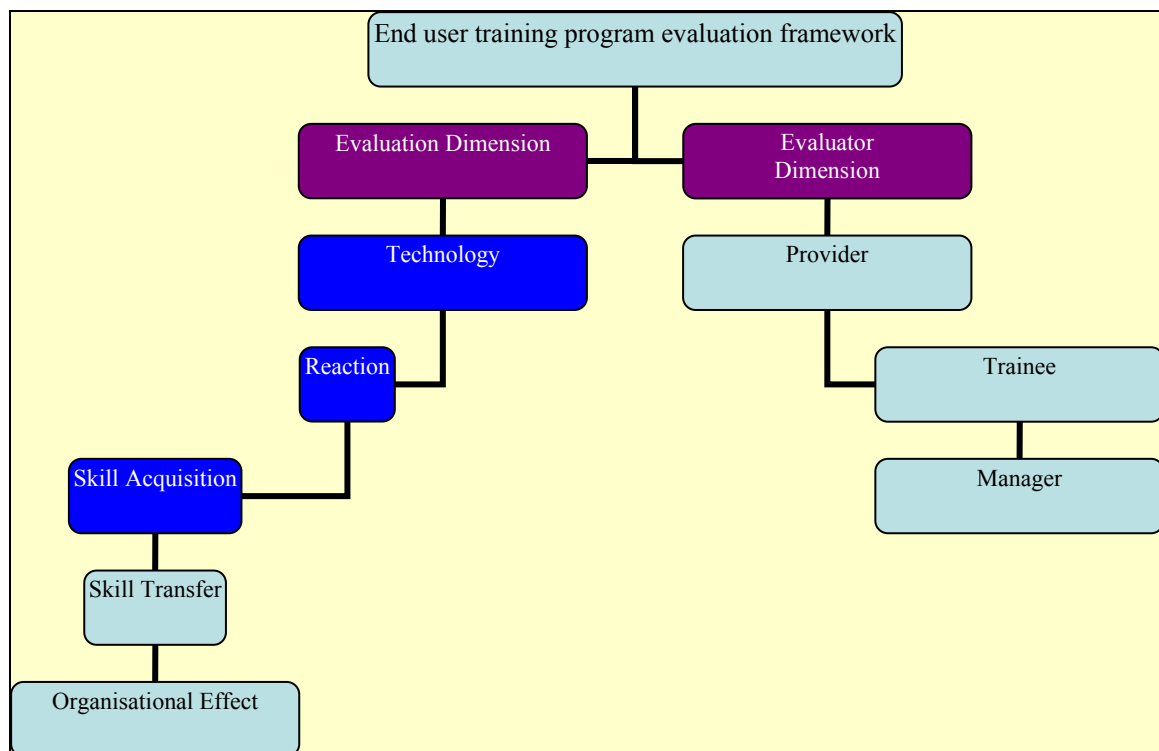
Figure 3.1. End user training program evaluation framework. Adapted from Mahapatra & Lai (2005).

For the purposes of this research study, only the levels highlighted in dark-blue (on the left-hand branch) will be used since the end-users assessed will be students of a tertiary learning institution. Skill transfer and organisational effect can therefore not be measured.

Mahapatra & Lai (2005) describe the first evaluation level to be used in the technology dimension, whereby the **effectiveness of information technology** is determined. This constitutes the evaluation of the IT-related design of the training and assessment process via a questionnaire (see chapter 7). The elements to be evaluated include the software design, its ease of use and quality of presentation, as well as its relevance to tasks related to training. This type of evaluation is vital to ensure the future improvement of information technology tools used for training, as well as for the enhancement of the training programs themselves. For a detailed description of the skills training software involved in this study, see chapter 5 and appendix A.

The second level of the evaluation dimension is reaction. Students must evaluate the quality of the skills assessment program in terms of the following criteria:

- The relevance of the software skills assessed to the tasks to be performed in an employment situation (or future academic use). In the light of the many functions a software program such as MS Word has, it is clear that the majority of users utilise only a small portion of these functions on a daily basis. Therefore, students must decide if an ample amount of ”necessary” skills were assessed.

- Students must evaluate how well the content of the assessment process was developed and presented. An example of this is **clarity**.
    - o Were questions formulated in a clear and unambiguous way?
    - o Was the method of question presentation adequately visible and legible?

- During this phase, students must evaluate the instructor as well as the location of the assessment process. Aspects of this include whether the instructor was readily available to answer questions if necessary.

- Were the students given enough time to complete the test?

Mahapatra & Lai (2005) suggest that questions like those mentioned above should be presented to users/students immediately after the completion of a software skills test. This could help determine any potential shortcomings of the e-assessment process. During the pilot test of WordAssessor, students were presented with an onscreen prompt directly following their e-assessment, whereby they could indicate whether any of the test aspects bothered them (see section 7.2).

The final level of the evaluation dimension is **skill acquisition**. The goal is to determine how effectively software users have been trained. Since this study focuses more on the assessment aspect of software skills training than the training process itself, the acquisition level determines whether actual software skills acquisition has taken place during the assessment phase. In this study, the skill acquisition level is used during the comparative study in chapter 7. Student preference with regard to how much learning (or skill acquisition) they feel has taken place during e-assessment, was established (see section 7.7).

3.2.2. Guidelines for electronic assessment

One of the main benefits of e-learning and e-assessment is that tedious clerical work such as marking, organising tests, etc. can be minimised (Amelung, Piotrowski and Rösner, 2006). This leaves tutors with more time to focus on the actual teaching aspect of their work (Amelung *et al*., 2006). The internet-based software skills assessment system (existing test system described in section 6.3.3.2), as well as the newly proposed intranet-based system (described in chapter 5 and appendix A), WordAssessor, are examples of e-assessment tools. To aid the development of the WordAssessor system, a broader perspective is needed with regard to e-learning and e-assessment strategies. The development of such a complex system needs guidelines and structure in every aspect of design.

The Qualifications and Curriculum Authority (QCA, 2007) provides a comprehensive review of some of the most important e-assessment strategies and regulations employed by modern systems. ***Some of the most important*** regulations are mentioned below together with how they fit into and need to be utilised in this particular study:

- With regard to general knowledge relevance, the Authority contends that the e-assessment structure should only examine facts and knowledge that is most essential to attain the required skills or qualification status. As mentioned in section 3.2.1, MS Word has hundreds of functions. For the purpose of comparing WordAssessor with the existing test system, it is therefore necessary to limit the complexity of assessment questions in order to best represent general word processing capabilities (the core aim of both systems). The details of these assessment questions, as well as how they are implemented to assess general word processing skills, can be seen in appendix A.

- Integrity is crucial in structuring e-assessment questions and criteria. This involves the assessment of student skills in a targeted way, without making longwinded or complex routes to obtain the solutions. The simplest path is always preferred. WordAssessor allows students to use any method to answer test questions, which even includes the use of keyboard shortcuts. This can minimise the steps needed towards the solution. The results of this implementation are seen in chapter 7.

- With regard to security, QCA (2007) mentions that the data involved with e-assessment systems must be secure and comply with current values and trends in the IT industry. In the development of WordAssessor, the Microsoft SQL Server, a greatly popular database management system, is used to store, update, transfer and secure data. Data are secured by means of encrypted passwords, ensuring the integrity of collected data (see section 5.3.1).

- In addition to protecting the integrity of collected test data, an e-assessment system must prevent any interference with the result of the assessment process (QCA, 2007). This includes any and all safeguards against possible plagiarism or copying from other students. Since WordAssessor and the existing test system are localised to single PC's, plagiarism and copying would be difficult to achieve, since tests are conducted in a closely monitored and relatively

isolated environment. See section 6.3.3.3 for details concerning the prevention of cheating.

- To ensure the security of the assessment process further, the authors mention that computers housing e-assessment programs should be protected by up-to-date firewalls and anti-virus software. The computers used for this comparative study are situated on campus at the University of the Free State, and are protected by virtually impenetrable firewalls and McAfee anti-virus software.

- Authentication is crucial in any e-learning or assessment process (QCA, 2007). Students should be granted access only to limited areas of the e-assessment tool. For this study, the following was implemented: The e-assessment program is unlocked for a once only use per PC by either the lecturer or assessment assistant, after which the student is required to enter his/her student number to confirm identity and create a unique test record. Another method (employed by the existing test system) is to request both a student number and pre-assigned password for access to the test. This method is slightly more cumbersome, but is generally accepted. See section 5.3.1 for details concerning the WordAssessor authentication method.

- With regard to data transmission, it is recommended that industry-standard encryption techniques be used while data are transmitted to and from the assessment locale (QCA, 2007). Since the University system is a closed, heavily protected network, the encryption provided by the MS SQL Server 2005 should provide adequately secure data transmission for the purpose of this study.

- Developers of the e-assessment systems must ensure that everything from marking to the presentation of final scores is handled internally and automatically (QCA, 2007). WordAssessor adheres to this principle, as detailed in chapter 5.

- Developers need to ensure that there is sufficient storage for all the data collected during software skills tests (QCA, 2007). Fortunately, the data storage format used with the new WordAssessor system is a simple SQL data table and requires little storage capacity on the server.

- Another vital component in any e-assessment system, as mentioned by the Qualifications and Curriculum Authority, is its ability to provide statistics. These include values such as the gross point average (GPA) for a large group of students, as well as the highest mark, lowest mark, fastest completed test, fastest completion time coupled with total score, the standard deviation in marks etc. WordAssessor uses a simple Microsoft Excel object to display all the marks and the different statistical values mentioned above (see section 5.3.2).

- A comprehensive period of testing must be carried out where any potential problems can be identified and repaired. In 2006, a pilot test of WordAssessor was carried out at the University of the Free State, in which approximately 350 students took a MS Word software skills test (see section 6.2). Students were asked to indicate to facilitators any problems encountered during the test. At the end of each test, students were also provided with an on-screen questionnaire to report any further problems and make suggestions and comments regarding the usability of the system. The response was overwhelmingly positive. Changes were made to the WordAssessor system according to suggestions and comments. Section 7.2 contains a detailed list of the problems encountered and changes made.

- The e-assessment system in question must be given a list of minimum operability specifications and tested on all viable platforms. For a full list of requirements and specifications for WordAssessor, see section 5.2.

- A vital guideline of QCA (2007) is familiarisation. Users wishing to partake in a software skills test must be familiar with the key operations of the program before they start the test. Due to the simplicity of the WordAssessor interface,

students were given verbal instructions prior to their software skills tests, in addition to a reference page detailing the most important functions of the WordAssessor interface (see appendix D).

- QCA (2007) encourages the use of a secure backup system for the protection of all data collected during software skills tests. WordAssessor stores all data on a central server and tables are easily backed onto removal storage as delimited text files (see section 5.2).

- With regard to the structure of questions, QCA (2007) advocates that all test questions should be comparable in terms of relevance and time required to complete them. In other words, if a certain question takes a significant amount of time longer to complete than another, the total score awarded for that longer question should be more than or equal to the score for the shorter one. WordAssessor incorporates questions that are relatively equal in length (in terms of time), and therefore avoids this pitfall. Instead of assessing an MS Word task that requires a multitude of steps, these questions are divided into smaller parts.

- Test facilitators and administrators must be aware of and be able to utilise all e-assessment program functions effectively, including administrative functions that can adjust any errors that might occur during a test session. The author of the WordAssessor system was the primary administrator of the e-assessment sessions, and this was not an issue.

- QCA (2007) also points out that, in order to optimise future tests, data with regard to question difficulty must be collected internally. A constantly updated record is for example necessary to indicate the questions causing the most difficulty among students. This can be used to structure future tests accordingly. From the results of the WordAssessor test, certain questions were identified as much more difficult than others (see sppendix A - section A.2.18).

The following section attempts to detail the various aspects of user learning in order to optimise the comparative assessment process discussed in chapter 6.

3.3. Users and their learning environment

Fincher (1994) describes learning as the process of gradually evolving from a lack of ability to a certain competence.

Gupta & Bostrom (2006) report that more than 70% of companies today require their middle and senior managers to be competent computer users. In addition, the authors note that computer literacy requirements have seen enormous growth in the majority of different end user classes/categories.

To develop the best method of assessing software skills, information on how users interact with computer software is needed. Users complete work-related tasks by accessing a variety of previously learned software functions and procedures.

In order to assess users' knowledge of these functions effectively, it is crucial to gain an understanding of how users obtain such knowledge (the actual learning process). In this way, techniques used in software training can be optimised and applied particularly to the practical study delineated in chapter 6. Only then can users' knowledge of software be assessed in the most comprehensive and fair way.

There are many types of learning methods. Two of the most prominent (mastery learning and discovery learning) are described below.

Davis & Sorrel (1995) describe mastery learning as follows: All students are able to learn with the right conditions. Students have to master every segment of a learning module before they are permitted to move on to the next segment. To take an arbitrary example, someone needs to be able to know how to structure a document in MS Word properly before they can design a presentation in MS PowerPoint.

Discovery learning, as defined by Van Joolingen (1999), is when students experiment on their own within a certain area of study, and consequently construct rules from the experimental results. In this sense, students construct their own knowledge, since they control the conditions of the experiments. An example is when someone learns how to use a new software program on their own by experimenting with functions until the desired result is achieved. Discovery learning takes place during the WordAssessor e-assessment, as described in section 6.3.3.3. During the software skills test, students are permitted to explore the entire MS Word 2003 interface (i.e. to experiment) to complete each test question. They only need to submit an answer to once they see the desired result on-screen. The outcome of this assessment approach will be compared to that of the existing test system approach, whereby students are told which methods to use when answering test questions. The above-mentioned comparison is detailed in section 7.3.

However, in light of the methods that apply best to this study, Gupta and Bostrom (2006) have reviewed current EUR (end-user training) literature and conducted an investigation into some of the areas that need to be focussed on. They mention *social cognitive theory* and *vicarious learning* as prominent and the most successful learning methods. These learning methods are discussed in greater detail in the following section. An explanation is also given to show why these methods of learning are most appropriate with regard to gaining software skills knowledge.

3.3.1. Social cognitive theory

Ryan, Bordoloi and Harrison, (2006) contend that social cognitive theory refers to the notion of social characteristics influencing the behaviour of learners in a certain situation, which in turn affects performance. The authors point out that the primary focus of social cognitive theory research has been on self-efficacy (a belief in one's own ability to complete a task successfully). In general, if a large amount of pressure (social or otherwise) is exerted upon a person, self-doubt can arise. This can lead to performance regression, since the person no longer believes as strongly in his or her ability to complete the task.

The role of social factors in the assessment process is a crucial element in yielding a comprehensive and effective representation of a software user's skills. This assessment process is detailed in chapter 7. Improving the self-efficacy of students is also however required to most successfully manipulate social factors in favour of the learners.

In this light, Gupta & Bostrom (2006) refer to Bandura's (1986) description of major sources of self-efficacy to increase a person's belief in his or her own ability.

Firstly, the authors mention the process of practicing a task until they can effectively complete it on their own. This practice session could be supervised by an instructor, a training manual can be used for guidance, or users could simply attempt a solution unaided until they succeed. The latter was mentioned in chapter 2, as trial-and-error learning. It is now apparent that trial-and-error could improve the self-efficacy of a student, even in an assessment environment. This is further substantiated in a later article by Bandura, in which he argues that, in order to stimulate brain development (i.e. learning), an individual should be "exploring, manipulating and influencing the environment" (Bandura, 2001). This approach is employed by the WordAssessor e-assessment system; students are allowed to "attempt at a solution by themselves until they succeed" (as mentioned above) during a software skills test. It will be determined whether or not attempting to increase students' self-efficacy in this way can lead to a more reliable assessment of word processing skills (see section 7.3).

Secondly, a person can learn a task by observing how **someone else** completes it. Marshall (2004) points out that hands-on time for students are crucial and equates this with the importance of watching a task be performed *before* trying it oneself. In this case, "*someone else*" does not necessarily refer to a person. Many interactive software solutions offer the possibility of video tutorials (animations). The company named **DebugMode** offers a freeware program (free to use) called **Wink** (see Figure 3.2)**,** with which videos can be created (DebugMode, 2007).

It is possible to improve a student's self-efficacy within a software skills assessment environment in the following way: After a user has been assessed, the program shows

the test results. In the case of an incorrect answer, the user is presented with an animation of the correct solution. Marshall (2004) notes that a problem could arise in the case of students who are less proficient than they realise, simply because they do not fully understand what the software does. If the instructor or animation clearly explains what is done *and* why it is necessary, the problem can be avoided. Chapter 7 (section 7.5) details how students respond to this approach, including whether or not they felt that this aided the effectiveness of the e-assessment process.
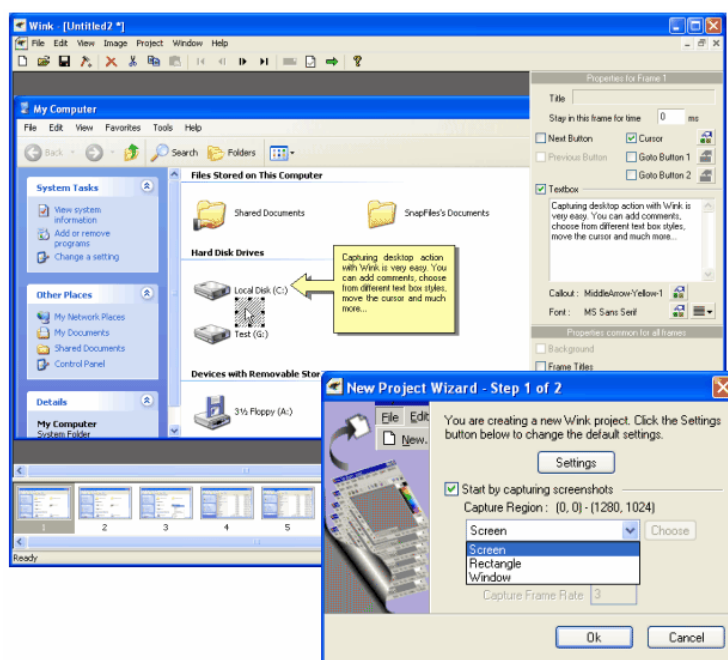
Figure 3.2. Screen capture from the animated tutorial creator, Wink.

The **third** source of self-efficacy, according to Gupta & Bostrom (2006), is motivation by verbal encouragement. Unfortunately, it is a fact that verbal encouragement from a computer is simply not as effective as that given by a person. It is therefore necessary that instructors encourage users as much as possible. In section 2.4.8 it is noted that users should only be interrupted in the beginning, middle or end of a task. Hence, instructors should offer words of encouragement either during these times or when a student asks a question (see section 6.3.3.3 for the implementation of this strategy).

3.3.2. Vicarious learning and behavioural modelling

Vicarious learning is the second type of learning mentioned by Gupta and Bostrom (2006), which is most suitable for learning software skills. Stenning, McKendree, Lee,

Cox, Dineen, Mayes, (1999) explicates this type of learning as students who learn by **observing** others. Basically, students **observe** tasks being performed, and then attempt these themselves. From section 3.3.1, it has already been seen that **watching** others perform a task is a major source of self-efficacy. According to social cognitive theory, this method of training is most suitable for complex tasks (Gupta & Bostrom, 2006; Bandura, 1986).

Vicarious learning adheres to the constructivist learning paradigm. According to Jonassen (1991), many teachers employ constructivism when developing learning environments. Muijs and Reynolds (2005, p62) explain that, in a constructivist learning environment, "learning is always an active process.......the learner needs to be active in order to learn effectively". In addition, Muijs & Reynolds (2005) state that constructivist learning does not merely consist of moving quickly from one topic to another, but also involves exploring and re-examining material in order to construct knowledge. Remembering information is easier when using a unique viewpoint or opinion regarding a certain subject as a basis, as opposed to simply reproducing information like a "parrot". More detail on the process of constructivist teaching/assessment, and how it relates to the goals of this study is discussed in chapter 4.

To further elaborate on vicarious learning, Gupta and Bostrom (2006) reveal that this method has consistently delivered superior training results as opposed to instruction by a lecturer or self-study via a manual. However, since the advent of *video tutorials* in the software training arena, it is clear that the audio-visual aspects of behavioural modelling can now apply to the computerised method of training in addition to mimicking "live" human behaviour. This aspect of vicarious learning is particularly useful for the purpose of this study. In section 5.3.5, the particulars of how video tutorials have been implemented in the WordAssessor system are described. Gupta and Bostrom (2006) mention the ***three most important improvements*** of behavioural modelling (BM) that has enhanced learning outcomes.

- Davis and Yi (2004) state that **Retention Enhancement** refers to the process of symbolic encoding that trainees (learning software skills) use to create concise

summaries of newly observed software skills during training. In MS Word, this type of concise summary includes the steps needed to perform a certain function. For example, if a student wishes to remember how to change the orientation of a document page from portrait to landscape in MS Word, the person would click on the file menu, click on page setup, and then in the dialog window select the landscape option under the orientation heading. A concise summary appears as: "Orientation: file, page setup, landscape".

- In a later article, Davis and Yi (2004) describe another significant improvement to behavioural modelling, known as **symbolic mental rehearsal** (SMR). This involves mentally linking visual images with memory codes (such as summaries) in order to improve memory retention. Following the concise summary example above, SMR is now used to link the four steps in order to change the orientation of a document page. Utilising the video tutorial technique of showing students why they had a certain question incorrect (at the end of a software skills test – as mentioned in section 3.3.1), retention enhancement and SMR can now be applied to the software skills test (refer to section 5.3.5 for implementation).

- **Enactive learning** can be explicated as learning by attempting a task and observing the consequences (Gupta & Bostrom, 2006). It is therefore another example of trial and error (see section 2.2, section 2.4.6, and section 3.3.1). Enactive learning is one of the key areas of this research study. As mentioned in chapter 1, an investigation is conducted (chapter 8) on whether this trial-and-error approach during a software skills test yields a more reliable assessment outcome than the strict and rigid means of assessment employed by programs such as the existing test system.

To further justify the inclusion of a behavioural modelling technique in this study, an experiment conducted by Simon (2000) can be useful. In this experiment, 450 members of the US Navy were studied, in their use of three different training methods, which included behavioural modelling. Supported by the results of his study, Simon contends that, in training scenarios where students employ many different learning styles, the use of behavioural modelling should consistently produce

the best training results. Some of the most prominent learning styles include: Learning by listening, learning by visual stimulation, such as written tests and pictures, learning from practical activities such as software usage, etc. (NDT, 2007). Furthermore, Simon states that the use of behavioural modelling will help to increase end user/learner satisfaction in addition to helping learners who struggle with the exploration-based learning approach.

Another example is given by Gupta (2006), with a BM experiment during which students are presented with an interactive simulated environment to stimulate enactive learning. The author found that a higher level of knowledge acquisition takes place when students are exposed to a simulation that **reacts to evaluate their behaviour**. In this way, students know immediately when they are on the wrong track in attempting a task. This motivates the assessment of students in the real MS Word Environment (as described in section 1.2.2), in its capacity to enhance the learning process (see section 7.3 for the results of this investigation). It is important to note that interactive simulations appear to function best in enhancing knowledge acquisition when "live" feedback is provided. This is particularly the case during enactive learning.

During software skills assessment, it is generally accepted that learning is no longer necessary, since previously acquired knowledge is being assessed. However, to demonstrate another use of BM, the use of interactive simulations, as mentioned above, can allow learning to continue, even during assessment. The basic premise is to allow a student access to an interactive solution to the test problems directly after completing the test. This enhances knowledge acquisition and clarifies any doubts or questions that students might have. Feng, Heffernan and Koedinger, (2006) employ this technique with very good results. They find that such a system increases the efficiency of the actual assessment process if learning is also involved during this phase. This method will be used in the WordAssessor application as described in section 5.3.5.

3.4. Chapter summary

As a result of changing technology, the software skills of users everywhere need to be kept up to date in order to perform in the business market. To assess a user's software skills successfully, an investigation is made of how this knowledge is accumulated.

With respect to learning methods, two of the most prominent and successful learning methods include **social cognitive theory** and **vicarious learning**. **Social Cognitive theory** refers to how social pressures of a certain environment can affect the performance of a user in a negative or positive way. It appears that social cognitive theory is most suited to enhancing student confidence. Vicarious learning is based upon this theory, as learning by observing is one of the primary ways of increasing a person's self-efficacy (see section 3.3.2).

In this study, the above-mentioned learning theories will be applied during the practical study to aid formative e-assessment. The results of investigations on how performance and other test-related factors are affected are detailed in chapter 8.

The most important guidelines for the creation of an e-assessment system are mentioned and in chapter 5, it is shown how these guidelines are used in the development of WordAssessor. The effect that the said guidelines have on the e-assessment process is detailed in chapter 7.

In the following chapter, the reasoning behind an optimal e-assessment paradigm is discussed, together with how constructivist teaching connects with software skills learning. Finally, the different types of feedback in e-assessment are detailed.

Chapter 4

The assessment process

This chapter focuses on the following aspects:

- **What assessment entails**
- **Paradigm shift in assessment**
- **Learning during assessment**
- **Assessment planning**
- **Giving students feedback**

4.1. Introduction

Companies throughout the world are constantly training their personnel to handle the implementation of new company procedures or tools. A crucial part of this process is assessment. Many companies use electronic assessment to automatically test the learned skills of their employees. To obtain a more comprehensive view of the importance of sound assessment practices in the international arena, consider the following:

> "An Italian job applicant is assessed at a test centre in France using an English language test. The test was developed in Australia by an international test developer and publisher, but is running from an ISP location in Germany. The testing is being carried out for a Dutch-based subsidiary of a multi-national corporation. The position the person is applying for is as a manager in the Dutch company's Tokyo office. The report on the test result, which are held on the multi-national corporation's intranet server in the US, are sent to the applicant's potential line-manager in Japan having first been interpreted by the company's out-sourced HR consultants in Belgium" (Bartam, 2006, p121).

The assessment process consists of many different components, as demonstrated by the above quote, and each component has its own crucial role. In the following

sections, each component of assessment and its function in higher education is discussed to determine their connection to the process of software skills assessment.

4.2. What assessment entails

The following two sections provide a definition for assessment as well as which aspects of assessment, as related to this study, are required by the University of the Free State.

4.2.1. Definition of assessment

Brown, Bull and Pendlebury, (1997) define assessment as follows: Take a sample of what students are involved with (these include tests, assignments, etc.); analyse results and draw conclusions regarding the value of the students' achievements. The conclusions can be used to determine student potential, propensity towards certain careers and other information, such as student intelligence or attitude. Assessment is vital, as it provides learners with proof that the accumulation of their knowledge is moving in a positive direction (Suleman, 2003). The comparative study in chapter 7 investigates different assessment methods on the campus of the University of the Free State, and hence assessment needs to be structured according the institution's regulations.

4.2.2. Assessment policy of the University of the Free State

To ensure that all important criteria for an effective assessment process is carried out according to the university's regulations, it is important to examine the requirements of the UFS assessment policy (see appendix B for complete assessment policy). The most significant aspects of the policy, as they apply to this study, are as follows:

- "***The purpose of assessment must always be communicated clearly.***" From the students' viewpoint, the purpose of assessment is to determine their proficiency level in the MS Word environment. With regard to the research aspect, the purpose of assessment in this study is to determine which methods of assessing students yield the most reliable representation of their word processing skills knowledge.

- *"The design of assessment to promote student learning is emphasised."* In addition, *"The feedback regarding assessment results must be clear, accurate, timely and meaningful."* As will be seen in chapter 5, the design of the WordAssessor automated assessment tool is based on student exploration (trial-and-error), as well as meaningful and comprehensive automated feedback.

- *"Assessment must be impartial."* The automated assessment systems used in this comparative study are machine-controlled, and thus offer neutral assessment for the students involved.

The following section clarifies the motivation of the research, and explains the purpose of investigating the optimum e-assessment paradigm for assessing users' word processing skills.

4.3. Paradigm shift in assessment

From the previous section, it is clear that assessment is involved in the process of enhancing the skills of the learner. The work of Guba (Ed. 1990) supports the belief that there has been a shift in focus from striving to obtain so-called pure objective results (summative assessment – section 4.4.1.3), to an acknowledgement and allowance of the subjectivity of human involvement and interpretation (formative assessment – section 4.4.1.1). This is true even in the field of Natural Science, and more specifically Information Technology.

Computer skills assessment has begun to incorporate the intricacies of student involvement and an awareness that nobody enters any scientific environment neutrally or objectively. The likelihood of "pure" scientific outcomes or of subscribing to one ultimate scientific method is extremely small. Students' background knowledge on what is to be assessed, their values, as well as their emotions always play a role, and might influence the results of a test, as discussed in section 2.3.1.

Murphy (1997) mentions Jonassen's (1991) contention that constructivist teaching is based on using realistic approaches to solve problems in real-world scenarios. She

also points out that constructivism allows an awareness of multiple perspectives on a certain topic (Murphy, 1997).

These aspects translate directly to the proposed 'realistic' software skills assessment environment (WordAssessor) in this study. Allowing students to use any method to accomplish a word processing task gives them the opportunity to observe problems from different perspectives. As Murphy (1997) points out, the focus is to be on the construction of knowledge rather than its reproduction.

The constructivist teaching paradigm suits the fast-moving world of ever-expanding bodies of information, with technological tools that enhance personal creativity (Guba, Ed. 1990). It also corresponds with the freedom offered by the available environments and their opportunities for communication. Seeing that such an approach puts the student at the centre rather than focusing on the accumulation of so-called hard, objective results, it results in freedom of choice and provides an opportunity for the exploration of new technological tools – also in the field of software skills assessment.

An example of constructivist teaching is a lecturer demonstrating to learners how to carry out a certain complex task while explaining the thinking behind each key step (Muijs & Reynolds, 2005). Another example that Muijs & Reynolds (2005) provide is motivating learners; after analysing their performance, the lecturer provides feedback and helps students determine how they went about solving certain problems.

With regard to software skills assessment, students should be free to use assessment tools resourcefully and explore the possibilities of a program independently. The introduction of this "exploratory" paradigm in WordAssessor determines its effect on students. This means that, from the outset of creating a computer skills assessment program, the student should be taken into account as a many-facetted person in a process of searching for and constructing knowledge.

In support of the above, Jonassen (2004, p1) contends that the new approach is already implemented: "These computer-based tools have tapped into students'

multiple intelligences, and enabled those with aptitude in visual learning, for example, to demonstrate knowledge creation more effectively." It also means that lecturers are transforming the learning environment so that students can become active producers rather than passive learners.

In an ITForum Paper, Jonassen (2001) points out that technologies of instruction have always focused on instructing, with students obliged to follow. The student's input in the interaction consisted of operating the tool as instructed and then waiting for a response in the form of an answer or judgment. Computers were used as mediators of instruction. As a result, the outcomes of learning were easier to control and predict. Jonassen objects to this approach, because it not only constrains creativity and learning, but also negates the complexity of being human. It leaves no room for students to construct and design their own world of knowledge. Learners should be able to use "technology as tools for analyzing the world, accessing information, interpreting and organizing their personal knowledge, and representing what they know to others." (Jonassen, 2001, p1).

Hand-in-hand with becoming more accustomed to the challenges of an open view on the possibilities of Human-Computer Interaction and assessment, is the realisation that an interdisciplinary culture should be encouraged. A report on a meeting on International Training and Support of Investigators in the Natural Sciences in Strasbourg (Human Frontier Science Programme, 2001), suggests that academic training programs should include diverse disciplines and various tertiary institutions, so that training and experience could be shared. The report pointed out that the NSF IGERT Programme and the Max Planck International Research Schools have already fruitfully worked to enhance cultivating a new interdisciplinary culture. Programs like these "... should stimulate a dialogue that, in due time, will lead to the restructuring of science education in different programs." (Human Frontier Science Program, 2001, p10).

In view of the fact that this study also incorporates an interdisciplinary aspect – dealing with assessment and focusing on human-computer interaction – one might mention that outcomes-based education in a sense also takes the alternative approach.

In both these approaches, the student is placed at centre; the lecturer becomes the facilitator; assessment is a learning experience and aids in future planning; individual differences of students are taken into account; students are encouraged to make their own decisions, solve problems on their own, and interpret knowledge, etc. (Bundaberg Curriculum Exchange, 2007).

In the case of this study, the bridge between assessment theory (i.e. the method of most reliably determining students' true knowledge) and its practical application in the field of information technology needs to be constructed effectively. In the following sections, learning during assessment, as well as a description of the various types of assessment, are discussed to more clearly define how these components will be integrated into the WordAssessor system (see chapter 5).

4.4. Learning during assessment

According to DiGesu (2007), assessment refers to a personal scenario where a person is beside someone and watches or studies the person's actions. She refers to the most important principles upon which assessment is based. The primary principle (and the one that will be emphasised in the assessment process of this study), is learning improvement; how assessment can aid the learning process. MacDonald (2002) supports this in stating that assessment is vital to ensure that students are given a chance to learn at significant points in the course.

When students are subject to situations where they need to solve problems, they need to retrieve accumulated knowledge. This in turn stimulates memory repetition, which is generally known to be an effective learning catalyst. Even though the amount of "actual" learning in this case might be minimal – due to the fact that the mind is only accessing information that already exists – this kind of stimulation further 'cements' knowledge into the longer-term memory. In this way, students learn while being assessed. In a software environment, students constantly see visual cues such as familiar button labels or menu items that can trigger certain memories and in turn, stimulate problem solving. This demonstrates that learning during assessment is present during this software skills assessment scenario (see formative assessment – section 4.4.1.1).

DiGesu (2007) refers to another important principle and requirement of assessment, namely that assessment has to be reasonable and impartial, and that communication about the assessment process has to be continuous. This type of communication should include how comprehensively course knowledge is assessed. If, for example, a student has to study ten chapters for a test and only one or two of these are tested, this could lead to confusion regarding information priority (as discussed above). In order to determine the progress of students successfully, the assessment system used should be refined regularly, and the student should receive well-timed feedback (Digesu, 2007). In section 4.6, feedback on demonstrated skills is discussed, as well as how it connects with the goals of this study. Before these aspects of feedback can be discussed, however, the various types of assessment need to be outlined.

4.4.1. Types of assessment and the implementation

There are various methods of assessing students: Summative assessment, formative assessment, as well as diagnostic, traditional and authentic assessment. Even though formative assessment is most widely used in e-assessment (Suleman, 2003), both summative and formative e-assessment are investigated to determine the methods with the most reliable representation of a user's true software skills knowledge. It should be noted that all the types of assessment discussed below have an impact on e-assessment.

4.4.1.1. Formative assessment

Formative assessment refers to the "frequent interactive assessments of student progress and understanding" (Holt, 2005, p1). According to Harlen (2007), this particular type of assessment is valuable, as it aids the learning process itself (by means of feedback from lecturers). She contends that if the effectiveness of formative assessment can be augmented, the level of student success can notably increase.

As one of the goals of this study is to facilitate learning during the assessment process itself, some formative guidelines are required for the successful implementation of this type of assessment. Holt (2005) outlines the most important strategies involved in optimising formative assessment.

She asserts that feedback with regard to the improvement of student work needs to be provided effectively. MacDonald (2002) noticed that students involved in learning to use a software package, incorrectly assigned priorities regarding program functions. This was directly related to the fact that students did not receive any feedback while attempting to perform tasks that might be foreign to them. Since practically all aspects of assessment in this study are automated, it appears logical that the incorporation of a feedback system will need to follow a similar path. The popular video-tutorial technique is a prime candidate, with its use in this regard discussed in chapter 5.

Holt (2005) stresses that students need to be made aware of learning goals and the conditions of assessment. In this way, post-assessment inquiries are minimised and students have a more thorough comprehension of what is expected. A basic example in a software skills test is to display the total weighting (score) of each question on the screen, during the test. In addition, the amount of time remaining for completing the test should be constantly displayed (see section 5.3.3). Simple elements such as these can keep the student informed on his or her progress and prevent excessive periods of nonproductivity. The above-mentioned features are incorporated in this study, and the results of the assessment process discussed in chapter 7.

The sections below describe two further types of assessment that are also vital in comprehensive student assessment.

4.4.1.2. Diagnostic assessment

Diagnostic assessment is used to estimate the potential of students, and can also be useful to measure how much help they need during a certain period of a course module (Suleman, 2003). This can ensure that they keep pace with regard to course requirements. Diagnostic assessment can also be used to aid the process of formative assessment. An example of diagnostic e-assessment is a software skills test that does not merely yield a final mark for a test, but also provides information regarding areas where errors occurred and areas that need focus on in order to improve.

4.4.1.3. Summative assessment

Summative assessment is the process of reporting an overview of accomplishments or results of learning to others (Harlen, 2007). This style of assessment is used to assign

weighted score totals for single or multiple courses. These scores can then determine if students are awarded their degree or granted access to later or more advanced courses. An example of this type of assessment is an examination with time restrictions, administered at the end of an academic year (Suleman, 2003). Summative assessment is the most common type of student assessment at academic institutions. It gives a broad overview of what has been accomplished. For this study, the summative aspect of assessment will play an important role in analysing the effectiveness of the different assessment methods described in chapter 7.

4.4.1.4. Traditional assessment vs. authentic assessment

Most new forms of software skills assessment require learners to perform tasks within a simulation of a software program or within the actual programme environment. To this end, Mueller's (2007) description of authentic assessment as a type of real-world testing can be applied. To clarify, this entails students performing tasks that simulate scenarios they would face in the real world.

Mueller (2007) compares this form of assessment to traditional assessment (whereby knowledge is tested in a more theoretical manner). For example, whereas the traditional method of assessment would ask the student to select a certain answer from a list of preset answers, authentic assessment would require students to demonstrate their knowledge in a practical manner (word processor example: instead of describing how to format a paragraph of text, the student has to demonstrate his or her formatting skills practically within the actual software environment).

Mueller (2007) also points out that, with traditional assessment, the expected answer is structured by the teacher, whereas responses in authentic assessment can be manipulated to a certain extent– thereby allowing for creativity. The core idea is that students must be willing to do more than memorise and recall relevant information (Pearson Education Inc., 2007); they should be able to apply their knowledge to solve practical problems. The method of authentic assessment as discussed above is the basis of assessment with the WordAssessor system (see chapter 5). Before the comparative study of different assessment methods can be conducted (chapter 7), the persons involved in the assessment process should be determined.

4.5. Assessment planning

According to Assessment Strategies, Inc. (2007), the following factors are central when selecting methods of assessment:

Firstly, the assessor should designate what is to be assessed: A decision is made regarding the aspects of ability to focus on in the assessment process. Assessment Strategies Inc. (2007) also state that the combined use of numerous assessment methods can sometimes yield more reliable results. Generally the most commonly included aspects of assessment in a software-skills environment include the analytical and problem solving ability, as software knowledge is used and applied in a time-constrained, practical environment. The pen-and-paper method of assessing word processing skills would therefore be unpractical.

Secondly, the assessor should choose the target assessment group (Assessment Strategies, Inc., 2007) and decide when and how they will be assessed. A summary detailing the skills and attributes of students, as well as the goals that students have achieved in the past, are valuable in the preparation of the assessment process. If skills in MS Word are to be assessed, the test questions need to be tailored in accordance with the skill level of the students to be assessed. This is to ensure that test questions are in line with actual knowledge and objectives.

Another factor that that can affect the assessment process is student experience (Assessment Strategies, Inc., 2007). Have students previously been assessed in the proposed manner? If this is not the case, provision should be made to acquaint students with the assessment protocol. With regard to e-assessment specifically, see section 3.2.2.

4.6. Giving students feedback

As stated above, the main purpose of this study is to determine the optimum e-assessment methods for assessing word-processing skills. This point is once again emphasised in order to link the main goal with a matter discussed in section 4.4.1.1: The notion that feedback is an essential part of any assessment system.

Wiggins (1998) contends that, learning cannot take place without feedback. Performance in any area of life can serve as a demonstrative example of this. If people want to improve performance, they need to know which performance-related aspects should be the focus of such improvement. These aspects can only be discovered by feedback, either from the physical responses of objects around us, or from human beings (Wiggins, 1998). For this study, the effect of an e-assessment system that gives comprehensive feedback directly after the assessment process is investigated.

To clarify another aim of this study, an example in HCI is useful. When users operate a certain software program, feedback is an integral part of the user interface (UI). A user can for example hover the mouse over a certain button to reveal its purpose, or click on a drop-down menu for its contents. Without this type of immediate feedback from the UI, users might not learn how to improve or optimise their performance. Without UI feedback such as pop-up tooltips, a user cannot quickly find the purpose of certain buttons. As mentioned, the WordAssessor system developed for this study is based on the real MS Word environment. It therefore serves as an example of a software environment that offers immediate UI feedback to the user. The existing test system (ETS) also used for student assessment in this study, is based on a simulated MS Word environment that does not include UI feedback. Student performance in both the above-mentioned systems are compared to determine which type of interface yields a better and more accurate indication software skills knowledge.

To further elaborate on the use of feedback, Skegg and Jones (2007) assert that, in order to improve the quality and effectiveness of assessing students' knowledge, they should not only receive feedback regarding how well they performed, but also which areas of their work can be improved. In the following pages, methods of giving students this type of feedback are investigated.

4.6.1. Types of feedback

Wiggins (1998) explains the difference between evaluation and feedback. He states that feedback should explain what the learner did or did not do correctly in a completely neutral manner. Evaluation, on the other hand, assigns praise or blame to a learner for following a certain course of action. Gerber (2008) clarifies Wiggins's

definition by stating that feedback is an indication of the difference between what was expected and what the student delivered during the assessment process. The type of feedback that is most suitable for the purpose of this study is investigated below.

Draper (2002) provides a comprehensive outline of the types of feedback currently used in education. This outline is used to determine of the best-suited feedback as applied to a software skills assessment environment.

To demonstrate how the different types of feedback apply to this study, the following example is useful: ***A learner is instructed to change the width of the left margin in a MS Word document to 1 cm.***

The first type of feedback that Draper (2002) mentions, is describing success or failure by means of a point score. This type of feedback instantly provides learners with their performance level in terms of what was expected. In terms of understanding errors, this type of feedback has no use; it is a form of summative assessment. It is used in most assessment environments, and is also included in the comparative study in chapter 7. For the above-mentioned example, a learner receives a mark out of a possible total of four (this is the amount of steps required for a correct answer).

The second type of feedback describes what is required for the correct answer. For the example above, this entails a step-by-step description of how to change the left margin width for a MS Word document. The solution would be:

- Click on the "File" menu
- Select "Page Setup".
- Change the left-margin width value to 1cm
- Click on OK.

The third type of feedback compares the learner's solution with the expected solution (similarities and differences). Since the left margin can only be changed successfully by completing all four steps mentioned above, this form of feedback entails indicating the steps that led to an incorrect response.

With regard to software usage, Draper (2002) mentions that an error message (refer to Figure 4.1)alone is not sufficient to inform a user of how and why they went wrong. A learner needs a demonstration of how to correct the error.
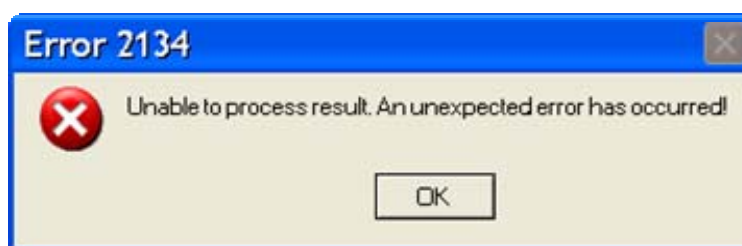
Figure 4.1. An example of an uninformative error message that MS Windows XP users are often faced with.

The second type of feedback, as explained above, is aptly suited for the video tutorial technique addressed earlier (section 3.3.1 and 3.3.2). However, in order to compensate for the ways in which a user can go wrong in a software environment, learners need to know why each crucial step in a certain chain of actions is necessary. Otherwise, when they are provided with the correct solution, they might not be able to remember each step effectively. This method of informing students of the importance of each step represents an important part of the learning process during formative assessment. This strategy is implemented in the study (see section 5.3.5) and the results discussed in chapter 7.

This leads to Draper's (2002) fourth type of learner feedback – explaining to the learner why the correct answer is indeed correct. While using the video tutorial technique, pop-up balloons or accompanying audio comments need to be used in the video to highlight the most important steps for the correct answer. This can also increase learner retention and the effectiveness of learning during assessment.

The images below (Figure 4.2) demonstrate how this technique can be used effectively.
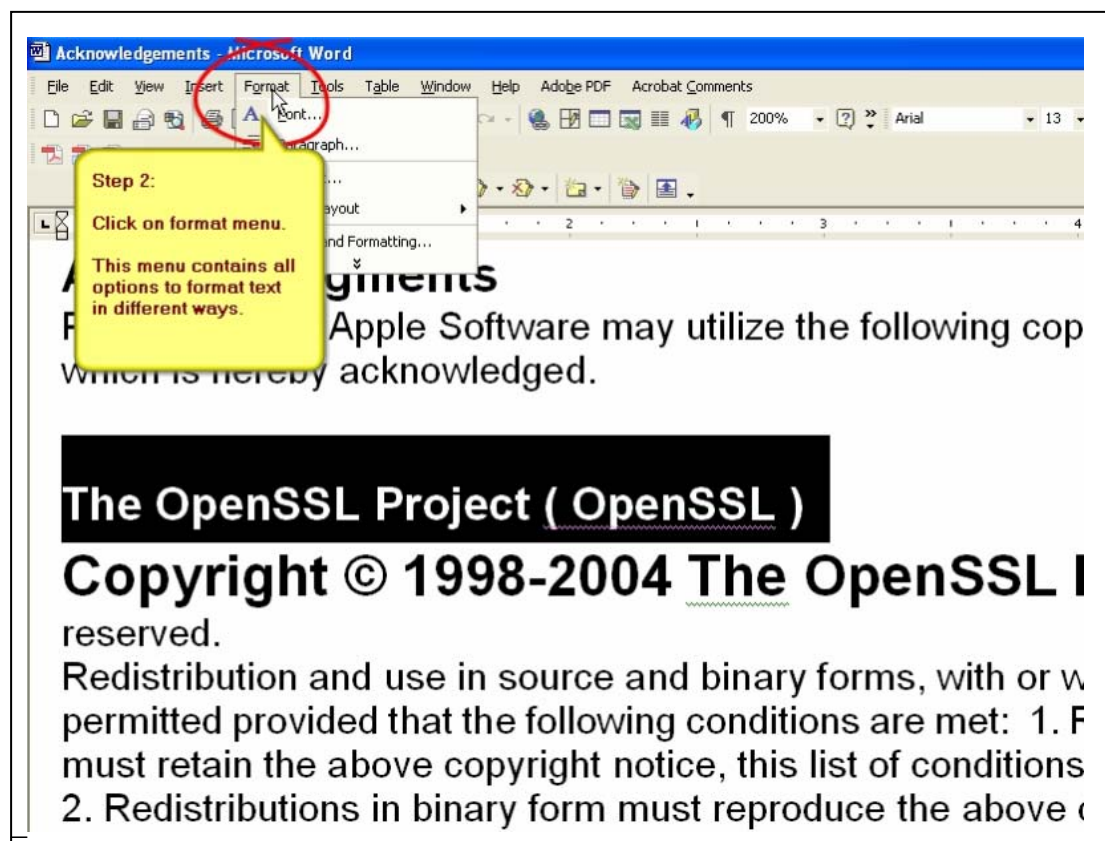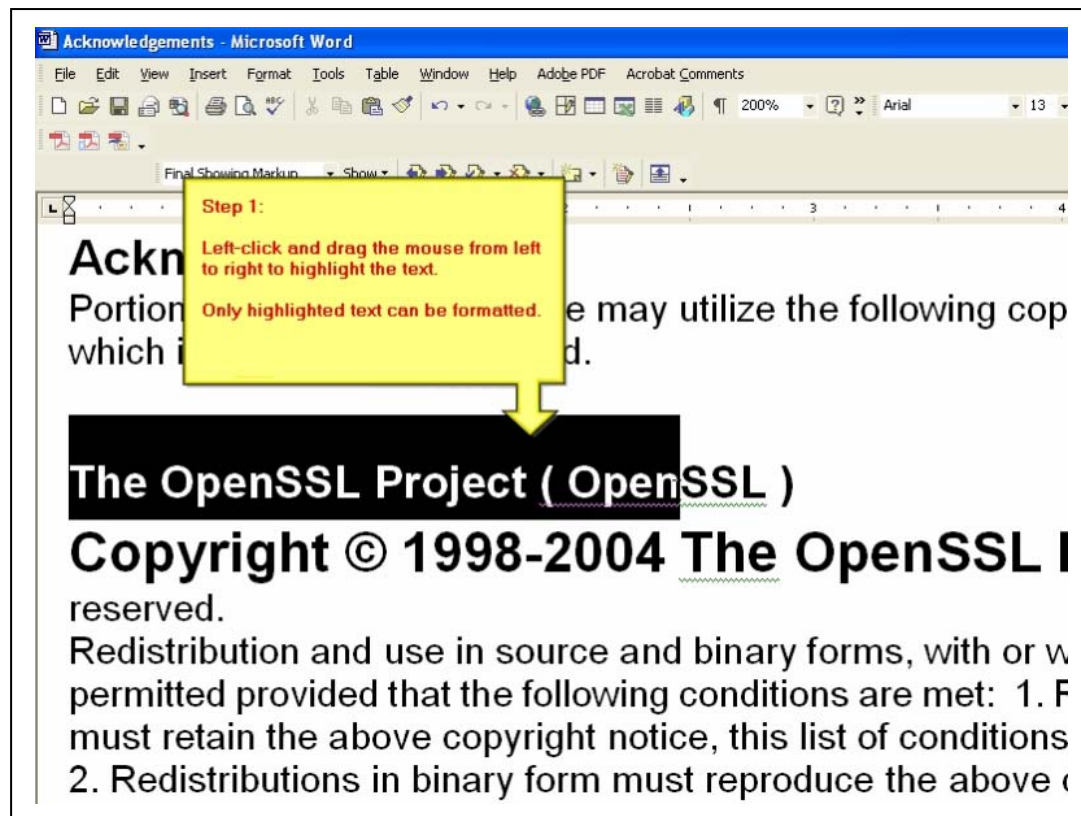
Figure 4.2. An example of balloon tips in an automated software skills test feedback system.

In terms of the images in Figure 4.2, it is vital to note that one instruction and one explanation of why the required step is necessary are provided in each balloon. This is in line with Draper's (2002) fourth type of learner feedback.

The implementation of the above-mentioned feedback methods for this study are based on the following criteria for quality feedback:

- Skegg and Jones (2007) assert that it is vital to realise that feedback will be most effective if students receive it as soon as possible after being assessed.
- MacFarlane and McKellar. (2007) recommend that feedback to students should be as detailed as possible.
- Bright (2007) notes that in such feedback, key points should be focused upon.
- Feedback should be honest and motivating, and can be provided before grades are allocated (Geyser, 2004).

In the implementation of the WordAssessor system described in chapter 5, feedback is automatically provided to students immediately after their software skills test, and before they see their final marks. The practical guidelines mentioned above are implemented in the WordAssessor software skills test and the results discussed in chapter 7.

4.7. Chapter summary

When attempting to determine how a learning module achieves its goals effectively, the various educational aspects involved should be considered. In this light, it is necessary to assess student knowledge, skills, attitude and values in combination with their reactions to instruction. It is only through this comprehensive understanding that improvements on learning and the assessment process in general are possible.

The educational goals of this study are described as follows:

1. Curriculum objectives

The objective of teaching in this study is to successfully implement comprehensive automated feedback for an e-assessment system, according to the standards described in section 4.6.

2. Assessment tasks

The assessment of word-processing skills is implemented by means of the WordAssessor and ETS systems. The assessment methods employed by both systems are compared and their effectiveness with regard to reliable software skills assessment determined (chapter 7 & 8). The WordAssessor system utilises assessment techniques that are authentic, formative, and summative in nature (see section 4.4).

3. Learning outcomes and 4. Learning activities

With regard to learning, the objective in this software skills assessment scenario is to implement a method of post-assessment feedback that provides students with the complete solutions to the questions they answered incorrectly. The solutions contain information regarding the importance of each step (see section 4.6.1). The broad purpose is to ensure that students understand where they went wrong, and are confident that they can correctly answer all the questions, should they redo the test in the future. In addition, the form of authentic assessment (the real MS Word environment) aims to determine whether trial-and-error (see section 4.2.2) plays a role in reliably determining true software skills knowledge.

All the above-mentioned facets of assessment play a vital role in determining the validity of the null-hypotheses described in chapter 1.

In the following chapter, the structure and functionality of the WordAssessor e-assessment tool is described in detail.

Chapter 5

The Research tool

This chapter focuses on the following aspect:

- **The WordAssessor components and technical description**

5.1. Introduction

In order to successfully compare the proposed assessment methods of this study with those currently in use at the UFS, it was necessary to develop a system employing such assessment methods.

As stated in section 1.2.2, the idea is to develop an assessment tool that automatically tests a user's skills within a real software environment. For this purpose, Microsoft Visual Studio Tools for Office were used to develop the WordAssessor system. The research was conducted over a two-year time period, and MS Word 2007 and Windows Vista were therefore not available during the developmental phase of WordAssessor. As a result, WordAssessor works within the MS Word 2003 and Windows XP environment.

It is important to note that WordAssessor was developed primarily to test the validity of the proposed assessment paradigms (see section 7.3 for results) and not necessarily as a commercially available system.

The main software skills test used in this study consisted of 25 questions asking students to perform tasks within the MS Word environment (see chapter 6 for details on student selection, preparation and execution of the software skills test). The 25 questions had to be programmed separately, as MS Word contains hundreds of different functions and programmable objects.

After a user has completed a certain task in accordance with the instructions of a test question, WordAssessor has to monitor the changes with the current MS Word document window. It has to check the conditions of the required end result of each question and compare it to the current condition of the objects of the MS Word environment. If the end result matches the requirements of the question, then the student is awarded one mark.

The basic structure of the system is explained below. The description of how WordAssessor programmatically handles the 25 questions is very extensive and thus not included in this chapter. See appendix A for such a description.

5.2. Technical description

WordAssessor is installed on each end-user PC. Each PC runs Windows XP and has MS Word 2003 installed. WordAssessor uses Microsoft primary interop assemblies to monitor what events take place within the MS Word software environment. Based on this, WordAssessor is able to determine if users (students) have correctly carried out tasks during a software skills test.

Following each question, WordAssessor saves the test data by updating data tables, located in an SQL server database. The results of the test are saved to these data tables during and directly following the test. The data tables are easily backed up as text delimited files (see section 3.2.2).

The basic functionality of the WordAssessor system is designed to work on computers that are compatible with Microsoft Office 2003. The minimum requirements for Office 2003 are as follows: Processor Pentium 233 MHz or higher processor; Pentium III recommended. Operating system: Microsoft Windows 2000 Service Pack 3 or later, or Windows XP or later (recommended). Memory: 128 MB or higher RAM (recommended). Monitor: Super VGA (800 x 600) or higher resolution with 256 colours.

The diagram in Figure 5.1 gives an overview of how the assessment tool works.

All WordAssessor data
sent to MS SQL Server
2005 database

All computers linked to campus network

Each PC contains a copy of WordAssessor linked to MS Word 2003

Users at PCs running Windows XP and MS Word 2003
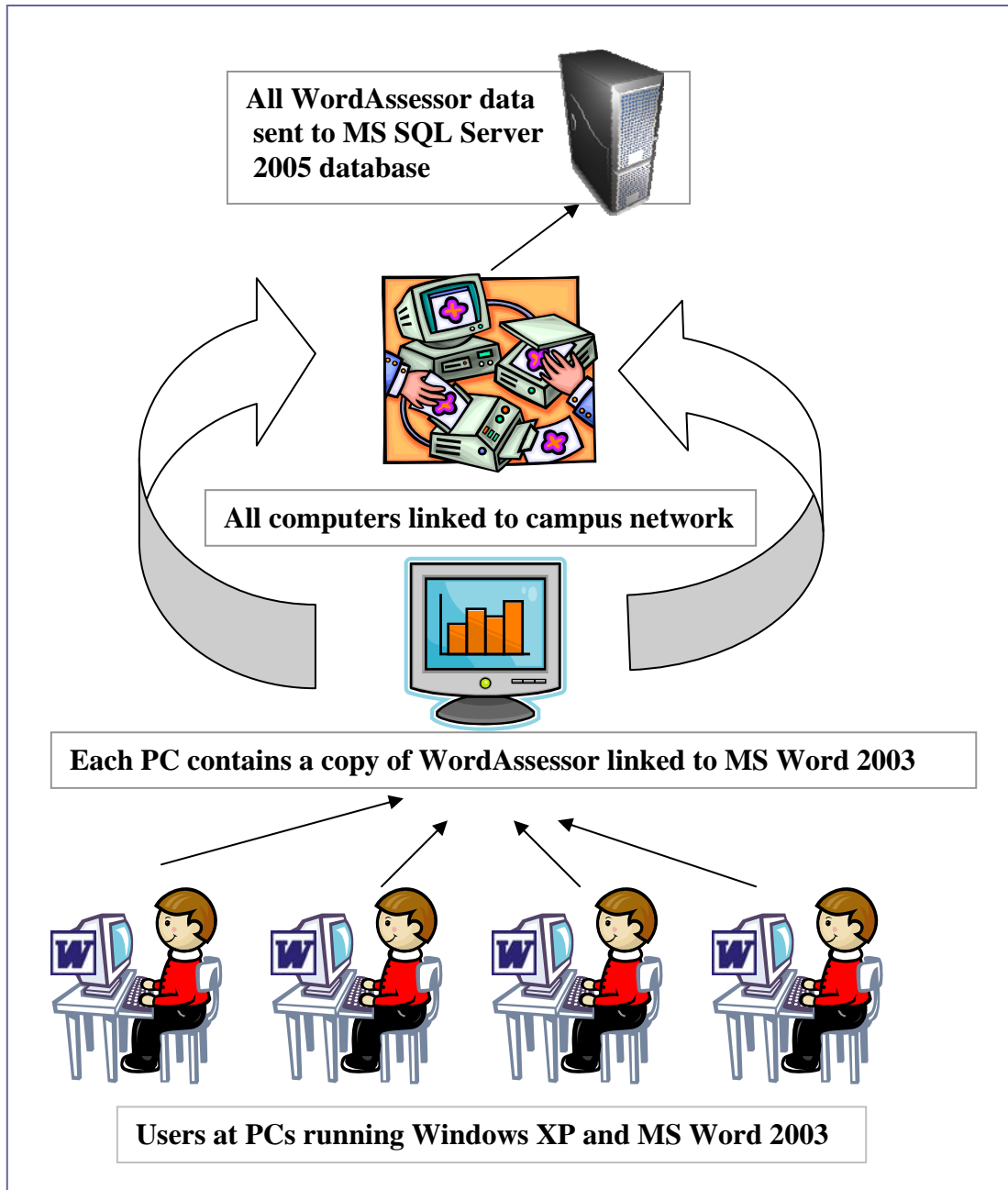
Figure 5.1. WordAssessor overview

5.3. The WordAssessor components

The WordAssessor user interface has five main components, namely:

1. The login screen.
2. The admin screens.
3. The questions screen.
4. The results screen.
5. The video tutorial screen.

5.3.1. The login screen

When the program is started, students are presented with a login screen (see Figure 5.2) where they are prompted to enter their student number and click OK to start the test. When the OK button is clicked, WordAssessor checks a SQL database to determine if the student has been registered for the software skills course (for this scenario). The SQL database is located on a central storage server. This use of data storage via a network on a central SQL server is fundamental in many modern software applications, and is related to Fogg's (2002) "fourth wave of focus" (see section 2.2), of the computing maturation process.
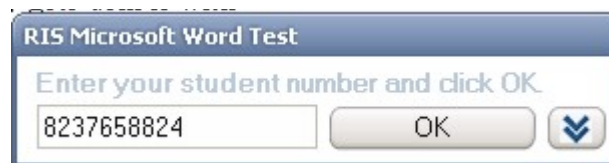
Figure 5.2. WordAssessor login screen

If the student is registered for the course, the skills test begins. If the student's details are, for some reason, not present in the database, an administrator clicks on the button to open a second login screen (see Figure 5.3).

Figure 5.3. WordAssessor Admin login screen

If the correct password is entered, the administrator clicks OK to open a small database editor (see Figure 5.4), where the student's details can be modified. The above (Figure 5.3) login and user authentication method conforms to the suggestion mentioned in section 3.2.2.

5.3.2. The admin screens

Each row of information (in Figure 5.4) contains a unique student number, a surname, initials, a mark (out of 25) and whether the student has completed the test. This is to prevent students from repeatedly logging into the test.
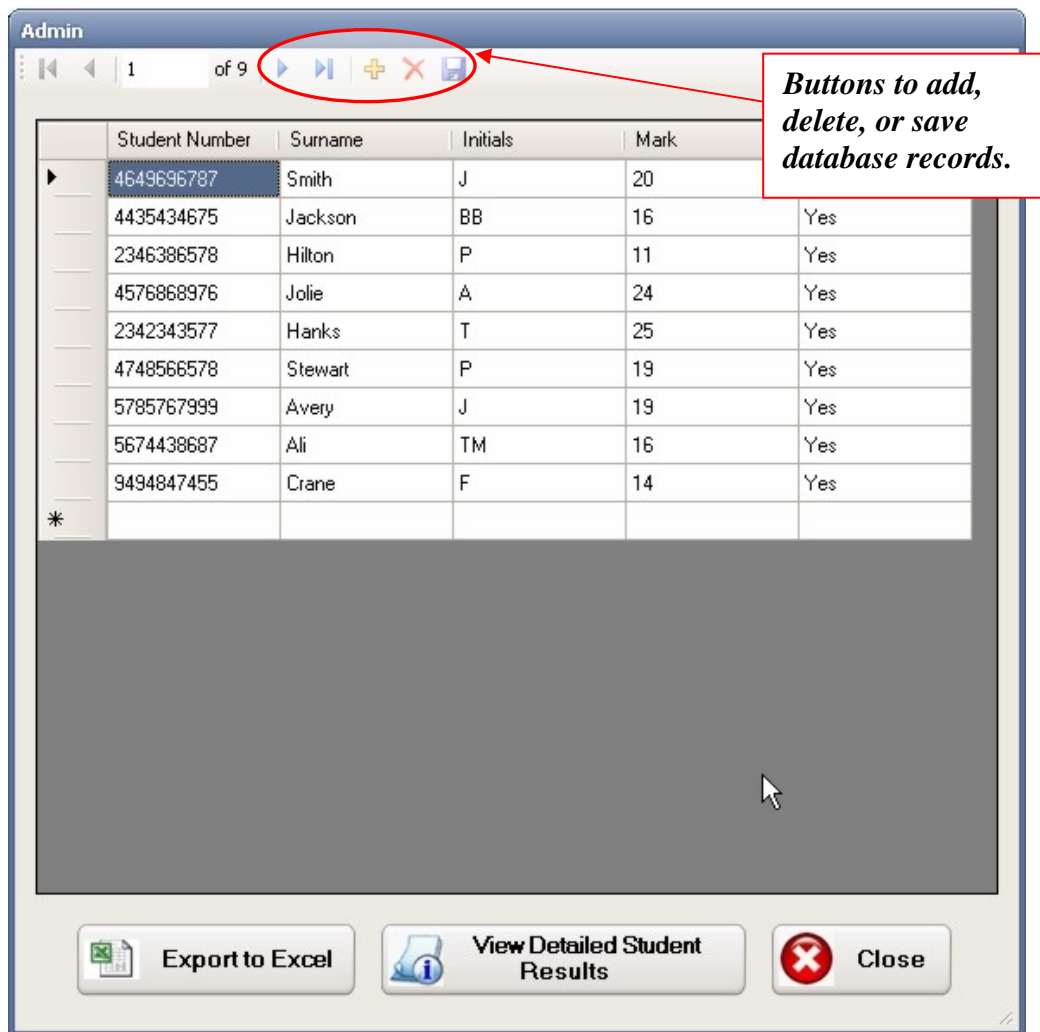
Figure 5.4. The main admin screen

The administrator (lecturer) clicks on "Export to Excel" to save the data in an MS Excel datasheet (as described in section 3.2.2). The lecturer can also click on "View Detailed Student Results" in order to view a secondary admin screen.

On the secondary screen (see Figure 5.5), the lecturer can see exactly which questions were answered correctly or incorrectly. In addition, the time taken for each question and the total time for each test are also displayed. Similar to the screen in Figure 5.4, the secondary screen includes an "Export to Excel" function (as described in section 3.2.2).

Figure 5.5. The secondary admin screen

As seen in Figure 5.5, the time taken to complete each question (measured in seconds) is recorded in the data table.

5.3.3. The question screen

When users start the software skills test, they are presented with an open MS Word document with a prepared body of text. At the bottom of the screen (Figure 5.6), they are presented with the first test question in a small, rectangular box. The users are requested to carry out the instructions presented in the question box. When they have finished, they click on the "Submit Answer" button. This demonstrates how WordAssessor employs a well-timed interruption strategy (as described in section 2.4.8). WordAssessor then automatically marks the question and presents the user with the next question. It must be noted that WordAssessor does not award marks for partially correct answers. In other words, WordAssessor only checks the task outcome and does not award marks according to which "path" was taken or methods used while answering the question.

The purpose of this is to investigate the validity of the null hypothesis $H_{0,2}$ (as described in section 1.4.1), whereby it is assumed that "When referring to an end user computer task, there is no difference in the assessment outcome when assessing the path/method followed as opposed to the task outcome."

For example, if certain text needs to be formatted "**Bold**", awarding marks for only clicking on a menu leading to the "**Bold**" setting does not seem to have any value when compared to real-world software usage standards. The assumption is that marks should only be awarded if the formatting change has been applied successfully (see section 7.6 for the validity of this assumption and the results of this investigation).

The users have to finish all 25 questions within 40 minutes. The amount of time left for the test is indicated at the bottom right corner of the questions screen. This is to constantly make users aware of the conditions of assessment, as mentioned in section 4.4.1.1. At certain times, the amount of time left is also displayed as a balloon-tip (see Figure 5.7) to provide the user with an extra visual cue. This aspect is described in section 2.4.5.

The MS Word software skills test environment is presented in Figure 5.6.
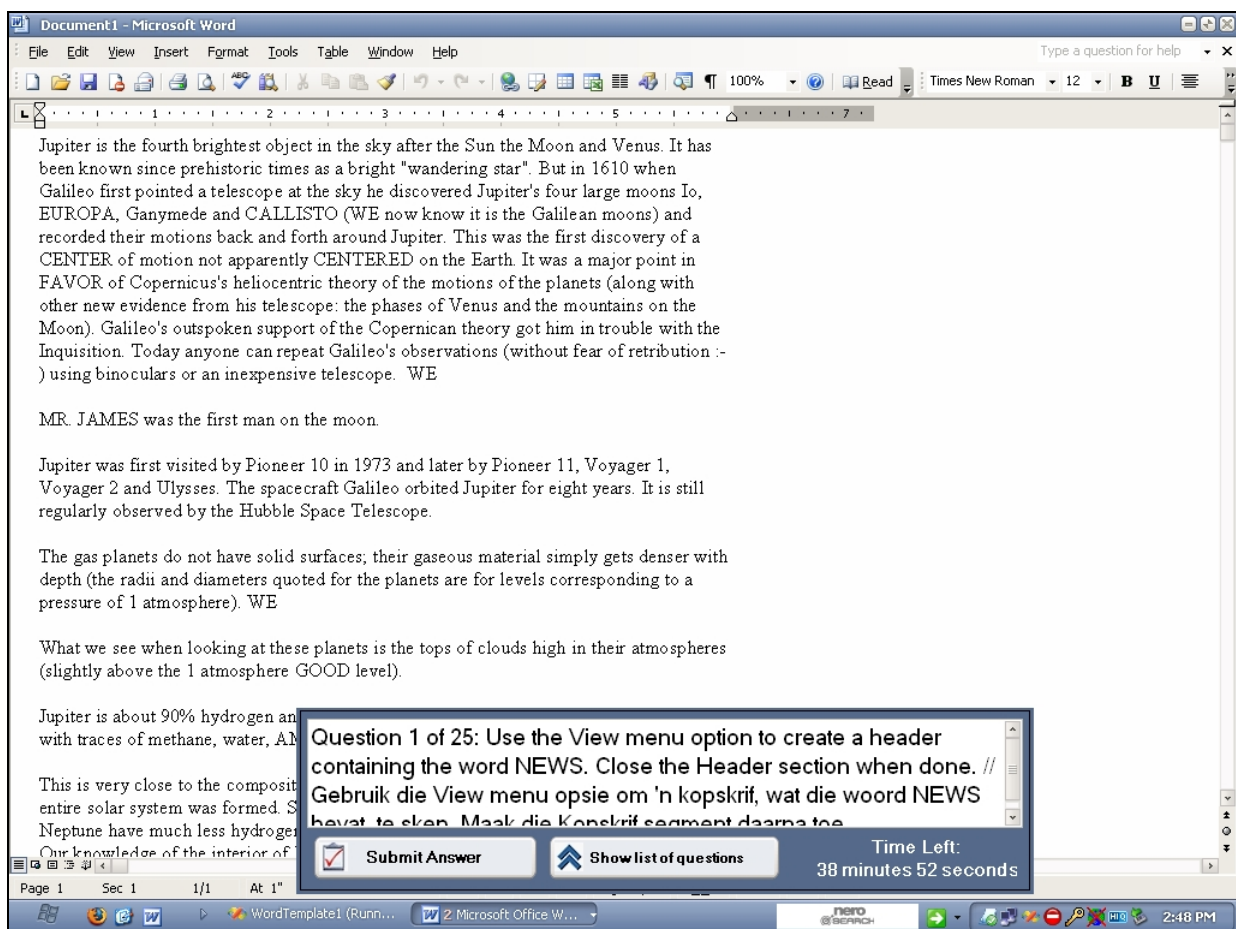
Figure 5.6. The WordAssessor software skills test environment
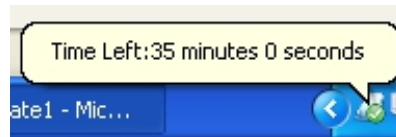
Figure 5.7. WordAssessor balloon tip

If users wish to skip ahead to another question, or retry a question, they can click on the "Show list of questions" button. This aspect demonstrates the non-linear questioning strategy of WordAssessor, as discussed in section 2.3.1. As shown below, this screen contains a blue link and a checkbox for the 25 questions of the main test. Users can click on any blue link to attempt the corresponding question.

A checkmark next to a question signifies that the question has been completed. This provides an indication of progress for the user during the test (as recommended by Faulkner (1998) - see section 2.3.1). This screen is also presented to users after they have finished all 25 test questions.

Figure 5.8. The question review screen

In Figure 5.8, users have the option of retrying questions, or ending the test and submitting all answers. This is done by clicking the "Exit / End Test" button.

5.3.4. The results screen

When the test has ended (either by finishing all the questions and clicking on Exit/End Test, or if 40mins has expired), users are immediately presented with the results of their test (see Figure 5.9). They are told which questions they had correct or incorrect, as well as the final percentage for the skills test. To see what the question statement was for a question, the user can hold the mouse cursor over one of the question numbers presented on the screen.

Figure 5.9. The question review screen

5.3.5. The video tutorial screen

Users are obliged to click on the "See Solutions" button. They are presented with the solutions to the questions they had incorrect in the form of short video tutorials. The duration of each video solution is typically less than one and a half minute per question. Each important step toward achieving the correct solution is highlighted and indicated with graphical indicators. As mentioned

in section 3.3.2, the purpose of presenting video solutions to students after their skills test is to promote vicarious learning (learning by observing others). In addition, the video solutions reaffirm the credibility of the final test results reported (as mentioned in section 2.4.7). The response of students to this approach is mentioned in section 7.7. Three example screen captures of the video tutorial mode are shown in Figures 5.10, 5.11 and 5.12. These demonstrate how symbolic mental rehearsal (linking visual images with summaries) is implemented in WordAssessor's video tutorials, as first mentioned in section 3.3.2.
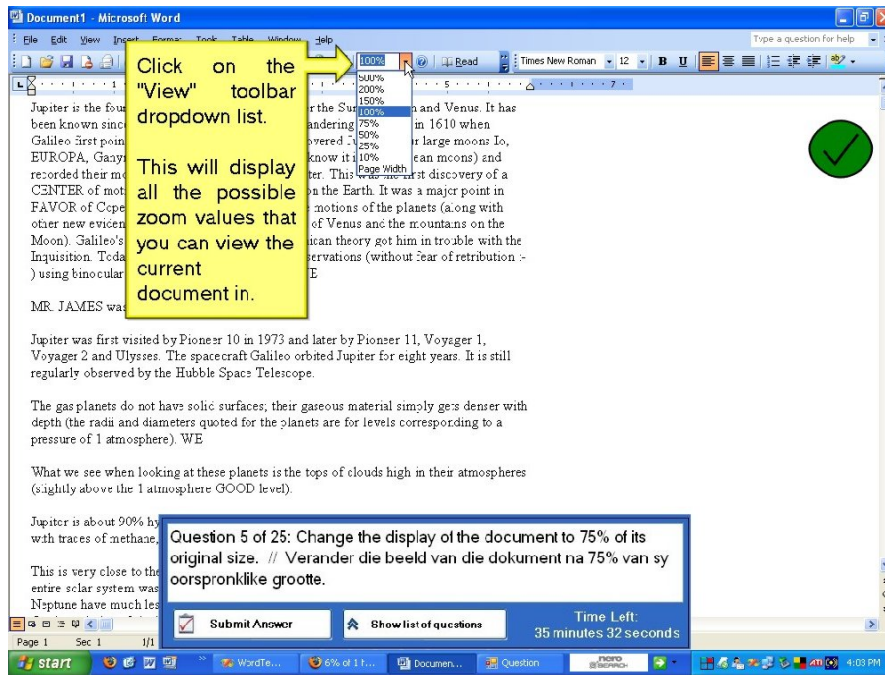

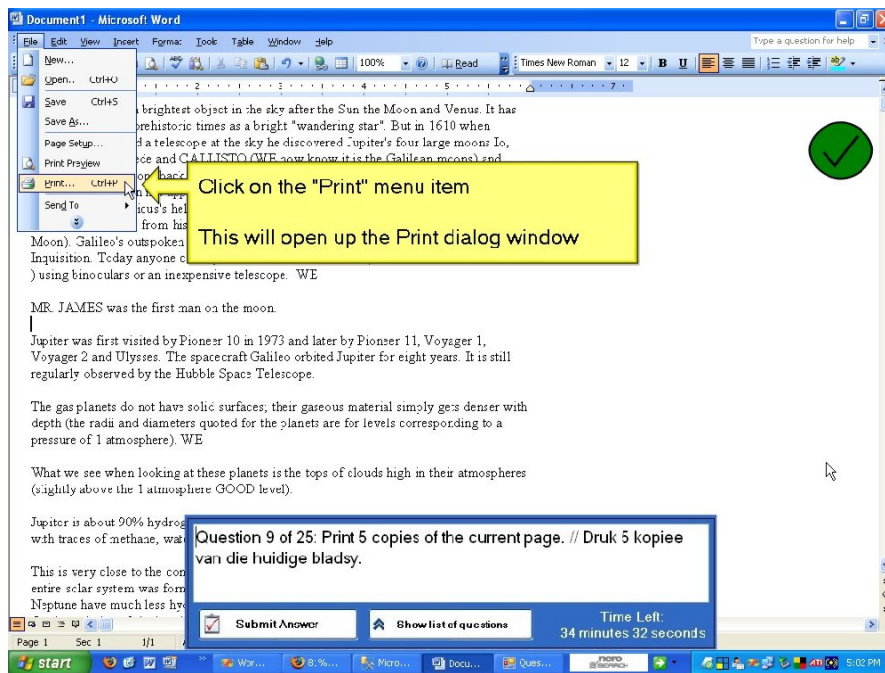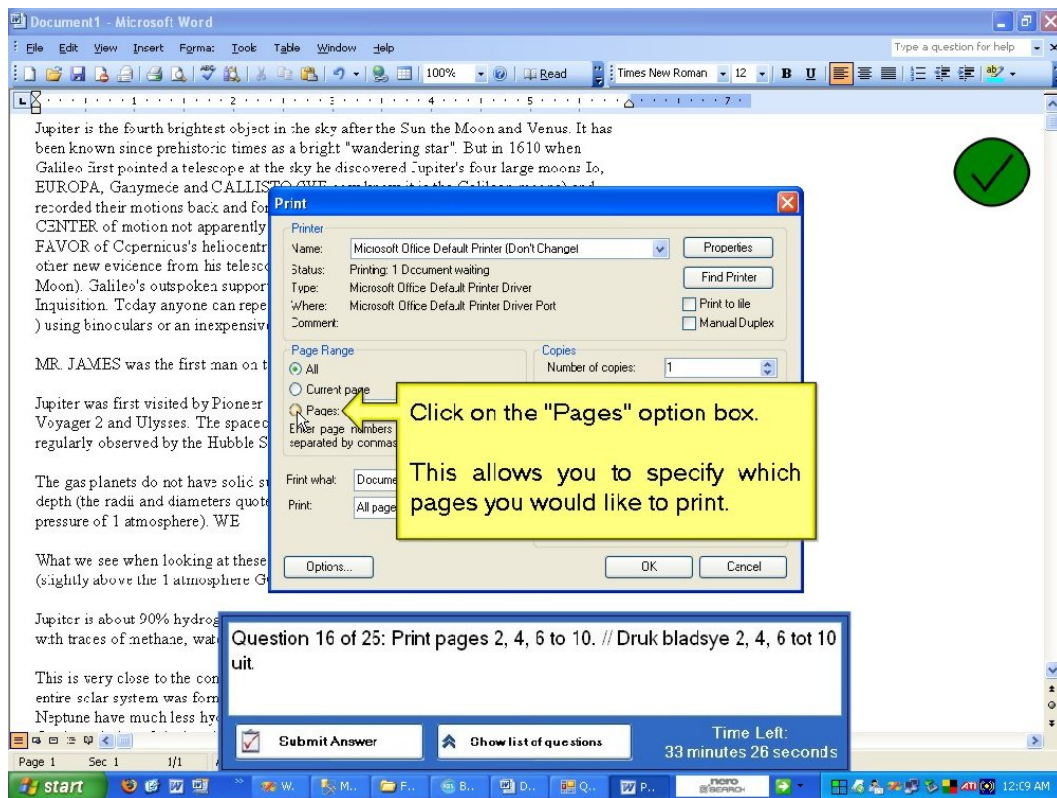*Figure 5.10. Video tutorial example screen 1*

Figure 5.11. Video tutorial example screen 2

Figure 5.12. Video tutorial example screen 3

5.4. Summary

This chapter shows how the WordAssessor system functions in order to test the proposed assessment paradigms (mentioned in section 1.3 and 1.4). The different parts of the system's user interface were detailed, as well as how they link to various aspects discussed in the literature study as provided in chapters 2, 3 and 4.

In the following chapter, the methodology for the pilot test of the WordAssessor system, as well as the comparative study are discussed.

Chapter 6

The pilot test and research study methodology

The chapter focuses on the following aspects:

- **The pilot test**
- **The main comparative study**

6.1. Introduction

This chapter considers the preparation and execution of the practical component of the comparative study. The pilot test of the WordAssessor e-assessment tool is described in order to determine whether there are any operational flaws. The preliminary pilot test serves the further purpose of ensuring that the subsequent comparative study will obtain valid results.

The criteria used for the selection of students participating in the study are explained. In addition, a detailed outline is given of the way in which the main comparative study was executed. Finally, an account is given of the way in which the comparative study was conducted in order to ascertain that it conforms to the perspectives obtained through the literature study (chapters 2, 3 and 4).

6.2. The pilot test

The following sections detail how and why the pilot test was carried out as well as which students were tested.

6.2.1. Introduction

In order to determine the optimum e-assessment paradigm for assessing word processing skills, it was necessary to carry out a practical study by which software skills could be assessed within a real-world context (by the WordAssessor e-assessment tool). However, before this study could be carried out, the validity of the WordAssessor tool had to be

determined. All potential bugs and interface problems that could hamper the e-assessment process had to be identified by means of a pilot test.

6.2.2. Basic premise

The pilot test involved assessing approximately 350 students at UFS, with the WordAssessor tool as part of their advanced computer literacy course. Students had to complete a software skills test, which required them to carry out 25 different word processing tasks. The WordAssessor tool would automatically check if the students carried out these tasks correctly, and assign a final mark (out of 25) to each student.

6.2.3. The participants

The ages of the participants ranged from eighteen to the mid-twenties. All the students involved had previously taken an introductory semester course on computer literacy. They were therefore familiar with the basic word processing interface structure.

6.2.4. Pilot test setting

The software skills tests were conducted in a large computer laboratory on the UFS campus. In order to maximise the possibility of identifying potential problems with the WordAssessor system, the assessment occurred over several practical sessions, lasting a total of four days.

6.2.5. Pilot test execution

During each practical session, administrators were on hand to answer questions. Observing the students during these sessions proved valuable indeed, since minor errors within the WordAssessor application were discovered (see section 7.2 for details).

Students also received an on-screen prompt directly after the test, in which they were asked whether they found any facet of the e-assessment test system problematic or confusing (see section 3.2.1). They were given the opportunity to mention any software-related bugs or errors during the e-assessment process (see section 7.2 for the results of this investigation). In addition, valuable lessons were learned regarding how users work

within a word processing environment. All the discoveries led to the subsequent modification and optimisation of the WordAssessor interface.

6.3. The main comparative study

After the pilot test had been completed, the necessary modifications were made to the WordAssessor system. Once this had been completed, the main comparative practical study was carried out at the UFS. The main aim of the study was to determine which e-assessment methods would provide the most reliable representation of students' true word processing skills knowledge. The results of the main practical comparative study are discussed in chapter 8, as well as how the hypotheses mentioned in section 1.4.1 are affected by the results.

6.3.1. Participants

At the UFS, students were assessed as part of their advanced computer literacy course. The same 40 minute test was administered in three different ways.

6.3.1.1. The automated e-assessment systems

Initially, a group of 160 students were assessed by the existing test system, and another group of 160 students by the WordAssessor system. Observing the guidelines specified in De Vos, Fouche, Delport and Strydom, (2007), these two population samples were selected from a total of 1117 students. This ensured a representative selection of the population in each sample group (in terms of skill level and expertise). In other words, students of all skill levels were assessed in each sample group.

6.3.1.2. The unautomated personalised test

A smaller group of students (25) were involved in a one-on-one assessment session. The administrator personally accompanied each student, one at a time, for the duration of the 40 minute test. Time constraints necessitated the smaller size of this group.

In order to provide a reliable representation of student skill level and expertise, students in the smaller group were selected based on their final average in the computer literacy course for the previous semester. Students were selected as follows:

- five students with an average below 40%
- five students with an average between 40-55%
- five students with an average between 55-70%
- five students with an average between 70-85%
- five students with an average above 85%.

Of this group, approximately half was also assessed with the existing test system. In addition to this personalised test, the remaining students were also assessed with the WordAssessor system. Students were selected in this manner to ultimately determine which automated e-assessment scenario yields the most reliable results with regard to word processing skills knowledge. See section 7.6 for a detailed presentation and analysis of the results.

6.3.2. The test questions

The word processing skills tests contained the same 25 questions (as described in appendix A) used during the pilot test of WordAssessor. It is important to note that these specific questions were chosen because they had already been used in the preceding year as part of the advanced computer literacy course at the UFS.

The course lecturer for that year implemented this list of questions as part of an "existing test system" test and hence the difficulty level of the questions conforms to the requirements of the advanced computer literacy course. The questions also conform to the QCA (2007) recommendation of minimising the complexity of questions (as mentioned in section 3.2.2).

Figure 6.1. gives an overview of how many students had to be assessed in each different test scenario.
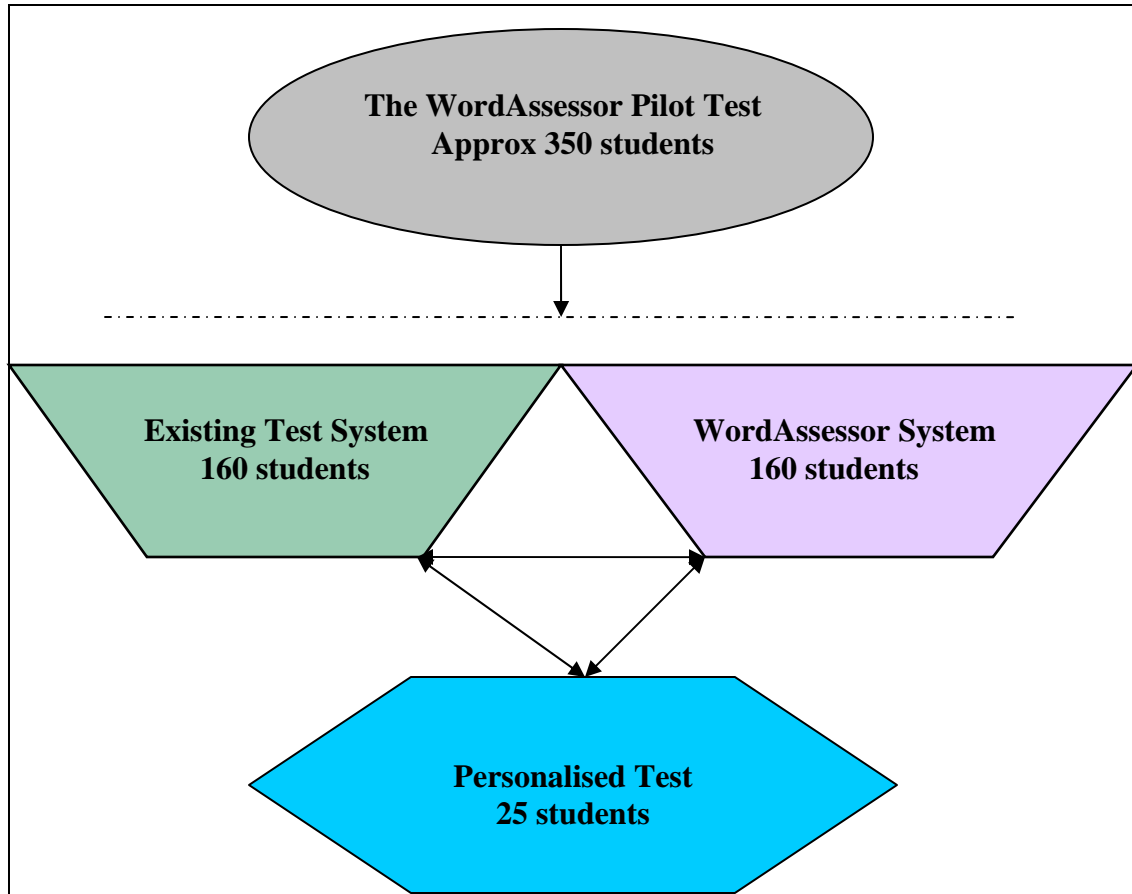
Figure 6.1. The pilot test and three different main software skills tests, all using the

same 25 questions.

6.3.3. Testing methodology

The following sections will detail how each of the three different software skills test
scenarios were carried out.

6.3.3.1. Personalised test

During the personalised, one-on-one assessment session, the following occurs for each of
the 25 students:

- Each student is assessed separately (i.e.not at the same time).
- The student is seated at specific workstations, where s/he is presented with a MS
  Word 2003 document window containing a body of text.

- The above-mentioned list of 25 questions is presented to the student on a piece of paper.

- The student is asked to perform the tasks as instructed by the questions in the document window.

- The administrator/assessor spends a maximum of 40 minutes with each student, closely monitoring everything that is done and awarding a single mark for a correctly answered question, and half a mark for partially correct ones.

The idea is to obtain the best possible indication of the students' true word processing skills knowledge in the absence of interference from an automated assessment tool.

6.3.3.2. Existing test system test

During the existing test system, students were asked to provide login details and complete the 25-question skills test within a period of 40 minutes. The environment of the skills test was simulated to look and function like a limited version of the MS Word 2002 environment (as discussed in section 1.2.2). Students were limited to specific methods of answering, and they were prevented from seeing the end result of their actions.

6.3.3.3. WordAssessor test

Students were asked to enter login details (as discussed in section 3.2.2), and given 40 minutes to complete the WordAssessor test. As opposed to the ETS scenario, they were now given the opportunity to use different methods of answering the test questions. Before the test commenced, students were given verbal instructions on the basic operations of the e-assessment test system. In addition, they were presented with basic instructions in printed form, on a piece of paper (refer to appendix D). Students were given a few minutes to review the instructions to become familiar with the system's functions (as discussed in section 3.2.2).

Additionally, students were given verbal encouragement prior to starting their test. They were assured that an administrator would be available to answer questions regarding the

test simply by raising their hand at any time. This is in line with Bandura's (1986) third source of self-efficacy, as mentioned by Gupta and Bostrom (2006) (see section 3.3.1).

During the test, students were closely monitored by administrators. This was to prevent plagiarism or copying (see section 3.2.2).

While students were taking the test, they were able to see the results of their actions, after which they could decide if they wanted to submit their answers. This is a form of discovery learning (as mentioned in section 3.3). Following the WordAssessor test, students were presented with compulsory video solutions (as described in section 5.2.5) to the questions they answered incorrectly. This was to ensure that they learn form their mistakes.

After students had finished the video solutions, they were asked to fill out a paper-based questionnaire (see appendix C), relating mostly to their preference with regard to the video tutorial method of feedback. The questionnaire also asked students which assessment methods they prefer in this type of environment. The results are presented in section 7.2.

It is important to note that all three versions of the same skills test (existing test system, WordAssessor, and personalised test) were presented to students during the same relative time frame of the UFS advanced computer literacy course. In other words, all students had been exposed to the same instructional material (as part of their advanced computer literacy course) before participating in the study.

6.4. Chapter summary

In this chapter, the methodology used during the pilot test of the WordAssessor system, as well as the comparative study, are discussed. A description is provided of the three software skills test scenarios. This includes an explanation of those features of the test designed in accordance with the recommendations deduced from the literature investigated in the earlier chapters.

In the following chapter, the results of the different word processing software skills tests are revealed, including student preferences regarding particular software skills assessment methods.

In addition, the effects of the video tutorials on this particular e-assessment process are noted. Finally, all remaining data are analysed and discussed.

Chapter 7

The comparative study

This chapter focuses on the following aspects:

- **Lessons learned from the pilot test**
- **Overall final results**
- **Detailed Chi-square analysis of final results**
- **ETS and WA marks correlation with PT**
- **ETS and WA marks correlation with BRS**
- **WordAssessor post-test questionnaire results**
- **Discussion of results**

7.1. Introduction

During the first part of this chapter, the lessons learned during the pilot test are discussed. Secondly, the results of the three software skills test scenarios (as described in the previous chapter) are detailed. The aim is to determine which of the two automated scenarios yields the most reliable indication of a user's true word processing skills knowledge. This should indicate the assessment methods (paradigms) to be recommended for use in this regard.

In order to reach the above-mentioned goal, a single factor ANOVA (analysis of variance) and post-hoc data analysis are performed for the scenario in which one group of students took the personalised test (PT), another group the ETS (existing test system) test and another group the WA (WordAssessor) test. Secondly, a detailed chi-squared analysis for the results of the 25 questions across all 3 test scenarios is performed and presented in tabular form. Also, a data correlation analysis is performed to determine the sets of marks that correlate best with students' true software skills knowledge. All the results are presented in either tabular or graphical form, and the purpose for each analysis explained.   .

Finally, the results of the questionnaires that were given to students after the WA test are revealed. This will determine the preferences of students with regard to feedback and e-assessment methods in a skills assessment scenario.

7.2. Lessons learned from the pilot test

The errors and bugs found during the pilot test of WordAssessor were mainly due to observing how students worked with the tool. In addition, questions that students asked during their e-assessment also proved valuable in identifying areas for improving the WordAssessor interface and functionality. This is line with the discussion in section 2.5.3, whereby the literature indicates that questions and actions by users during the pilot test should be recorded. Finally, students were given an on-screen prompt (as mentioned in section 3.2.1 and section 6.2), whereby they could point out any bugs or flaws that they encountered while being assessed. The problems identified in this way are mentioned below.

In the MS Word environment, it was found that certain program functions would result in the appearance of certain dialog windows. The problem was that, sometimes dialog windows would appear slightly behind the WordAssessor question box. The reason for this was that the main question box (as shown in the Figure 5.5.), had a permanent interface setting that made it "Always-On-Top". When a dialog window appears behind the question box, this would lead to confusion and result in incorrect answers.

The WordAssessor question box would cover up certain buttons on the MS Word dialog window form, preventing the student from clicking on them. The dialog window would need to be moved manually to a position higher on the screen. An example of this error is shown in Figure 7.1.

Additionally, some users found the font type of the question box to be bothersome, as the computer screens in the laboratory were quite small. As a result, font type was changed and increased in size.
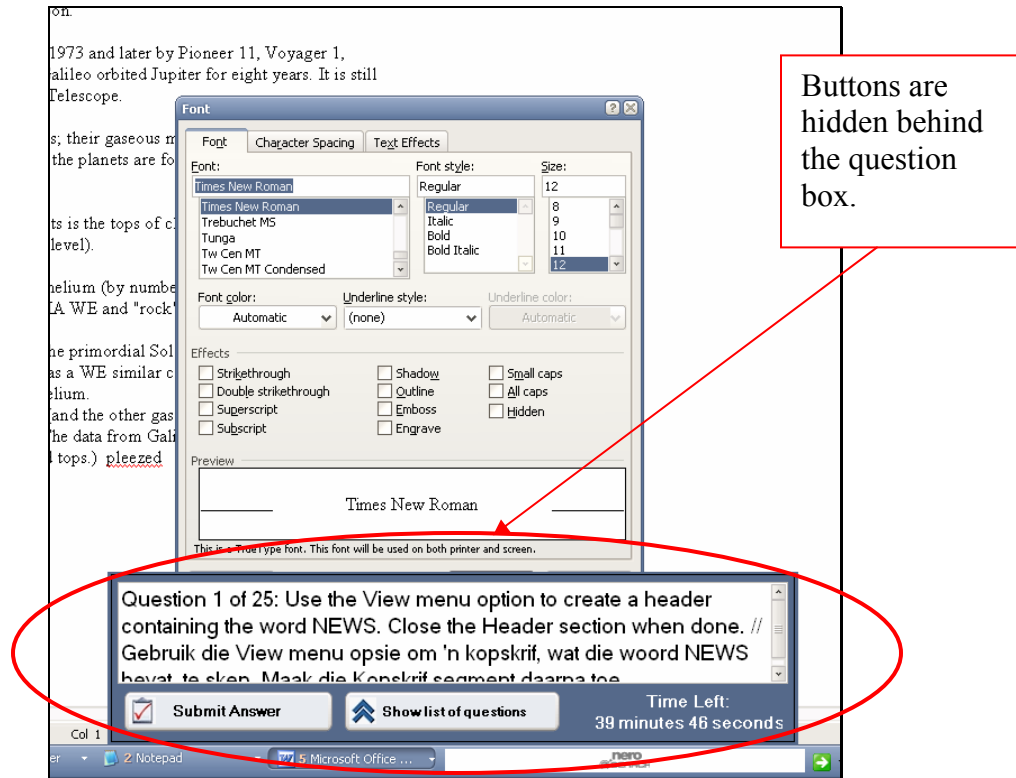
Figure 7.1. WordAssessor error discovered during the pilot test.

An example of a distracting interface component (see section 2.4.3) was also discovered. When users clicked on the button that "showed all questions", they sometimes forgot to minimise that panel by clicking on the "hide list of questions" button. This form would then remain open and distract students during the test.

Another issue was that certain students wanted clearer instructions prior to the test. Other students wanted to see the correct way of responding to a question when they had finished either the question or the test. Consequently, video tutorials were implemented for the main comparative study (as explained in chapter 5).

Body language was monitored (as suggested in section 2.4.3) and sometimes signs of exasperation were observed. This revealed that speed of the WordAssessor program was occasionally slow and unresponsive. Most importantly, students noted that the speed of the program was not always as fast as it should be, particularly when submitting an answer. The 25 questions were therefore modified to enable a much quicker response

time. The network and PC speeds were also factors in the performance of the WordAssessor system.

Another noteworthy problem was that users wanted an easier way to redo questions or to skip ahead. This was implemented with better efficiency in the final version of the program and users were given clear instructions on how to accomplish this (see section 5.3.3).

Another noteworthy flaw was that some computers in the laboratory were configured for US English and some for UK English. In order for some questions to work correctly, WordAssessor requires a computer on which the regional settings are correct. The necessary enhancements were made to the final version of the program.

The discovery of the above flaws helped ensure that the main comparative study could be carried out with minimal difficulties.

7.3. Overall final results

The following sections give an outline of the final results for all three test scenarios.

7.3.1. Is the PT a reliable benchmark?

Firstly, it had to be determined if the results of the PT could be used as a reliable benchmark for determining students' true word processing software skills knowledge. In this respect, two sets of marks from the same group of students were collected for analysis.

The first set contained the final scores achieved during the PT. The second set contained the final marks achieved by the same students during the basic computer literacy course (during the same year of study, some months prior to taking the PT). The above-mentioned basic computer literacy course (henceforth referred to as BRS) aimed to teach students basic software skills knowledge in the Microsoft Windows XP environment. Students were taught to use the core software applications of the Microsoft Office

package. Most notably, MS Word skills would also be taught and assessed as a significant part of the course. The assessment of students at the end of the BRS course was designed to estimate student's true software skills knowledge.

The correlation between the two sets of marks is used to determine whether the PT yields a reliable indication of a student's true word processing knowledge.

Below (Figure 7.2) is a graphical representation of the final mark correlation between the PT and the BRS module.
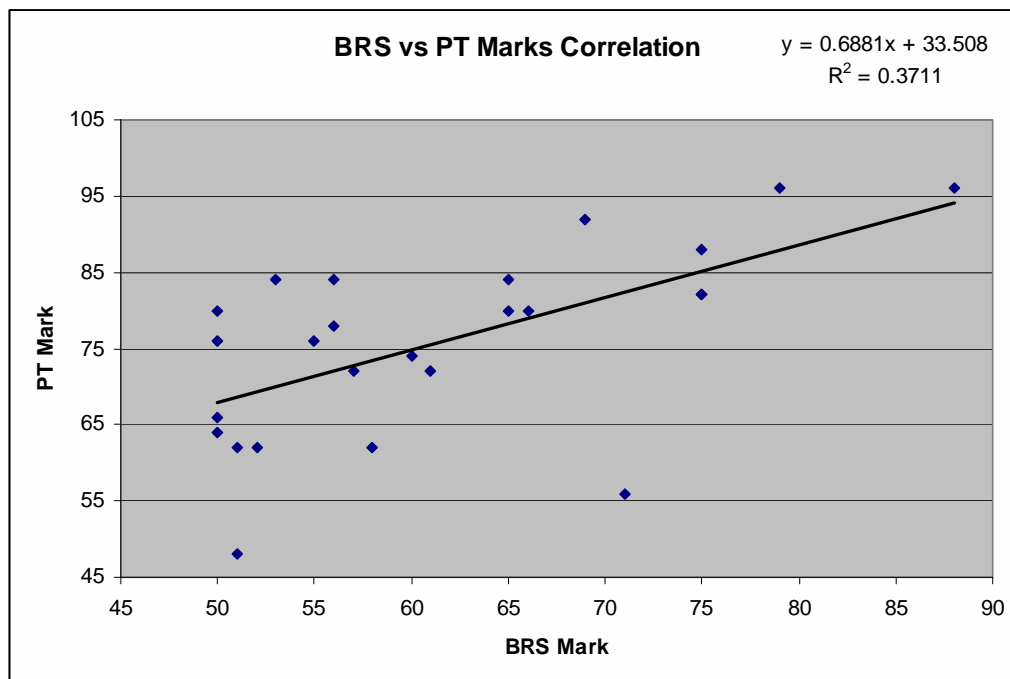
Figure 7.2. PT vs. BRS final marks correlation

For the above correlation, a two-tailed correlation test was used with df = 23. The correlation coefficient was found to be **r = 0.609** and **p = 0.0001**. This shows, that with α = 0.05, the correlation is significant and thus the PT can be used as a benchmark with which ETS and WA can be compared.

7.3.2. Final results graphical ANOVA

Table 7.1. contains the results mean (average final score for students), standard deviation, and number of measurements (students used) for each test scenario. The results are based on three different groups of students, with one group taking the ETS test, a second group taking the WA test, and the final group taking the PT (as mentioned in Section 7.1).

For an overview of the final results, the following three null hypotheses can be formulated:

- $H_{0,1}$ = There is no difference between the final results average of ETS and PT.
- $H_{0,2}$ = There is no difference between the final results average of WA and PT.
- $H_{0,3}$ = There is no difference between the final results average of ETS and WA.

	Mean Score	**Standard deviation**	**Number of students**
ETS Test	69.58	13.09	160
WA Test	73.83	8.22	160
Personal Test	75.76	12.09	25

Table 7.1. Summary of final results for all three tests
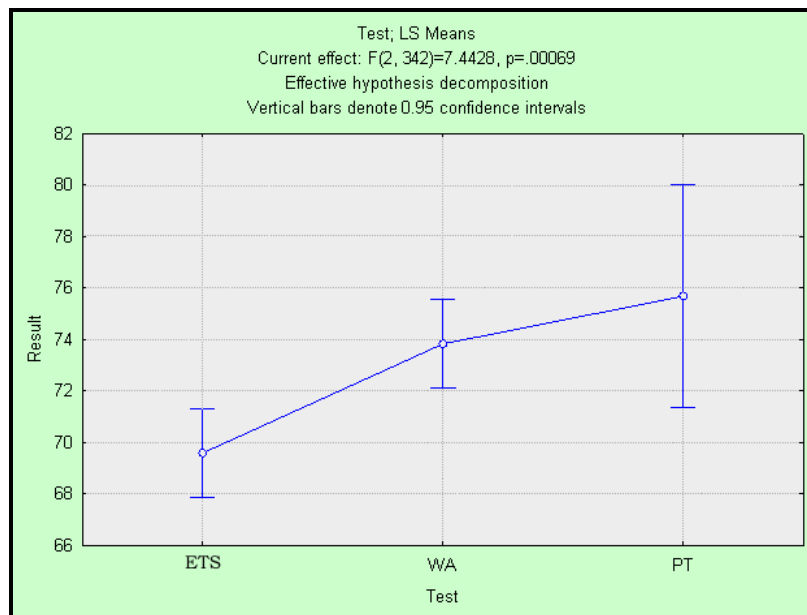
Figure 7.3. Graphic representation of final marks for all three tests

Figure 7.3. is a graphic representation of the final results for the three scenarios in terms of variance analysis. From the graph in the said figure, it is clear that the PT has the highest score average, closely followed by the WA and the ETS score average further behind.

An analysis of variance shows that the tests had a significant influence on the mark obtained ($F(2, 342) = 7.44$, $p = 0.001$). A post-hoc test (Bonferroni method) was performed to determine the significance of differences between individual tests. The results are listed in table 7.2.

Comparison	Significant? (p <0.05?)	t	p
1: ETS vs. PT	Yes	2.621	0.009
2: WA vs. PT	No	0.845	0.399
3: ETS vs. WA	Yes	3.415	0.001

Table 7.2. Data used for post-hoc test

Table 7.2. shows that the final marks of the ETS test differ significantly from those of the PT and WA. Conversely, the final marks from the WA test do not differ significantly from those of the PT. Therefore, $H_{0,1}$ and $H_{0,3}$ (as described at the beginning of this section) can be rejected, while $H_{0,2}$ cannot, i.e. there is no (*significant*) difference between the final results average of WA and PT.

To see in detail how the results of the three tests compare, the results of all 25 questions across all three test scenarios need to be examined (see appendix A for more detail on these questions).

7.4. Detailed Chi-square analysis of final results

Since WA and ETS use different methods of assessing students, it is necessary to determine which methods allow for the most reliable assessment. WA only awards marks for fully correct answers, while ETS awards marks for partially correct answers as well. WA allows students to see the end result or their actions, while ETS does not. For this reason, a chi-square analysis of the results was conducted for each question with regard

to the number of students who achieved more than 50% for a certain question. Table 7.3. is an example of a contingency table for Question 19 in a WA vs. PT comparison:

The hypothesis for this analysis can be formulated as follows:

- H_0 = There is no difference between the assessment outcome of WA and PT for question 19.

Observed Table (Frequencies per Question) Row variables: Test(2) Column variables: Pass(2)			
	Pass (>=50)	**Fail (<50)**	**Total**
WA	32	128	160
PT	7	18	25
Total	39	146	185

Table 7.3. Contingency table for question 19, with $\chi^2 = 0.832$, df = 1 and p = 0.3617

It could therefore be concluded that the difference between the assessment outcomes of WA and PT for Question 19 was not significant ($p > 0.05$), i.e. students do not perform significantly better in either of the two scenarios for this question.

In the following sections, a summary of the results of all the contingency tables is provided in tabular form for WA vs. PT, ETS vs. PT, and WA vs. ETS. See section 7.8 for a discussion of the results.

7.4.1. WordAssessor vs. Personalised test

A chi-square analysis was performed for **WA vs. PT** (see Table 7.4), according to the example for Question 19 above. The hypothesis for the analysis below can be formulated as follows:

- H_0 = There is no difference between the assessment outcome of WA and PT for each of the questions

	WA Pass	WA Fail	PT Pass	PT Fail	χ^2	p	Result
Q1	149	11	24	1	0.295	0.587	No difference
Q2	87	73	22	3	10.1	0.001	PT better -> Reject H_0
Q3	124	36	23	2	2.785	0.095	No difference
Q4	129	31	22	3	0.784	0.376	No difference
Q5	136	24	25	0	4.309	0.038	PT better -> Reject H_0
Q6	152	8	24	1	0.047	0.829	No difference
Q7	154	6	25	0	0.969	0.325	No difference
Q8	157	3	25	0	0.476	0.49	No difference
Q9	36	124	17	8	21.898	0	PT better -> Reject H_0
Q10	154	6	24	1	0.004	0.951	No difference
Q11	89	71	20	5	5.307	0.021	PT better -> Reject H_0
Q12	99	61	24	1	11.3	0.001	PT better -> Reject H_0
Q13	147	13	20	5	3.471	0.062	No difference
Q14	156	4	24	1	0.185	0.667	No difference
Q15	157	3	25	0	0.476	0.49	No difference
Q16	56	104	18	7	12.333	0.000	PT better -> Reject H_0
Q17	133	27	20	5	0.148	0.701	No difference
Q18	1	159	5	20	25.865	0	PT better -> Reject H_0
Q19	32	128	7	18	0.832	0.362	No difference
Q20	106	54	13	12	1.913	0.167	No difference
Q21	128	32	8	17	25.585	0	WA better -> Reject H_0
Q22	159	1	23	2	7.372	0.007	WA better -> Reject H_0
Q23	132	28	11	14	18.263	0	WA better -> Reject H_0
Q24	148	12	23	2	0.008	0.930	No difference
Q25	102	58	12	13	2.268	0.132	No difference

Table 7.4. Chi-square analysis for Question 1 through 25 WA vs. PT

Result	PT BETTER	WA BETTER	No difference
No. of Questions	7	3	15

Table 7.5. Results summary for Table 7.4.

From the results in Table 7.5, it is clear that students performed better in the PT for 7 out of the 25 questions. For 15 of the 25 questions, there was no statistically significant difference in the assessment outcome between the two scenarios. WA only performed significantly better for 3 out of the 25 questions.

7.4.2. Existing test system vs. Personalised test

A chi-squared analysis was done for **ETS vs. PT** (see Table 7.6) according to the example for question 19 above. The hypothesis for the analysis below can be formulated as follows:

- H_0 = There is no difference between the assessment outcome of ETS and PT.

	ETS Pass	ETS Fail	PT Pass	PT Fail	χ^2	p	Result
Q1	154	6	24	1	0.004	0.9514	No Difference
Q2	108	52	22	3	4.349	0.037	PT better -> Reject H_0
Q3	152	8	23	2	0.381	0.5372	No Difference
Q4	156	4	22	3	5.36	0.0206	ETS better -> Reject H_0
Q5	148	12	25	0	2.005	0.1567	No Difference
Q6	154	6	24	1	0.004	0.9514	No Difference
Q7	112	48	25	0	10.128	0.0014	PT better -> Reject H_0
Q8	154	6	25	0	0.969	0.3249	No Difference
Q9	152	8	17	8	19.95	0	ETS better -> Reject H_0
Q10	152	8	24	1	0.047	0.8288	No Difference
Q11	104	56	20	5	2.201	0.1379	No Difference
Q12	117	43	24	1	6.241	0.0124	PT better -> Reject H_0
Q13	119	41	20	5	0.366	0.545	No Difference
Q14	152	8	24	1	0.047	0.8288	No Difference
Q15	152	8	25	0	1.306	0.253	No Difference
Q16	66	94	18	7	8.247	0.004	PT better -> Reject H_0
Q17	115	45	20	5	0.724	0.3949	No Difference
Q18	6	154	5	20	10.209	0.0013	PT better -> Reject H_0
Q19	52	108	7	18	0.202	0.6534	No Difference
Q20	3	157	13	12	68.76	0	PT better -> Reject H_0
Q21	119	41	8	17	18.039	0	ETS better -> Reject H_0
Q22	154	6	23	2	0.944	0.3312	No Difference
Q23	85	75	11	14	0.721	0.3957	No Difference
Q24	154	6	23	2	0.944	0.3312	No Difference
Q25	67	93	12	13	0.332	0.5647	No Difference

Table 7.6. Chi-square analysis for Question 1 through 25 ETS vs. PT

Result	PT BETTER	ETS BETTER	No Difference
No. of Questions	6	3	16

Table 7.7. Results summary for Table 7.6.

From the results table (see Table 7.7), it is clear that students performed better in the PT for 6 out of the 25 questions. For 16 of the 25 questions, there was no statistically significant difference in the assessment outcome between the two scenarios.

7.4.3. WordAssessor vs. Existing test system

Since it is clear that the students scored highest with the PT test, it would be interesting to see how students performed when using the ETS in comparison with the WordAssessor system.

A chi-square analysis was performed for **WA vs. ETS** (see Table 7.8)**,** according to the example for Question 19 above. The hypothesis for the analysis below can be formulated as follows:

- H_0 = There is no difference between the assessment outcomes of WA and ETS.

	WA Pass	WA Fail	ETS Pass	ETS Fail	χ^2	p	Result
Q1	149	11	154	6	1.553	0.213	No difference
Q2	87	73	108	52	5.79	0.016	ETS better -> Reject H_0
Q3	124	36	152	8	20.659	0.000	ETS better -> Reject H_0
Q4	129	31	156	4	23.386	0.000	ETS better -> Reject H_0
Q5	136	24	148	12	4.507	0.034	ETS better -> Reject H_0
Q6	152	8	154	6	0.299	0.585	No difference
Q7	154	6	112	48	39.298	0.000	WA better -> Reject H_0
Q8	157	3	154	6	1.029	0.310	No difference
Q9	36	124	152	8	173.514	0.000	ETS better -> Reject H_0
Q10	154	6	152	8	0.299	0.585	No difference
Q11	89	71	104	56	2.937	0.087	No difference
Q12	99	61	117	43	4.615	0.032	ETS better -> Reject H_0
Q13	147	13	119	41	17.466	0.000	WA better -> Reject H_0
Q14	156	4	152	8	1.385	0.239	No difference
Q15	157	3	152	8	2.354	0.125	No difference
Q16	56	104	66	94	1.325	0.250	No difference
Q17	133	27	115	45	5.806	0.016	WA better -> Reject H_0
Q18	1	159	6	154	3.651	0.056	No difference
Q19	32	128	52	108	6.457	0.011	ETS better -> Reject H_0
Q20	106	54	3	157	147.61	0.000	WA better -> Reject H_0
Q21	128	32	119	41	1.438	0.231	No difference
Q22	159	1	154	6	3.651	0.056	No difference
Q23	132	28	85	75	31.626	0.000	WA better -> Reject H_0
Q24	148	12	154	6	2.119	0.145	No difference
Q25	102	58	67	93	15.361	0.000	WA better -> Reject H_0

Table 7.8. Chi-square analysis for Question 1 through 25 WA vs. ETS.

The results of the above analysis are summarised in Table 7.9.

Result	ETS BETTER	WA BETTER	No difference
No. of Questions	7	6	12

Table 7.9. Results summary for Table 7.8.

From table 7.9., it is clear that students performed better in the ETS for 7 out of the 25 questions and 6 out of 25 for WA.

For 12 of the 25 questions, there was no statistically significant difference in the assessment outcome between the two scenarios.

7.5. WordAssessor post-test questionnaire results

As explained in the methodology chapter, students were provided with a questionnaire (see appendix C) directly after finishing the WordAssessor skills test and video solutions. The results of how students responded for each questionnaire question are listed below. Six questions were video tutorial-related and three dealt with the software skills assessment environment. Tables 7.10 and 7.11 summarizes the students' responses.

Row nr	Question / Statement	Response				
		Strongly Disagree	*Disagree*	*Neutral*	*Agree*	*Strongly Agree*
1	After viewing the video tutorials, I now understand where I went wrong in the test questions I had incorrect.	4.60%	1.50%	7.60%	37.90%	48.50%
2	I will be able to obtain all the correct answers if I had to redo the test.	1.50%	3.00%	8.30%	47.70%	39.40%
3	After attempting the test by myself and then viewing the video tutorials of the solutions, I feel that I've learned better in the process.	3.00%	3.00%	10.60%	38.60%	44.70%
4	The video tutorials were straightforward and easy to follow.	2.30%	4.60%	11.40%	40.20%	41.70%
5	I prefer receiving feedback though video tutorials rather than static methods like paper-based feedback or on-screen text feedback.	5.30%	6.82%	11.40%	45.50%	31.10%
6	I prefer being tested in the real MS Word environment and not a simulated environment.	0.80%	9.10%	29.60%	38.70%	22%
7	I prefer being able to use any method to answer a question (as opposed to being told what method to use).	1.50%	8.30%	15.20%	41.70%	33.30%
8		**Response**				
9		**Sometimes**		**No**	**Yes**	
10	Were the video tutorials distracting in any way?	9.10%		74.30%	16.70%	

Table 7.10. Student responses for first 8 questionnaire questions

Row nr	Question: What aspect of the video tutorials did you like the most?	
1	Response	Percentage of Students
2	Quality of Presentation	19.30%
3	The fact that it showed the solution effectively	17.80%
4	The fact that it explained each step	20.70%
5	The fact that the solution was shown by video instead of text	21.50%
6	The fact that I learned while being tested	20.70%

Table 7.11. Student responses to last questionnaire question

From Table 7.11 (see row 1), it is apparent that the majority of students felt that they knew where they went wrong after viewing the video tutorials directly after the test. In addition, most students were confident that they had learned more in the process (refer to section 3.2.1) and would be able to correctly answer all the questions, if they were to retake the test (see row 2 and 3). In other words, students indicated that their self-efficacy was increased in the process (see section 3.3.1). Approximately 87% of students either agreed or strongly agreed with the fact that they prefer video tutorials over paper or text-based forms of post-test feedback (see row 5).

60.7% of students preferred being tested in a real software environment as opposed to a simulated one. 29.6% of the students were neutral on the issue (see row 6 of table 7.11).

In row 7 of table 7.11, it is shown that 75% of the students either agreed or strongly agreed with the fact that they prefer being able to use any method to answer a question (as opposed to being told what method to use). When asked which aspect of the video tutorials they liked most, there was not an overwhelming majority for any of the responses. However, most of the respondents revealed that they most liked the fact that the solution was shown by video instead of text (see table 7.12, row 5).

Students were asked two more questions (questions 10 and 11 in Appendix C) whereby they could indicate if they disliked anything about the video tutorials, or if they had any suggestions on how the learning aspect of the video tutorials could be enhanced. The only dislike some students had, was that the pace of the video tutorials were slightly slower

than they might have preferred. No noteworthy suggestions were made with regard to improving the learning aspect of the video tutorials.

7.6. Discussion of results

Table 7.12. shows a summary of the comparison results for all three scenarios.

	PT vs. ETS	**PT vs. WA**	**WA vs. ETS**
Q1	No Difference	No difference	No difference
Q2	PT better -> Reject H_0	PT better -> Reject H_0	ETS better -> Reject H_0
Q3	No Difference	No difference	ETS better -> Reject H_0
Q4	ETS better -> Reject H_0	No difference	ETS better -> Reject H_0
Q5	No Difference	PT better -> Reject H_0	ETS better -> Reject H_0
Q6	No Difference	No difference	No difference
Q7	PT better -> Reject H_0	No difference	WA better -> Reject H_0
Q8	No Difference	No difference	No difference
Q9	ETS better -> Reject H_0	PT better -> Reject H_0	ETS better -> Reject H_0
Q10	No Difference	No difference	No difference
Q11	No Difference	PT better -> Reject H_0	No difference
Q12	PT better -> Reject H_0	PT better -> Reject H_0	ETS better -> Reject H_0
Q13	No Difference	No difference	WA better -> Reject H_0
Q14	No Difference	No difference	No difference
Q15	No Difference	No difference	No difference
Q16	PT better -> Reject H_0	PT better -> Reject H_0	No difference
Q17	No Difference	No difference	WA better -> Reject H_0
Q18	PT better -> Reject H_0	PT better -> Reject H_0	No difference
Q19	No Difference	No difference	ETS better -> Reject H_0
Q20	PT better -> Reject H_0	No difference	WA better -> Reject H_0
Q21	ETS better -> Reject H_0	WA better -> Reject H_0	No difference
Q22	No Difference	WA better -> Reject H_0	No difference
Q23	No Difference	WA better -> Reject H_0	WA better -> Reject H_0
Q24	No Difference	No difference	No difference
Q25	No Difference	No difference	WA better -> Reject H0

Table 7.12. Results summary across all three scenarios

When considering the WA, ETS and PT test scenarios, table 7.12 shows that the different assessment methods yielded different results for most of the 25 test questions (see section 7.4.4). In the PT vs. ETS scenario (column 1), there was no significant difference in the results for 16 of the 25 test questions. However, the PT yielded significantly better scores for 6 of the 25 questions, while the ETS only yielded significantly better scores for 3 questions. The difference between the final scores of the two scenarios is clearly significant. This is probably due to the fact that the PT test environment (see section 6.3.3.1) does not contain any e-assessment interface to interact with, and thus distractions are kept to a minimum.

The PT scenario assesses students' word processing skills according to a real-world standard, and as mentioned in section 7.3.1, this can be used as a reliable benchmark for estimating students' true word processing skills knowledge. In the PT vs. WA scenario (column 2 of table 7.12), there was no significant difference in the results for 15 of the 25 test questions. However, the PT yielded significantly better scores for 7 of the 25 questions, while the WA only yielded significantly better scores for 3 questions. This is similar to the PT vs. ETS scenario mentioned above.

Finally, when considering the WA vs. ETS scenario (column 3 of table 7.12), it is found that the ETS test had significantly higher results for 7 out of the 25 questions and the WA system had significantly higher results for 6 out of the 25 questions. There was no significant difference in the results of 12 out of the 25 test questions. According to the results of this scenario, it can be only be demonstrated how the "per-question" results differ and cannot be used to determined which of the two systems yield a more reliable representation of students' true word processing knowledge. In order to do this, observe Figure 7.4. The figure shows the final test marks correlation ($p = 0.489$ and $r = 0.234$ - the correlation is not significant) for students who participated in both the ETS and PT test scenario.
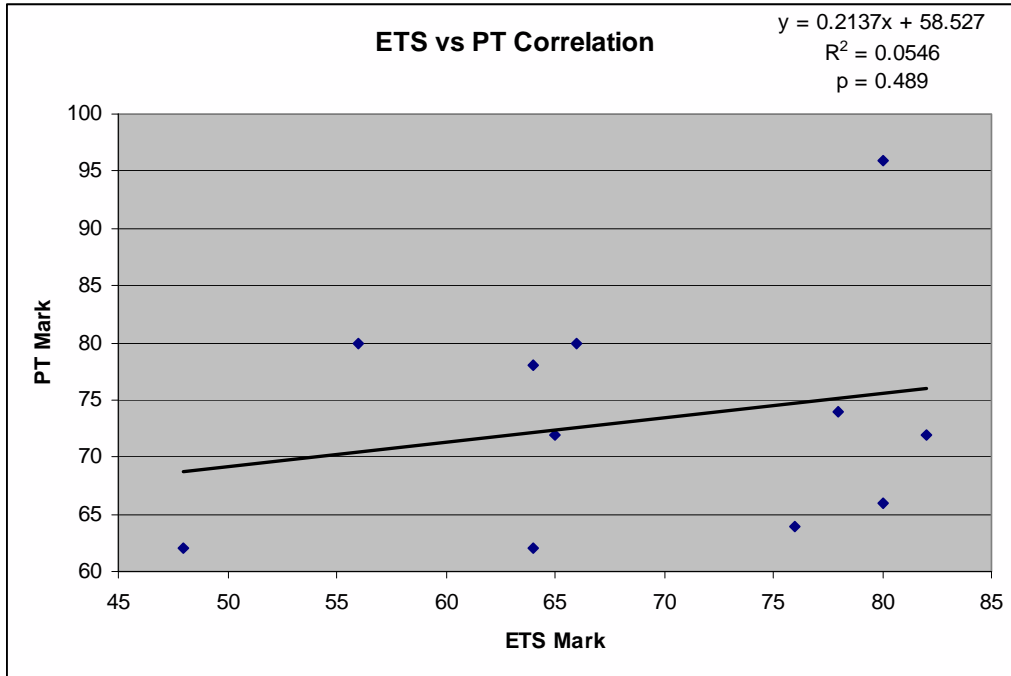
Figure 7.4. ETS vs. PT final marks correlation

On the other hand, Figure 7.5. shows the final test marks correlation ($p = 0.008$ and $r = 0.616$ - the correlation is significant) for students who participated in both the WA and PT scenario.
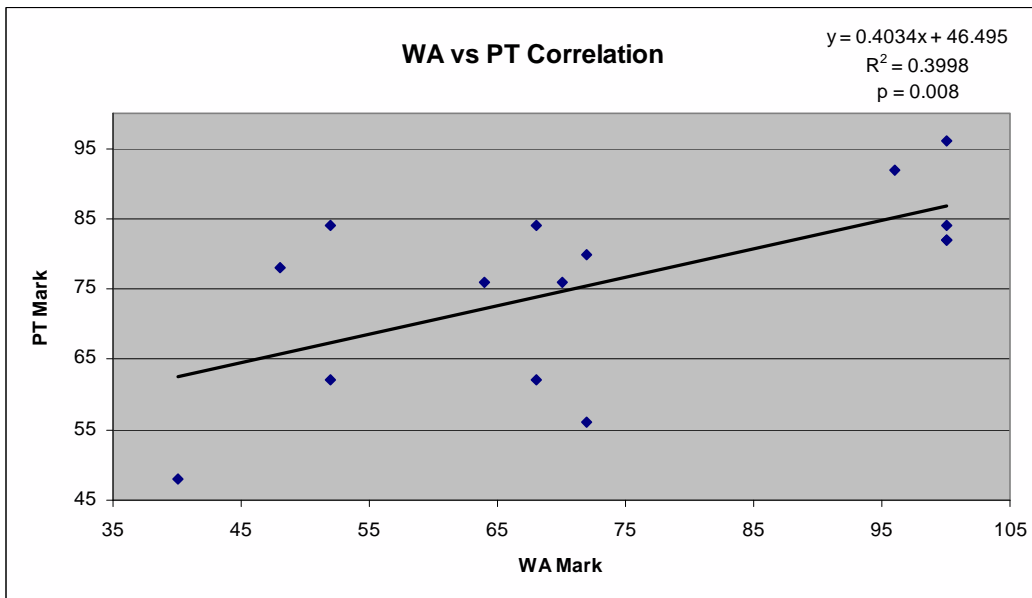
Figure 7.5. WA vs. PT final marks correlation

94

It therefore seems that WA yielded a more reliable indication of the students' true word processing skills knowledge (as opposed to the ETS), when compared to the PT benchmark. The reason for the high correlation of the WordAssessor test scenario results with those of the PT benchmark cannot be attributed to a single factor, but is most likely due to the combination of assessment methods utilised by the WordAssessor system.

To further solidify the claim that the assessment methods employed by WordAssessor yield a more reliable representation of students' true software skills knowledge, a further investigation needs to be undertaken. Figure 7.6 shows the final marks correlation ($r = 0.271$ and $p = 0.106$) between students' final marks for their basic computer literacy course (BRS) and their final marks for the ETS test. Since the correlation is not significant, this could possibly imply that the assessment methods employed by the ETS are not reliable in estimating word processing knowledge.
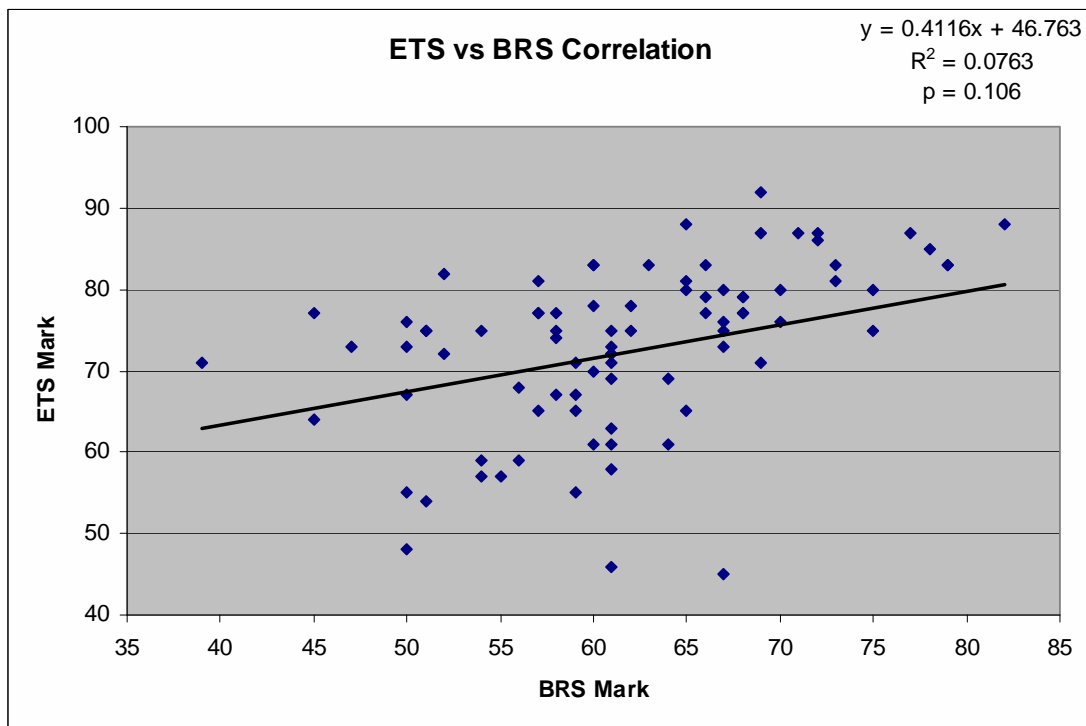
Figure 7.6. ETS vs. BRS final marks correlation

However, Figure 7.7 indicates the final marks correlation (r = 0.437 and p = 0.000) between students' final marks for their basic computer literacy course (BRS) and those for the WA test. This correlation is highly significant, and further supports the claim that the e-assessment methods employed by WordAssessor are reliable in estimating word processing skills knowledge.
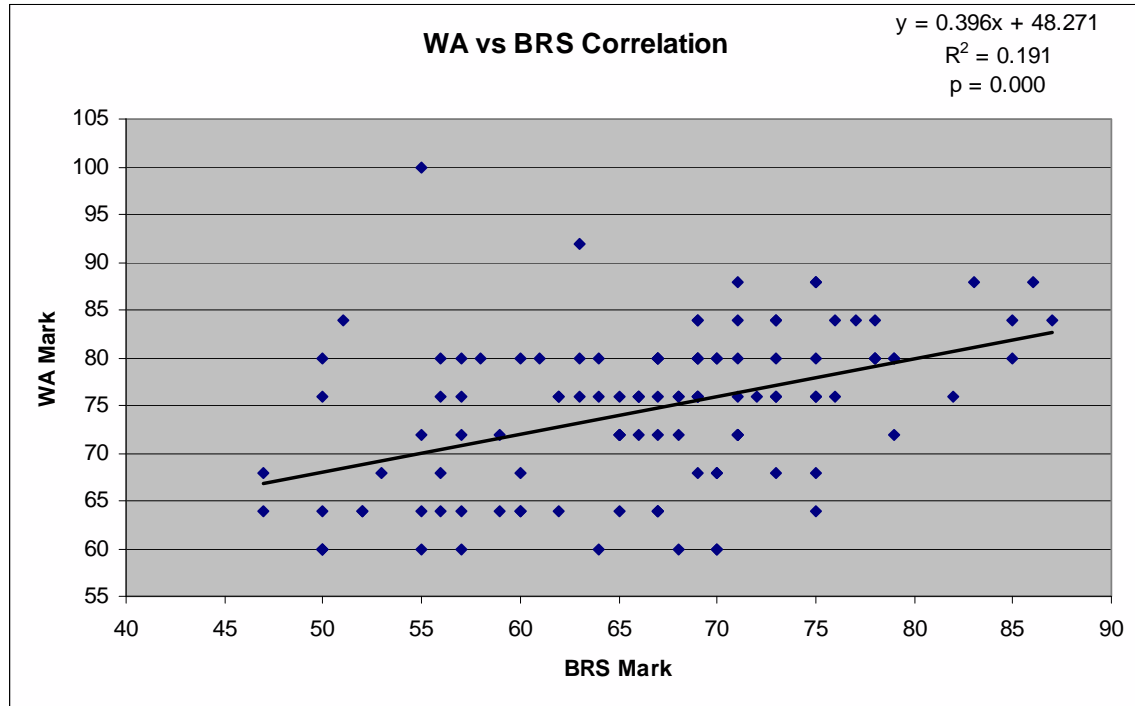
Figure 7.7. WA vs. BRS final marks correlation

This also shows that the method of awarding marks for partially correct answers during e-assessment (as mentioned in section 5.3.3) does not seem to have a particular advantage in terms of results.

Finally, it appears that WordAssessor's interruption strategy was effective in executing a reliable e-assessment process (as mentioned in section 5.3.3 and section 2.4.8).

7.7. Chapter summary

In this chapter, the final test results of students subjected to the ETS, WA and PT test scenarios were investigated. It was determined that the different assessment methods

mostly yield different results for each of the 25 test questions. It was also shown that the PT results could be used as a viable benchmark with which the results of the other two scenarios could be compared.

This comparison was administered in the form of data correlations, firstly between the final marks of the ETS and PT tests and secondly between the final marks of the WA and PT tests. By means of this final marks correlation, it was found that WA yielded a more reliable indication of these students' true word processing skills knowledge (as opposed to the ETS), when compared to the PT benchmark. This indicates that increasing students' self-efficacy (see section 3.3.1) by allowing them to explore the software environment while being assessed, increases the reliability of the assessment.

Furthermore, data correlation was performed for the final marks of the ETS test versus the final BRS (computer literacy course) module marks. This correlation was also performed for the final marks of the WA test versus the final BRS module marks. It was found the WA marks correlated more closely with the BRS marks than the ETS marks.

From the results of the post-test questionnaire, it was found that, according to students, the compulsory post-test WA video tutorials had an overwhelmingly positive effect on them with regard to learning during the e-assessment.

The majority of students also indicated that they prefer being assessed in a real software environment and being able to use any method to obtain an answer, as opposed to being told what method to use. Finally, students indicated that they prefer receiving feedback in video form, as opposed to plain text-based feedback.

An overview of this comparative study is given in the following chapter, with an interpretation of the above results. From this interpretation, conclusions are drawn to determine the findings of this study.

Chapter 8

Conclusion

8.1. Introduction

This research study was conducted to determine the optimum e-assessment methods for assessing users' word processing skills. The different strategies involved in assessing users' word processing skills were compared with one another to determine which method gave the most reliable representation of a user's true word processing skills knowledge. In addition, the process of assessing word processing skills electronically (e-assessment) was investigated in order to determine which guidelines could ultimately enhance the effectiveness of formative e-assessment (learning while being assessed). For this study, the bridge between assessment theory and its practical application in the field of information technology needed to be constructed effectively.

8.2. Aims and motivation

One of the reasons that motivated this investigation is as follows: Students at the University of the Free State, who were assessed as part of their computer literacy course, reported that they were dissatisfied with the automated assessment tool that tested their word-processing skills. The main problem was that this program (referred to as the existing test system (ETS) in previous chapters), uses assessment methods that seemed, according to the students, questionable in its usefulness and efficiency (it must be mentioned that this study has shown that the traditional pen-and-paper method of assessing software skills is unpractical – see section 4.5).

As a result, a new computer assessment tool was developed specifically for this study (referred to as WordAssessor in previous chapters). Its purpose was to employ newly proposed assessment methods (as first described in section 1.3 and 1.4), to assess users'

word processing skills and potentially enhance the effectiveness of the software skills assessment process.

8.3. User testing

To gauge students' true word processing skills knowledge in the most reliable way, a personal, un-automated software skills test was administered to students. During this test, students were presented with a list of 25 questions on a sheet of paper. Each question instructed the student to do a certain word processing software task within the presented software environment. An administrator sat next to each student and graded the performance of each student for each question.

Both the ETS and WordAssessor systems presented students with a software skills test consisting of the same 25 questions as the personal test (PT). The e-assessment program then automatically marked students' answers to the questions, and presented them with a mark (percentage).

Out of all three test scenarios, it was found that the mark average was highest (and significantly so) for the PT, and when the students' marks for this test were correlated with the marks of their basic computer literacy course (of which word processing skills played a significant role), it was found that the correlation was significant. This fact demonstrates that the PT marks could be used as a reliable representation of the students' true word processing skills knowledge.

The challenge was to determine which of the automated e-assessment methods would be most suitable for reliably assessing students' true word processing skills.

The main differences in assessment strategy between the methods employed by the ETS and WordAssessor (refer to section 1.2) are as follows (see Table 8.1):

ETS	WordAssessor
Simulated MS Word Environment.	Actual MS Word Environment.
Students were limited in methods they could use to answer questions.	Students can use any method to answer questions.
Students do not see end-result of their actions.	Students see end-result of their actions
Students obtain marks for partially correct answers.	Students only receive marks for fully correct answers[1].

Table 8.1. Program comparison between ETS and WA

In addition to comparing the effect of the above-mentioned assessment methods, this study also attempted to determine which methods of post-assessment feedback appear to best stimulate learning, in terms of student preference.

The various types of assessment, as well as the constructivist teaching paradigm, were discussed in chapter 4 to provide a foundation for the purpose and execution of this study.

8.4. The research tool

Before the different assessment methods could be implemented, compared and analysed, the WordAssessor e-assessment tool was developed. To aid its development, the various technical and human-related elements surrounding the e-assessment process were researched (chapters 2, 3 and 4).

Firstly, the various human elements that could adversely affect interaction with an e-assessment tool were examined. With regard to regulating the users' emotion, visual design principles were discussed to prevent frustration for users taking a software skills test. In addition, recommendations were made that could accommodate the user's memory. Also, guidelines for the development of an e-assessment tool were discussed,

---

[1] The WordAssessor tool was developed with the intent of only checking whether students had fully and correctly completed a given task during a test. The tool only checks if students had the end-result of a question correct, and does not check which path students follow to achieve the correct answer or which method they used to achieve the end-result.

including usability focus points during the development of the e-assessment tool. Finally, suggestions that could help identify potential user interface problems during the pilot test of the e-assessment were outlined.

On the technical side, the most important guidelines for the creation of an e-assessment system were also explored and used during the development cycle of the WordAssessor system. Once all relevant information was gathered, the WordAssessor system was developed (see chapter 5 for details on how WordAssessor was designed to conform to e-assessment guidelines, as gathered from literature).

Once the WordAssessor system was developed, a full-scale software testing study was conducted. A few hundred students at the University of the Free State were assessed (as part of their computer literacy course) to determine any possible shortcomings or errors in the system (see section 7.2). All observations were recorded and the main comparative study conducted.

8.5. Findings

To reiterate section 1.4.1, the main null-hypotheses for this study were as follows:

$H_{0,1a}$: There is no difference in the assessment outcome of a computerised assessment tool that evaluates the path/method followed, and a personalised software skills test.

$H_{0,1b}$: There is no difference in the assessment outcome of a computerised assessment tool that evaluates the task outcome, and a personalised software skills test.

$H_{0,2}$: When referring to an end-user computer task, there is no difference in the assessment outcome when assessing the path/method followed, as opposed to the task outcome.

$H_{0,3}$: Allowing a user to see the end result of a performed action does not result in a more reliable indication of his/her skills.

$H_{0,4}$: There is no difference in the assessment outcome of a computerised assessment tool that restricts users to use certain methods to perform a task, as opposed to allowing the users to use any method to perform a task.

$H_{0,5}$: There is no difference in the preference of students to work in a simulated, scaled-down software environment, and the actual (possibly complex) software environment.

$H_{0,6}$: There is no difference in the preference of students when receiving plain text-based feedback after a software skills test, as opposed to receiving feedback via video tutorials.

$H_{0,7}$: Directly following a software skills test, students do not feel that they have learned more effectively if the solutions of incorrectly answered test questions are presented to them via video tutorials.

The results of the main comparative study were detailed in the previous chapter. After analysing these results, the following was found:

By means of the final marks correlation, it was found that WA yielded a more reliable indication of students' true word processing skills knowledge (as opposed to the ETS), when compared to the PT benchmark. Also, it was found the WA marks correlated more closely to the BRS marks than the ETS marks.

This suggests that $\mathbf{H_{0,1a}}$ can be rejected in favour of $\mathbf{H_{0,1b}}$. Also, $\mathbf{H_{0,2}}$, $\mathbf{H_{0,3}}$, and $\mathbf{H_{0,4}}$ can also be rejected. In other words, in this scenario, allowing a student to see the end-result of an action during a software skills test yields a more reliable assessment of word processing skills. This conclusion also holds when students are allowed to use any method to complete a word processing task, instead of being told what method to use. Similarly, in this scenario, awarding marks for partially correct test answers does not seem to have an advantage over simply awarding full marks for fully correct answers.

When administering a software skills test, users' opinions are vital. In this comparative study, students preferred to use any method to answer a question. In addition, it seems that students overwhelmingly preferred vicarious learning (see section 3.3.2 and section 5.3.5), as they revealed that they prefer receiving video feedback instead of text feedback (as a result, $H_{0,6}$ can be rejected). In addition, students preferred working in a real software environment as opposed to a simulated one, hence $H_{0,5}$ can be rejected.

Finally, after viewing the video tutorials following the test, an overwhelming majority of students felt that they knew where they went wrong. They believed they would be able to obtain perfect marks if they had to redo the test and most importantly, they reported that they experienced much better learning in the process (consequently $H_{0,7}$ can be rejected).

8.6. Recommendations

Based on the findings mentioned above, it appears that a more flexible e-assessment approach (as demonstrated by the WordAssessor system) could be beneficial on various levels. Students' software skills should be assessed within the real software environment to allow for real-life circumstances such as *trial-and-error*. This process is central in the learning of any new software package, in addition to being a part of day-to-day usage of software programs. In addition, it is a vital source of self-efficacy (as mentioned in section 3.3.1)

Also, students should be able to answer test questions in a way that suits them, rather than being limited with respect to the method of use (the students that participated in this study made this very clear). Assessing software skills knowledge via any method not based on a real software environment possibly denies students the opportunity of being assessed reliably. As a result, this approach can be recommended for effective use in the e-assessment arena.

Another aspect that can be recommended is that marks only be awarded for fully correct answers during an e-assessment process where software skills are involved (refer to

section 5.3.3 and section 7.8 for details on this recommendation). Furthermore, with regard to video feedback, students showed an overwhelming interest in this approach. Every key step leading to a test answer was clearly explained by video and students felt more involved with the learning process while being assessed than they ever did with paper-based forms of feedback.

It is here that, for any field of study, the Internet could be used for the centralisation of video forms of feedback. Seeing how the answer to a test question is achieved step-by-step via video (even while possibly being narrated by a teacher), had a much greater effect on students participating in this study than the static, text-based form of feedback so widely used in the past.

8.7. Further research

The study has revealed some interesting opportunities for further research. In this study, it was found that allowing students to be assessed in a real software environment yielded a more reliable assessment outcome, as opposed to a simulated software environment. However, the extent to which students learn while being assessed in a real software environment still needs to be determined. Do students retain knowledge acquired during such an assessment (e.g. from trial-and-error), over time, or is the extent of memory retention regarding such newly acquired skills negligible?

In addition, the extent to which students retain the knowledge acquired from watching video-tutorials over time needs to be determined. The question remains: Do video tutorials offer an advantage over text- or paper-based e-assessment feedback in terms of long-term memory retention?

References

A.S. (Assessment Strategies, Inc.) (2007). *Assessment Methods*. Retrieved 21 June 2007 from http://www.asinc.ca/en/methods/index.html

Adamczyk, P. D., & Bailey, B. P. (2004). If not now, when?: the effects of interruption at different moments within task execution. *Proceedings of the SIGCHI conference on Human factors in computing systems CHI '0*. ACM Press

Allen, R. A. (2001). A History of the Personal Computer: The People and the Technology. *Allan Publishing*. Retrieved on 16 November from http://www.retrocomputing.net/info/allan/eBook12.pdf

Amelung, M., Piotrowski, M., Rösner, D. (2006). EduComponents: experiences in e-assessment in computer science education. ACM SIGCSE Bulletin. *Proceedings of the 11th annual SIGCSE conference on Innovation and technology in computer science education ITICSE '06*, 38(3). ACM Press.

Bandura, A. (2001). SOCIAL COGNITIVE THEORY: An Agentic Perspective. Annual *Review of Psychology*, 52(1), 1-26.

Bandura, A. (1986). *Social Foundations of Thought and Action: A Social-Cognitive Theory*. Englewood Cliffs, NJ: Prentice Hall.

Bartam, D. (2006). The internationalisation of Testing and New Models of Test Delivery on the Internet. *International Journal of Testing*, 6(2), p121-131.

Boyle, A. (2007). *The Formative Use of e-Assessment: Some Early Implementations, and Suggestions for How We Might Move On*. Qualifications and Curriculum Authority (QCA).

Brave, S., & Nass, C. (2002). Emotion in Human-Computer Interaction. *The human-computer interaction handbook: fundamentals, evolving technologies and emerging applications.* Lawrence Erlbaum Associates, Inc. Mahwah, NJ, USA.

Bright, K. (2007). *Giving students effective feedback, The Higher Education Academy*, Retrieved 3 October 2007 from
http://www.ukcle.ac.uk/resources/assessment/effectivefeedback.html

Brown, G., Bull, J., Pendlebury, M. (1997). *Assessing Student Learning in Higher Education*. Routledge.

Bundaberg Curriculum Exchange (2007). Retrieved 26 June 2007 from
http://www.teachers.ash.org.au/bce/obe.htm

Carrol, J. M. (2002). *Human-Computer Interaction in the New Millennium*. ACM Press, New York, New York.

Danino, N. (2001). *Human-Computer Interaction and Your Site*. SitePoint. Retrieved 5 September 2007 from http://www.sitepoint.com/article/computer-interaction-site

Davis, D., & Sorrell, J. (1995). *Mastery learning in public schools. Paper prepared for PSY 702: Conditions of Learning*. Valdosta, GA: Valdosta State University. Retrieved 18 September 2007 from: http://chiron.valdosta.edu/whuitt/files/mastlear.html

Davis, F.D., & Yi, M.Y. (2004). Improving Computer Skill Training: Behavior Modeling, Symbolic Mental Rehearsal, and the Role of Knowledge Structures. *Journal of Applied Psychology*, 89 (3), 509-523.

DebugMode. (2007). *Wink Software Package*. Retrieved 3 May 2007 from
http://www.debugmode.com/wink/

De Vos, A.S., Fouche, C.B., Delport, C.S.L., Strydom, H. (2007). *Research at grass roots: For the social sciences and human service professions*. Van Schaik Publishers.

DiGesu, E. (2007). *Purpose of Assessment*. Retrieved 18 June 2007 from http://www.gecdsb.on.ca/d&g/onlinepd/Assessment%20&%20Evaluation/Purpose.htm

Dix, A., Finlay, J., Abowd, G. D., Beale, R. (2004). *Human-Computer Interaction*. Pearson Education Limited, CM20 2JE, England, Third Edition.

Draper, S. W. (2002). *Feedback*. University of Glasgow Department of Psycology. Retrieved 25 June 2007 from http://www.psy.gla.ac.uk/~steve/feedback.html.

Edwards, S (2004). Using Software Testing to Move Students from Trial-and-Error to Reflection-in-Action. *In Proceedings of the 35th SIGCSE technical symposium on Computer science education (SIGCSE '04),* Norfolk, VA, USA, ACM Press.

Escobedo II, M. (2007). *When to use Microsoft Office, and when to use something else*. Retrived 16 November 2007 from http://www.helium.com/tm/113554/doubt-microsoft-office-widely

Faulkner, C. (1998). *The Essence of Human-Computer Interaction*. Hertfordshire, Prentice Hall.

Feng, M., Heffernan, N. T., Koedinger, K. R. (2006). E-learning & scientific applications: Addressing the testing challenge with a web-based e-assessment system that tutors as it assesses. *Proceedings of the 15th international conference on WWW '06.*

Fincher, C. (1994). "*Learning Theory and Research," in Teaching and Learning in the College Classroom*. (Feldman, K. A., & Paulson, M., (Eds)). Ashe Reader Series, Needham, MA: Ginn Press.

Flexner, S.B. (Ed.) (1993). *Random House Unabridged Dictionary*. New York: Random House.

Fogg, B.J. (2002). Motivating, Influencing and Persuading Users. *The human-computer interaction handbook: fundamentals, evolving technologies and emerging applications.* Lawrence Erlbaum Associates, Inc. Mahwah, NJ, USA.

Galitz, W. O. (2002). *The Essential Guide to User Interface Design*. Wiley Computer Publishing, Ney York.

Gerber, R. (2008). *Assessment, Evaluation and Feedback*. Retrieved 26 February 2008 from http://www.petech.ac.za/robert/assessme.htm

Geyser, H. (2004). Learning from Assesment. *Teaching and learning in higher education*. Pretoria. Van Schaik.

Gievska, S., & Sibert, J. (2005). Analyzing user activities: Using task context variables for selecting the best timing for interrupting users. *Proceedings of the 2005 joint conference on Smart objects and ambient intelligence: innovative context-aware services: usages and technologies sOc-EUSAI '05*. ACM Press.

Grudin, J. (2005). Three Faces of Human-Computer Interaction. *IEEE Annals of the History of Computing*. IEEE, Inc.

Guba, E. G. (Ed, 1990). *The Paradigm Dialog*. Sage Publications, Inc. 2455 Teller Road. Newbury Park, California.

Gupta, S. (2006). *Longitudinal Investigation of Collaborative e- Learning in an End User Training Context*. MIS department, University of Georgia, Athens, GA.

Gupta, S., & Bostrom, R. P. (2006). Session 6.1: End-user training methods: what we know, need to know. *Proceedings of the 2006 ACM SIGMIS CPR conference on computer personnel research: Forty four years of computer personnel research: achievements, challenges & the future.* SIGMIS CPR '06, ACM Press

Harlen, W. (2007). *Age Appropriate Goals and Assessment*. Retrieved 1 October 2007 from http://scienceonline.terc.edu/harlen_conversation/WH3.html

Harvard Glossary (2007). *Harvard Law Glossary of Terms*. Retrieved 16 November 2007 from http://cyber.law.harvard.edu/readinessguide/glossary.html

Hewett, Baecker, Card, Carey, Gasen, Mantei, Perlman, Strong and Verplank. (1992, 1996). *ACM SIGCHI Curricula for Human-Computer Interaction.* Retrieved 28 January 2007 from http://sigchi.org/cdg/cdg2.html

Holt, J. (2005). Assessment for Learning. *Curriculum Corporation Conference 1*, Brisbane.

HUMAN FRONTIER SCIENCE PROGRAMME (2001). *Report on a meeting held in Strasbourg.* France, November 29-30, 2001, on International Training and Support of Investigators in the Natural Sciences, Towards a New Paradigm for Education, Training and Career Paths in the Natural Sciences. Retrieved 13 June 2007 from http://www.hfsp.org/pubs/Position_Papers/FundersReport2002.pdf

Iqbal, S. T., & Bailey, B. P. (2006). Using knowledge to predict and manage: Leveraging characteristics of task structure to predict the cost of interruption. *Proceedings of the SIGCHI conference on Human Factors in computing systems,* CHI '06. ACM Press.

ISO 9241-11 (1998). *Guidance for Usability, International Standards Organisation.* Retrieved 19 September 2007 from http://www.idemployee.id.tue.nl/g.w.m.rauterberg/lecturenotes/ISO9241part11.pdf

Jonassen, D. H. (1991). Evaluating constructivistic learning. *Educational Technology*, 31, 28-33.

Jonassen, D.H. (2001). Technology as Cognitive Tools: Learners as Designers. *ITForum Paper #1*. Retrieved 21 June 2007 from http://itech1.coe.uga.edu/itforum/paper1/paper1.html

Jonassen, D.H. (2004). Construction as Paradigm for Teaching and Learning Concept to Classroom. *Educational Broadcasting Corporation Workshop*. Retrieved 30 June 2007 from http://www.thirteen.org/edonline/concept2class/constructivism/explor_sub5.html

Leach, R. A. (2004). *The Chiropractic Theories: A Textbook of Scientific Research*. Lippincott Williams & Wilkins.

Macdonald, J. (2002). Developing competent e-learners: the role of assessment. *Learning Communities and Assessment Cultures Conference*. University of Northumbria, Newcastle.

MacFarlane & McKellar. (2007). *Useful Pointers for Feedback*. The Higher Education Academy. Retrieved 3 October 2007 from http://www.ukcle.ac.uk/resources/trns/mcqs/eleven.html

Mahapatra, R., & Lai, V. S. (2005). Evaluating end-user training programmes. *Communications of the ACM*, 48 (1). ACM Press.

Marshall, L. (2004). Technological Tools: The Need to Situate Software Skills in the Implementation of Design Concepts, *Fourth IEEE International Conference on Advanced Learning Technologies (ICALT'04)*, IEEE Inc.

Mueller, J. (2007). *What is Authentic Assessment, Authentic Assessment Toolbox*. Retrieved 28 June 2007 from
http://jonathan.mueller.faculty.noctrl.edu/toolbox/whatisit.htm

Muijs & Reynolds (2005). *Effective Teaching: Evidence and Practice*. Sage Publications.

Munduca, C., Savina, M., Merritts, D. (2007). *Starting Point – Teaching Entry Level Geoscience*. Retrieved 29 January 2007 from
http://serc.carleton.edu/introgeo/assessment/glossary.html

Murphy, E. (1997). *Characteristics of Constructivist Learning & Teaching*. Retrieved on 3 October 2007 from http://www.stemnet.nf.ca/~elmurphy/emurphy/cle3.html

NDT Resource Centre (2007). *Teaching Resources*. Retrieved 17 June 2007 from http://www.ndt-ed.org/TeachingResources/ClassroomTips/Learning_Styles.htm

Nielsen, J. (2003). *Usability 101: Introduction to Usability.* Retrieved 21 March 2008 from http://www.useit.com/alertbox/20030825.html

Pearson Education Inc. (2007). *Authentic Assessment Overview*. TeacherVision. Retrieved 28 June from http://www.teachervision.fen.com/teaching-methods-and-management/educational-testing/4911.html?page=2&detoured=1

Pollson, K. (2006). *Chronology of Personal Computer Software*. Retrieved 6 February 2007 from http://www.islandnet.com/~kpolsson/compsoft/soft1983.htm

Preece, J., Rogers, Y., Sharp, H., Benyon, D., Holland, S., Carey, T. (1994). *Human-Computer Interaction*. Addison-Wesley.

Proctor, R. W., & Vu, Kim-Phuong L. (2002). *The human-computer interaction handbook: fundamentals, evolving technologies and emerging applications*. Lawrence Erlbaum Associates, Inc. Mahwah, NJ, USA.

111

QCA (2007). *Regulatory Principles for E-assessment*. 83 Piccadilly, London. Retrieved 17 July 2007 from http://www.qca.org.uk/libraryAssets/media/Final_regulatory_principles_document_-_PRINTED.pdf.

QMSS (2007). Hypothesis Testing. *QMSS e-Lessons*. Columbia Center for New Media Teaching and Learning. Retrieved 17 July 2007 from http://ccnmtl.columbia.edu/projects/qmss/ht_about.html

Ryan, S. D., Bordoloi, B., Harrison, D. A. (2006). *Acquiring Conceptual Data Modeling Skills: The Effect of Cooperative Learning and Self-Efficacy on Learning Outcomes.* ACM Press.

Simon, S. J. (2000). The Relationship of Learning Style and Training Method to End-User Computer Satisfaction and Computer Use: A Structural Equation Model. *Information Technology, Learning, and Performance Journal*, 18(1).

Skegg, D., & Jones, G. (2007). *Key Points for Good Feedback*. University of Otago Quality Advancement. Retrieved 2 October 2007 from http://www.otago.ac.nz/quality/feedbackwp/key.html

Smith, S. M. (2004). Software Skills Acquisition: Confidence vs. Competence. [electronic version]. *Information Technology, Learning, and Performance Journal*. 22(2).

Stenning, K., McKendree, J., Lee, J., Cox, R., Dineen, F., Mayes, T. (1999). Vicarious learning from educational dialogue. *Proceedings of the 1999 conference on Computer support for collaborative learning CSCL '99*, International Society of the Learning Sciences.

Suleman, S. (2003). *E-assessment*. Ferl. Retrieved 23 June 2007 from http://ferl.qa.org.uk/display.cfm?resID=5367

Tohidi, M., Buxton, W., Baecker, R., Sellen, A. (2006). *Proceedings of the SIGCHI conference on Human Factors in computing systems CHI '06*. ACM Press.

Van Joolingen, W. (1999). Cognitive tools for discovery learning. *International Journal of Artificial Intelligence in Education, 10*. Retrieved 18 September 2007 from http://aied.inf.ed.ac.uk/members99/archive/vol_10/joolingen/paper.pdf

Watzman, S. (2002). Visual Design Principles for Usable Interfaces. *The human-computer interaction handbook: fundamentals, evolving technologies and emerging applications*. Lawrence Erlbaum Associates, Inc. Mahwah, NJ, USA.

Wiedenbeck, S., Zila, P. L., McConnell, D. S. (1995). *End-user training: an empirical study comparing on-line practice methods*. ACM Press.

Wiggins, G. (1998). Tips for teaching: Feedback - How learning occurs. *Centre for teaching excellence*. Retrieved 25 June 2007 from http://www.dean.usma.edu/cte/TipFeedback.cfm

Yi, M.Y., & Davis, F.D. (2003). Developing and Validating an Observational Learning Model of Computer Software Training and Skill Acquisition. *Information Systems Research*, 14 (2), 146-170.

Appendix A

The WordAssessor questions

A.1. Introduction

The comparative study for this software skills test uses 25 unique questions to test students' word processing knowledge. The design of the 25 questions was in line with the QCA's (2007) recommendation of minimizing complexity (see section 3.2.2) and consisted of realistic real-world task scenarios. The methods involved in answering each question, involves utilising different objects, methods and strategies within the MS Word environment in order to obtain the correct answers.

As a result, WordAssessor's marking mechanism for each of the 25 questions had to be designed, programmed and implemented separately. The questions, their solutions and a small portion of the marking mechanism necessary for each question are described on the following pages.

Even though this skills testing tool was developed specifically for this study, there exists a very real possibility that the program can be expanded to suit commercial use. In that case, should lecturers of a software skills course like to set up a test, they can choose from a wide variety of pre-programmed questions to suit their needs.

A.2. The Questions

A.2.1. Question 1

This question deals with the Header and Footer object. It asks the user to add a header to the current document:

Question: **Use the View menu option to create a header containing the word NEWS. Close the Header section when done.**

Solution

The conventional way of accomplishing the above task is the following:

1. Click on the View menu in the current MS Word window and click on the Header and Footer Option.
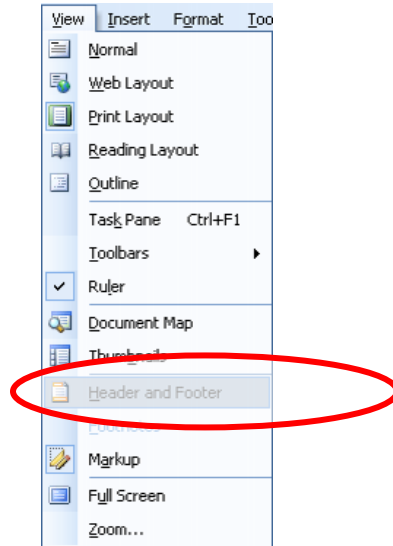
2. Click in the Header field and type "NEWS".
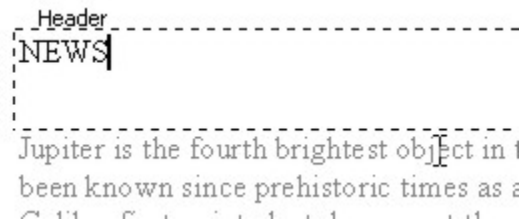
3. Close the Header section by clicking on the "Close" button on the Header and Footer toolbar.

WordAssessor Marking Procedure

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button.

WordAssessor then has to go through the following process (all done in the background, invisible to the user):

- Find the current user's student number in the Student Results data table and select that row to edit.
- Take the time variable (measured in seconds) to see how long the user has worked on the current question and write it into the selected data row.
- Reset the time variable, so it could be used in the next question.
- Start the timer mechanism for the next question.
- Set the current MS Word window's view to the print view setting.
- Open the primary header of the document.
- Set the selection to the start of the header field and extend it to the end of the field so that everything in the Header field is selected.
- Check to see if the selected string contains the word "NEWS".

- **If the word "NEWS" is present**, do the following:
  - Reset the current window view to the normal view.
  - Check the student results data table to see if the user has previously submitted a correct answer for this question. If not, do the following:
    - Increase the total score for the test by one.
  - Update the "Question 1 Result" value in the student results data table with the value "Correct".
- **If the word "NEWS" is not present** in the Header field, do the following:
  - Reset the current window view to the normal view.

   o Update the "Question 1 Result" value in the student results data table with the value "Incorrect".

- If the current question hasn't been answered before, increase the variable of the total number of questions answered by one.
- If the question hasn't been answered before, place a checkmark next to question 1 on the questions form.
- Set the default text for the current document window, in case the user has made incorrect changes to the body of the text.

Note: The above marking procedure has similar elements for each question. For the marking-procedure descriptions of the following questions, only the elements that are unique to that specific question will be described (in order to minimise repetition). The generic elements above, that will be excluded in questions 2-25 have been marked in a blue font.

A.2.2. Question 2

This question deals with the Insert Date Function:

*Question:* **Insert the current date (in the form 11/23/2001) into this document. The date should update automatically.**

Solution

The correct solution is obtained by completing the following steps:

1. Click on the Insert menu in the current window's main menu bar and click on the Date and Time menu item.
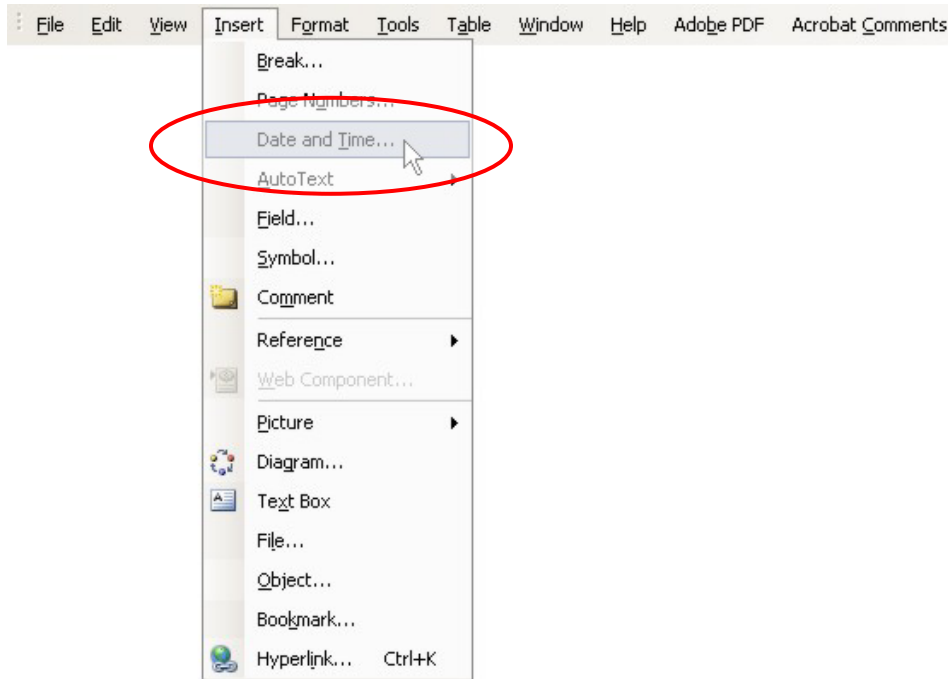
2. In the Date and Time window that appears, select the first option (i.e. the date in the format referred to in the question (MM/DD/YYYY) and check the box that says Update Automatically and click OK.
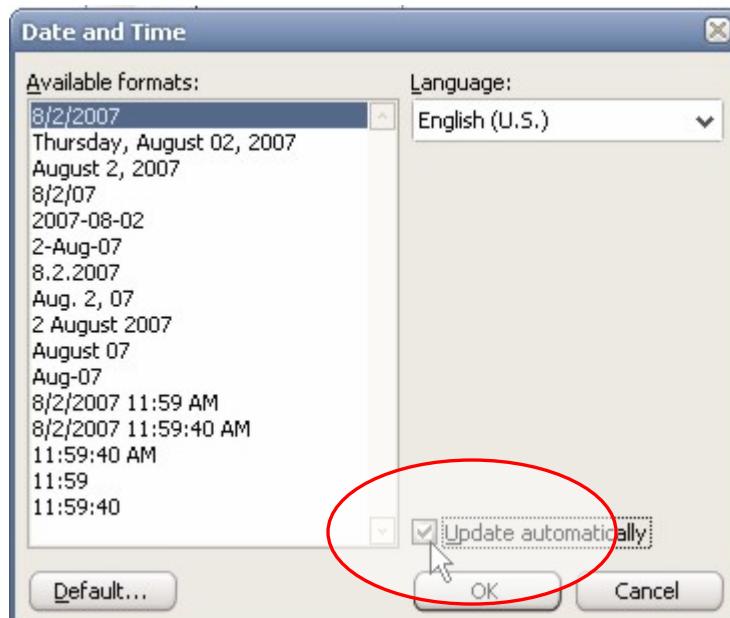
WordAssessor Marking Procedure

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button.

WordAssessor then goes through the following process (all done in the background, invisible to the user):

- Process similar to question 1.
- Select the entire contents of the current MS Word window.
- Check the first field in the current document.

- **If the field exists**, and contains the current date in the form (MM/DD/YYYY), do the following:
    o Delete the field.
    o Check the student results data table to see if the user has already submitted a correct answer. If not, do the following:
        ▪ Increase the total score for the test by one.
    o Update the "Question 2 Result" value in the student results data table with the value "Correct".
- **If the field does not exist**, or if it exists and does not contain the current date in the form (MM/DD/YYYY), do the following:
    o Update the "Question 2 Result" value in the student results data table with the value "Incorrect".
- Reset the current window's selection to empty.
- Process similar to question 1.

A.2.3. Question 3

This question deals with the Replace function, whereby the user must replace certain occurrences of text with other text:

Question: **Replace all occurrences of the word WE with the word YOU.**

Solution

The correct solution is obtained by completing the following steps:

1. Click on the Edit menu of the current window and click on the Replace menu item (or simply press Ctrl+H).
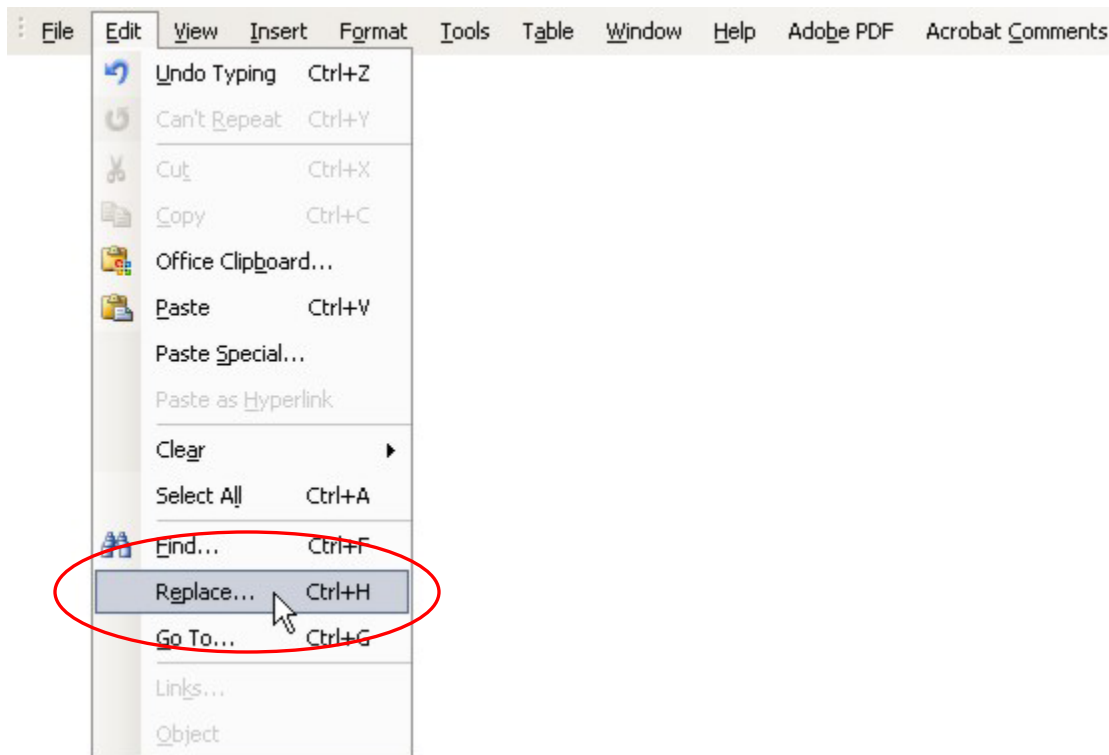
2. When the Find and Replace dialog window appears, type the word "WE" in the top edit box and the word "YOU" in the bottom edit box and click Replace All.
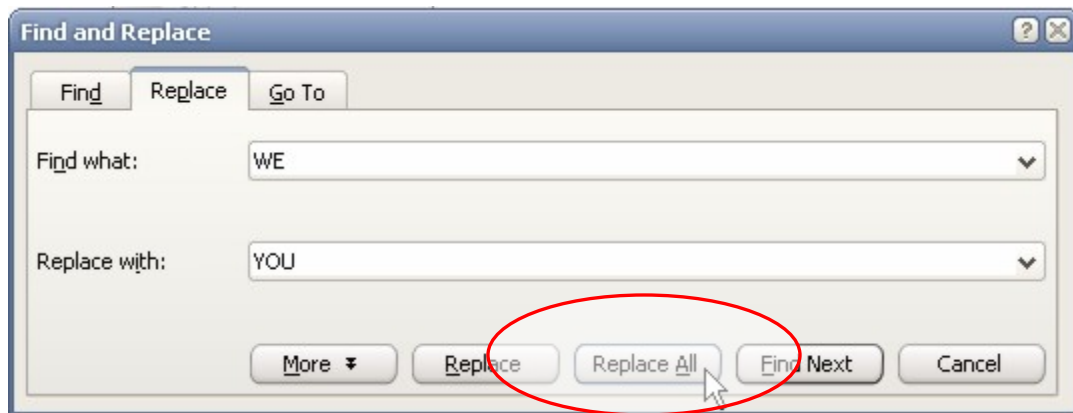
WordAssessor Marking Procedure

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button.

WordAssessor then goes through the following process (all done in the background, invisible to the user):

- Process similar to question 1.
- Select the entire contents of the current window.
- Assign the current selection to a string variable and search the string as follows:
  - If the string contains the word "YOU" seven times (there were originally seven occurrences of the word "WE" in the document) and the word "WE" zero times, then the question is correct.

- **If the question is indeed correct**, do the following:
  - Check the student results data table to see if the user has already submitted a correct answer. If not, do the following:
    - Increase the total score for the test by one.
    - Update the "Question 3 Result" value in the student results data table with the value "Correct".
- **If the question is not correct**, do the following:

       o Update the "Question 3 Result" value in the student results data table with the value "Incorrect".

- Process similar to question 1.
- Set the selection for the next question so that no text is selected.

A.2.4. Question 4

This question deals with typing or inserting regular text into an open document:

Question: **Insert the text HENLEY after MR. JAMES. Press the ENTER key after typing HENLEY.**

Solution

The correct solution is obtained by completing the following steps:

1. Locate the phrase "MR. JAMES" in the current MS Word window.

2. Type or insert the phrase "HENLEY" immediately after this occurrence.

WordAssessor Marking Procedure

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button.

WordAssessor then goes through the following process (all done in the background, invisible to the user):

- Process similar to question 1.
- If the third paragraph of the current document **contains the phrase "MR. JAMES HENLEY"**, do the following:
  - o Check the student results data table to see if the user has already submitted a correct answer. If not, do the following:
    - ▪ Increase the total score for the test by one.
    - ▪ Update the "Question 4 Result" value in the student results data table with the value "Correct".
- If the third paragraph **does not contain the phrase "MR. JAMES HENLEY"**, do the following:
  - o Update the "Question 4 Result" value in the student results data table with the value "Incorrect".

- Process similar to question 1.
- Set the selection for Question 5 so that no text is selected.

A.2.5. Question 5

This question deals with the view, or zoom function:

Question: **Change the display of the document to 75% of its original size.**

Solution

The correct solution is obtained by completing the following steps:

1.   Click on the View menu and click on the Zoom menu item.
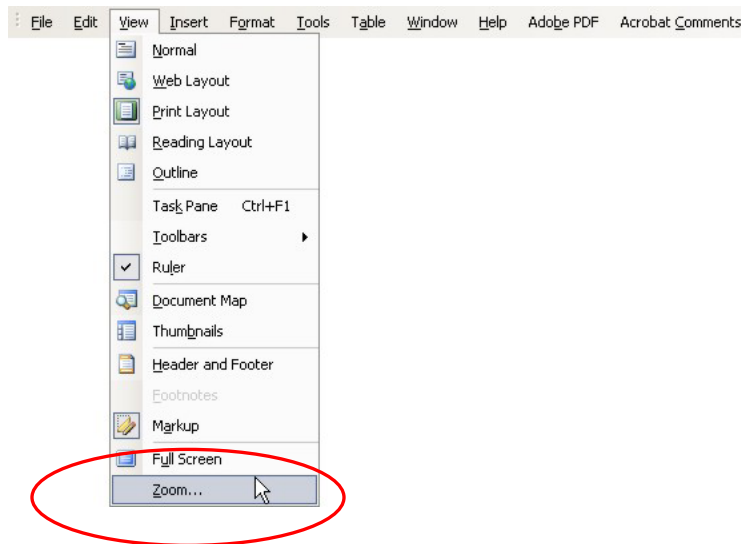
2.   In the window that opens up, click on the 75% option.
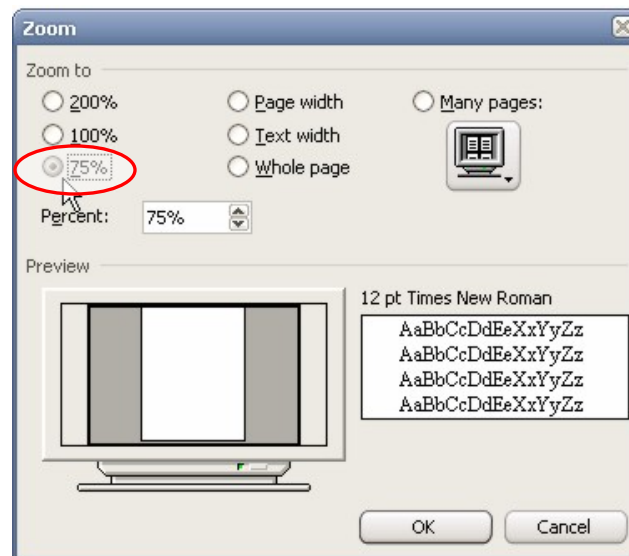
Another way of accomplishing this is just to select the 75% option from the zoom drop-down list on the main toolbar.
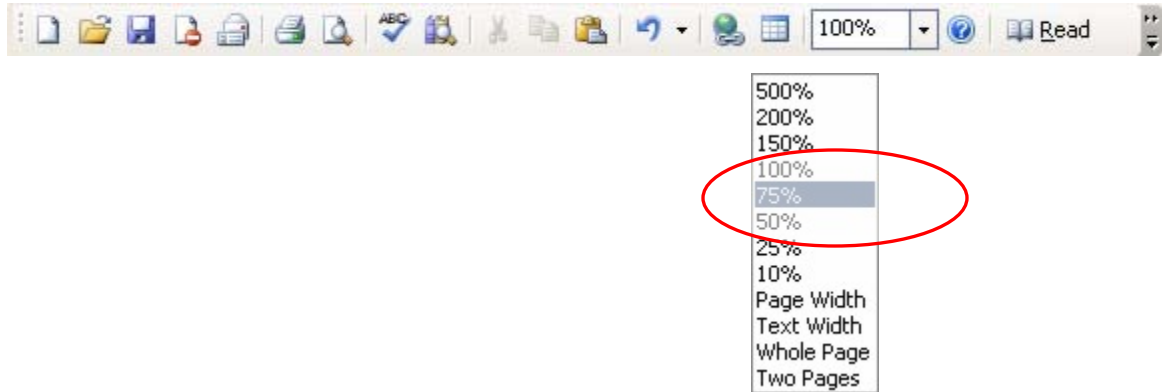
Either way is accepted by the WordAssessor marking procedure, as it only checks the end-result.

WordAssessor Marking Procedure

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button. WordAssessor then goes through the following process (all done in the background, invisible to the user):

- Process similar to question 1.
- If the zoom percentage of the current document's view **is set to 75%**, do the following:

    o Check the student results data table to see if the user has already submitted a correct answer. If not, do the following:
        - Increase the total score for the test by one.
        - Update the "Question 5 Result" value in the student results data table with the value "Correct".

- If the zoom percentage of the current document's view **is not set to 75%**, do the following:

o   Update the "Question 5 Result" value in the student results data table with the value "Incorrect".

- Process similar to question 1.
- Reset the view for the current window to 100% zoom level.

___

A.2.6. Question 6

This question deals with the "Insert Picture from File" function.

Question: **Insert the picture named LOGO into this document. LOGO is not in the clip art gallery.**

Solution

The conventional way of accomplishing the above task is the following:

1. Click on the Insert menu.
2. Select the Picture side-menu and click on From File:
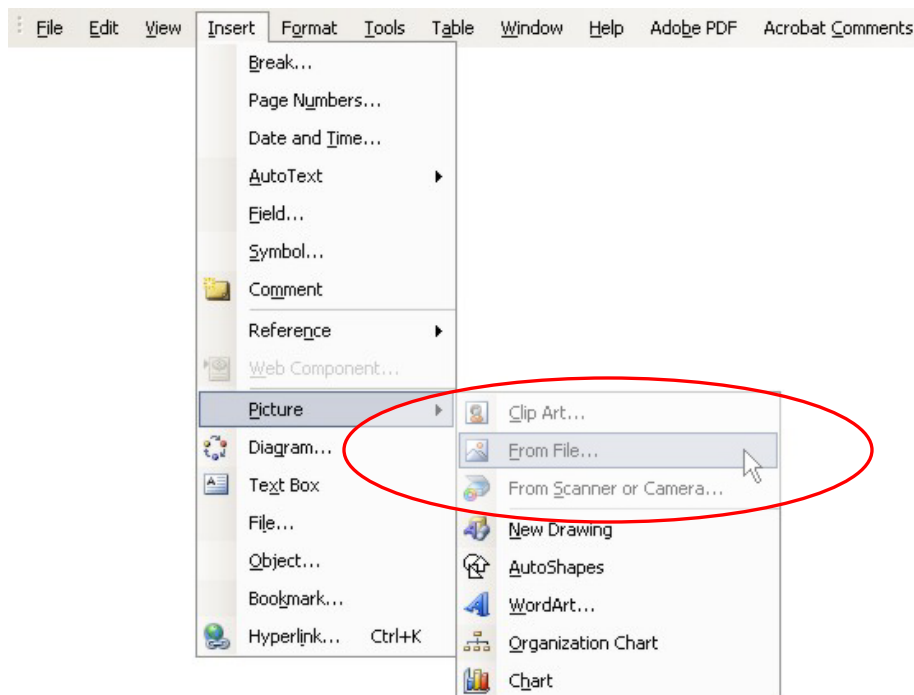
3. From the window that is opened, select the LOGO.jpg file and click on the Insert button:
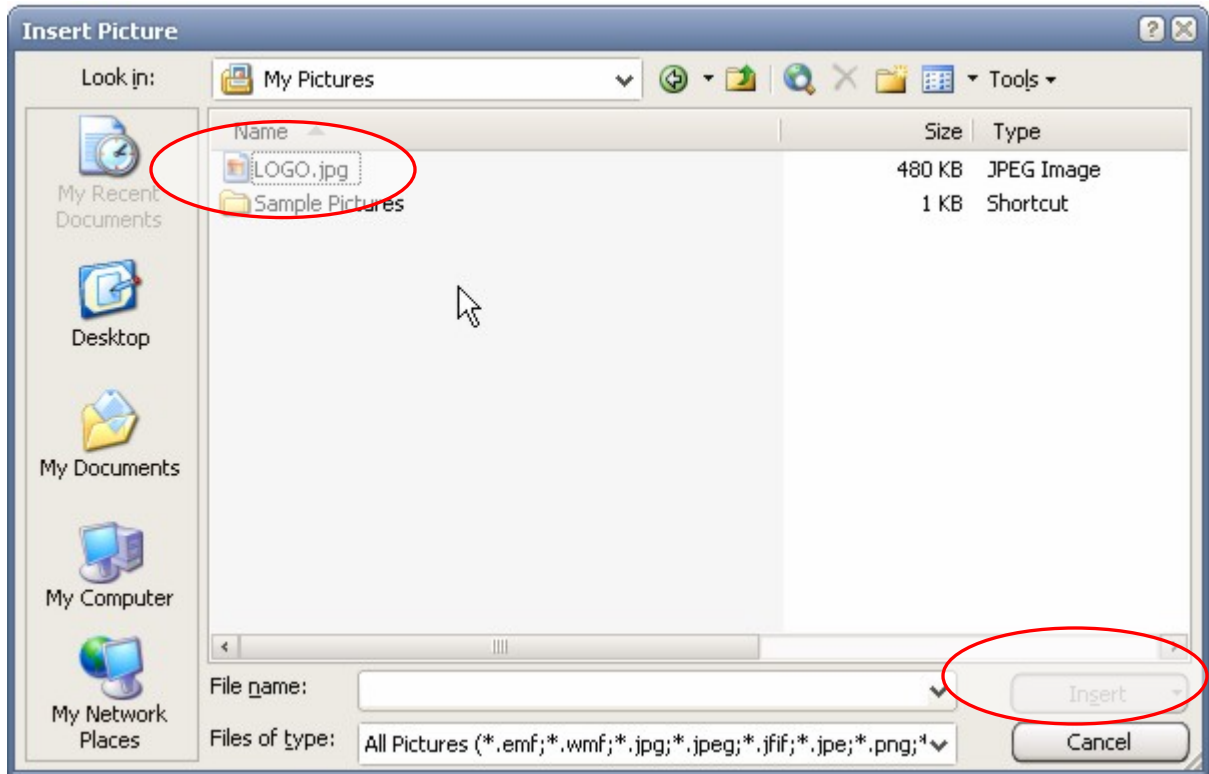
<u>WordAssessor Marking Procedure</u>

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button.

WordAssessor then goes through the following process (all done in the background, invisible to the user):

- Process similar to question 1.
- Select the entire contents of the current document window.

- If the selection **contains an inline shape object with a scale width of 37.5 units** (the properties of the LOGO.jpg image), do the following:
  - Check the student results data table to see if the user has already submitted a correct answer. If not, do the following:
    - Increase the total score for the test by one.
    - Update the "Question 6 Result" value in the student results data table with the value "Correct".

- If the selection **does not contain an inline shape object with a scale width of 37.5 units** (the properties of the LOGO.jpg image), do the following:
    - Update the "Question 6 Result" value in the student results data table with the value "Incorrect".

- Process similar to question 1.

- Select the phrase "**it has been known since**" in the default body of text by starting the current selection from character position 84 (of the default body of text) and ending the selection 5 words later (MS Word's word object is used). This is in preparation of Question 7.

A.2.7. Question 7

This question deals with the Delete function:

Question: **Delete the selected text.**

Solution

The correct solution is obtained by completing the following steps:

1. Press the delete key on the keyboard to delete the text that has been pre-selected:

WordAssessor Marking Procedure

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button.

WordAssessor then goes through the following process (all done in the background, invisible to the user):

- Process similar to question 1.
- Select the entire contents of the current document window.

- If the selection **does not contain the phrase "it has been known since**, do the following:
    - o Check the student results data table to see if the user has already submitted a correct answer. If not, do the following:
        - Increase the total score for the test by one.
        - Update the "Question 7 Result" value in the student results data table with the value "Correct".
- If the selection **does contain the phrase "it has been known since**, do the following:
        - Update the "Question 7 Result" value in the student results data table with the value "Incorrect".

- Process similar to question 1.

- Select the phrase "**wandering star**" in the default body of text by starting the current selection from character position 139 (of the default body of text) and ending the selection 2 words later (MS Word's word object is used). This is in preparation of Question 8.

A.2.8. Question 8

This question deals with the Bold text-formatting function:

Question: **Bold the selected text.**

Solution

The text is pre-selected as shown below:
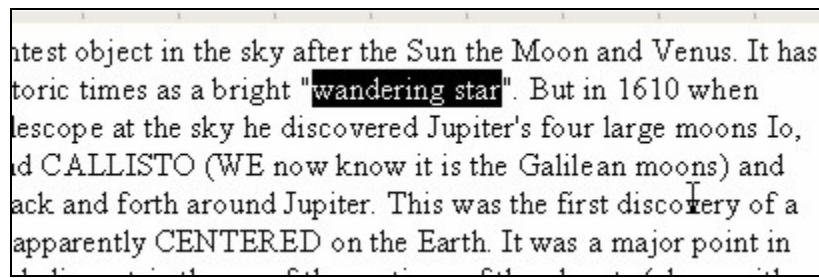
The correct solution is obtained by completing the following steps:

1. Click on the "Bold" icon in the Formatting Toolbar.
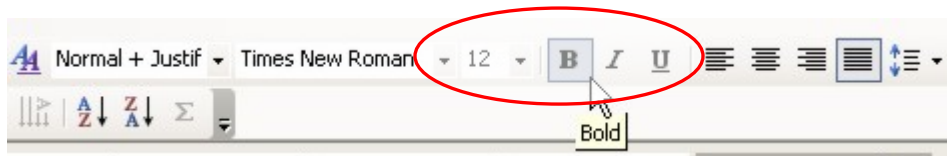
OR

1. Click on the Format Menu and click on the Font menu-item.
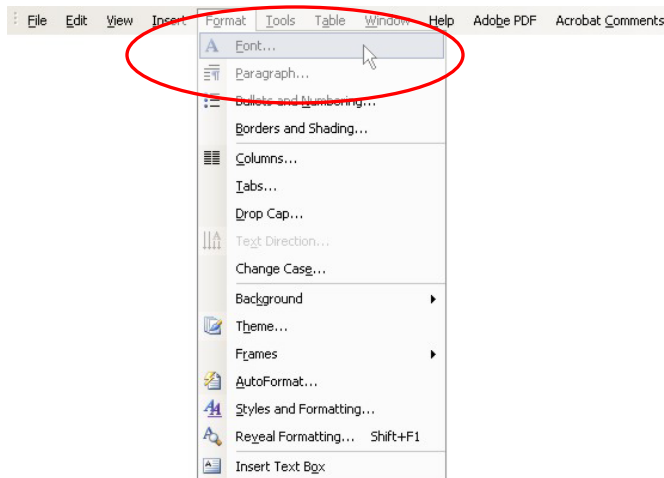
2. In the window that appears, select the "Bold" item in the Font style list:
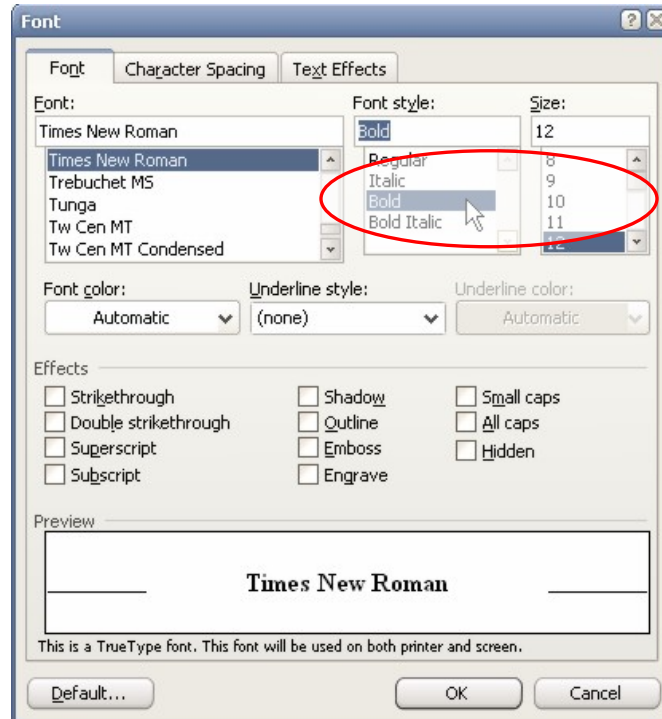
3. Click OK.

WordAssessor Marking Procedure

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button.

WordAssessor then goes through the following process (all done in the background, invisible to the user):

- Process similar to question 1.

- If the phrase "wandering star" in the first paragraph of the document **is formatted "Bold"**, do the following:
  - Check the student results data table to see if the user has already submitted a correct answer. If not, do the following:
    - Increase the total score for the test by one.
    - Update the "Question 8 Result" value in the student results data table with the value "Correct".
- If the phrase "wandering star" in the first paragraph of the document **is NOT formatted "Bold"**, do the following:
    - Update the "Question 8 Result" value in the student results data table with the value "Incorrect".

- Process similar to question 1.

A.2.9. Question 9

This question deals with the Print command:

Question: **Print 5 copies of the current page.**

Note: WordAssessor intercepts the MS Word Print command so that, when the print command is given, no actual pages are printed. Instead, a message is shown stating "Printing Successful!".

Solution

The correct solution is obtained by completing the following steps:

1. Click on the File menu and then click on the Print menu item:
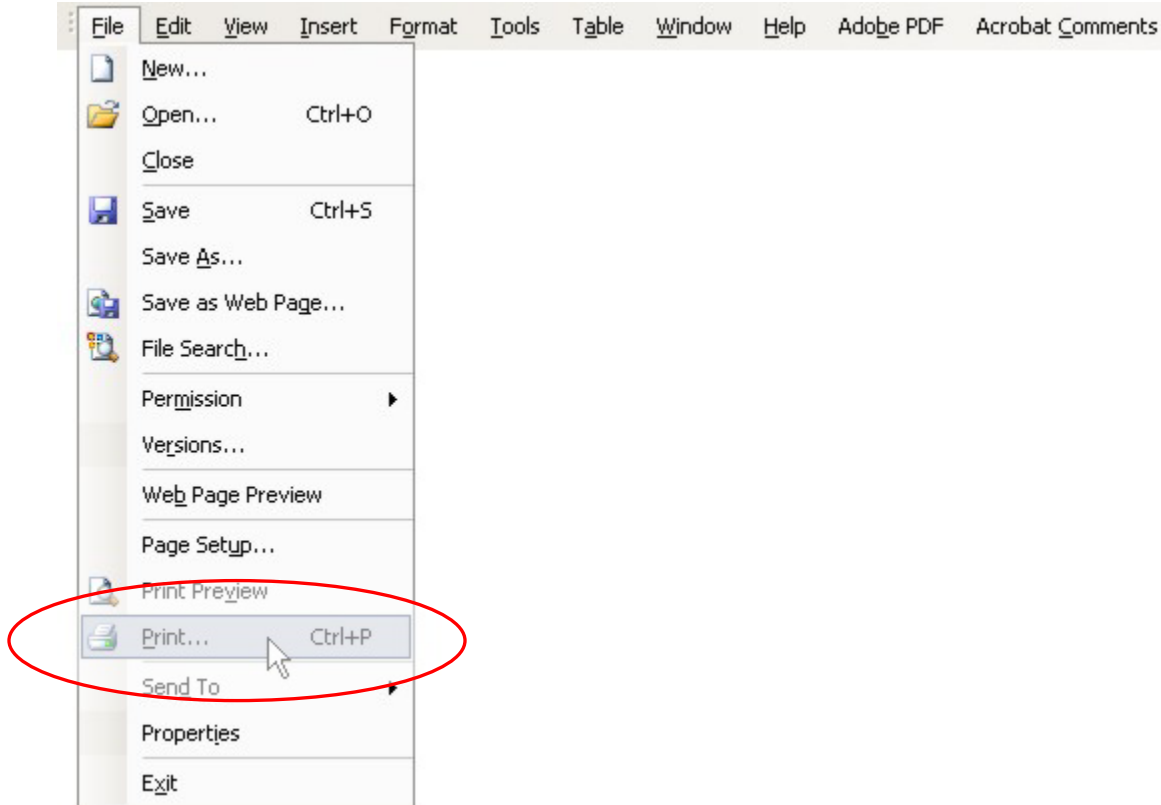
2. When the Print dialog appears, do the following:

    a. In the "Page range" group-box, select the "Current page" option.

    b. In the "Copies" group-box, enter the number 5 in the text-box that is labelled "Number of copies".

    c. Click OK

    (*An image of the dialog is displayed below*)
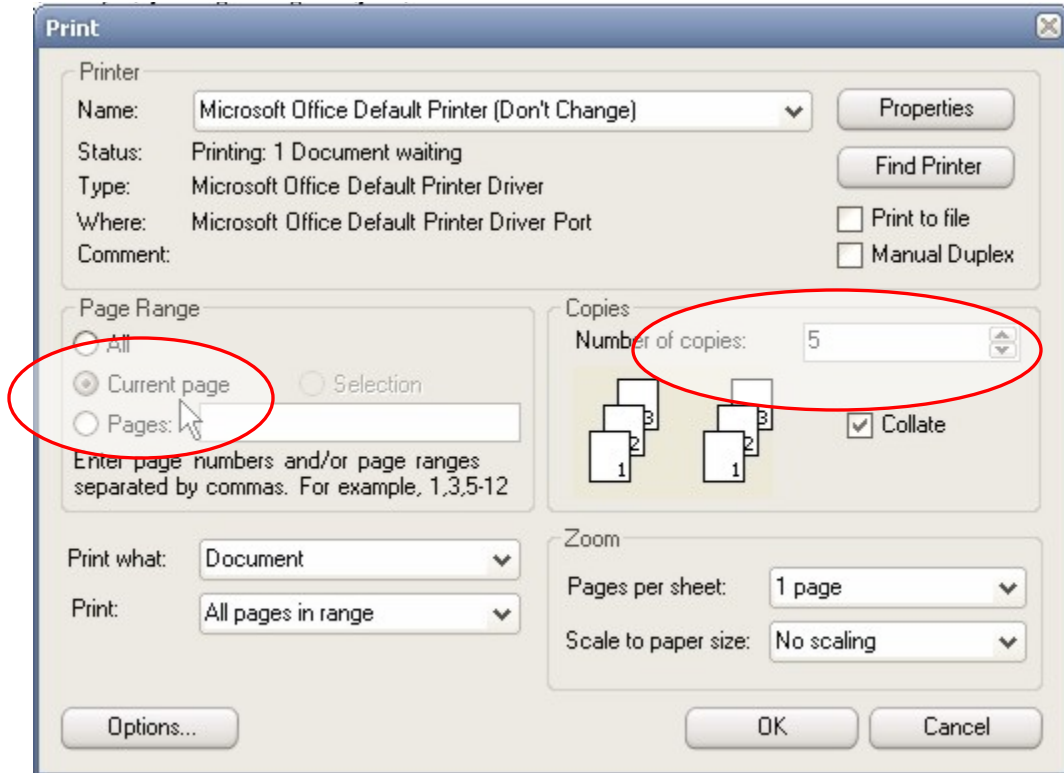
WordAssessor Marking Procedure

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button.

WordAssessor then goes through the following process (all done in the background, invisible to the user):

- Process similar to question 1.

At this point, it is important to note what happens when the student opens the print dialog and correctly selects the "Current Page" and "5 Copies". If the student successfully selected these parameters and clicked on the OK button in the print dialog window, WordAssessor automatically writes a text configuration file (Question9.txt) to the local hard drive, to indicate that the question has been correctly answered.

- **If the text configuration file "Question9.txt" is present** in the preset directory after the student has submitted the answer, do the following:

  - Check the student results data table to see if the user has already submitted a correct answer. If not, do the following:

    - Increase the total score for the test by one.
    - Update the "Question 9 Result" value in the student results data table with the value "Correct".

- **If the text configuration file "Question9.txt" is NOT present** in the preset directory after the student has submitted the answer, do the following:

    - Update the "Question 9 Result" value in the student results data table with the value "Incorrect".

- Process similar to question 1.
- Select the phrase "Galileo first pointed" in the first paragrpah of the current document in preparation of Question 10.

A.2.10. Question 10

This question deals with the Font text-formatting function:

Question: **Change the font for the selected text to Arial Narrow.**

## Solution

For this question, the text in the current document window has been selected as follows:
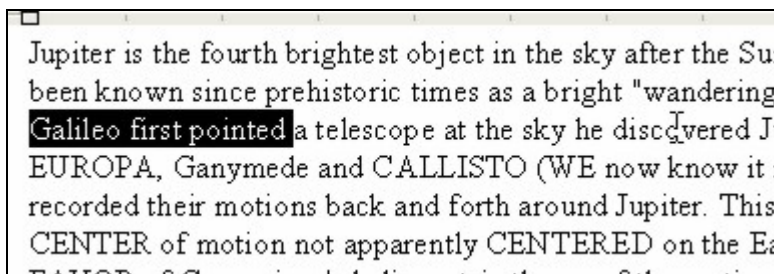
The correct solution is obtained by completing the following steps:

1. In the formatting toolbar, click on the font selection drop-down list and select the "Arial Narrow" item.
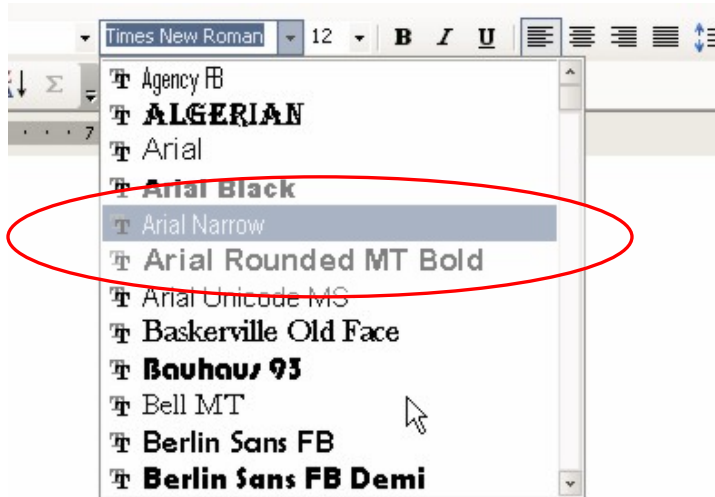
Another way of accomplishing the above solution is as follows:

1. Click on the Format menu and click on the Font menu-item.
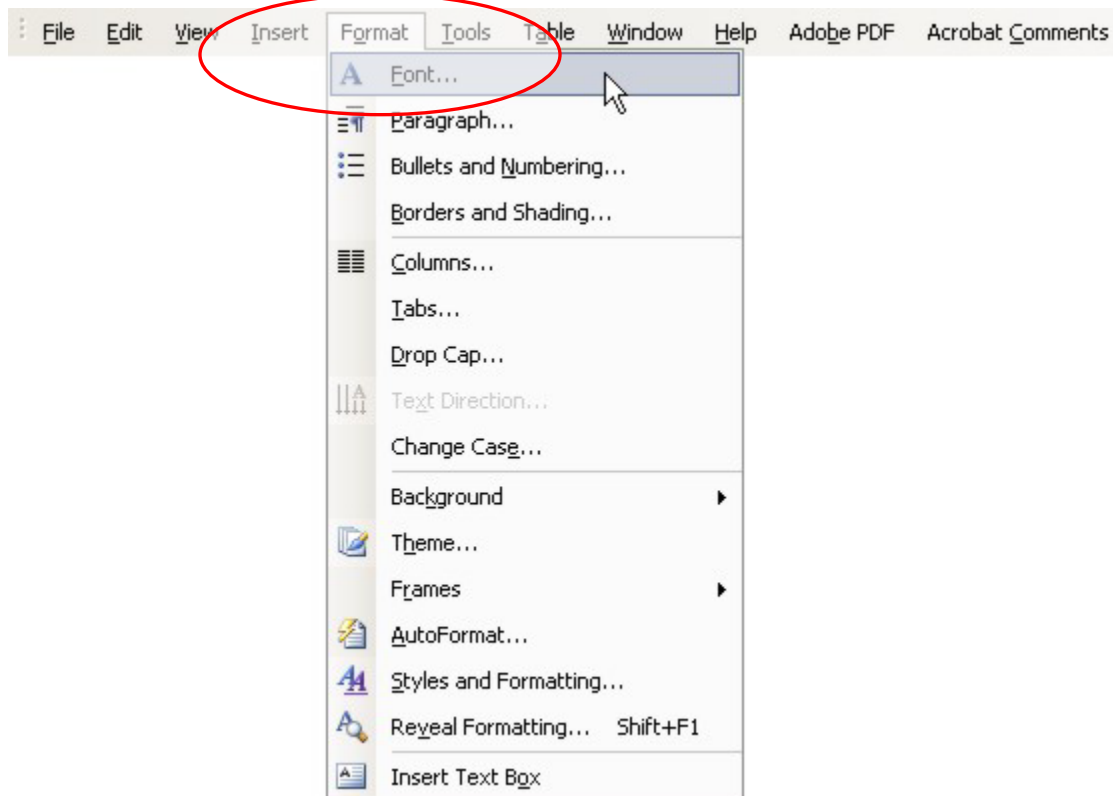
2. In the window that appears, select the "Arial Narrow" font from the Font listbox:
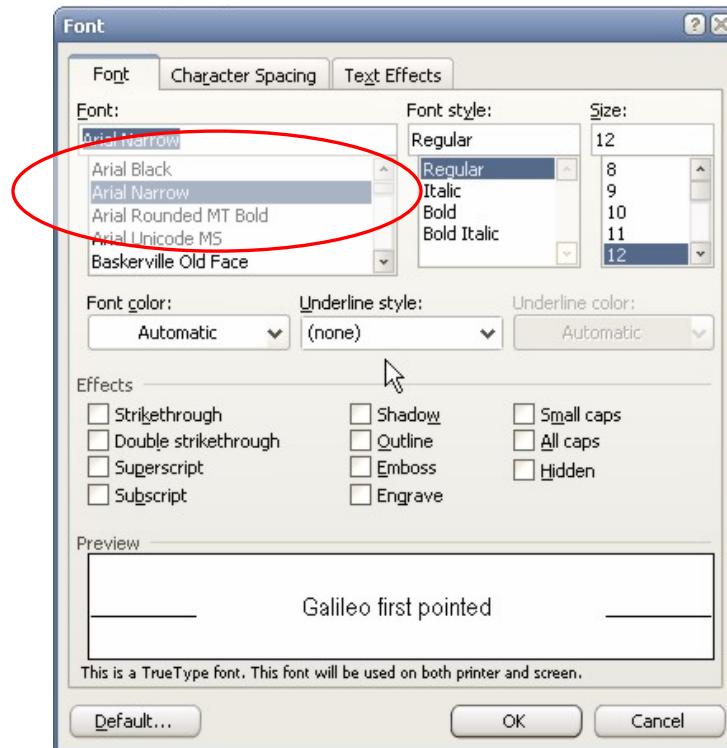
3. Click OK

WordAssessor Marking Procedure

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button.

WordAssessor then goes through the following process (all done in the background, invisible to the user):

- Process similar to question 1.
- If the phrase "Galileo first pointed" **is formatted with the "Arial Narrow" font-type**, do the following:
  - o Check the student results data table to see if the user has already submitted a correct answer. If not, do the following:
    - Increase the total score for the test by one.

- Update the "Question 10 Result" value in the student results data table with the value "Correct".
- If the phrase "Galileo first pointed" **is NOT formatted with the "Arial Narrow" font-type**, do the following:
  - Update the "Question 10 Result" value in the student results data table with the value "Incorrect".

- Process similar to question 1.
- In preparation of Question 11, select the first paragraph of the current document's text.

A.2.11. Question 11

This question deals with the Indent paragraph-formatting function:

Question: **Indent the selected paragraph by one stop.**

Solution

For this question, the text in the current document window has been pre-selected as follows:

Jupiter is the fourth brightest object in the sky after the Sun the Moon and Venus. It has been known since prehistoric times as a bright "wandering star". But in 1610 when Galileo first pointed a telescope at the sky he discovered Jupiter's four large moons Io, EUROPA, Ganymede and CALLISTO (WE now know it is the Galilean moons) and recorded their motions back and forth around Jupiter. This was the first discovery of a CENTER of motion not apparently CENTERED on the Earth. It was a major point in FAVOR of Copernicus's heliocentric theory of the motions of the planets (along with other new evidence from his telescope: the phases of Venus and the mountains on the Moon). Galileo's outspoken support of the Copernican theory got him in trouble with the Inquisition. Today anyone can repeat Galileo's observations (without fear of retribution :-) using binoculars or an inexpensive telescope. WE

MR. JAMES was the first man on the moon.

The correct solution is obtained by completing the following steps:

1. Click on the "Increase Indent" toolbar button in the Formatting toolbar:

WordAssessor Marking Procedure

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button.

WordAssessor then goes through the following process (all done in the background, invisible to the user):

- Process similar to question 1.
- **If the first paragraph is indented by one stop**, do the following:
    - Check the student results data table to see if the user has already submitted a correct answer. If not, do the following:
        - Increase the total score for the test by one.
        - Update the "Question 11 Result" value in the student results data table with the value "Correct".
- **If the first paragraph is NOT indented by one stop**, do the following:
        - Update the "Question 11 Result" value in the student results data table with the value "Incorrect".

- Process similar to question 1.
- In preparation of Question 11, select the phrase "telescope at the sky" and increase its font size to 13.

Note: To avoid becoming too convoluted, the technical particulars of selecting a phrase within an open MS Word document, by means of the program , is not discussed. This is due mainly to the fact that, in the C# Visual Studio programming environment, this particular action can require about 12 full lines of code.

A.2.12. Question 12

This question deals with applying multiple text formatting options:

Question: **Simultaneously apply multiple formatting options (Size 12, Underline, Blue colour) to the selected text.**

Solution

For this question, the text in the current document window has been pre-selected as follows:

Jupiter is the fourth brightest object in the sky after the Sun the Moon and Venus. It has been known since prehistoric times as a bright "wandering star". But in 1610 when Galileo first pointed a telescope at the sky he discovered Jupiter's four large moons Io, EUROPA, Ganymede and CALLISTO (WE now know it is the Galilean moons) and recorded their motions back and forth around Jupiter. This was the first discovery of a CENTER of motion not apparently CENTERED on the Earth. It was a major point in

The correct solution is obtained by completing the following steps:
1. In the Formatting toolbar, select the following parameters:
    a. In the font-size drop-down list, select 12.
    b. Click on the Underline icon.
    c. From the Font-Colour dialog drop-down box, select the colour Blue.
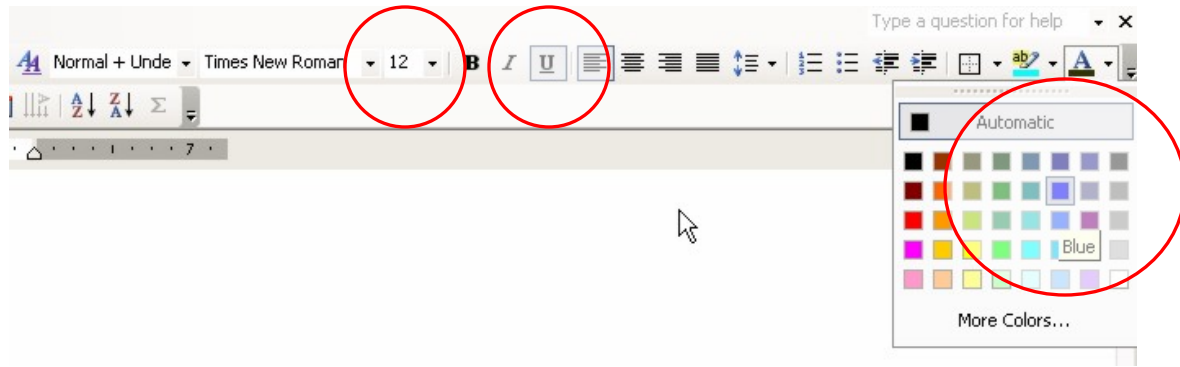
Another way of accomplishing this is as follows:

1. Click on the Format menu and then click on the Font menu-item:
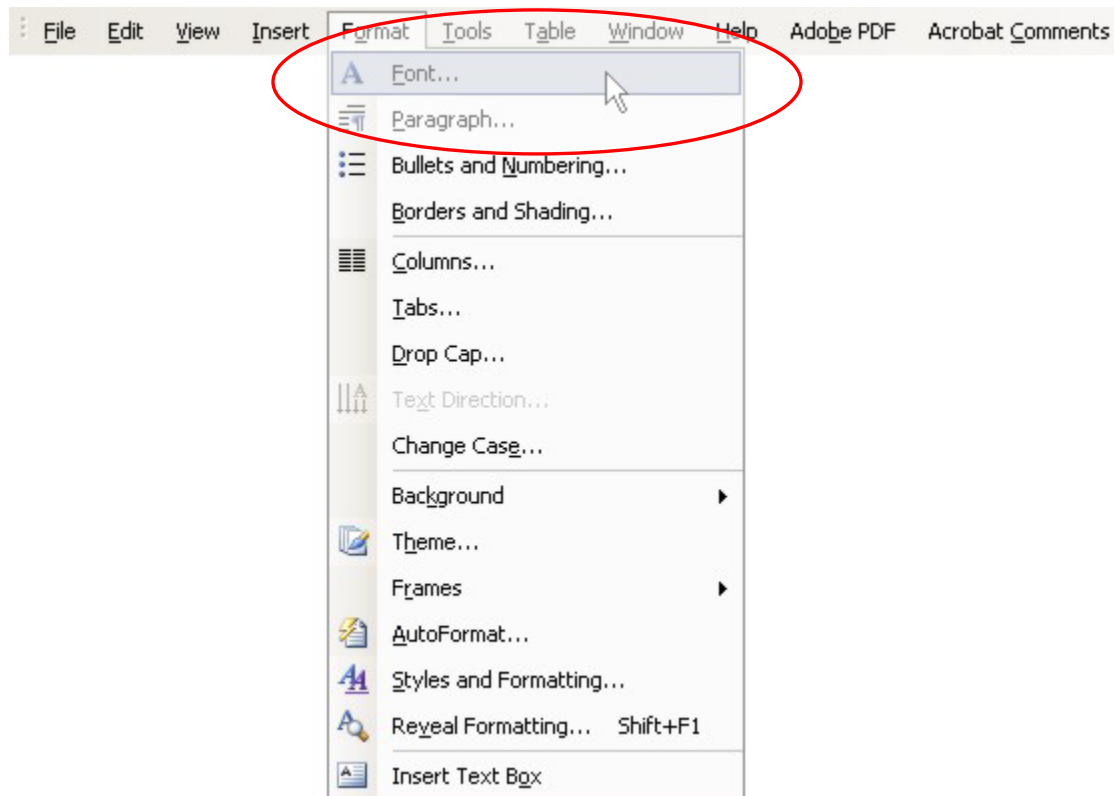
2. In the Font dialog screen that appears, select the parameters as shown in red in the image below:
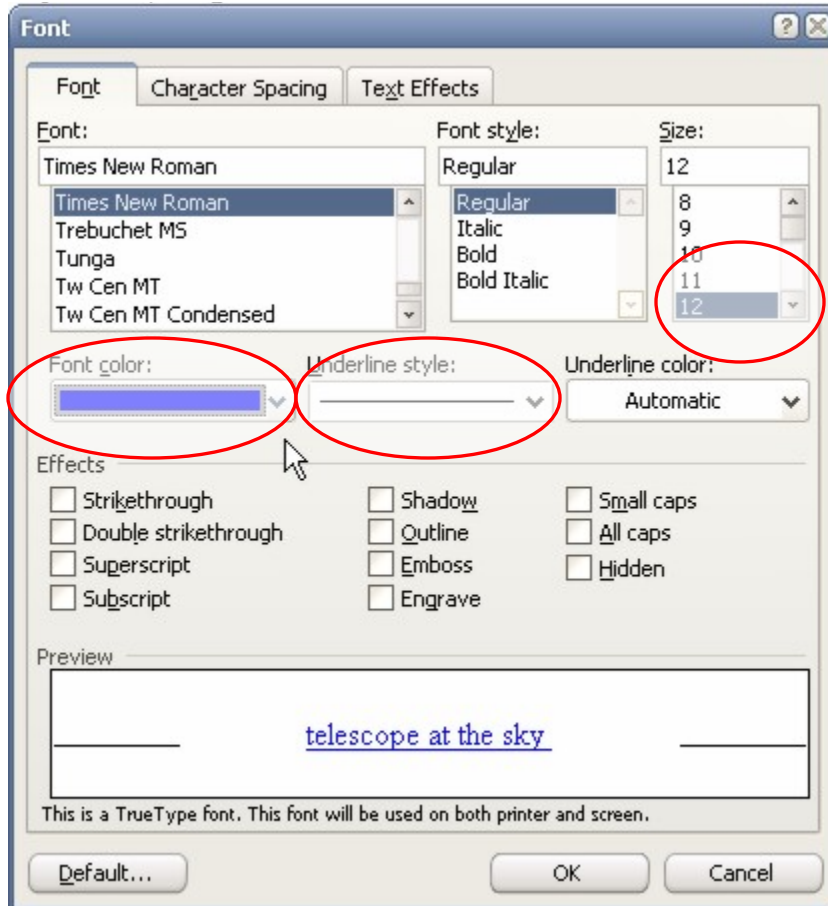
WordAssessor Marking Procedure

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button.

WordAssessor then goes through the following process (all done in the background, invisible to the user):

- Process similar to question 1.
- **If the phrase "telescope at the sky" is formatted as size 12 font, single-underlined and has font colour Blue**, do the following:
  - Check the student results data table to see if the user has already submitted a correct answer. If not, do the following:
    - Increase the total score for the test by one.

- Update the "Question 12 Result" value in the student results data table with the value "Correct".

- **If the phrase "telescope at the sky" is NOT formatted as size 12 font, single-underlined and has font colour Blue**, do the following

  - Update the "Question 12 Result" value in the student results data table with the value "Incorrect".

- Process similar to question 1.
- In preparation of Question 13, select the third paragraph of text.

A.2.13. Question 13

This question deals with applying the bullet-type formatting option:

Question: **Insert the check mark bullet in front of the selected text.**

## Solution

For this question, the text in the current document window has been pre-selected as follows:

MR. JAMES was the first man on the moon.

Jupiter was first visited by Pioneer 10 in 1973 and later by Pioneer 11, Voyager 1, Voyager 2 and Ulysses. The spacecraft Galileo orbited Jupiter for eight years. It is still regularly observed by the Hubble Space Telescope.

The gas planets do not have solid surfaces; their gaseous material simply gets denser with depth (the radii and diameters quoted for the planets are for levels corresponding to a

The correct solution is obtained by completing the following steps:

1. Click on the Format menu and click on the Bullets and Numbering menu-item.
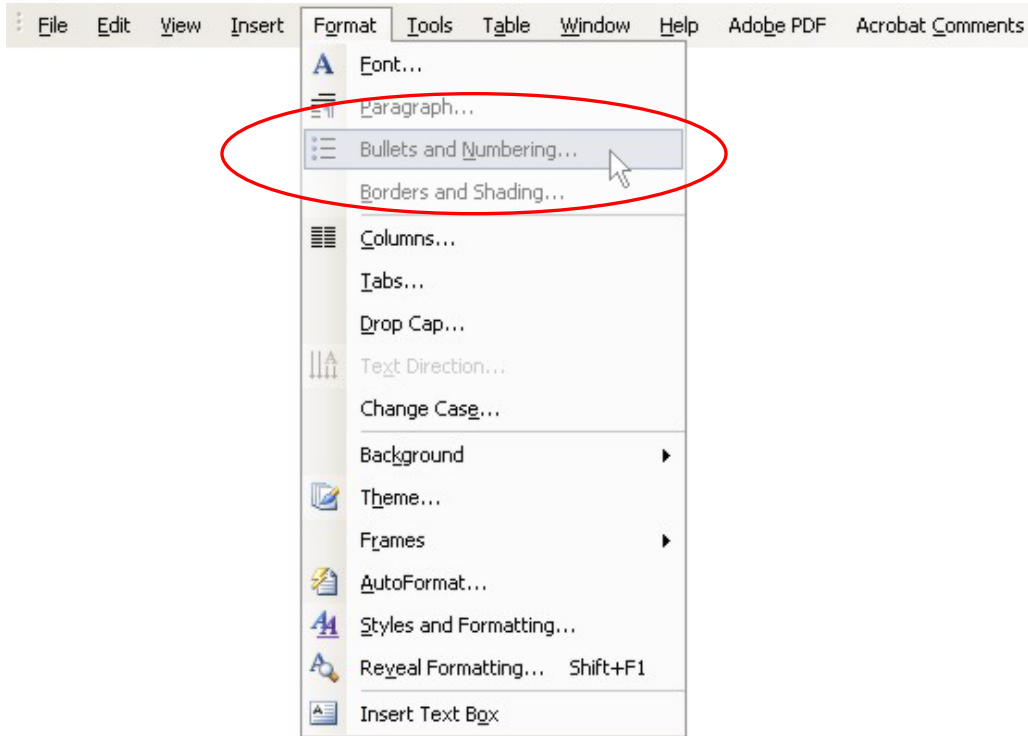
2. In the dialog that appears, select the option with the checkmark bullets and click OK.

    (See image below)
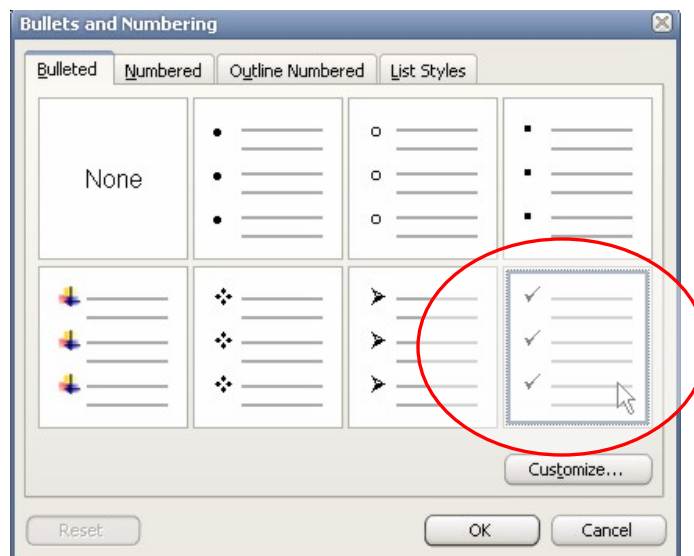
WordAssessor Marking Procedure

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button.

WordAssessor then goes through the following process (all done in the background, invisible to the user):

- Process similar to question 1.
- **If the paragraph beginning with the phrase "Jupiter was first visited by…" is indented by the checkmark bullet**, do the following:
    o Check the student results data table to see if the user has already submitted a correct answer. If not, do the following:
        ▪ Increase the total score for the test by one.
        ▪ Update the "Question 13 Result" value in the student results data table with the value "Correct".
- **If the paragraph beginning with the phrase "Jupiter was first visited by…" is not indented by the checkmark bullet**, do the following
        ▪ Update the "Question 13 Result" value in the student results data table with the value "Incorrect".

- Process similar to question 1.
- In preparation of Question 14, select the text "Pioneer 10 in".

A.2.14. Question 14

This question deals with applying the italic text-formatting option:

Question: **Italicize the selected text.**

Solution

For this question, the text in the current document window has been pre-selected as follows:

Jupiter was first visited by Pioneer 10 in 1973 and later by Pioneer 11, Voyager 1, Voyager 2 and Ulysses. The spacecraft Galileo orbited Jupiter for eight years. It is still regularly observed by the Hubble Space Telescope.

The correct solution is obtained by completing the following steps:

1. Click on the Format menu and click on the Font menu-item.
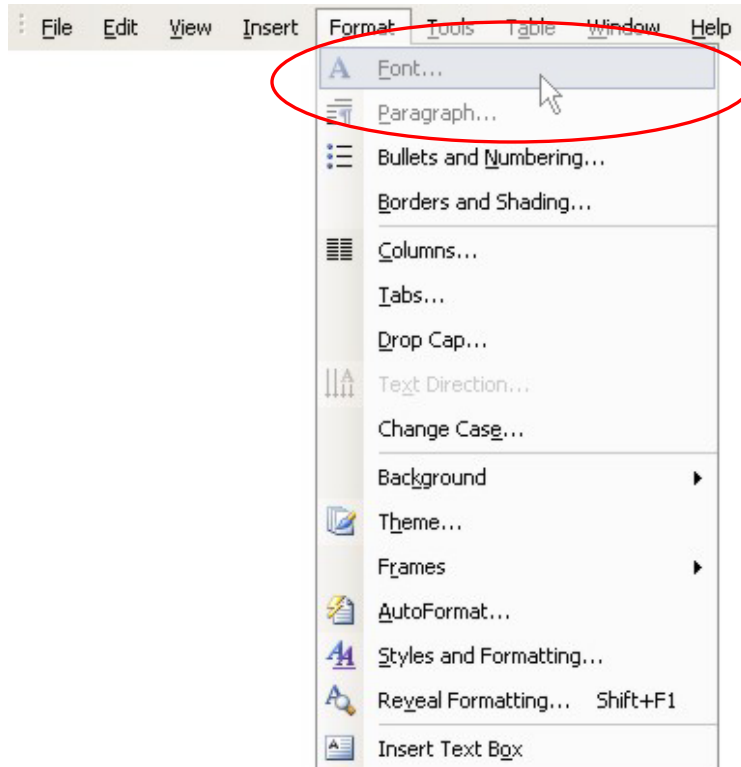
2. In the dialog window that appears, select the "Italic" option in the "Font style" list-box.
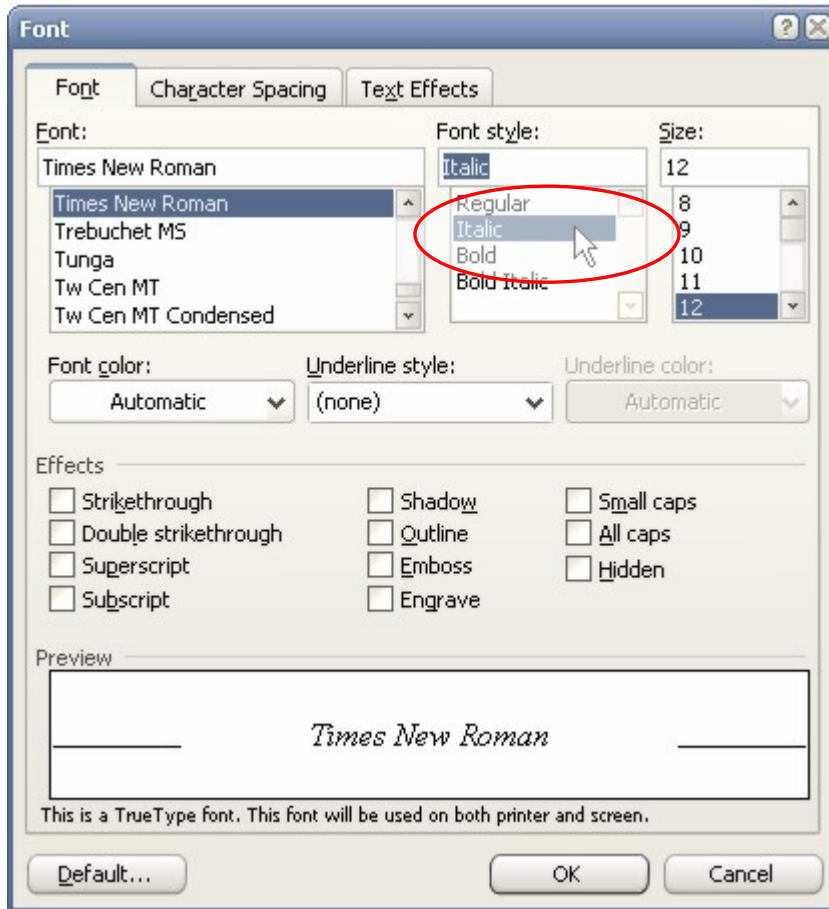
3. Alternatively, either click on the "Italic" toolbar button to format the selected text or press the "Ctrl+I" shortcut key on the keyboard.

WordAssessor Marking Procedure

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button.

WordAssessor then goes through the following process (all done in the background, invisible to the user):

- Process similar to question 1.
- **If the phrase "Pioneer 10 in" is italicised, do the following**:
  - o Check the student results data table to see if the user has already submitted a correct answer. If not, do the following:
    - ▪ Increase the total score for the test by one.
    - ▪ Update the "Question 14 Result" value in the student results data table with the value "Correct".
- **If the phrase "Pioneer 10 in" is not italicised, do the following:** Update the "Question 14 Result" value in the student results data table with the value "Incorrect".

- Process similar to question 1.
- In preparation of Question 15, select the paragraph beginning with the text "The gas planets do not have…".

A.2.15. Question 15

This question deals with applying centre alignment to text:

Question: **Centre the selected text between the left and right margins.**

Solution

For this question, the text in the current document window has been pre-selected as follows:

The gas planets do not have solid surfaces; their gaseous material simply gets denser with depth (the radii and diameters quoted for the planets are for levels corresponding to a pressure of 1 atmosphere). WE

The correct solution is obtained by completing the following steps:

1. Click on the Centre Text toolbar button.

OR
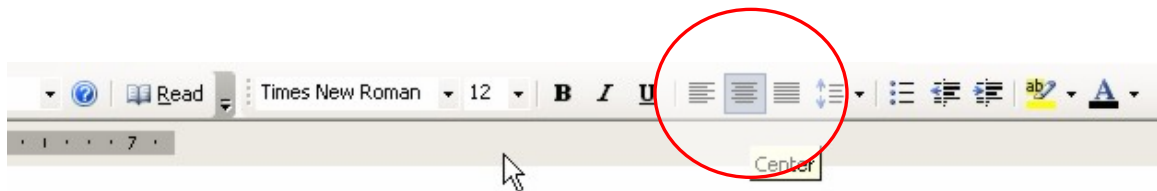
2. Click on the Format menu and select the Paragraph menu item.
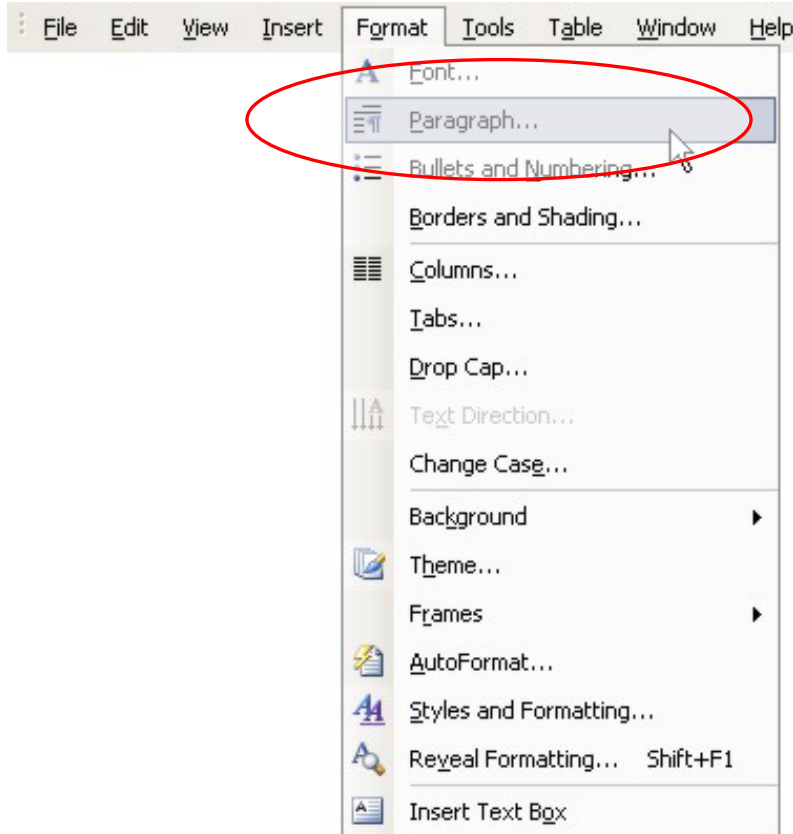
3. In the dialog window that opens up, select the "Centred" option from the
    Alignment drop-down list, as shown below.
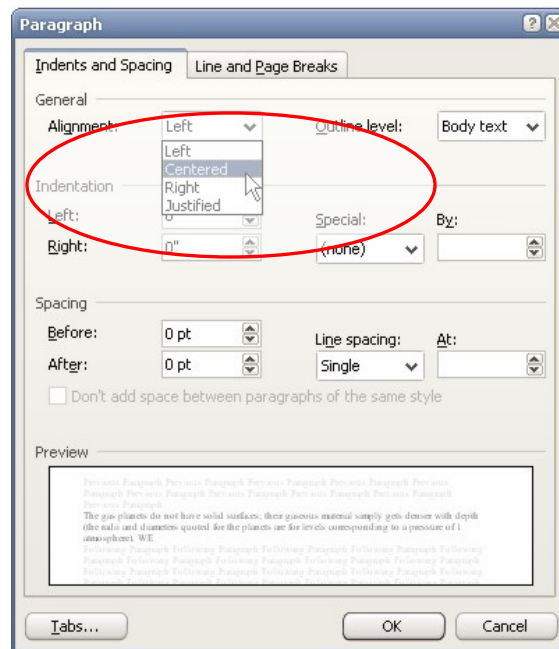
WordAssessor Marking Procedure

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button.

WordAssessor then goes through the following process (all done in the background, invisible to the user):

- Process similar to question 1.

- **If the paragraph beginning with "The gas planet do not have…" is centred, do the following**:
    - o Check the student results data table to see if the user has already submitted a correct answer. If not, do the following:
        - ▪ Increase the total score for the test by one.
        - ▪ Update the "Question 15 Result" value in the student results data table with the value "Correct".

- **If the paragraph beginning with "The gas planet do not have…" is not centred, do the following:** Update the "Question 15 Result" value in the student results data table with the value "Incorrect".

- Process similar to question 1.
- In preparation of Question 16, check to see if Question 16 has been done before. If it has, reset the text configuration file connected to Question 16.

A.2.16. Question 16

This question deals with printing certain pages of a document:

Question: **Print pages 2, 4, 6 to 10.**

Note: WordAssessor intercepts the MS Word Print command so that, when the print command is given, no actual pages are printed. Instead, a message is shown stating "Printing Successful!".
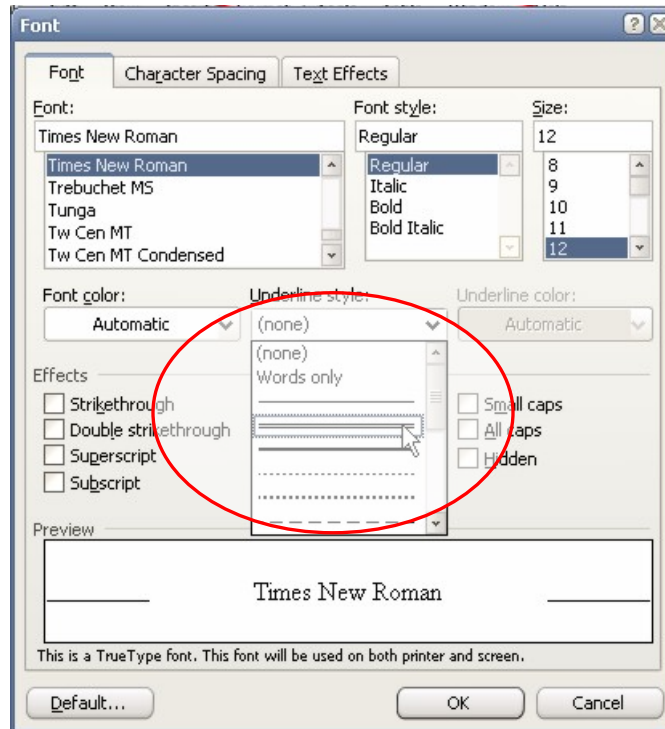
Solution

The correct solution is obtained by completing the following steps:

1. Click on the File menu and select the Print menu item.
   (See image below)
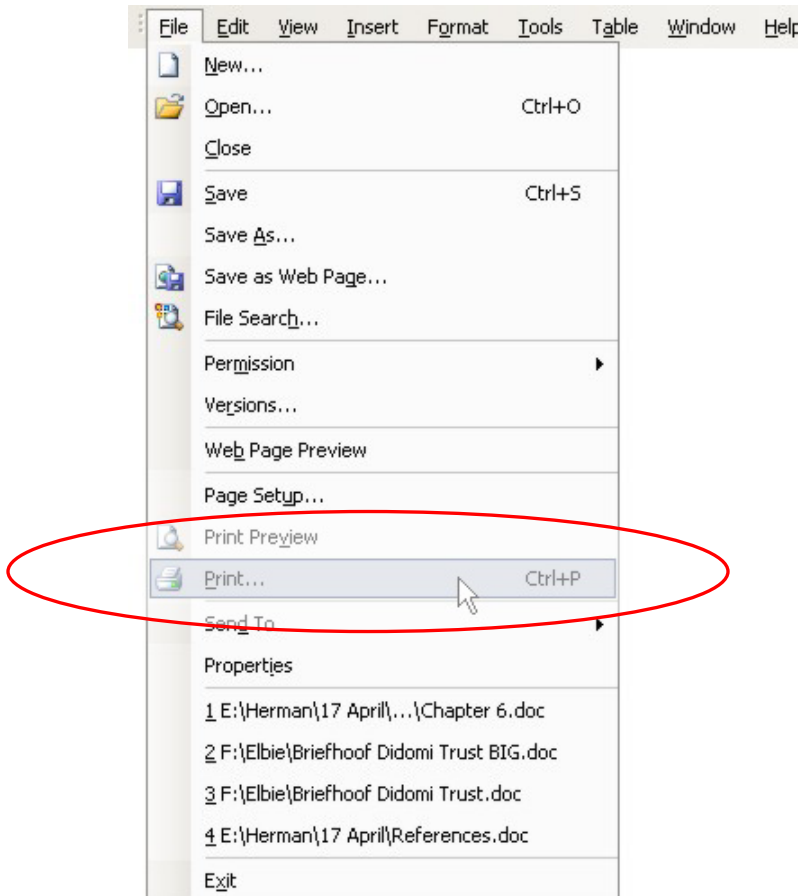
2. In th dialog window that appears, select the "Pages" option and type either "2,4,6-10" or "2,4,6,7,8,9,10" in the "Pages:" edit-box (As shown below).
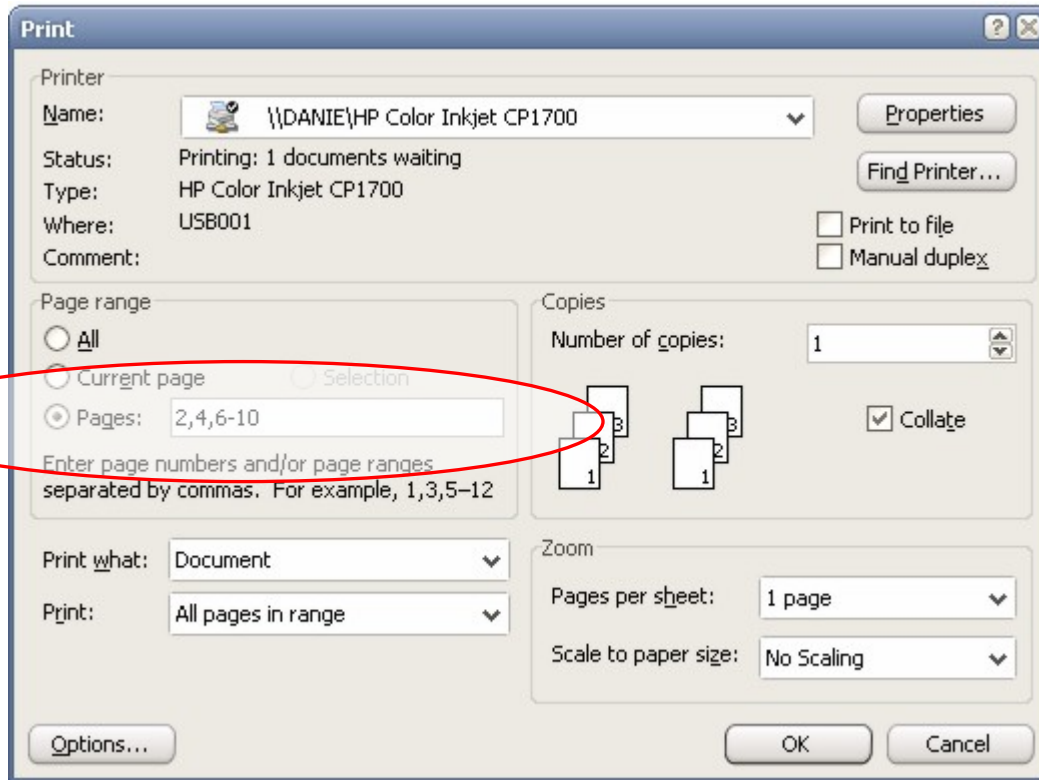
OR
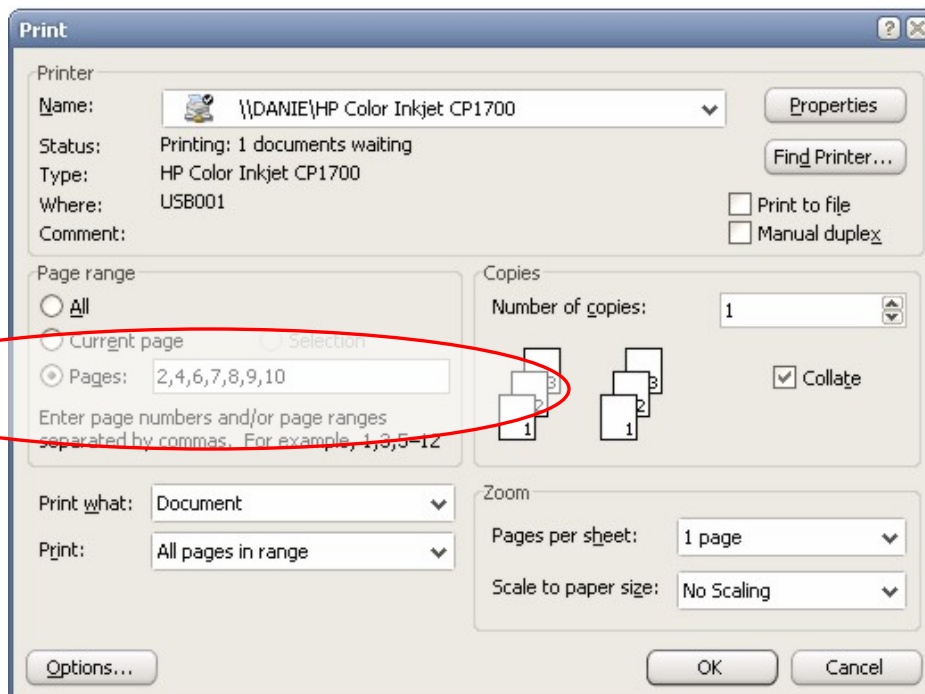
WordAssessor Marking Procedure

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button.

WordAssessor then goes through the following process (all done in the background, invisible to the user):

- Process similar to question 1.

If the user correctly prints the said pages, a configuration file is written to the user's hard drive as "Question16.txt".

- **If the "Question16.txt" configuration file exists on the user's hard drive, do the following**:
    o Check the student results data table to see if the user has already submitted a correct answer. If not, do the following:
        ▪ Increase the total score for the test by one.
        ▪ Update the "Question 16 Result" value in the student results data table with the value "Correct".
        ▪ Delete the "Question16.txt" configuration file from the user's hard drive, in case s/he would like to redo the question later.

- **If the "Question16.txt" configuration file does not exist on the user's hard drive, do the following:**
    o Update the "Question 16 Result" value in the student results data table with the value "Incorrect".

- Process similar to question 1.
- In preparation of Question 17, select the paragraph beginning with "The gas planets do not have…".

A.2.17. Question 17

This question deals with double-underlining of certain text:

Question: **Double underline the selected text.**

Solution

For this question, the text in the current document window has been pre-selected as follows:

The gas planets do not have solid surfaces; their gaseous material simply gets denser with depth (the radii and diameters quoted for the planets are for levels corresponding to a pressure of 1 atmosphere). WE

The correct solution is obtained by completing the following steps:

1. Click on the Format menu and select the Font menu item.
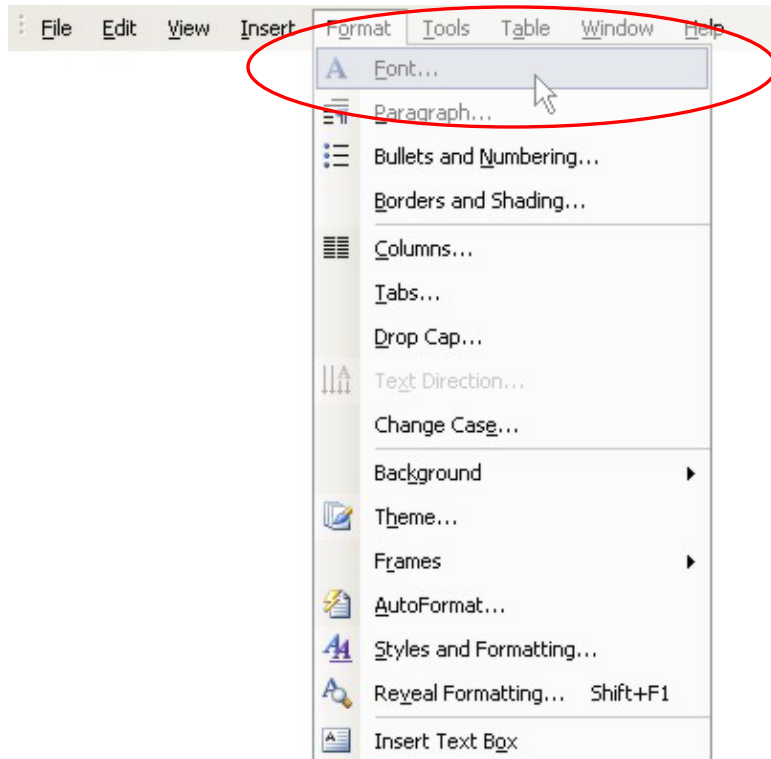
2. In the dialog window that appears, select the option with the two lines from the "Underline style" drop-down list.

WordAssessor Marking Procedure

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button.

WordAssessor then goes through the following process (all done in the background, invisible to the user):

- Process similar to question 1.
- **If the paragraph starting with "The gas planets do not have…" is double-underlined, do the following**:
  - o Check the student results data table to see if the user has already submitted a correct answer. If not, do the following:
    - ▪ Increase the total score for the test by one.
    - ▪ Update the "Question 17 Result" value in the student results data table with the value "Correct".

- **If the paragraph starting with "The gas planets do not have…" is double-underlined, do the following:**
  - Update the "Question 17 Result" value in the student results data table with the value "Incorrect".

- Process similar to question 1.
- In preparation of Question 18, select the paragraph beginning with "The gas planets do not have…"
- Also for Question 18, enable a timer that constantly checks the state of the mouse cursor.

A.2.18. Question 18

This question deals with the format-painter tool:

Question: **Click on the shortcut button that will copy the formatting of the selected sentence.**

Solution

For this question, the text in the current document window has been pre-selected as follows:

The correct solution is obtained by completing the following steps:
1. Click on the "Format Painter" toolbar button.
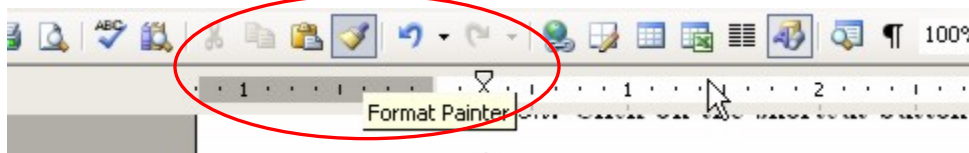
WordAssessor Marking Procedure

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button.

WordAssessor then goes through the following process (all done in the background, invisible to the user):

- Process similar to question 1.
- **If the mouse cursor is in the format painter "state", do the following**:
  - o Check the student results data table to see if the user has already submitted a correct answer. If not, do the following:
    - ▪ Increase the total score for the test by one.
    - ▪ Update the "Question 18 Result" value in the student results data table with the value "Correct".

- **If the mouse cursor is not in the format painter "state", do the following:**
  - o Update the "Question 18 Result" value in the student results data table with the value "Incorrect".

- Process similar to question 1.
- Disable the timer that constantly checks the state of the mouse cursor.

It is interesting to note that, from the results in section 7.4, only about 4% of the students, who attempted this question, had it correct. It seems as if the format painter tool is very rarely used.

A.2.19. Question 19

This question deals with the tab-stop function:

Question: **Add a right tab stop at 3 inches on the ruler bar**

Solution

The correct solution is obtained by completing the following steps:

1. Double-click on the ruler bar at the 3 inches position.

OR
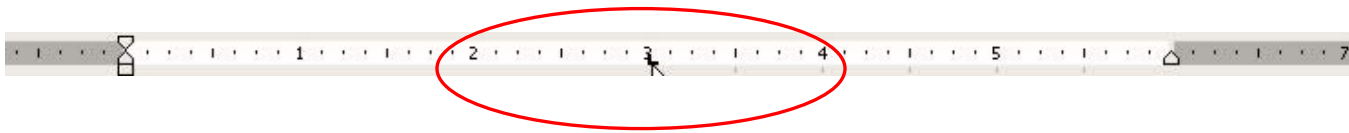
2. Click on the Format menu and select the "Tabs" menu item.

3. In the dialog window that opens, click on the "Right" alignment option and click OK.
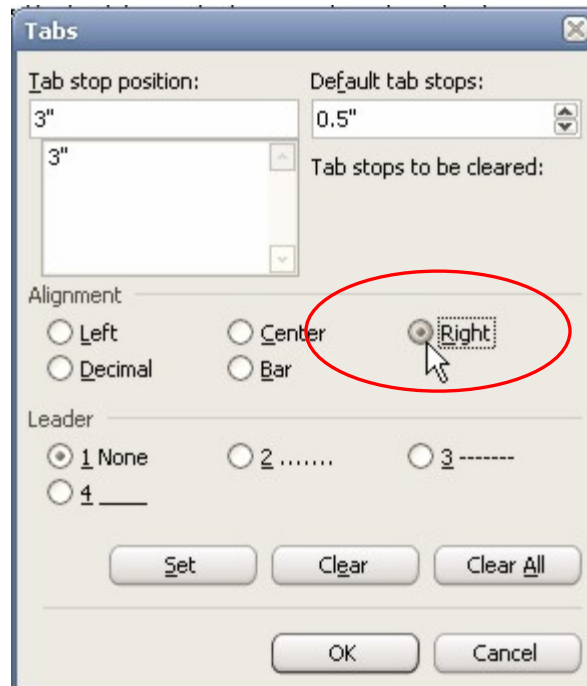
WordAssessor Marking Procedure

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button.

WordAssessor then goes through the following process (all done in the background, invisible to the user):

- Process similar to question 1.

- **If the current document contains a right tab-stop at the 3 inches position, do the following**:
    o Check the student results data table to see if the user has already submitted a correct answer. If not, do the following:
        ▪ Increase the total score for the test by one.
        ▪ Update the "Question 19 Result" value in the student results data table with the value "Correct".

- **If the current document does not contain a right tab-stop at the 3 inches position, do the following:**
    o Update the "Question 19 Result" value in the student results data table with the value "Incorrect".

- Process similar to question 1.
- In preparation of the next question, set the view of the current document to print view and set the zoom level of the document to 100%.

A.2.20. Question 20

This question deals with the zoom function:

Question: **Display the whole page of this document.**

Solution

The correct solution is obtained by completing the following steps:

1. Select the Zoom drop-down list from the main toolbar and choose the "Whole Page" option.
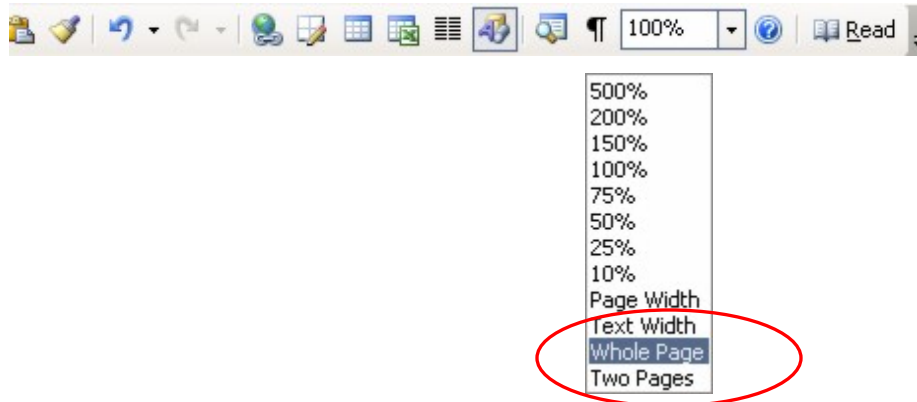
OR

2. Select the View menu and choose the "Zoom" option.

162
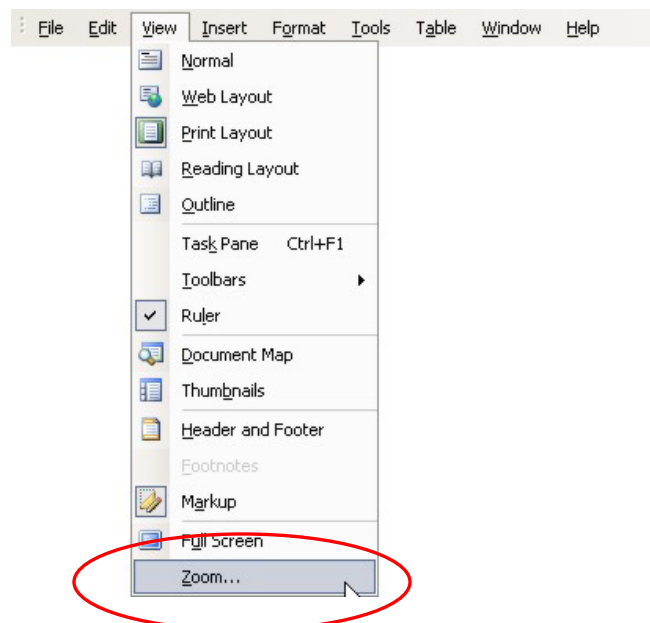
3. In the dialog window that appears, select the "Whole page" option and click OK.
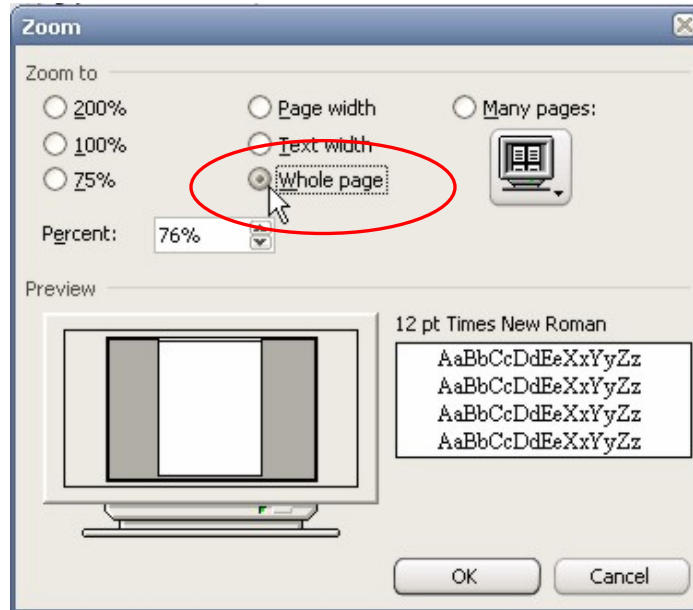
WordAssessor Marking Procedure

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button.

WordAssessor then goes through the following process (all done in the background, invisible to the user):

- Process similar to question 1.

- **If the view of the current document window is set to "Whole page", do the following**:
  - Check the student results data table to see if the user has already submitted a correct answer. If not, do the following:
    - Increase the total score for the test by one.

- Update the "Question 20 Result" value in the student results data table with the value "Correct".

- **If the view of the current document window is not set to "Whole page", do the following:**
  - Update the "Question 20 Result" value in the student results data table with the value "Incorrect".

- Process similar to question 1.
- Reset the view of the current document to normal view and set the zoom level of the document to 100%.

A.2.21. Question 21

This question deals with MS Word's page margins setting:

Question: **Set the left margin to 0.5 inches.**

Solution

The correct solution is obtained by completing the following steps:

1. Select the File menu and choose the "Page Setup" option.
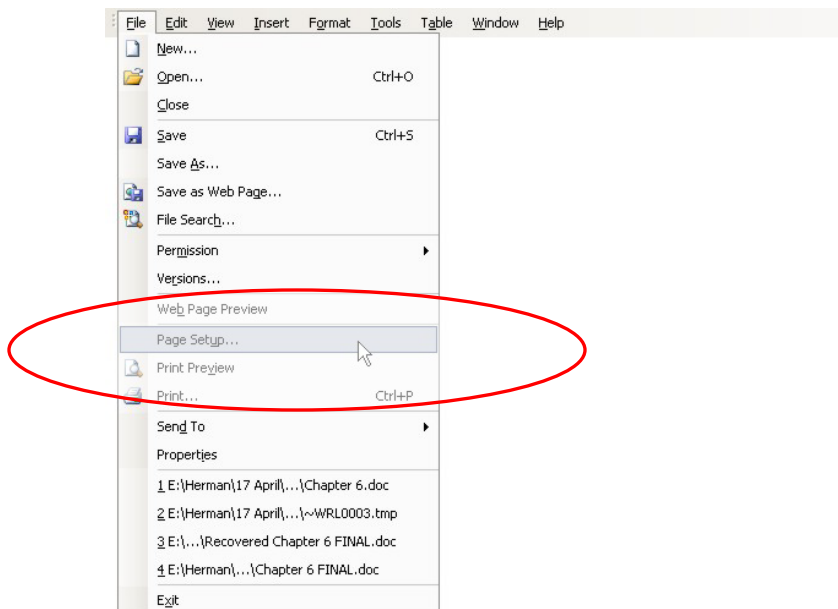
2. In the dialog window that appears, type 0.5" in the left margins edit box.

(See image below)
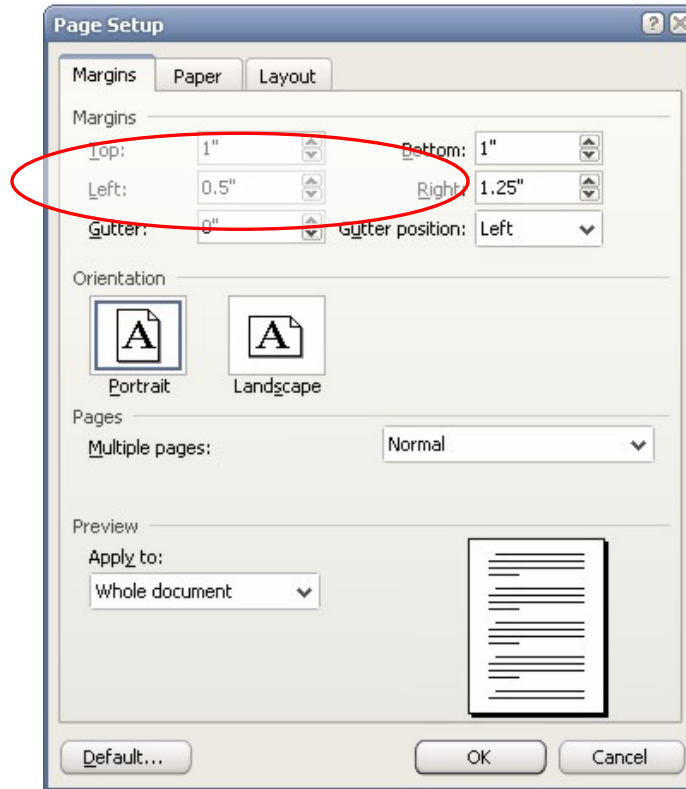
WordAssessor Marking Procedure

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button.

WordAssessor then goes through the following process (all done in the background, invisible to the user):

- Process similar to question 1.

- **If the left margin of the page is set to 0.5 inches, do the following**:
  - o Check the student results data table to see if the user has already submitted a correct answer. If not, do the following:
    - Increase the total score for the test by one.

- Update the "Question 21 Result" value in the student results data table with the value "Correct".

- **If the left margin of the page is not set to 0.5 inches, do the following:**
  o Update the "Question 21 Result" value in the student results data table with the value "Incorrect".

- Process similar to question 1.
- Reset the left margin of the current document to its default state.

A.2.22. Question 22

This question deals with the page break function:

Question: **Add a page break at the insertion point.**

Solution

The correct solution is obtained by completing the following steps:

1. Click on the Insert menu and select the "Break" option.
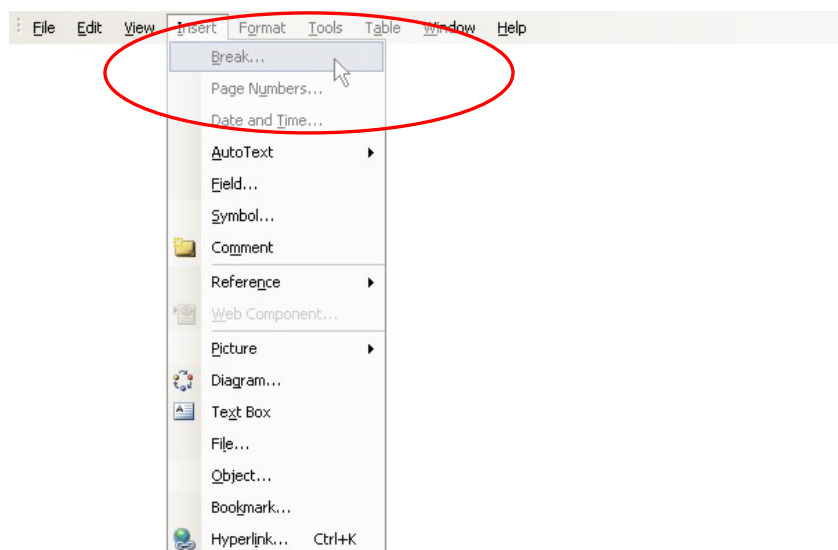
2. In the dialog window that appears, select "Page break" and click OK.
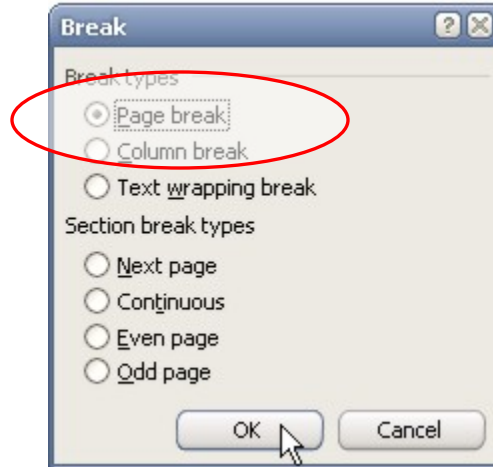
WordAssessor Marking Procedure

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button.

WordAssessor then goes through the following process (all done in the background, invisible to the user):

- Process similar to question 1.

- **If a page-break is present, do the following**:
    - o Check the student results data table to see if the user has already submitted a correct answer. If not, do the following:
        - Increase the total score for the test by one.
        - Update the "Question 22 Result" value in the student results data table with the value "Correct".
- **If a page-break is not present, do the following:**
    - o Update the "Question 22 Result" value in the student results data table with the value "Incorrect".

- Process similar to question 1.
- In preparation of Question 23, select the word "GOOD" from the main body of text.

A.2.23. Question 23

This question deals with synonym function:

Question: **Find a synonym for the selected word GOOD. Replace the word GOOD with the word EXCELLENT**

Solution

The correct solution is obtained by completing the following steps:

1. Right click on the word "GOOD", and select "excellent" from the synonyms pop-up menu.
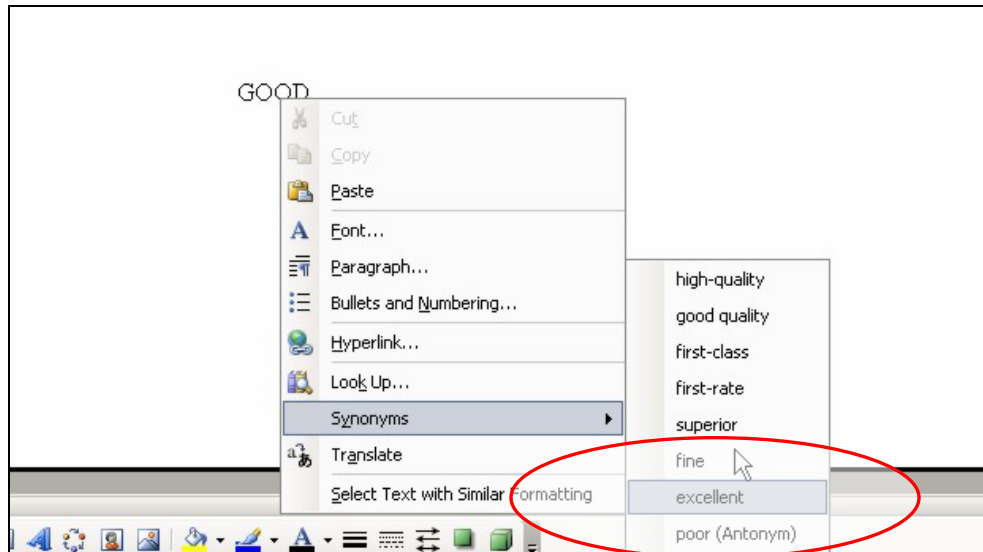
There are numerous ways to replace the word "GOOD" with its synonym "EXCELLENT". Any way is accepted by WordAssessor as the programme only checks the end-result.

WordAssessor Marking Procedure

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button.

WordAssessor then goes through the following process (all done in the background, invisible to the user):

- Process similar to question 1.

- **If the word "GOOD" has been replaced by the word "EXCELLENT", do the following**:
  - o Check the student results data table to see if the user has already submitted a correct answer. If not, do the following:
    - ▪ Increase the total score for the test by one.
    - ▪ Update the "Question 23 Result" value in the student results data table with the value "Correct".
- **If the word "GOOD" has not been replaced by the word "EXCELLENT", do the following:**
  - o Update the "Question 23 Result" value in the student results data table with the value "Incorrect".

- Process similar to question 1.
- In preparation of Question 24, set the language of the document to "UkEnglish".

A.2.24. Question 24

This question deals with the spell-check function:

Question: **Spell check the document. The word PLEEZED is spelled incorrectly and should be PLEASED.**

Solution

The correct solution is obtained by completing the following steps:

1. Either press F7 on the keyboard or select "Spelling and Grammar" from the Tools menu
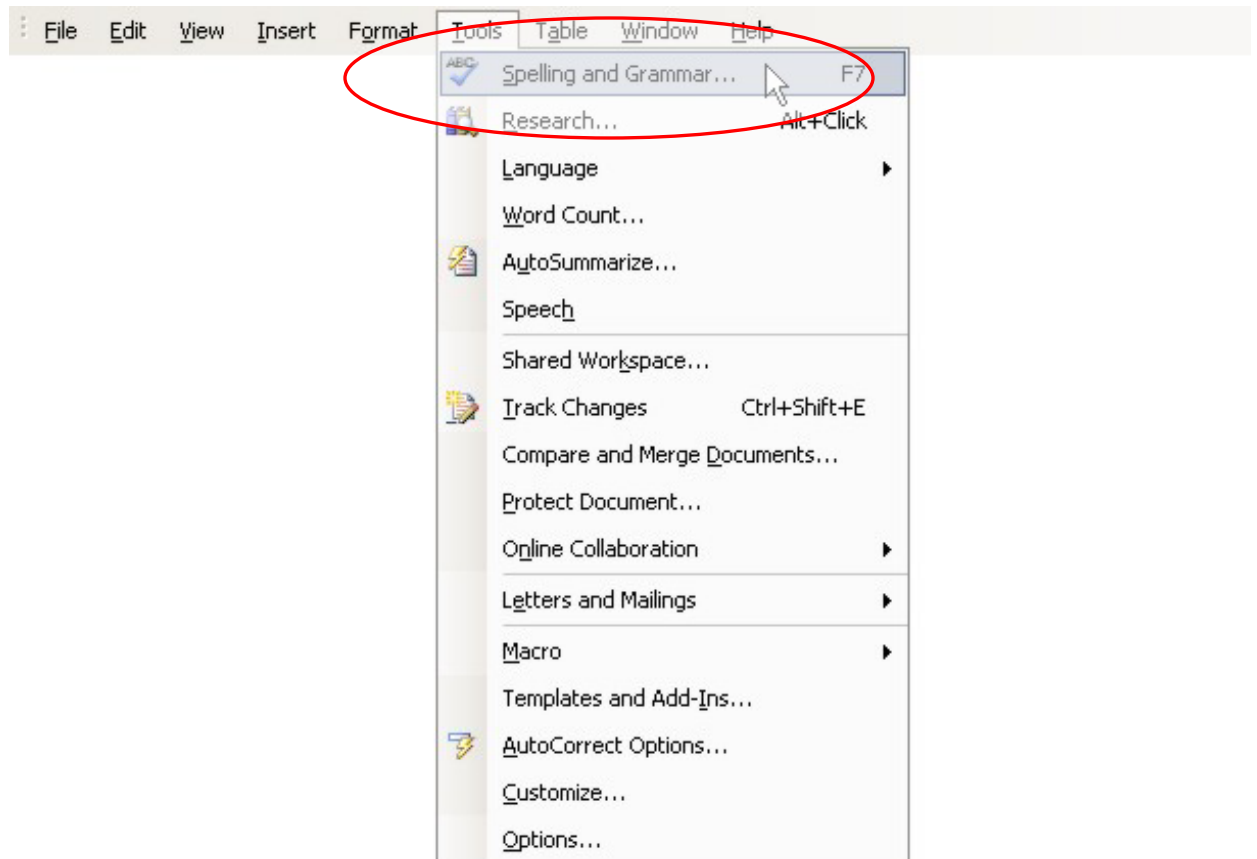
2. In the dialog window that appears, click on "Change"

(See image below)

OR

3. Right click on the word "Pleezed" and select the spell-checked word "Pleased".
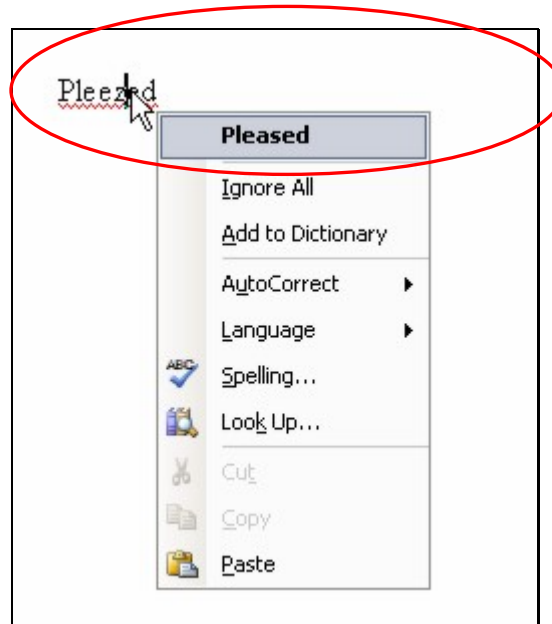
WordAssessor Marking Procedure

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button.

WordAssessor then goes through the following process (all done in the background, invisible to the user):

- Process similar to question 1.

- **If the word "PLEEZED" has been replaced by the spell-checked word "PLEASED", do the following**:
    - o Check the student results data table to see if the user has already submitted a correct answer. If not, do the following:
        - Increase the total score for the test by one.
        - Update the "Question 24 Result" value in the student results data table with the value "Correct".

- **If the word "PLEEZED" has not been replaced by the spell-checked word "PLEASED", do the following:**
    - o Update the "Question 24 Result" value in the student results data table with the value "Incorrect".

- Process similar to question 1.
- Check if the configuration files for Question 25 exist. If they do, delete them so that users can repeat the question if necessary.

A.2.25. Question 25

This question deals with the page orientation setting:

Question: **Print the current document in landscape mode. Do not use the "File - Page Setup" option.**

The reason this question has a limiting effect on the method the user must use to answer the question is that the existing test system (see section 1.3) employs the same question, word for word.
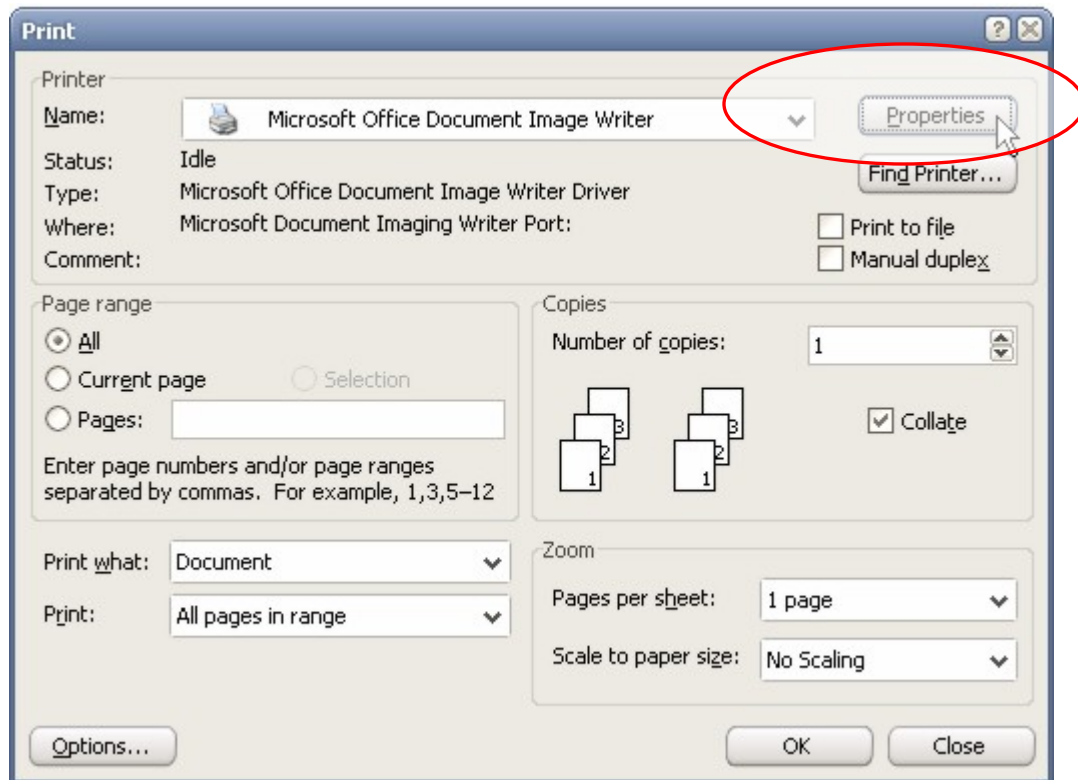
Solution

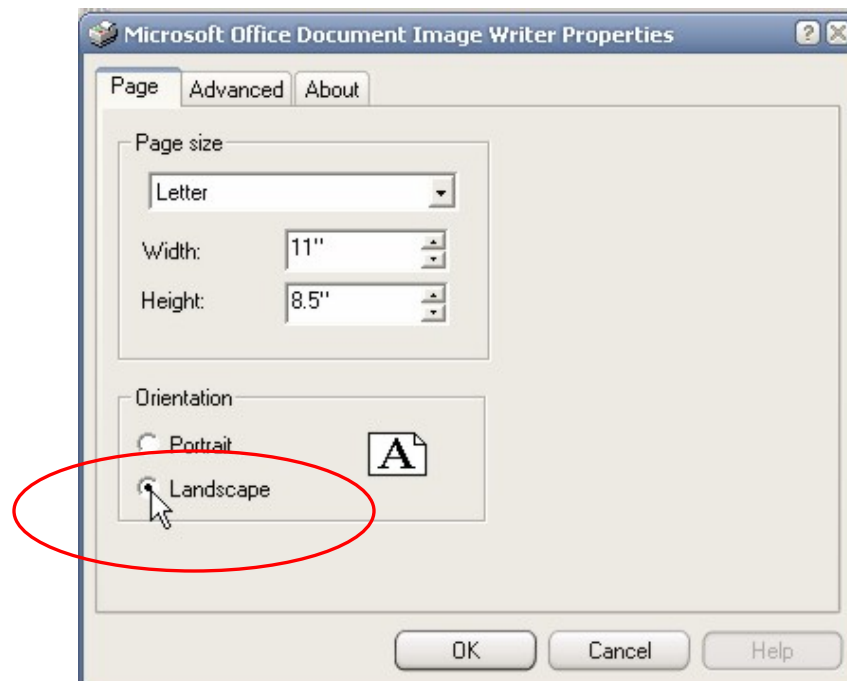The correct solution is obtained by completing the following steps:

1. Select "Print" from the File menu.
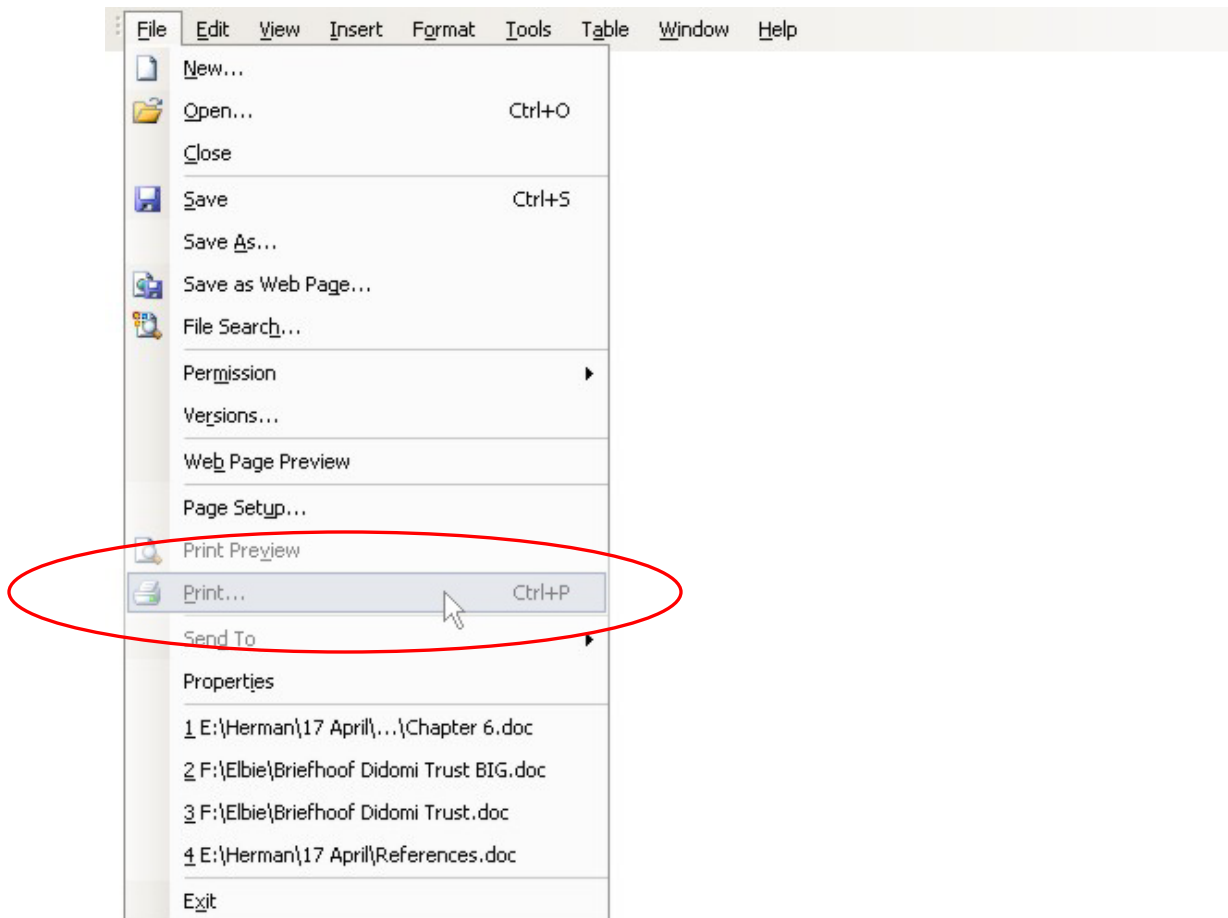
2. In the dialog that appears, click on the "Properties" button.

3. In the second dialog window that appears, select the "Landscape" option from the "Page" tab and click OK twice.

WordAssessor Marking Procedure

When the user has completed the question and wishes to submit his/her answer, they click on the Submit Answer button.

WordAssessor then goes through the following process (all done in the background, invisible to the user):

- Process similar to question 1.

- **Check the Question 25 configuration files. If the print orientation was changed to landscape by using the print properties option, do the following**:
    - o Check the student results data table to see if the user has already submitted a correct answer. If not, do the following:
        - ▪ Increase the total score for the test by one.
        - ▪ Update the "Question 25 Result" value in the student results data table with the value "Correct".

- **Check the Question 25 configuration files. If the print orientation was not changed to landscape by using the print properties option, do the following:**
    - o Update the "Question 25 Result" value in the student results data table with the value "Incorrect".

- Process similar to question 1.

Appendix B

ASSESSMENT POLICY OF THE UNIVERSITY OF

THE FREE STATE

1.   Aanhef

Die assesserinsbeleid van die Universiteit van die Vrystaat (UV) neem as vertrekpunte die nasionale hoëronderwys-wetgewing asook die visie, missie en waardes van die UV. Die assesseringsbeleid moet in samehang met ander UV-beleidsdokumente gelees word.

Die UV erken Uitkomsgerigte Onderwys (UGO) as onderwysbenadering **soos vervat in dokumente van die Nasionale Departement van Onderwys. Uitkomsgerigte onderwys is 'n leerdergesentreerde, resultaatgerigte onderwysbenadering wat gebaseer is op die aanname dat alle studente in staat is om hulle volle potensiaal te bereik, maar dat dit nie noodwendig op dieselfde wyse en op dieselfde tydstip sal geskied nie.**

Die implikasie hiervan is dat:
- **wat studente moet leer, duidelik omskryf word;**

- **elke student se vordering op die demonstrering van vaardighede en bevoegdhede moet berus;**
- **studente se behoeftes geakkommodeer word deur middel van veelvuldige onderrig- en leerstrategieë asook assesseringsinstrumente;**
- **elke student genoegsame tyd, geleenthede en ondersteuning verskaf behoort te word ten einde sy/haar potensiaal te laat realiseer.**

Uitkomsgerigte assessering word omskryf as die identifisering, versameling en interpretering van 'n student se prestasie gemeet teen die uitkomste van die spesifieke kwalifikasie. Die fokus in deurlopende assessering behoort op die demonstrering van duidelik geformuleerde uitkomste binne konteks te wees. Om dit te verwesenlik kan toepaslike assesseringsinstrumente gebruik word wat lewenslange leer kan bevorder.

2.   Doel van die beleid

Die oorkoepelende doel van hierdie beleid is om 'n raamwerk daar te stel waarbinne assesseringspraktyke aan hierdie universiteit sal meewerk tot die:

- **ondersteuning van maksimale ontwikkeling van studente;**
- **versekering van effektiewe leer binne**

1.   Preamble

The assessment policy of the University of the Free State (UFS) takes as points of departure the national higher education legislation, as well as the vision, mission and values of the UFS. The assessment policy must be read together with other UFS policy documents.

The UFS acknowledge Outcomes-based Education (OBE) as educational approach **as contained in documents by the National Department of Education. Outcomes-based education is a learner-centred, result-oriented educational approach based on the assumption that all learners are capable of realising their full potential, but that this will not necessarily occur in the same way and at the same point in time.**

The implication is that:
- **what students have to learn must be clearly defined;**
- **every student's progress must be based on the demonstration of achievements;**
- **student's needs must be accommodated by means of multiple teaching and learning strategies as well as assessment instruments;**
- **every student should be afforded sufficient time, opportunities and support to be able to realise his/her full potential.**

Outcomes-based assessment is defined as the identification, collection and interpretation of a student's performance measured against the outcomes of the specific qualification. The focus in continuous assessment should be on the demonstration of clearly formulated outcomes within context. To achieve this, appropriate assessment instruments can be used which can promote lifelong learning

2.   Aim of the policy

The overarching aim of this policy is to establish a framework within which assessment practices at this university will contribute towards the:

- **support of maximum development of students;**

konteks;
- verbetering van onderrigpraktyke;
- gradering van studenteprestasies.

- assurance of effective learning within context;
- improvement of teaching practices;
- grading of student performance.

Ten einde hierdie doelstellings te verwesenlik, behoort hierdie beleid in fakulteitsverband vertolk te word in prosedures wat die volgende beginsels onderskryf.

In order to achieve these aims, this policy should be interpreted within each faculty as procedures which endorse the following principles.

3.   Beginsels van assessering wat beleid onderskryf

3.   Principles of assessment endorsed by the policy

Omdat assessering een van die kragtigste invloede op die aard en omvang van studenteleer uitoefen, plaas dié beleid veral klem op die ontwerp van assessering om studenteleer te bevorder. Deurgaans word in ag geneem dat effektiewe assessering gegrond is op gesonde programontwerp, -ontwikkeling, -implementering en -evaluering.

As assessment exerts one of the most powerful influences on the nature and extent of student learning, this policy especially emphasises the design of assessment to promote student learning. It is taken into account throughout that effective assessment is based on sound programme design, development, implementation and evaluation.

Hierdie beginsels behoort as riglyne vir akademici betrokke by assessering dien, waaraan hul assesseringspraktyke, in terme van individuele assesseringsgeleenthede en die prosesse op module- en programvlak gemeet kan word. Fakulteite is verantwoordelik vir die vertolking van hierdie beginsels binne eie konteks in terme van reglemente en regulasies wat ook in studiegidse opgeneem en toegepas moet word.

These principles should serve as guidelines to academics involved in assessment by which their assessment practices can be measured in terms of individual assessment opportunities and the processes at module and programme level. Faculties are responsible for the interpretation of these principles within own context in terms of rules and regulations which must also be taken up in study guides and applied.

Daar word verwag dat alle assesseringsgeleenthede en –prosesse aan hierdie beginsels sal voldoen. Akademici betrokke by die assessering van studenteleer behoort hulself vir alle vlakke van assessering (modulevlak, programvlak, webgebaseerde toeste, ens.) teen dié beginsels te kan verantwoord.

It is expected that all assessment opportunities and processes will comply with these principles. Academics involved in the assessment of student learning should be able to justify their actions in respect of all levels of assessment (module level, programme level, web-based tests, etc.) against these principles.

Hierdie beginsels moet egter nie in isolasie oorweeg of toegepas word nie, maar sover as moontlik in balans tot mekaar.

However, these principles must not be considered or applied in isolation, but as far as possible balanced against one another.

Effektiewe assessering is gebaseer op die volgende beginsels:
- Die doel van assessering moet altyd duidelik gekommunikeer word.
- Assessering moet holisties en kriteriumgerig eerder as normgerig wees.
- Assessering moet outentiek, deurlopend, multidimensioneel, gevarieerd en gebalanseerd wees.
- Assessering moet deurlopend met die onderrig- en leerproses geïntegreer word.
- Assessering moet deursigtig, geldig, betroubaar en regverdig wees.
- Assessering kan verskeie vorme aanneem, inligting vanuit verskeie kontekste versamel en verskeie metodes aanwend na gelang van dit wat geassesseer word en wat die behoefte van die student is.
- Assessering moet onpartydig, sensitief teenoor ras, geslag, kulturele agtergrond en vermoëns wees.
- Die terugvoer van assesseringsresultate moet duidelik, akkuraat, tydig en betekenisvol wees.
- Vordering moet gekoppel wees aan die demonstrering van die uitkomste binne konteks.

Effective assessment is based on the following principles:
- The purpose of assessment must always be communicated clearly.
- Assessment must be holistic and criterion referenced rather than norm referenced.
- Assessment must be authentic, continuous, multidimensional, varied and balanced.
- Assessment must be integrated continuously with the teaching and learning process.
- Assessment must be transparent, valid, reliable and just.
- Assessment can assume various forms, gather information from various contexts and use various methods depending on what is being assessed and the needs of the student.
- Assessment must be impartial, sensitive towards race, gender, cultural background and ability.
- The feedback regarding assessment results must be clear, accurate, timely and meaningful
- Progress must be linked to the

- **Resultate van assesserings-geleenthede moet gebruik word om ondersteuning en ortodidaktiese leiding aan studente te verskaf waar nodig.**
- **Die proses en volume van assessering moet realisties en hanteerbaar vir beide studente en dosente wees.**
- **Sekuriteit moet gehandhaaf word deur die nodige prosedures daar te stel om oneerlikheid sover moontlik te verhoed, te identifiseer en te hanteer.**

- **demonstration of the outcomes within context.**
- **Results of assessment opportunities must be used to provide students with support and orthodidactical guidance where necessary.**

- **The process and volume of assessment must be realistic and manageable for both students and lecturers.**
- **Security must be maintained through the introduction of procedures to prevent, detect and handle dishonesty.**

4. Geïntegreerde assessering

Verskillende soorte assessering kan in die uitvoering van hierdie beleid aangewend word. Hierdie soorte assessering is nie onderling uitsluitend nie en behoort geïntegreerd in die oorkoepelende assesseringsproses aangewend te word en sluit die volgende in:

- **Vormende assessering sodat die positiewe prestasies van studente herken, bespreek en toepaslike opvolgstappe beplan kan word.**

- **Diagnostiese assessering, waardeur leerprobleme geïdentifiseer en geklassifiseer kan word sodat toepaslike ortodidaktiese leiding voorsien kan word.**

- **Summatiewe assessering om die oorkoepelende prestasie van 'n student op 'n sistematiese wyse aan te teken en verslag te doen.**

- **Evaluerings ten einde inligting van studente se prestasie te versamel en te vergelyk, sodat dit in kurrikulumontwikkeling en die evaluering van onderrig en leer aangewend kan word.**

Uitkomste word geassesseer in terme van toegepaste bevoegdheid, naamlik praktiese bevoegdheid, gefundeerde bevoegdheid en reflektiewe bevoegtheid wat in konteks gedemonstreer moet word.

Geïntegreerde assessering kom tot sy reg as daaruit voortvloei dat die oorkoepelende uitkomste van 'n program/kwalifikasie behaal is.

4. Integrated assessment

Different types of assessment can be applied in the implementation of this policy. These types of assessment are not mutually exclusive and should be applied in an integrated manner in the overarching assessment process, and include the following:

- **Formative assessment in order that the positive achievements of students can be recognised, discussed and appropriate follow-up steps planned.**

- **Diagnostic evaluation, by means of which learning problems can be identified and classified in order that appropriate orthodidactical guidance can be provided.**

- **Summative assessment in order to record and report on the overarching performance of a student in a systematic manner.**

- **Evaluations aimed at collecting and comparing information regarding students' performance, in order that it may be applied in curriculum development and evaluation of teaching and learning.**

Outcomes are assessed in terms of applied competence, namely practical competence, foundational competence and reflexive competence which must be demonstrated in context.

Integrated assessment comes into its own if it results in overarching outcomes of a programme/qualification having been achieved.

5. Assessering as deel van gehalteversekering

Gehalteversekering behoort volledig deel te wees van onderrig en leer ten einde 'n effek op onderwys en opleiding te kan hê. Moderering is 'n belangrike instrument wat kwaliteitstandaarde vir die insette, prosesse en uitsette sal verseker. Moderering vind dus nie net aan die einde van 'n proses plaas nie, maar vorm deel van die sikliese aard van gehalte.

Moderering word intern en ekstern op verskeie vlakke uitgevoer. Dit verseker dat studente konstant, akkuraat en op 'n goed ontwerpte wyse geassesseer word. 'n Interne modereringstelsel vorm deel van 'n

5. Assessment as part of quality assurance

Quality assurance should be integral to teaching and learning to have an effect on education and training. Moderation is an important tool that will ensure that quality standards for the inputs, the processes and outputs, are maintained. Moderation should thus not take place only at the end of the process, but is part of the cyclical nature of quality.

Moderation is executed internally and externally at different levels. It ensures that students are assessed in a consistent, accurate and well-designed manner. An internal moderation system forms part of the

gehaltebestuurstelsel van hierdie instelling (sien die beleidsdokument t.o.v. gehalteversekering).

Gehalteversekering vir die proses van onderrig- en leerassessering word gedefinieer in terme van toepaslikheid ("fitness for purpose"). In gehalteversekering van onderrig- en leerassessering moet vasgestel word:

- **of studente die module- en kwalifikasie-uitkomste op die betrokke NKR-vlak bemeester het; en**
- **of die assesserings geldig, betroubaar, verantwoordbaar en uitvoerbaar is.**

Gehalteversekering, op die relevante NKR vlak, kan geskied deur middel van:
- **evaluering van dosente en leermateriaal deur studente aan die hand van vraelyste;**
- **interne skriftelike moderering;**
- **volledige memorandums gebaseer op die assesseringskriteria en/of -standaarde vir die gestelde uitkomste;**
- **eksterne moderering van assessering op uittreevlakke soos omskryf in die gehalteversekeringsbeleid;**
- **evauering van uitkomste deur buitevennote in gevalle waar sodanige partye betrokke is; en**
- **die analisering en interpretering van die bewyse van die assesserings-proses.**

6.   Opleiding van assessore/ moderatore

- **Van nuutaangestelde doserende personeel word verwag om die Universiteit van die Vrystaat se assessore-/moderatore-opleidings-program te deurloop om te kan kwalifiseer vir 'n vaste aanstelling. Soortgelyke opleiding, wat aan ander instansies deurloop is, mag erken word aan die hand van die riglyne in hierdie assesseringsbeleid en erkenning van voorafleer.**
- **Deurlopende "opskerpwerks-winkels" oor assesserings-/ modererings- verwante onderwerpe behoort op gereelde basis aan akademiese personeel aangebied te word.**
- **Dit is die verantwoordelikheid van departementele voorsitters/ departementshoofde om te verseker dat assessore/moderatore toepaslike opleiding deurloop.**

7.   Bestuur en implementering van die beleid

Die departementshoof/departementele voorsitter tesame met die programdirekteur is verantwoordelik vir die bestuur en implementering van hierdie beleid rakende die betrokke modules in die program. Die bestuur word gerig deur 'n vasgelegde prosedure.

8.   Aantekening en verslagdoening van vordering en prestasie

quality management system of this institution (see the policy document on quality assurance).

Quality assurance for the process of teaching and learning assessment is defined in terms of fitness for purpose. In quality assurance of teaching and learning assessment it must be determined:

- **whether students mastered the module and qualifications outcomes at the NQF level concerned; and**
- **whether the assessments were valid, reliable and feasible.**

Quality assurance, at the relevant NQF level, can occur by means of:
- **evaluation of lecturers and learning material by students on the basis of questionnaires;**
- **internal written moderation;**
- **comprehensive memorandums based on the assessment criteria and/or standards for the set outcomes;**
- **external moderation of learning assessment at exit levels as defined in the quality assurance policy;**
- **evaluation of outcomes by external associates in cases where they are involved; and**
- **analysis and interpretation of the evidence of the assessment process.**

6.   Training of assessors/ moderators

- **It is expected from newly appointed lecturing staff to complete the assessor/moderators training programme of the University of the Free State in order to qualify for a permanent appointment. Similar training undergone at other institutions may be recognised on the basis of the guidelines of this policy on assessment and prior learning.**
- **Ongoing "refresher workshops" on assessment/moderation-related subjects should be offered to academic staff on a regular basis.**
- **It is the responsibility of departmental chairpersons/heads of departments to make sure that assessors/moderators undergo appropriate training.**

7.   Management and implementation of the policy

The head of the department/departmental chairperson, together with the programme director, is responsible for the management and implementation of this policy with regard to the relevant modules in the programme. The management is directed by a fixed procedure.

8.   Recording of and reporting on progress and performance

Bemagtigde akademici sal instaat wees om te kan besluit hoe hierdie faset van assessering binne hul verantwoordelikheidsarea behoort te geskied en dit moet dus nie as voorskriftelik gesien word nie. Die bedoeling is eerder om ruimte te skep waarbinne akademici verantwoordbare keuses ten opsigte van die aantekening en verslagdoening van vordering en prestasie binne hul eie omgewings kan maak.

Empowered academics will be able to decide how this facet of assessment should take place within their area of responsibility and it must therefore not be regarded as prescriptive. The intention is rather to create scope within which academics can make justifiable choices regarding the recording of and reporting on progress and performance in their environments.

8.1   Aantekening van vordering en prestasie

Kumulatiewe getuienis van studente se vordering en prestasie moet aangeteken word en kan in die geval van professionele grade deel van die student se portefeulje vorm. Dit is die verantwoordelikheid van die student om die portefeulje volgens voorskrifte op datum te hou. Verder is dit die verantwoordelikheid van dosente om rekord te hou van die assesseringsproses en -uitslag

Recording   of   progress   and performance

Cumulative evidence of students' progress and performance must be recorded and, in the case of professional degrees, this can form part of the student's portfolio. It is the responsibility of the student to keep the portfolio up to date according to the instructions. It further is the responsibility of lecturers to keep record of the assessment process and the – results.

8.2   Verslagdoening

Effektiewe kommunikering van studente se prestasie is 'n voorvereiste vir die voorsiening van gehalte-onderwys. Verslagdoening is 'n integrale deel van onderig, leer en assessering.

Reporting

Effective communication of students' performance is a precondition for the provision of quality education. Reporting is an integral part of teaching, learning and assessment.

8.2.1 Die interne verslagdoeningsproses moet:
- **dien as geleentheid vir die dosent om voortdurend terugvoer aan studente te gee aangaande hulle vordering in die onderrig- en leerproses;**
- **'n akkurate weergawe wees van die student se vordering en prestasie;**
- **'n aanduiding gee van die student se sterk punte en ontwikkelingsbehoeftes, asook opvolgstappe vir die onderrig- en leerproses aandui;**
- **deur 'n konstruktiewe benadering, motivering aanmoedig.**

8.2.1 The internal reporting process must:
- **serve as an opportunity for the lecturer to provide students with feedback regarding their progress in the teaching and learning process on an ongoing basis;**
- **be an accurate reflection of the student's progress and performance;**
- **provide an indication of the student's strong points and development needs, as well as indicate follow-up steps for the teaching and learning process;**
- **encourage motivation through a constructive approach.**

8.2.2 In gevalle waar eksterne vennote betrokke is (bv. diensleermodules), moet die verslagdoeningsproses:
- **dien as geleentheid vir dosente en studente om aan die vennote terugvoer te gee oor vordering met die aktiwiteite;**
- **'n geleentheid bied vir die vennote om aan te dui hoe hulle die uitkomste beoordeel;**
- **'n aanduiding gee van die student (en vennote) se ontwikkelingsbehoeftes, asook opvolgstappe vir hierdie doel aandui;**
- **'n konstruktiewe bydrae lewer tot interafhanklike, gesamentlike leer en ontwikkeling; en**
- **deursigtigheid en verantwoordbaarheid teenoor die vennote bevorder.**

8.2.2   In instances where external associates are involved (e.g. in-service learning modules) the reporting process must:
- **serve as opportunity for lecturers and students to give feedback to the associates regarding progress of the activities;**
- **provide an opportunity to the associates to show how they judge the outcomes;**
- **give an indication of the development needs of the students (and associates) as well as follow up steps for this purpose;**
- **provide a constructive contribution to interdependent, joint learning and development; and**
- **promote transparency and accountability towards the associates.**

9.   Implementering en monitering van dié beleid
- **Elke akademiese lynbestuurder van die**

9.   Implementation and monitoring of this policy
- **Every academic line manager at the**

Universiteit van die Vrystaat is verantwoordelik vir die implementering van en toesighouding oor die assesseringsbeleid.

- **Die akademiese lynbestuurders moet via die relevante komitees aan die Dekaan en die Fakulteitsraad verslag doen aangaande die uitvoering en bevindings van die assesseringsproses op vasgestelde tydsintervalle.**

10.   Dispute en die bylê van dispute

Enige akademiese personeellid/student aan die Universiteit van die Vrystaat kan, vir verdere aksie, enige relevante klag, vraag of getuienis ten opsigte van die implementering van die assesseringsbeleid, sonder vrese vir benadeling, by die betrokke Dekaan of persoon deur die Dekaan/Fakulteitsbestuur aangewys, indien.

University of the Free State is responsible for the implementation of and supervision of the assessment policy.

- **The academic line managers must report to the Dean and Faculty Board via the relevant committees on the completion and findings of the assessment process during fixed time intervals.**

10.   Disputes and the settling of the disputes

Any academic personnel member/student at the University of the Free State can, for further action, lay any relevant complaint, ask any question or submit evidence pertaining to the implementation of the assessment policy to the Dean concerned or person appointed by the Dean/Faculty Management, without fear of impairment.

Appendix C

WordAssessor post-test questionnaire

Dear Student

Thank you for completing your test with the new MS Word Skills testing program!

PLEASE take a few quick minutes to complete the questionnaire below. Your responses will provide important information that will help UFS in planning better ways to test your computer skills and improve the quality of your education.

This research study will fail without your help. PLEASE be honest with your answers and don't rush through the questions.

Your help will be GREATLY APPRECIATED!!

Thanks!

--- Questionnaire ---

IN EACH CASE, PLEASE CIRCLE THE APPROPRIATE OPTION

VIDEO TUTORIALS

1. **After viewing the video tutorials, I now understand where I went wrong in the test questions I had incorrect.**

- strongly disagree
- disagree
- neutral
- agree
- strongly agree

2. **I will be able to obtain all the correct answers if I had to redo the test.**

- strongly disagree
- disagree
- neutral
- agree
- strongly agree

3. **After attempting the test by myself and then viewing the video tutorials of the solutions, I feel that I have improved my learning in the process.**

- strongly disagree
- disagree
- neutral
- agree
- strongly agree

4. **The video tutorials were straightforward and easy to follow.**

- strongly disagree
- disagree
- neutral
- agree
- strongly agree

5. **Were the video tutorials distracting in any way?**

- yes
- no
- sometimes

6. **I prefer receiving feedback though video tutorials rather than static methods like paper-based feedback or on-screen text feedback.**

- strongly disagree
- disagree
- neutral
- agree
- strongly agree

SOFTWARE ENVIRONMENT

7. I prefer being tested in the real MS Word environment and not a simulated environment.

- strongly disagree
- disagree
- neutral
- agree
- strongly agree

8. I prefer being able to use any method to answer a question (as opposed to being told what method to use).

- strongly disagree
- disagree
- neutral
- agree
- strongly agree

9. What aspect of the video tutorials did you like the most? Please arrange the following items from best to worst (Using the numbers 1 to 5).

- Quality of Presentation

- The fact that it showed the solution effectively

- The fact that it explained each step

- The fact that the solution was shown by video instead of text

- The fact that I learned while being tested

10. Did you dislike anything about the video tutorials?

- Yes
- No

If Yes, please specify:

_____

_____

_____

11. **Do you have any suggestions on how we could improve the video tutorial technique to enhance your learning?**

Please specify:

_____
_____
_____

Appendix D
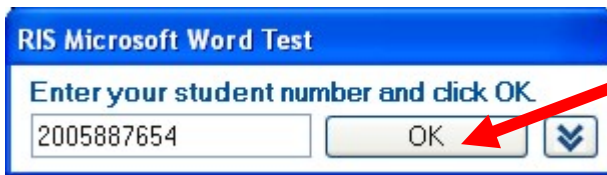
WordAssessor test instructions
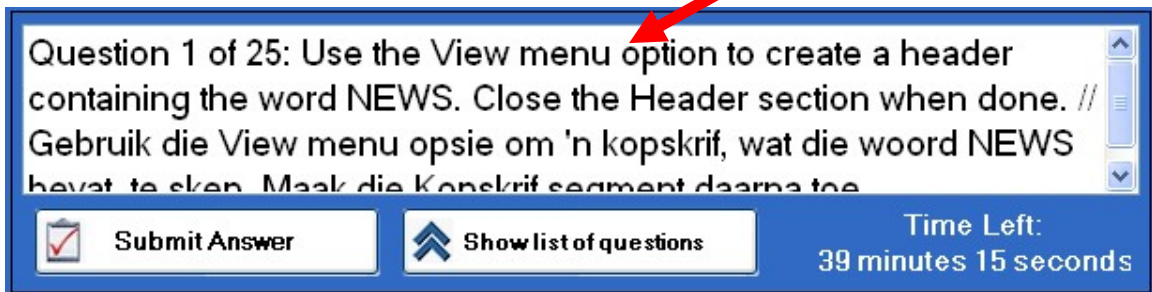
Use the MS Word skills testing program on your computer as follows:

1. **Type in your student number and click OK**

2. **Read the question in the box**

3. **Do what the question asks in the open MS Word window.**
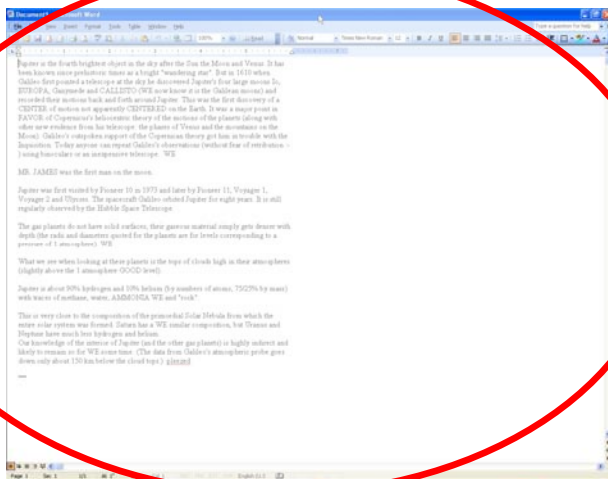
4. When you have completed the question, click on the "Submit Answer" button as shown below. Wait a moment for the program to mark your answer.

Note: Make sure that any open dialog windows like the "Print" or "Find and Replace" or "Insert Picture" etc. are closed before you click on "Submit Answer".

OTHER IMPORTANT INSTRUCTIONS

- **To skip to a different question or to redo a certain question, do the following:**

1. Click on the "Show list of questions" button.
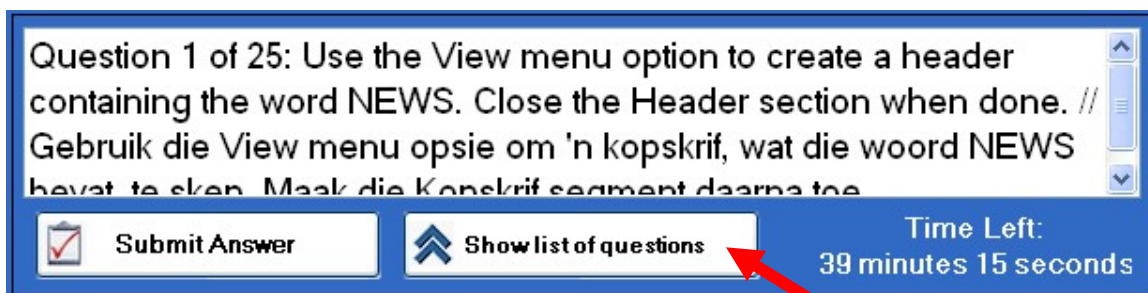
2. Wait for a moment, and then click on any of the blue question links to try a certain question.

3. After you have clicked on one of the blue links, it will open up that specific question. Click on the "Hide list of questions" button and proceed to answer the question (see image below).

_____

- **When you have finished all of the questions, click on the "Exit / End Test" button as shown in the above picture.**
- **Follow the on-screen instructions and you will be shown the solutions to the questions you had incorrect via Short Video Tutorials.**

_____

- **Lastly, please complete the questionnaire given to you.**

Summary

In recent times, people have become more and more reliant on computers on a daily basis. As a result, the need has arisen to optimise the task-related experience in terms of time efficiency, which demands effective training in software skills. To be more specific, word processing skills are currently considered essential in any field of work and are in high demand. This study focuses on determining the optimal paradigm (methods) to assess users' word processing skills.

One of the main reasons for this research was the fact that students at the University of the Free State (UFS) reported to the computer literacy course lecturer that they were dissatisfied with the **virtual, simulated** MS Word software environment used to assess (e-assess) their word processing skills electronically. This existing test system (ETS) at the UFS requires students to perform certain tasks and automatically checks whether the required end-result is obtained. However, this system is based on a simulated interface with limited functionality. As a result, the relevant information on software e-assessment systems was researched and a new software skills e-assessment application developed accordingly. The aim was to develop a tool that would be able to assess students' word processing skills in the most reliable way possible. Another aim was to find methods of stimulating the learning process during the e-assessment of word processing skills.

Therefore the newly developed e-assessment system, WordAssessor, is designed to be based on the real MS Word environment. It requires students to perform certain tasks and automatically checks whether the said tasks have been correctly performed. WordAssessor allows students to explore the MS Word interface fully while being assessed. It even allows students to use trial-and-error to solve certain problems (tasks). To stimulate learning potentially further during e-assessment, WordAssessor presents students with a video solution for the questions they had incorrect, directly after the test.

In order to assess the validity of the e-assessment methods employed by the WordAssessor system, comparative user testing was conducted. Students' word processing skills were assessed as part of their advanced computer literacy course, using the ETS, the WordAssessor system and a personalised test scenario (where no e-assessment tool was used). In addition, participating students were provided with a questionnaire to determine their reaction and preference with regard to the various elements of the assessment methods.

By analysing the results in detail, it was found that the results of the personalised test scenario (PT) yielded the most reliable indication of students' true word processing skills, and could be used as a benchmark. Hereafter, the results of the WordAssessor test scenario were analysed to determine the correlation (relationship) with the results of the personalised test. The same type of correlation was performed between the results of the ETS and PT. It was established that the WordAssessor results correlated significantly more closely with the PT results than the ETS.

In the end (and after additional analysis) it was found that the methods employed by WordAssessor yielded the most reliable indication of students' true word processing skills knowledge when compared to the ETS. In addition, from the results of the post-assessment questionnaire, it was determined that students felt they learned more as a result of the video tutorials. Furthermore, they stated that they preferred video feedback over text- or paper-based feedback. They also stated that they preferred being assessed in a real software environment, as opposed to a simulation.

It was recommended that a more flexible and realistic e-assessment approach (as demonstrated by the WordAssessor system) could be beneficial to students on several levels. Also, it was recommended that students be able to answer test questions in a way that suits them rather than being instructed as to the method of use. Finally, the use of highly detailed video tutorials directly following e-assessment (only for questions students had incorrect), was recommended.

Keywords: E-assessment

Software skills

Formative assessment

Word processing

Video tutorials

Assessment paradigm

Opsomming

In die jongste verlede het die mens toenemend afhanklik geraak van 'n daaglikse gebruik van rekenaars. Dit het daartoe gelei dat 'n behoefte ontstaan het om die taak-gerigte ervaring ten opsigte van doeltreffende tydsbenutting te optimaliseer, wat op sy beurt effektiewe opleiding in programmatuur-vaardighede vereis. In die besonder word woordverwerkingsvaardighede tans as onontbeerlik beskou binne enige arbeidsveld en is dus hoog in aanvraag. Dit is die fokus van hierdie studie om vas te stel wat die optimale paradigma (metodes) is om die woordverwerkingsvaardighede van gebruikers te assesseer.

Een van die hoofredes vir hierdie navorsing is gegee in die feit dat studente aan die Universiteit van die Vrystaat (UVS) by 'n dosent, wat rekenaargeletterdheid aanbied, gekla het dat hulle ontevrede is met die *virtueel-gesimuleerde MS Word* programmatuur-omgewing wat gebruik word om hul woordverwerkingsvaardighede elektronies te assesseer (e-assessering). Die bestaande toets-sisteem (BTS) by die UVS verwag van studente om sekere take uit te voer en gaan dan outomaties na of die verwagte eind-resultate verkry is. Hierdie sisteem is egter gebaseer op 'n gesimuleerde koppelvlak met 'n beperkte funksionaliteit. Dit het daartoe gelei dat die relevante inligting rakende die e-assessering van programmatuur ondersoek is en dat 'n geskikte nuwe programmatuur e-assesseringstoepassing ontwikkel is.

Hierdie nuut-ontwikkelde e-assesseringsisteem, *WordAssessor*, is so ontwerp dat dit op die werklike *MS Word*-omgewing gebaseer is. Dit verg van die studente om sekere take uit te voer en om outomaties na te gaan of hierdie take behoorlik voltooi is. *WordAssessor* laat studente toe om die *MS Word* koppelvlak volledig te benut terwyl die assessering plaasvind. Dit laat studente selfs toe om deur middel van die probeer-fouteer metode bepaalde probleme (take) op te los. Om leer-potensiaal tydens die assessering te

193

stimuleer, bied *WordAssessor* onmiddellik na die toets aan studente 'n video-oplossing vir die probleme wat hulle verkeerd gehad het.

Om die geldigheid van die e-assesseringsmetodes wat deur die *WordAssessor*-sisteem gebruik word te bepaal, word gebruik gemaak van vergelykende gebruikertoetsing. Studente se woordverwerkingsvaardighede is geassesseer as deel van hul gevorderde rekenaargeletterdheidskursus, met gebruikmaking van die BTS, die *WordAssessor*-sisteem en 'n persoonlik-begeleide toets-scenario (PB) (waar geen e-assesseringsinstrument gebruik is nie). Hierbenewens, is deelnemende studente voorsien van 'n vraelys om hul reaksie en voorkeur te bepaal met betrekking tot die verskillende elemente van die assesseringsmetodes.

Deur middel van 'n detail-analise van die resultate is bevind dat die persoonlik-begeleide toets-scenario die mees betroubare aanduiding van die werklike woordverwerkingsvaar-dighede van die studente is en dat dit as standaard (maatstaf) gebruik kan word. Vervolgens is die resultate van die woordverwerkingstoets-scenario ontleed om die korrelasie met die resultate van die persoonlik-begeleide toets vas te stel. 'n Soortgelyke korrelasie is toegepas op die resultate van die BTS en die PB. Daar is vasgestel dat die *WordAssessor*-resultate op 'n betekenisvolle wyse nader korreleer met die PB-resultate as met dié van die BTS.

Uiteindelik (en na verdere analise) is bevind dat, in vergelyking met die BTS, die metodes wat deur *WordAssessor* aangewend is, die mees betroubare aanduiding van studente se werklike woordverwerkingsvaardighede en –kundigheid daarstel. Bykomend, is uit die resultate van die post-assesseringvraelys vasgestel dat studente van mening is dat hulle as 'n uitvloeisel van die video tutoriale meer geleer het. Daarbenewens het hulle gestel dat hulle die video terugvoer verkies bo teks- of papier-gebaseerde terugvoer. Hulle het ook vermeld dat hulle verkieslik binne die egte programmatuuromgewing geassesseer wil word, in teenstelling met 'n simulasie.

Daar is aanbeveel dat 'n meer soepele en realistiese e-assesseringsbenadering (soos deur die *WordAssessor*-sisteem gedemonstreer) op verskillende vlakke voordelig vir studente sal wees. Verder is daar aanbeveel dat studente in staat gestel moet word om die toetsvrae op 'n wyse te beantwoord waarmee hulle gemaklik is, eerder as om voorskrifte oor die metode van gebruik te ontvang. Laastens is die gebruik van hoogsgedetaileerde video tutoriale regstreeks ná die e-assessering (slegs ten opsigte van daardie vrae waar studente gefouteer het) aanbeveel.

Sleutelwoorde: E-assessering

Sagteware vaardighede

Formatiewe assessering

Woordverwerking

Video tutoriale

Assesseringsparadigma