INTEGRATING STUDENT SELF-ASSESSMENT AND FEEDBACK IN E-LEARNING APPLICATIONS: A PROPOSED EDUCATIONAL MODEL

Iman Sadek Zainy Alansari

Submitted for the Degree
of Doctor of Philosophy

Department of Computing
School of Computing, Informatics and Media

University of Bradford

2009

Abstract

There is a large demand for the use of e-learning tools to support student learning, in the form of distance or blended learning. The need for e-learning environment that encourages learners to learn independently or in groups in virtual settings is crucial. Some e-learning environments provide repositories of ‘resources’. They neither facilitate a strategy for learning or teaching, nor they guide students through the resources, and tutors in constructing their courses.

E-learning environments need to incorporate pedagogical practices which support and allow students to learn by removing any barriers that might inhibit their learning. Therefore, one of the most important aspects in developing e-learning environments is defining appropriate models where technology and pedagogy are integrated.

This thesis provides such a framework for developing e-learning applications; it aims to make it easier for tutors to implement their lesson content and engage learners to achieve the course objectives. The proposed model incorporates constructive alignment, assessment and feedback and unlike other e-learning environments guides the tutor to construct lessons and help learners to use effective learning environment. Furthermore, the thesis investigates on how supported learning can help students adapt to the different approaches to learning. The empirical work undertaken investigates the role of constructing a well designed self-assessment and feedback unit within a learning environment.

Dedicated to my great parents, loving husband and lovely children ...
keep inspiring me!

Acknowledgements

My deepest thanks go to God, who created me, helps me, and has given me the ability to reach this stage.

I would like to acknowledge, with great pleasure, Dr. Anastasia Konstadopoulou and Dr. Rod Fretwell, who have provided me with guidance and encouragement throughout this journey. Thank you Anastasia and Rod, I have completed this thesis with the support from your expertise and seemingly endless reservoirs of patience.

I would also like to thank the technical support team and John Dermo who supported and helped me conduct the experimental work, using Question Mark Perception. I would like also to thank the members of the Computing Department: Miss Rona Wilson and Mrs. Bev Yates.

Many thanks to the lecturer Arwa Linjawi and her students, at King Abdulaziz University, Jeddah, who helped me by partaking in the first and second online experiments.

I give loving thanks to my husband, Majdi Alyas, my lovely children, and my friend Noorjanah. It would not have been possible to complete my PhD research without their encouragement and support throughout my personal life and career.

I would also like to thank my friends Amina Alswaai, Ibtihal Nafea, Rasha Fares, Rasha Osman, and Rogayya Abdul-Rahman who supported and encouraged me throughout this process.

Last, but not least, I give gratitude to my loving parents and family who have continually encouraged and supported me throughout this journey.

Publications

1. I. Alansari, A. Konstadopoulou, R. Fretwell, “Prompt and Relevant Feedback in e-Learning”, International Conference on Excellence in Education 2008: Future Minds and Creativity, *Proceedings of the Annual Conference of the International Center for Innovation in Education (ICIE)* held in Paris-France, ISBN:978-9957-476-03-8, 1-4 July 2008.

2. I. Alansari, A. Konstadopoulou, R. Fretwell, “Assessment and Instant Feedback in Learning Java Programming”, *PGNET:The* 9^{th} *Annual Postgraduate Symposium,The Convergence of Telecommunications, Networking and Broadcasting* Liverpool John Moores University, ISBN:978-1-902560-19-9, 23rd-24th June 2008.

3. I. Alansari, A. Konstadopoulou, R. Fretwell, “Generalized, Instant and Effective Feedback in Learning Java Programming”, *Proceedings of the* 9^{th} *Informatics Workshop for Research Students*, University of Bradford, ISBN:978-1-85143-251-6, 13 June 2008.

4. I. Alansari, A. Konstadopoulou, R. Fretwell, “Modeling Effective Self Assessment and Feedback to Enhance Distance Learning”, *Proceedings of the* 8^{th} *Informatics Workshop for Research Students*, University of Bradford, ISBN:978-185143-246-2, 28 June 2007.

Contents

List of Tables

List of Figures

List of Abbreviations

CAA	Computer Assisted Assessment
CRC	Class Responsibility Collaborator
JDI	Java Debug Interface
JPDA	Java Platform Debugger Architecture
JSP	Jackson Structured Programming
JT	Java Translator
JVMDI	Java Virtual Machine Debug Interface
SOLO	Structure of Observed Learning Outcomes
UI	User Interface
VLEs	Virtual Learning Environments
VM	Virtual Machine
WBT	Web Based Training

Chapter 1

Thesis Introduction

1.1 Introduction

The contribution of this thesis is the design of an e-learning system in which the emphasis is upon prompt relevant feedback to students. The design is aligned with established theory of learning together with the results of the empirical investigation conducted as part of the design precess. The system is general in that its application is not restricted to any particular academic discipline.

The literature review investigates different information communication technologies which could be introduced to support learning. Currently, e-learning environments are heavily dependent on the delivery of the course content and in providing information rather than focusing on educational concepts. However, when e-learning environments incorporate the established theory of learning, student learning is enhanced.

1.2 Motivation

The use of e-learning, for teaching, has increased dramatically over recent years. The e-learning environment needs to meet the student's needs and permit deep learning of the subject matter. Many of the recent virtual learning environments (VLEs), that permit e-learning, do not actually support student learning. These environments are generally designed to manage the course delivery through the use of a supported graphical interface design. Designers should however design and maintain e-learning environments which support learning processes and encourage the removal of any barriers that may inhibit the student from learning.

In designing an e-learning environment many factors must be considered, such as: communication between the instructors and the students; the delivery of learning resources; self assessment; summative assessment; and feedback. Self-assessment supports student learning. The feedback maintains the learning process by keeping students motivated. Therefore, designers must consider self-assessment and feedback because these are critical factors affecting the instructor, student and the course content.

It is important to investigate the effect that assessment and feedback have on student learning; this will require considerable attention. A persistent problem in e-learning application is how to choose the most appropriate model to support student learning, especially when there are large numbers of students on the course. The thesis ultimately aims to identify how e-learning applications can support each student individually, by focusing on how different and new e-learning applications can guide instructors to support students to learn and to meet their course objectives.

1.3 Aims and Objectives

The aims of this thesis are:

- To provide general solution for designing e-learning applications which will support e-learning and distance learning environments.
- To discuss and evaluate the role of constructing a well designed self-assessment and feedback unit within a learning environment.
- To investigate how self-assessment can support learning and help students to adapt to different approaches to learning.
- To demonstrate how the proposed design aims to help instructors, based on assessments, to design course contents that is aligned with course objectives.

1.4 Research Methodology

Given the objective of the research is to design a system for e-learning, the approach initially was to review what is known in the area of

- established theory of learning.
- distance learning.
- virtual learning environments.
- computer assisted assessment.

That review presented in chapter 2 exposes two important main points. Firstly, the need for a system that leads the tutor to apply constructive alignment theory as introduced by Biggs (2003, p. 25). Secondly a system that provides the students continual self-assessment with prompt, relevant feedback. The second point is especially important for distance learning and e-learning where the tutor is not accessible by the student. Consequently, the next step was to discover how prompt, relevant feedback can be provided by software. A series of practical experiments was designed, conducted and evaluated as shown in chapter 3. Finally, all that had been established was combined and incorporated in the design of the e-learning system.

- A model was formulated and tested. The model incorporates constructive alignment, assessment and feedback. It was tested for generality constructing lessons in two different subjects to show that the resulting lessons are congruent with model.
- The model was developed further (see chapter 5) by considering the use of the system
    a. by the tutor in constructing lessons,
    b. by the student in learning from a lesson. Again particular attention is given to constructive alignment, assessment and feedback.

1.5 Thesis Organization

- **Chapter 1** provides a context for the research area; it provides a brief introduction and outline of the problem domain. The structure of the thesis is also presented.
- **Chapter 2** presents the literature review.
- **Chapter 3** presents the experimental works that have been conducted; different methods of self assessment and feedback were used and discussed to identify how they support student learning.
- **Chapter 4** introduces a proposed educational model that could be generally applied to support e-learning applications.
- **Chapter 5** discusses the design, the implementation and the adoption of the proposed model.

- **Chapter 6** summarises the contributions of this thesis and suggests future areas of work and interest.

Chapter 2

Literature Review

2.1 Introduction

In e-learning or distance learning, courses shift from being instructor-led to learner-led as they progress. Thus instructors play a large role at the beginning of the course; they set the pace, assignments, and present information. Their role then gradually reduces in the sense of physical presence, thus encouraging the learner to construct their own knowledge rather than gaining transitory knowledge (Horton, 2006). According to Oliver et al. (2001), this approach is a learner-centred paradigm.

Throughout the learning process students need advice and support, especially if the learner and teacher cannot have synchronous meetings; this could be as a result of different time zones or communication difficulties. KeenGwe et al. (2009) argued the importance of technology in restructuring teaching and learning practices that supports learner-centered approach by providing learning tasks with tools that actively engage learners to develop higher order skills, such as problem solving and critical thinking.

Macdonald (2003, p. 378) stress the role of self assessment and feedback within the learning process; assessment that is appropriate to the course content and supporting course pedagogy can then play a significant role in student learning . Ramsden argued that assessments offer a means of helping students to learn by reporting on student achievements, the teachers can then make decisions about their future teaching practices. It is simply helping students to learn, and helping teachers to learn the best way to teach them (Ramsden, 2003, p. 177).

The following sections will further present varied research and findings that offer different viewpoints for supporting student learning. Firstly, the aim is to understand the concepts of recent theories on learning. Secondly, a brief review of current e-learning environments will be made. A comparison will then be made concerning the consequences of the usage of self assessment to support student learning. Peer assessment and the integration of assessment with text books will also be reviewed. Furthermore, recent computer assessments will be discussed and finally an over view for cognising leaning sequence used in e-learning.

2.2 Theories on Learning

There are large amounts of research concerning how students learn; these theories of learning are important to the development of educational software. To design the best e-learning systems, these types of research and theories that support student learning must be considered. In the following section, some of the most influential theories that have affected the educational system will be highlighted.

2.2.1 The Theory of Approaches to Learning

The theory ‘approaches to learning’ perceives learning from the student’s perspective; this approach was first introduced by Marton and Saljo (1976). Later, it was further elaborated by Ramsden, among others. According to Entwistle, Marton and Saljo tried to identify different levels concerning the processing of information among university students (Entwistle, 2000). Students were asked to read an academic article and they were told that they would be asked questions at a later time. It was found that the students interpreted the instructions differently, and their ability to answer the questions about the meaning of the article depended on how they had decided to tackle the task. The research focused on what was learned rather than how much the student had learned. It was found that the students tackled the task in one of two ways: some focused on the author's message; while the rest relied on 'question-spotting', thus trying to focus on what would be asked in the test. Their research introduced the concept of two categories of learning; these were deep and surface approaches to learning.

Biggs' research, 'student approaches to learning and studying', focused on how students go about learning. He identified that student's approach learning based on two factors: strategy which is how the student approaches a task; and motive which is why the student wants to approach it. His research concluded that students approach learning based on one of the following three approaches. The surface approach is where the student follows a surface strategy and is surface-motivated. The deep approach is the second approach where the student has deep internal motivations and is following a deep strategy to learning, such as searching for analogies, relating to previous knowledge, and theorising about what is learned. Finally, the achieving approach is where the student intends to reach the highest possible mark and their strategy focuses on how to maximise their chances of achieving this (Biggs, 1987).

Biggs also identified two influential factors affecting how students adapt their approach to learning: personal factors, including the student's background or personality; and the teaching factor/context, such as the time pressures in exams which encourage students to follow surface approaches rather than deep approaches to learning (Biggs, 1987).

Entwistle (1988) argued that students develop different ways of following the deep approach to learning by either relying more on facts and previous knowledge or by building up personal meaning. He presented a model of study orientations and outcomes that related deep levels of understanding to deep approaches to learning. In contrast, he related surface level understanding to the surface approach to learning.

Marton and Saljo (1984) performed a generalised insight into the manner in which students conduct and accomplish learning tasks. When taking into consideration everyday learning approach the ‘Text’ metaphors becomes pivotal. These generalized learning approaches were further defined by (Ramsden, 2003, p. 40), which in this study form the basis of understanding the different learning approaches. A summary of different learning approaches can be seen in table 2.1.

Ramsden identified that teachers must understand the concepts of the different approaches to learning to ensure that they can create better teaching strategies. The deep approach to learning involves the student practicing, understanding and relating what they have learnt in real life situations, while the surface approach makes the student focus on isolated parts of information for assessment purposes, these cannot be practiced in real life situations. Although students are capable of both approaches, educators must consider how students learn and select the most appropriate context which will increase the quality of learning. By changing the student’s approach to learning the student will not change, however, their experience, perception and conception of the thing that they want to learn will.

Ramsden (2003, pp. 119-120) acknowledges that there are many issues that the teacher must address, within any educational environment, to increase deep approaches. These include:

- Representing clear goals and structures.
- Selecting the best teaching strategies.
- Providing appropriate assessment tools.
- Addressing and watching the student progress and conduct course evaluations.
- Teachers evaluating the above issues to develop their future educational work?

Table 2.1: Deep and Surface Approaches to Learning Ramsden (2003, p. 47)

Deep	**Surface**
Focus is on 'what is signified' (e.g. the author's intention)	Focus is on the 'signs' (e.g. the word-sentence level of the text)
Relate previous knowledge to new knowledge	Focus on unrelated parts of the task
Relate knowledge from different courses	Memorise information for assessments
Relate theoretical ideas to everyday experience	Associate facts and concepts unreflectively
Relate and distinguish evidence and argument	Fail to distinguish principles from examples
Organise and structure content into a coherent whole	Treat the task as an external imposition
Internal emphasis: 'A window through which aspects of reality become visible and more intelligible'. (Entwistle and Marton 1984)	External emphasis: Demands of assessments, Knowledge cut off from everyday reality.

These points are critical for distance learning; where no face-to-face communication exists, methods need to be applied to assist the isolated students. Instructors must consider these points when deploying learning environments to assist student learning.

Another important issue that must be considered concerns knowing and understanding the student's style of learning. In Pask's theory, students can be either holistic or serial learners. Serial learners prefer to engage in learning in a step-by-step process progressing from beginning to end, to build the big picture (Pask, 1988). Conversely, holistic learners prefer to build a picture of the whole task linking the inner parts to the overall framework. Some students have versatile learning styles and can use holistic or serialistic styles, as appropriate, as an effective way to learn.

2.2.2 Constructing Learning by Aligning Teaching (Constructive Alignment)

One of the most important learning theories is the constructive alignment theory, introduced by Biggs, it encourages teaching practices to incorporate deep approaches to learning (Biggs, 1996). The theory, based on constructivism, emphasises how the learner creates knowledge rather than how they represent knowledge; their knowledge is then created by their approach to learning. The surface approach focuses on low cognitive levels of engagement which can yield fragmented outcomes; students intend to get the work done with the minimum effort needed to meet the course requirements. In contrast, deep approaches to learning represent high cognitive levels of engagement, with the learning task; these students are exited about learning, when they learn they investigate to reach understanding, reflect on possibilities, implications, applications, and consequences of what they learn. These students use high cognitive processes and are therefore able to teach themselves.

Biggs identified three student factors which can affect and encourage deep learning (Biggs, 1999). They are the academic orientation of the student, the student's level of engagement, and the degree of learning-related activity. Figure 2.1, presented below describes this concept; Biggs notes that to encourage student learning and to promote higher level thinking, students require active engagement with the learning task. In passive teaching, point A, student factors appear to make a large difference, whilst at point B, active teaching, the differences between the student factors appear to be less significant.

2.2.2.1 The 3P Model of Learning and Teaching

John Biggs' 3P model describes a relationship between teaching and learning in terms of three phases of time, which is before, during and after learning. The model focuses on the ways students choose to process academic tasks and has three stages: presage,

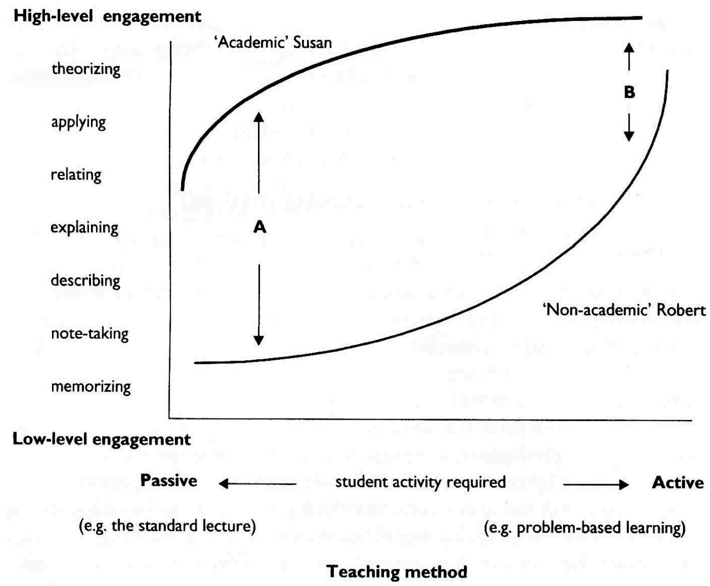


Figure 2.1: Encouraging Deep Learning (Biggs, 2003, p. 4)

process and product. Firstly, student presage factors occur prior to any learning taking place, at this phase there are two kinds of factors, student based and teaching context based. The teaching context refers to the course structure environment, set by the teacher and the institution. This environment represents: the teacher's intentions, how the course will be taught, the teacher's ability, the climate of ethos and so on. Student factors relate to the student's prior knowledge, their interest in the topic and their abilities. All these factors interact at the process phase to determine the student's learning activities and approaches to learning. Each student will behave differently depending on the factors at the presage level, see Figure 2.2 below. The product phase of the 3P model suggests that learning approaches are related to qualitative, quantitative and affective differences in the learning outcomes. A deep approach would produce high quality learning outcomes, while a surface approach would result in poor quality outcomes. The 3P model highlights three levels of thinking about teaching that might affect the learning outcomes. These levels are based on the student factors, teaching factors, and from the 3P system as whole.

- **Level 1** focuses on the student (i.e. blame the student theory of teaching). At this level the teacher focuses on the student's presage factors; the teacher then classifies

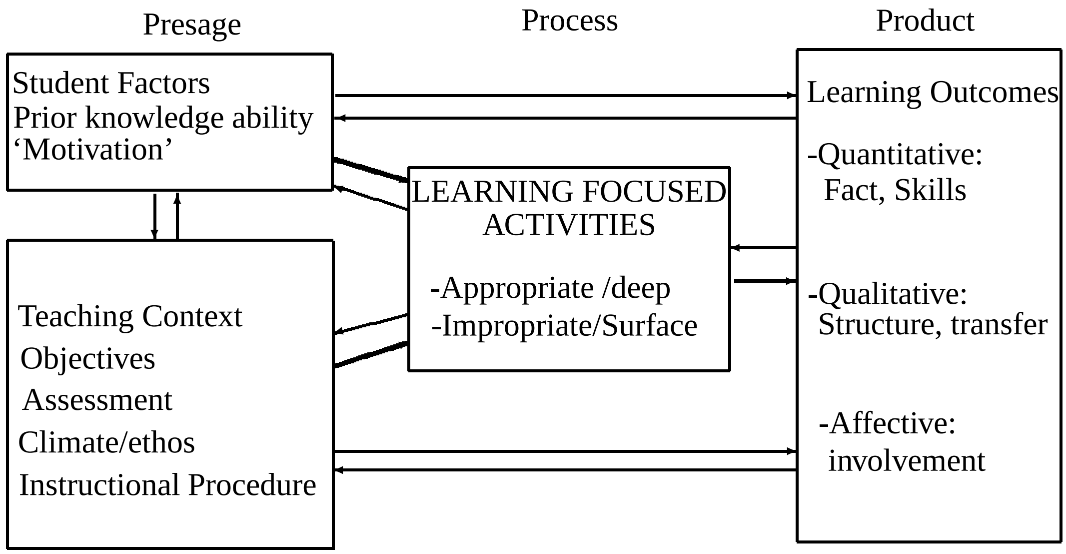


Figure 2.2: The 3P Model of Teaching and Learning (Biggs, 2003, p. 19)

the student as being good or bad. At this level, the teacher's role is to clearly set the learning environment and it is up to the student to take notes, learn or even attend lectures. Teaching at this level is constant and utilises transferring of knowledge, assessment is then used as a measurement for identifying the student's level.

- **Level 2** focuses on what the teacher does and classifies the teacher as either a good or poor teacher (i.e. blame the teacher perspective). This level results in passive learners because the teachers focus on what they are doing and not what the student is learning.

- **Level 3** focuses on what the student does, this is the most advanced level and concerns what the student does before, during and after teaching. Teachers are concerned about the product of the learning outcomes of their teaching; teaching at this level is to support learning. The level of understanding must be defined in terms of what level is required and how learning can be assessed to diagnose whether they have learned or not.

2.2.2.2 Understanding (Knowledge Perspective)

Understanding of a subject is related to how knowledge is activated and remembered. Psychologists claim that humans have poor memories and that their short-term memory retaining capability alters from about three or four elements (i.e. words, digits, or letters) to about nine elements; the memory capacity that is generally quoted is seven $7\pm$ two elements (Atkinson and Shiffrin, 1968).

Humans learn by: associating new information with knowledge that is already known; or by building new knowledge in place of older information. In teaching, teachers are rarely aware of what their students know or of how they know it. In terms of knowledge we now link this to understanding; knowledge is constructed as a result of the learning activity. Thus conflicting with the older theories that identify knowledge as being constructed as simply a transmission of knowledge from teacher to the passive learner.

According to Biggs, Tyler explained that ‘learning takes place through the active behaviour of the student, it is what they do that determines their learning, and not what the teacher does’ (Biggs, 2001, p. 224). Activation of the student itself is not enough; to understand this further, a theory explaining how a student is activated is needed. According to SOLO taxonomy, which stands for ‘structure of the observed learning outcomes’, five levels of cognitive processes are identified which describe the differing levels of understanding (Biggs and Collis, 1982):

- The first and lowest level is the **pre-structural** level which refers to the student as having no understanding or as missing the point of the learning outcome.
- The second is the **uni-structural** level, at this level students are able to identify or recite something, such as a simple procedure in a class.
- Thirdly is the **multi-structural** level where students are able to classify, combine and outline factors independently.

- At the fourth level, the **relational level**, students are able to relate, compare and analyse the learning tasks independently.
- The last and highest level is the extended-abstract level where students are able to generalise regarding the structure, beyond the information that has been given, and they can produce a hypothesis and theory.

Levels four and five refer to deep understanding and levels two and three refer to surface understanding. As a teacher, how do you get the student to learn what you want them to and how do you get the student activated so as to match the teachers' intentions? John Biggs introduced a theory to solve this problem, the constructive alignment theory, which aims to:

1. Clearly state the learning objectives of the course in terms of the SOLO taxonomy.
2. Make the exam measures precisely match the students' ability to explain, relate, prove and apply their learning.
3. Choose proper learning activities.
4. Train students' skills.

As Biggs describes in his 3P model, learning occurs best when there is a balanced system between the teaching and the learning and where all the components work together. It is important to determine how this system can be controlled best to make it work properly? The theory of constructive alignment provides the answer to this question. All of the system components need to align to work towards deep learning; constructive alignment encourages deep engagement with the learning task. Biggs' theory is based on this; learners construct knowledge through learning activities and assessment tasks. Figure 2.3 represents these concepts.

By specifying different levels of understanding of the content, and by aligning student activities, teaching methods, and assessment tasks targeted at these levels, students are

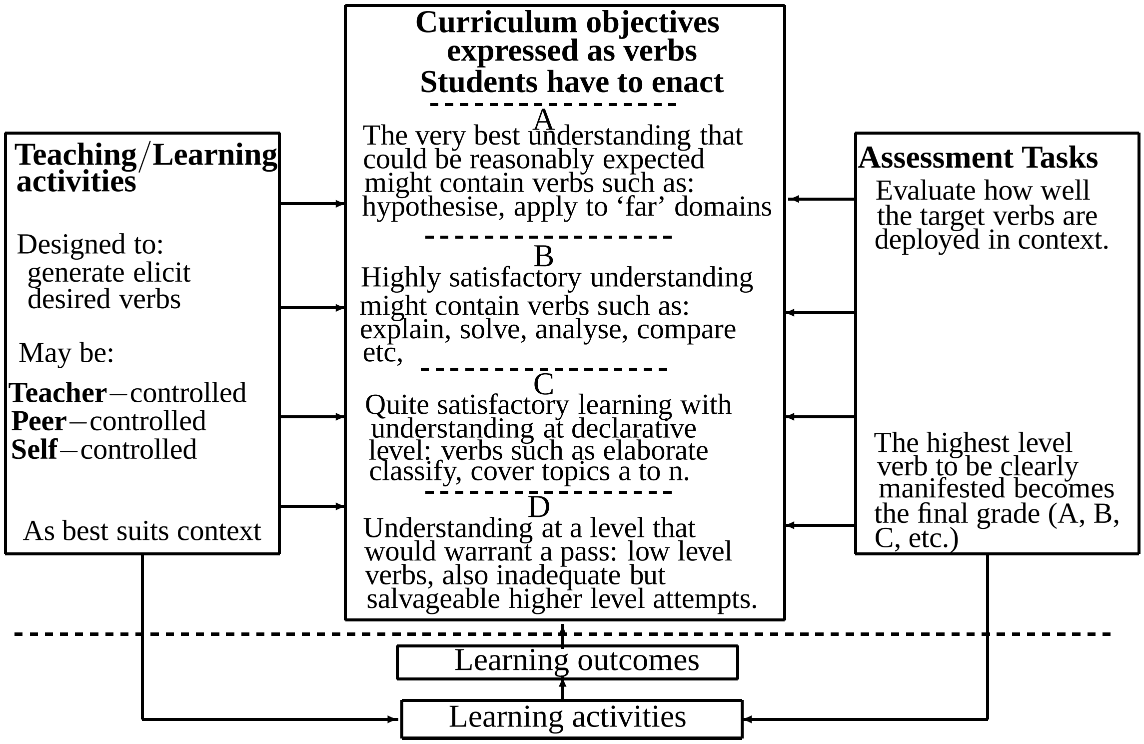


Figure 2.3: Aligning Curriculum Objectives, Teaching/Learning Activities (TLAs), and Assessment Task (Biggs, 2003, p. 28)

involved in deep learning. The teacher's role is to act as a medium between the student and the learning activities. This can be simply performed, the teacher needs to: specify the curriculum and clear state the course objectives to determine the level of understanding rather than a list of topics to be covered; choose the proper teaching methods to meet the objectives; and select assessment tasks that address the objectives and which test whether the students' learning has met the set objectives.

The constructive alignment model contains two systems, the teaching system and the learning system. The main part of the system is the middle part, presented in figure 2.3, which contains the curriculum objectives and which specifies the level of understanding. The key point of this theory is to ensure that the teaching methods and the assessment tasks are aligned with the learning activities and the intended outcomes. The learner should not be given any opportunities to escape from the intended learning outcomes.

2.3 Brief Introduction to E-Learning Technologies

E-learning technologies vary in usage and depend on the requirements of the course. They are sometimes used completely as a medium between the instructor and the student, this is known as distance learning, or they are combined with regular teaching methods, known as blended learning. Blogs, Podcasting and Wiki can be used as stand-alone e-learning tool or combined with other virtual learning environments like Blackboard and Moodle. The following sections briefly explain some of the important definitions and some of the e-learning technologies used as teaching and learning tool s.

2.3.1 E-Learning

The term e-learning has many definitions which tend to refer to learning using technological resources. Distance education is sometimes referred to as e-learning as it utilises a range of technological devices (Lance, 2000). These generally emerged in the 1980s (Keegan, 1990). In the near future, it is likely that there will be no separation between distance learning and regular courses (Turoff, 2000). E-learning can be synchronous where communication is instant between the learner and the instructor, or it can be asynchronous where the communication does not occur in real time (Broadbent, 2002).

E-learning is usually delivered using: a computer network, the internet, CDs, radio, television or even via the telephone. E-learning can be:

- Informal, where learners can access information on the web for obtaining pertinent information.
- Self-paced, where learners access web-based training materials, self directed in their own time.
- Leader-led, where instructors or facilitators lead the learning process in either a synchronous or asynchronous mode.

- Blended, where the technology is combined with regular classes for learning. Most universities currently utilise regular classes with a blended learning style. They use delivery systems, such as Black Board, for students to download course notes, submit assignments, or view any necessary announcements for that particular class.

2.3.2 Blogs

In recent years, the usage of blogs, as a technology, have grown significantly (Blogs, n.d.). A blog can be defined as a personalised diary where individuals can reveal their own thoughts and share them to the outside world. According to Blogs (n.d.), blogs have many features and provide the ability to:

- Publish and organise personal thoughts.
- Share information with others, access to a personalised blog can be controlled.
- Be easily designed, for personal blogs.
- Post pictures to the personal blog through computers, mobiles or the internet.

It is easy to start a blog, the software is free and the only restriction is that a Google account is required by the individual. The software utilised to create a blog could prove to be very useful as it could help students, as a tool, for course work projects. An educational blog can be an electronic binder or notebook, a classroom forum for discussion between the tutor and students or between classmates, support classroom activities such as class notes and quizzes (Folkerts, 2008). Like other e-learning environments, blogs are easy to start, hard to maintain, insufficient to develop more complex and dynamic learning environment.

2.3.3 Podcasting

Podcasting is a new technology where MP3 and other types of audio files can be downloaded automatically to a computer or a generic MP3 player. Software such as iTunes, iPodder, and Odeo all support podcasting. Podcasting software can be used to subscribe to chosen and favourite podcasts; when connected automatic downloads from these subscribed podcasts will occur to the computer. Audio files of learning materials has been given to students to enhance student learning experience (Janossy, 2007). Students can benefit from the lecture audio files by listening before, during or after the lecture to reinforce their learning.

2.3.4 Wiki

Wiki *Wikipedia* (n.d.) is software that allows different users to create and modify interlinked web page content. It was created by Ward Cunningham in the mid 1990s and has been described as a collaborative technology for organising information on web sites. Wikipedia, the online encyclopaedia is probably the best known Wiki on the internet.

Augar, Raitman, and Zhou described that Wiki is used as a collaborative ‘tool for facilitating online education’ (Augar et al., 2004, p. 95). It is used for interaction between members of online learning groups where participation is flexible. Participants can use Wikis to share, reflect and exchange knowledge. It can be used to collaborate on projects with students in distance, to create a discussion between teachers and students on a particular topic, or discussing brainstorm ideas for projects (Lipkewich, 2008).

To use Wiki as a collaborative effective learning tool, adequate and effective security measures must be taken because the main disadvantages of Wiki is that the web content can be created and amended incorrectly by users. Another problem is inappropriate posting of content or accidental deletion.

2.4 Virtual Learning Environments

With the recent developments in technology, virtual learning environments (VLEs) are becoming widely used for educational purposes. VLEs are developed systems that are used as a medium between the learner and the instructor. They are used by instructors to manage their courses for their students by introducing the course content, assignments and keeping track of learner progress. Conversely, the learner uses it throughout the course and in submitting given assignments. Although VLEs are mainly used for distance learning, they are also used in blended learning.

Most universities now adapt VLEs to: deliver traditional courses, and to communicate with the students. The VLE is usually hosted on a server and accessed as web pages; they regularly include templates for content pages, discussion forums, and quizzes and exercises, such as: multiple choices, true/false and one word answers. Instructors use these templates to identify course content and other information to their learners. There are many e-learning platforms developed such as Blackboard, WebCT, Moodle, ANGEL, Futurelab, and COSE. The most commonly used are Blackboard and Moodle (Kybartaite et al., 2007, p. 4).

The major limitation in using existing VLEs is the experience and knowledge of the instructor. Lack of usage and technical problems leads instructors to look for technical and creative experts for help, which will consume time and hinder the learning process. On the other hand, those systems do not provide immediate response to students questions or submissions upon their needs (Taylor, 2002). The following sections will briefly introduce some of the most recently developed ones.

2.4.1 Blackboard

Blackboard offers a virtual learning environment; it was founded in 1997 and was first released at Cornell University (Inc, 2007). The Blackboard system allows the creation and management of web based teaching materials, extensive technical knowledge is not needed as the system is relatively easy to use. Blackboard server software runs under Windows and UNIX environments and offers cross platform browser support for many users (including: Windows, Macintosh, UNIX). This has made it widely used and it has become one of the main market leaders for VLE in higher education. In summary, Blackboard software supports the following features:

- Managing course materials.
- Helping to organise students.
- Communicating with students.
- Assisting student group work.
- Creating online quizzes and exams.
- Managing student marks.

Although Blackboard is a powerful system and is widely used, it can suffer from organisational issues if it is not adopted correctly.

2.4.2 WebCT

WebCT is a web based online platform similar to Blackboard; it began in 1994 as a faculty project at the University of British Columbia Inc (2007). It was released for use in 1997 and has been adopted by 2200 colleges and universities. Its functionalities compete with the blackboard platform. It was designed to have the following features:

- It creates and maintains course materials interactively through the web.
- The course is accessible from the internet.
- It supports multiple media types.

- It supports communication tools.
- It contains a Global Calendar feature.
- It supports evaluation tools

In October 2005, BlackBoard and WebCT announced their plans to merge. In reality, Blackboard purchased its bigger competitor WebCT; the merger promised to provide the best e-learning environment within education.

2.4.3 Moodle

Moodle (Dougiamas, n.d.) is a course management system, similar to the Blackboard and WebCT platforms. Moodle is a free, open source, software which was created by Martin Dougiamas in Perth, Australia. Moodle software offers some special features which are not available in other e-learning environments, such as: the tracking of specific student activities; single question voting tools (polls); and glossaries that allow student entries, evaluations and comments.

2.5 Comparison Studies

Gayo-Avello and Fernandez-Cuervo (2003) introduced hypotheses which states that the use of web based self assessment tools can improve student learning of the theoretical concepts of computer programming . The study included a comparative between three groups that were taught the same subject: the control group, who did not receive any support; and two experimental groups: one received traditional self assessment; and the other utilised Duck *Duck* (n.d.), an online self assessment tool. The research concluded that self assessment tools improved student learning of the theoretical concepts of computer programming. Thus, signifying the importance of using self assessment as a learning tool, aimed at supporting student learning.

Kleinman and Entin (2002)introduced a comparison between online learning and in-class learning, of programming courses. They identified that many students drop out because of the technological hurdle. The online students were asked to have computers with specific minimum requirements, in terms of the hardware and software. They also needed to download specific software, such as FirstClass email software, Visual Basic Compiler and student exercise files. Many students were unable to fulfil the entire course requirements at the beginning of the term and therefore decided to withdraw from the course. The solution for solving these technical assistance problems was suggested and included the provision of a consistent and knowledgeable technical support team, available via toll free numbers, or via email, to help these students and staff.

Going beyond the authors' suggestions, this problem could be solved by providing a well organised and user friendly e-learning environment that would provide an easy way to communicate with the students. An online compiling environment, made available at the beginning of the course, could also be provided to help students to overcome these initial technological hurdles. The students could also be asked their choice of compiler (C++, Java, etc).

The authors reported that a PowerPoint presentation was provided to all of the in-class students, for each class. The students submitted their assignments in a timely fashion and they had access to a discussion group to post questions and reply to others. The authors also posted specific questions to simulate student discussion. These questions were designed 'from student mistakes in programming, from student questions or from excerpting different solution to homework exercises and asking for a comparison and evaluation of these' (Kleinman and Entin, 2002, p. 209).

The authors appear to attempt to assess the students by posting questions relating to their progress, thus forming a kind of indirect feedback. If the students reacted to the teacher's posted questions, they would increase their educational progress and reduce their weaknesses on the course. The questions posted by the students could be used as indirect tools for discovering student weaknesses and self assessment questions could be designed to support these.

Another suggestion would be to take the most beneficial questions and post them under a link called ‘frequently asked questions’ (FAQ). In this way, a search tool could be provided to make it easier for students to target the answer. This method would need to be designed in a way would not restrict the student from participating with others.

Another major problem reported by the authors was the ‘overhead in terms of instructor’s time’ (Kleinman and Entin, 2002, p. 214). The instructors would download the students’ homework attachments, type comments, save the work and then send it back to the student. This involved a lot of work for the instructor, especially when they had large classes. Instructors should be focusing on the learning process be providing quality class material and not large amounts of time marking work. This problem could be resolved by helping the instructors to utilise software, which can be designed to run automatically on submitted pieces of work, to automatically correct the programming assignments. If the program runs correctly, results could be generated immediately, with instant feedback being sent to the student identifying that the program was working perfectly and they have met the program objectives. If it was not working as expected, the work could then be forwarded to the instructor and a message sent informing the student that their work was being submitted for manual grading.

The manual grading should not be limited to what they have achieved as grades, or simple comments on their programming errors, because it might leads them to surface approach of learning. To encourage students to deep learning, instructor should interprets the misconception on individual submission and provide the learner with sufficient feedback that target the code learning objectives. This can be done by giving the students a learning activity that encourages them to practice and tackle similar programming problems that leads to the same learning outcomes.

To minimise instructor writing, a form with check boxes could be used to grade the work manually; these forms could contain formulated programming comments used by the instructor, including the following:

- Program is not well commented on. Try to use useful comments to make your program understandable and clear.
- Your program should not be resulting with fixed values, but should be run with different inputs.
- Include only proper header files into your program.

These are some of the mostly used comments in programming courses. This mechanism could reduce instructor grading time and would eliminate the instructors need to type comments to the students.

2.6 Computer Assisted Assessment and Feedback

Computer assisted assessment **(CAA)** refers to the use of computers in assessing student progress. CAA can have many purposes, they can be: *diagnostic*, to test the student's knowledge or background prior to starting a course; *formative*, to test and support the student's knowledge during the course; and *summative*, to test the student's overall grade at the end of the year. The main advantages of CAA can be summarised by the following:

- Supports student learning by offering instant access to CAA as they finish a lesson.
- Allows the instructor to: monitor student progress during the course and offer support as needed.
- Makes adjustments to the course plan based upon student progress and by detecting any areas of weakness.

- Saves instructor time in terms of marking and also reduces the number of human correction errors.
- Both the student and instructor gain immediate feedback. The student can view their: final grades from the tests; feedback on the errors that they have made, which are linked to the corrected questions and; level against the class average grade. The instructor can obtain feedback on student progress, weaknesses in specific questions, and statistical evaluation of coursework.

The above advantages indicate that well designed CAAs not only support student understanding, but also increase student confidence during a course. Furthermore, they provide the instructor with insight into the students' understanding of the course. The following sections describe different assessment types and their functionalities in supporting student needs.

2.6.1 Formative Assessment

Formative assessments can contribute to enhancing student learning during a course. They are designed to help students to practice and learn, rather than to provide a grade. Their main features can be summarised as:

- Convenient for students; they can be conducted at anytime and anywhere.
- Motivating students to learn; they can be repeated until the student is satisfied with their score.
- Helping students to measure their understanding and progress.
- Providing formative and immediate feedback.

Computer assisted formative assessments are also useful for instructors, because they provide an indication of student progress without the need for formal grading. The formative assessments are useful for indicating the individual students who have encountered complications whilst practising. Formative assessments are limited in that large amounts of time are needed to create objective questions with appropriately detailed feedback. Although some books provide question banks, it is crucial that teachers provide their learners with the correct ones.

2.6.2 Summative Assessment

Summative assessments (tests) measure and evaluate student performance in the form of a grade. They can be taken regularly throughout a course after the completion of certain parts of a course module. They are similar to formative assessments, in their format, however they differ in terms of their rules; usually completion dates and times are restricted, the results are monitored and feedback is rarely provided for the student. Summative assessments are very useful measurement tools, utilised for evaluating student progress and for targeting specific achievements. Summative CAAs are automatically marked, thus saving the instructor's time

2.6.3 Diagnostic Assessment

Diagnostic assessments, also called pre-assessments, usually occur prior to the start of a course and, determine the existing level of student knowledge. Appropriate course notes can then be introduced and the student can be directed to the most relevant level of study.

2.6.4 Surveys

Computer assisted assessments are also used in the form of surveys; instructors utilise them to evaluate course content and student satisfaction at the end of a course.

2.6.5 Peer Assessments

Sitthiworachart and Joy (2004) have developed special modules to support specific learning; they introduced software to develop cognitive skills. They claimed that peer assessments provided many uses, because they exposed students to evaluate other students' work, therefore they would exchange knowledge, understand more and improve their programming skills by looking at different solutions to the same problem. They identified the following features that they would include in their system:

- Automatic test results (students would gain an appreciation for correctness of the programs that they were viewing).
- Strict marking guidelines.
- Tutor support throughout the process.

The suggestion would be that the system would generate 50% of the marks from the teacher through automatic testing, and the final 50% of the marks would come from an average of three peers. Each peer would give a maximum of 30% for the quality of the program and a maximum of 20% for the quality of marking. 'A 5 point Likert scale was recommended in order to help students in making accurate and fair judgments' (Sitthiworachart and Joy, 2004). The author suggested that these peer assessments should be anonymous to avoid marking friends.

Evaluating other students' work has advantages: self evaluation skills will be increased and the explanations provided with the corrections will encourage the student to react. The student reactions will help the educator to evaluate the learning process to determine and estimate how well their course is understood. Conversely, fairness, in terms of evaluating other students work, needs to be considered. Who will guarantee that the three peers will be fair in their grade estimation? More attention needs to be given here: the grade either needs to be reviewed by the instructor, or the classification has to be judged by the instructor, in terms of tests, to guarantee fairness to all students.

2.6.6 CAA Software Features

Most of the recent CAA software deploy varied types of assessment and therefore provide the instructor with different options to customise and control the following:

- Timed assessments: control the time period of the assessment.
- Multiple attempts: the instructor has the option to allow unlimited attempts at a test, ‘open’, or only allow a ‘one off’ attempt.
- Test completion: the instructor has the option to force completion of the assessment, once it has been initiated, or they can provide the option to save answers and resume the test at a later stage.

Once the test or the assessment is completed, students can be provided with their total score and view any feedback given. They could also check their scores via a ‘view grades’ interface option.

2.6.7 Question Types

Most CAA systems provide different types of questions to test learners knowledge and understanding. Carter et al. (2003, p. 114,p. 159) reported that multiple choice, multiple answers, textual answers can be used to assess low and high learning skills if they adopted correctly. Bryan and Clegg (2006, p. 67) argued that although multiple choice questions are widely used in online assessments and can be used to test high-level cognitive skills depending on how they are constructed.

(Biggs, 2003, p. 84) differentiates questions types to convergent, divergent, high level and low level questions. As Ramsden argued that convergent questions usually used to solve problems that have a unique answer not necessarily low-level, where divergent questions give a range of answers, enforce student’s input and best for high-level learning. In the assessment design both convergent and divergent questions to be considered to cover all aspects of learning. For example foundation year students require to learn facts and basics before they proceed to a higher level of learning. This encourages the use of more convergent questions than divergent one.

2.7 Deployed Computer Assisted Assessment

Some of the educational software has started to adapt computer assisted assessments like Blackboard with other independent internet services, such as WebMCQ (*WebMCQ*, n.d.) and Question Mark Perception. CAAs can be based on peers and called peer assessment or self assisted assessment. Some CAA supports designing assessments for general use like Blackboard assessments, others are made spcific for certain displance like QuizPACK & QuizGuide. Each of the CAA use a different format; however if these systems could adapt a united template for the questions provided by experienced academics, and provide a way to integrate specific ones on the need like QuizPACK & QuizGuide, more beneficial and powerful learning environments could be provided. The following sections describe some general and specific CAAs.

2.7.1 Question Mark Perception

Question Mark Perception is a software utilised to support computer assisted assessments (*Question Mark Perception*, n.d.). It supports different types of assessments and enables a variety of question types to be delivered to the student. Different types of reports about the student or their assessments can be produced; these reports can be customised using different filters.

King and Duke-Williams (2001), examined the design of objective 22 questions to assist in the assessment of higher learning outcomes for two post-graduate courses. They have used two different CAAs software, Question Mark Perception and Half-Baked Hot

Potatoes. They have concluded that Question Mark Perception is secured, flexible in scoring, and for large scale delivery. In contrast, there are a considerable overhead in constructing effective objective questions in terms of time, training, and experience. To deliver higher learning outcomes CAAs need sophisticated question types, systematic and selective feedback, and linking to other software. Dalziel (2000) reported that Question Mark Perception has the feature to be integrated with question banks and can be used over the Internet for questions file upload, which gives the advantages for non-specialist assessment users by spreading up-to-date material, and ease of distribution .

Bertolo and Lambert (2007) and Lambert have used QMP software to implement CAA in Chemistry concepts to provide students with prompt feedback, engage students with the subject and reducing the lecturer's marking load. They have provide the students by two types of assessments; formative with feedback and summative with no feedback. They concluded that creating assessments using CAA system took considerable time but has made a positive impact upon the students' learning experience and save lecturer's time as no need for usual marking. In contrast, the authors reported that the students found technical problems in accessing the assessment from outside the university network due to differences in Internet providers. To solve this problem, they encouraged the students to access the formative assessment from home and summative one from a university where they can be technically supported.

Dermo argued the drivers and obstacles for assessments using Question Mark Perception software in Bradford university for his conducted initial research (Dermo, 2007). He concluded that CAAs save human financial resources improve reliability in marking, provide assessment results, item analysis data, and encourage good assessment practise. He also perceived the obstacles such as assessment limitation to assess higher level skills, high expenses of software licenses, technical failure, limitation of Internet connection, and security issues such as cheating.

Although Question Mark Perception is a very powerful tool for assessment construction and delivery, it suffers from minor complications in terms of the delivery of questions and some of its functions are not without faults, such as the ability to send broadcast emails to students. In order to use it effectively, it requires administrative technical support to solve technical problems during assessment construction.

2.7.2 Hot Potatoes

Hot Potatoes is a free assessment software (*Hot Potatoes*, n.d.). It is free for non-profit educational institutions, or for people who make their pages available to the public. It supports multiple choices, short answers, jumbled sentences, crosswords, matching/orderings and gap filling exercises.

2.7.3 QuizPACK & QuizGuide

Brusilovsky and Sosnovsky (2005) introduced two web based self assessments called QuizPACK (*QuizPACK*, n.d.) (Quizzes for Parameterised Assessment of C Knowledge) and QuizGuide (*QuizGuide*, n.d.). They introduced a system with two different approaches: the organisational approach, and the technical approach. Both approaches promise to help students to understand C++ code execution. The system provides a C++ code fragmentation and asks the student to submit the values of the parameters in the code, see Figure 2.4. The same code can be executed many times with different, randomly generated, values. After submitting the answer, the system provides either a correct or incorrect answer as feedback to the student. The QuizPACK system is an organisational approach because the instructor organises and decides which quizzes the students can take in accordance with the lectures. QuizGuide is an adaptive guide for students, so they can decide which quiz best suits their needs, see Figure 2.5. The authors identified that both systems improved student understanding and performance, although, the latter was more effective and highly reusable. The reason for this was that the adaptive style motivated the student to use more quizzes. Such a system is powerful and helpful in supporting

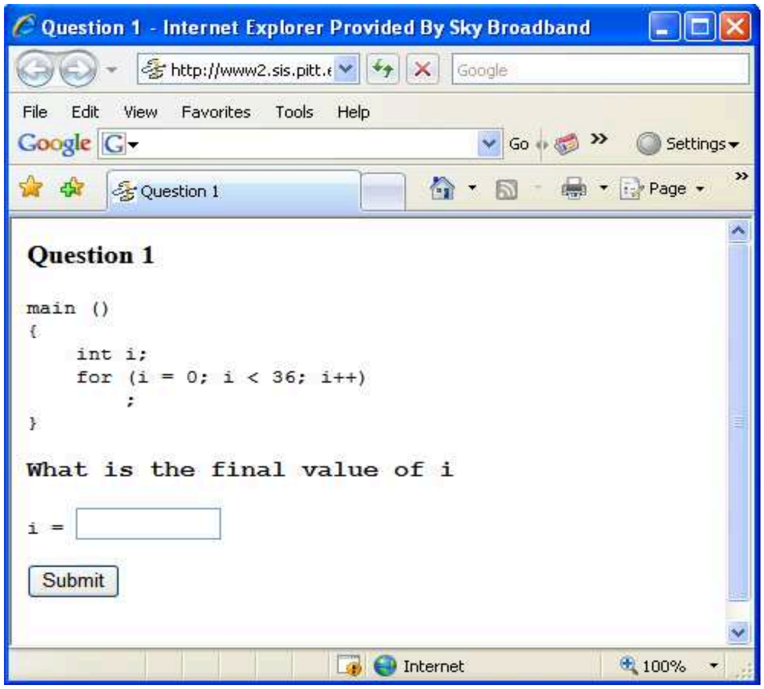


Figure 2.4: Quiz PACK Interface

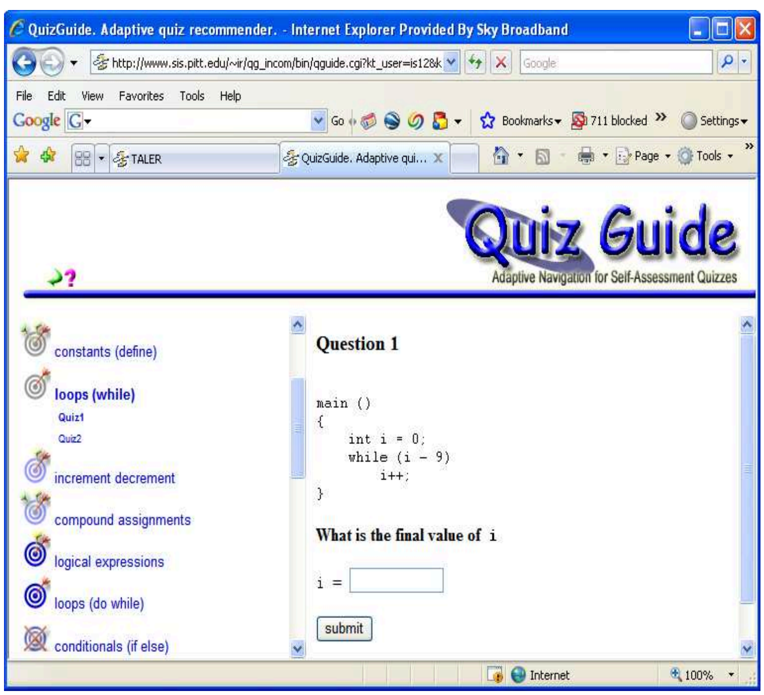


Figure 2.5: Quiz Guide Interface

student understand in programming codes. However, proper detailed feedback should be included and added for when a student submits a wrong answer. Providing the word 'incorrect' with the correct answer is not sufficient in helping the student to understand the code fragment. A suggestion for the feedback would be that it should either involve a detailed explanation of the code steps or a working applet that visualises the work of the code fragment.

2.7.4 Computer Assisted Assessment in Practice

A recent research by Iahad et al. (2004), on computer assisted assessment have proven to enhance student learning . The following type of assessment was designed to help students' understanding and give them immediate feedback regarding the quality of their work.

They developed an online test that helps students to prepare for final examinations; the research identified the role of feedback in a learner centred paradigm (Iahad et al., 2004). The lecture material was provided to the students throughout the semester. The system was developed with 50 multiple choice questions, with four optional answers; after answering the 50 questions, the students were provided with feedback containing the following:

- The assessment mark.
- Verbal suggestions about what to do next based upon the range of the mark.
- Questions that they answered incorrectly were provided with the correct answers.
- Hyperlinks to PowerPoint slides were provided to explain the correct answers.

Finally, the student completed a compulsory evaluation questionnaire to review the test; this was used later as direct feedback with the indirect feedback from the students' mark. Figure 2.6 shows an implementation chart for the online test process for this system.

The students found the system very helpful for preparing for their final exam. However,

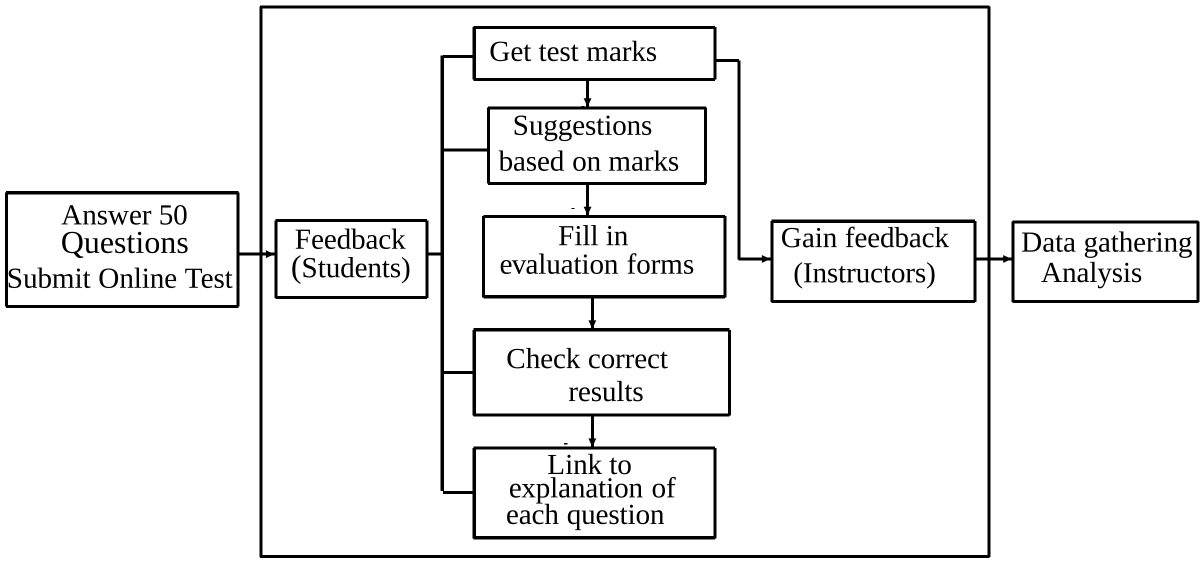


Figure 2.6: Online Test Process. Source (Iahad et al., 2004, p.4)

they reported many comments regarding the software design, asking for more detailed feedback and requesting a reduction in the length of the exam. Simple, effective self assessment and feedback systems could be developed from the above design, with some modifications, to ensure the system is more effective and useful for both the instructor and the student. Eliminating some of the unnecessary features that the above diagram contains and adding other features for the students will provide a very helpful tool that should support student understanding. The list below identifies some of the elements that are considered as weakness and which may deter students from using self assessment as a learning tool:

- Too long self assessments.
- A lack of use of graphical user interfaces.
- Too long evaluation responses.

The effective implementation of supportive self assessments may take much time and effort, however in the long term the input time and effort from the instructor could be reduced. If the instructor correctly inputs information from a qualitative database for the subject, including the utilisation of filling the question bank with effective self assessments and informative class notes, then the whole system will provide a very supportive learning aid.

2.7.5 Integrating CAA with Text Books

Dalziel (2000) reported that computer assisted assessments (CAA) can play both formative and summative roles in teaching and learning. He proposed the use of CAA via the internet, with the integration of text book question banks within web based CAA applications. Thus providing flexibility for teachers in their editing and sharing of questions amongst other teachers. This would save instructors' time, improve the quality of feedback and produce greater cost benefits.

It appears to be a good idea to devise a question bank for instructors to use for their learners which will save a lot of time. However, it is critical to discuss the quality of the question bank created, to ensure that the system is correct, updated, current and not limited.

2.8 Simple Editor and Compiler as a Learning Tool

It is noted that when studying any programming language, beginners regularly have difficulties in downloading and running specified programs. In a taught computer lab, students have their instructors support to guide them through the necessary steps to write, compile and run program codes. In distance learning courses, it is more difficult to support the student to implement and compile the examples provided. These students may be able to follow the instructor's instructions to download the specific compiler and run the codes, but they may fail to follow the instructor's overall plan and fail to compile and run the programming code. These difficulties can hinder students from learning; therefore, adequate network resources should be developed to eliminate these barriers. A simple interface could be developed that would allow students to edit and compile their codes, this would be an easy process to adopt, especially for students who are beginners to learning programming.

One of the existing internet resources is JXXX (Tschalar, n.d.). This remote compiling service was developed to enable users to compile multiple java programs using one of the platforms to which the JDK has been ported, specified compiler options are available. Figure 2.7 contains a snap shot of the JXXX interface, available on the internet.

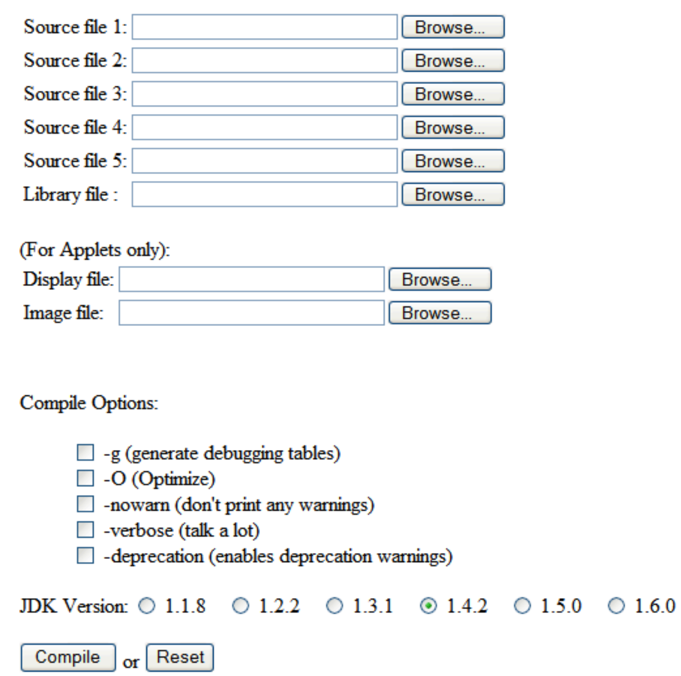


Figure 2.7: JXXX Compiler Service Snap Shot

2.9 Organise Learning Sequence

In order to organise a learning sequence, the designer must consider the different experiences that cause a learner to learn. A course is simply a collection of individual lessons, consisting of: objectives, contents, activities, assessments and feedback. The following explains Horton classifications for learning sequences as classical, activity centred, learner centred, knowledge centred, exploratory or generated (Horton, 2000):

Classical Tutorials:

are used to teach basic knowledge and skills, they are widely available as WBT. Figure 2.8 represents the structure of a classical tutorial.

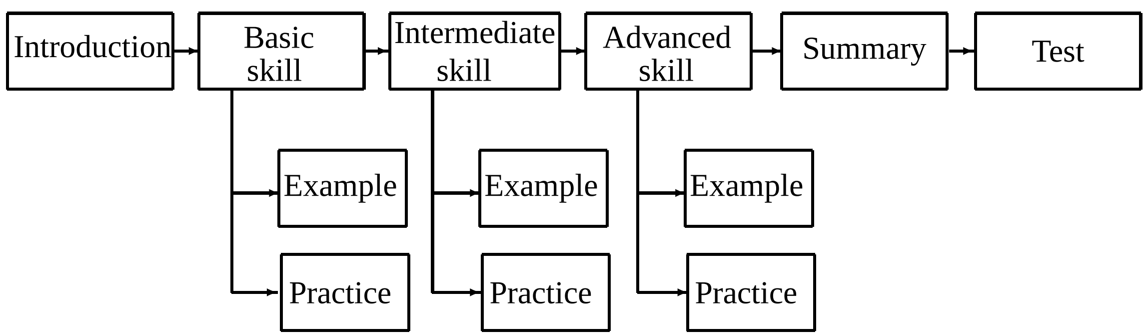


Figure 2.8: Classical Tutorial Structure (Horton, 2000, p. 136)

Activity Centred Tutorials:

are used for subjects that require rich interaction with the computer or the learner. Figure 2.9 identifies the structure of an activity centred tutorial.

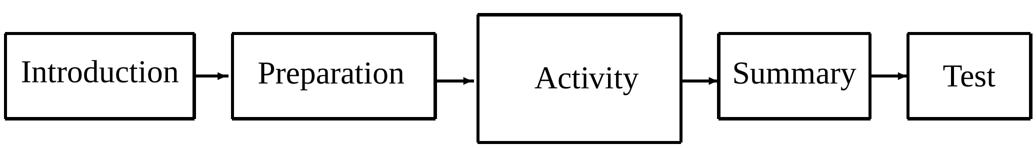


Figure 2.9: Activity Centred Tutorial Structure (Horton, 2000, p. 138)

Learner Customised Tutorials:

are used when learners need to customise the tutorial for their needs. They suit learners with high levels of knowledge and with differing learning needs. Figure 2.10 represents the learner customised tutorial structure.

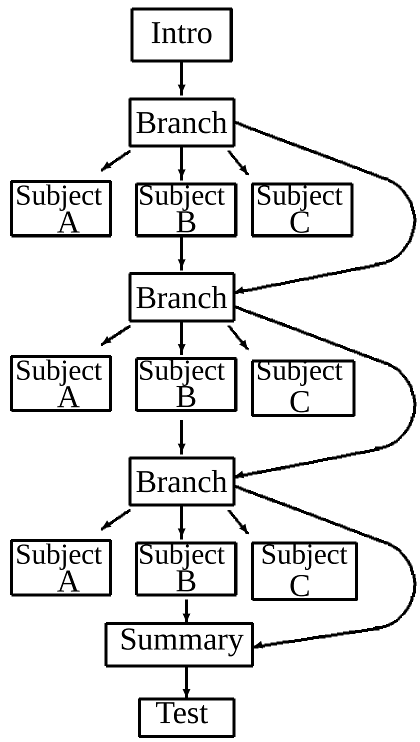


Figure 2.10: Learner Customised Tutorial Structure (Horton, 2000, p. 139)

Knowledge Based Tutorials:

are used when students have some knowledge of the subject. This structure allows the learner to skip areas where the knowledge is already known. Tests guide the learner to reach the level of knowledge that is required, they can then continue through the tutorial and end with the summary and test. Figure 2.11 represents the structure of a knowledge base tutorial.

Exploratory Tutorials:

are used by learners who are interested in finding the knowledge on their own. Usually learners self navigate and explore through the tutorials using menus, once they

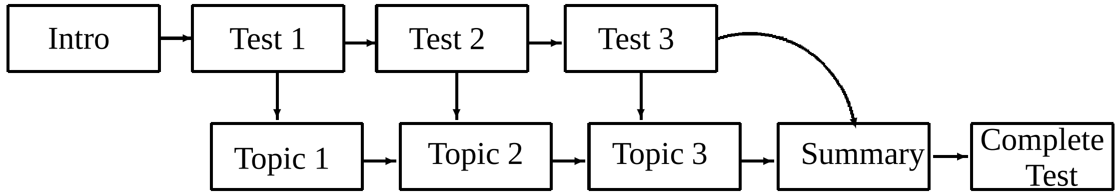


Figure 2.11: Knowledge Based Tutorial Structure (Horton, 2000, p. 141)

have finished they can view the summary and take the test. This type of structure can put the learner at risk because they can be misguided through the learning process. Figure 2.12 represents the structure of the exploratory tutorial.

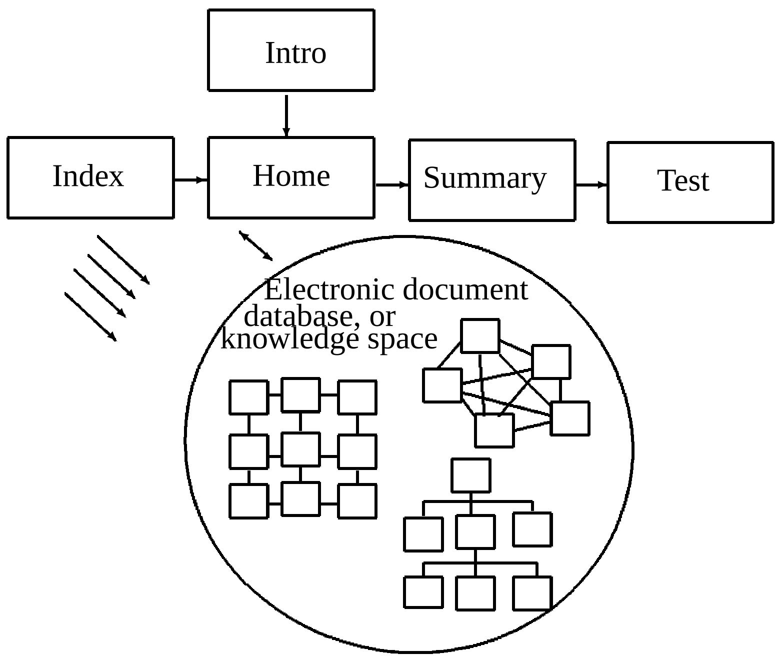


Figure 2.12: Exploratory Tutorial Structure (Horton, 2000, p. 143)

Generated Lessons:

This type of structure is similar to the knowledge based tutorial; however, the learner initially takes one test, based on the test answers, a sequence of lessons are generated for the learner. Figure 2.13 represents the structure of a generated lesson tutorial.

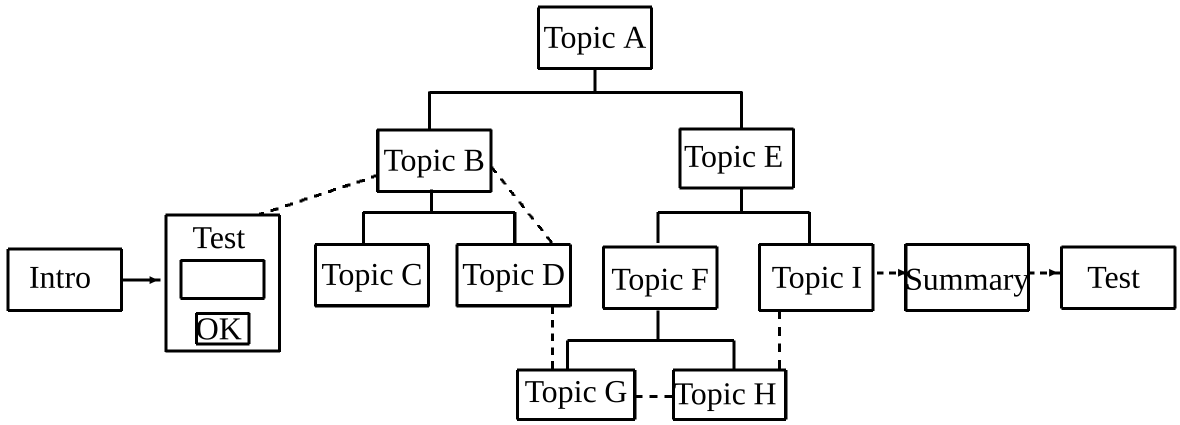


Figure 2.13: Generated Lessons Structure (Horton, 2000, p. 146)

2.10 Conclusion

The above research clearly identifies that students need assessments during the learning process, either through the use of existing systems such as Blackboard assessment or through the use of software created assessments, like Question Mark Perception (QMP). Each of these systems has its own strengths and weaknesses. Although Blackboard assessments are easy to use, they have limitations in reporting student weaknesses and in detecting misconceptions in the students' learning. Other systems, like QMP Perception helps instructors to create on and off line assessments for their students; however they are limited due to: the fixed assessment appearance, the lack of instructor input, and the lack of analysis of student data.

The next chapter will discuss a number of experiments that support e-learning environments; the role of self assessment and feedback will be emphasised as a powerful tool for enhancing student learning and as a synchronous tools for the instructor to monitor student progress.

Chapter 3

Design and Methodology for Developing Prompt and Relevant Assessment and Feedback

3.1 Introduction

The aim of this experimental work is to identify ways to increase the performance and learning of students who are isolated from their instructors. Distance learning relies heavily on the role of self assessment and feedback; the design of self assessment and feedback will be the main focus of these experiments. It will investigate what should be made for making better assessments that support students'learning.

3.2 Designing issues

Throughout the teaching process, the teacher must consider the pedagogical issues affecting a distance learner. Thus, distance learning courses should be developed in a way that incorporates: clearly stated standards and goals; the best course structure; useful course content; appropriate assessment tools; and the provision of immediate feedback throughout the learning process. The distance learning system, presented in Figure 3.1, identifies two aspects of communication: the teacher's side and the learner's side, both of which are connected.

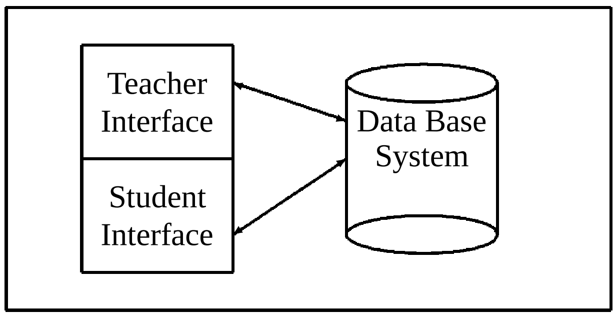


Figure 3.1: General Distance Learning System Design

When designing a distance learning course, the teacher must be able to customise the layout of the interface to ensure the student needs and subject matter requirements are accomplished. This design will affect the student's interface; the design interface should be simple and helpful for the student to ensure that the learners can easily use the system in a way that will not interfere with their learning. Designers must consider qualitative self assessment and feedback tools, because they provide important monitoring between the instructor, student and the course content.

3.3 Experiment I: The Use of Applets

3.3.1 Introduction

Iahad et al. Iahad et al. (2004) discussed the four fundamental elements within learner centred assessment introduced by Huba and Freed Huba and Freed (2000). As shown in Figure 3.2, the process starts by formulating intended outcomes. For the student, the learning outcomes help them to determine what they need to learn. The instructor will utilise these intended outcomes to evaluate whether the goals for the course were achieved. The next steps involve the development of: direct assessment measures, such as exams

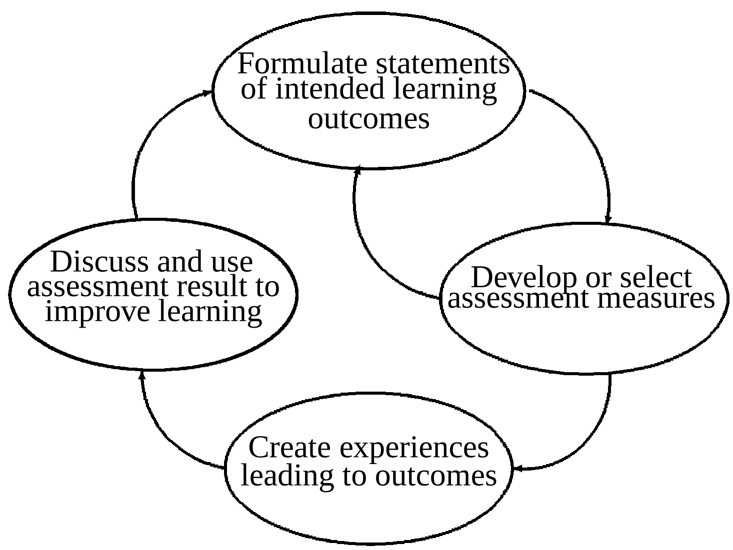


Figure 3.2: The Assessment Process Huba and Freed (2000).

and quizzes; and indirect assessment measures, such as evaluations at the end of the semester. The third element of the assessment process involves creating assessments that lead to specific outcomes, thus identifying and applying assessment techniques that will help students to attain high cognitive skills and achieve the required goals. Finally, it is important to ask how these assessment results can be used to improve learning.

This experiment focuses on most of these elements, especially the last two parts. First, formative assessments, providing proper feedback, will be formulated for the students; these will cover the weekly class objectives. The students will be notified by email when each assessment becomes available; they can practise and learn at their own pace and in their own time. Students will then be tested by summative assessments, the students will be asked about their confidence in the answers they provide; this tool will identify any unachieved goals and report any student weaknesses. The following sections describe the experiment in terms of it's: scope, objectives, procedures and results.

3.3.2 Experiment Scope and Objectives

The first experiment will be applied to a group of foundation year students, who are on an introductory Java programming course. This student group is suitable because they are beginners in the area of programming and therefore need intensive support during the course. They need proper and immediate feedback to ensure that they remain highly motivated. Moreover, the instructors need to monitor their students' performance closely throughout the course. This experimental group simulates distance learning; the group requires an environment that not only engages them in the learning process but continually ensures that their instructor is kept informed regarding student progress.

Java programming involves problem solving. The experiment aims to engage the student to acquire and gain knowledge by practising, and also to help them to adopt deep approaches to learning. As Ramsden argued that assessment is about helping student to learn, a way of reporting students' progress, a way of making decision about teaching and diagnosing students' misunderstanding in order to learn effectively (Ramsden, 2003, p.177, p.205), the objectives of this experiment are to:

- Find the best ways to help learners to 'attain higher cognitive skills as well as factual information' (Steinberg, 1991, p.100).
- Report the achieved goals.

- Help instructors to monitor students' progress during the course.
- Allow the teacher to be continually informed regarding the learning process and to ensure that they can make any required judgments during the coursework.
- Help the students to demonstrate their levels of understanding and determine any weaknesses that they might have.
- Improve student learning, by providing immediate feedback that clarifies any misunderstandings or misconceptions.

3.3.3 Procedure

The experiment will be conducted with 25 students who are, in their first year, studying Java at King Abdulaziz University. All of the students are volunteers and will receive extra course credit for participating. Students will be given regular classroom lectures and then will practise using the online assessments on the web browser and **QMP** software.

Five formative assessments and two summative assessments will be provided during the course. Each formative assessment will follow the structure illustrated in Figure 3.5. The formative assessments will have the following features:

- The assessments will be provided to the student for completion in their own time and at their own pace. This will simulate helping a distance or e-learner.
- Feedback will be supported by **interactive applets** and **textual answers** for the questions that are answered incorrectly; and their final scores will be provided. This will help them to gain formative feedback to improve their learning and get their level of understanding.
- For each programming question, there is a programmed applet that describes the behaviour of the programmed code. The applet helps the learner to trace the code,

line by line, and provides a verbal description for each line in the code, until the code is finished.

- The students can restart the applet at any time; some of the programmed code will have varied values to illustrate to the student the behaviour of the given code with different values. This will help to understand the code from different perspective.
- If their score is below 75%, the student will be asked to repeat the assessment and review the feedback provided. Scoring below 75% means that they are not yet have targeted the required goals.

The first summative assessment will be conducted part way through the course, after the second formative assessment. The last summative assessment will be delivered at the end of the course. The structure of each summative assessment is illustrated in Figure 3.6. To ensure quality learning occurs and to review student performance:

- Timed summative assessments will be given to the student, these assessments will review the same objectives covered in the formative assessments.
- The summative assessments will be conducted in the lab during a timetabled session.
- Students will be asked to answer the questions and also report their confidence in their answers.
- No textual or applet feedback will be provided with the summative assessments. The students will obtain their final scores after their submission.
- The first summative assessment will occur after the first two formative assessments and the second will occur after the last three formative assessments.
- The two summative assessments will target the same objectives, in terms of what the student has learned, as the formative ones.

3.3.4 Assessment Design

Figure 3.3 shows an enhanced self assessment design that promises to help student understanding and at the same time informs the instructor of the students' progress.

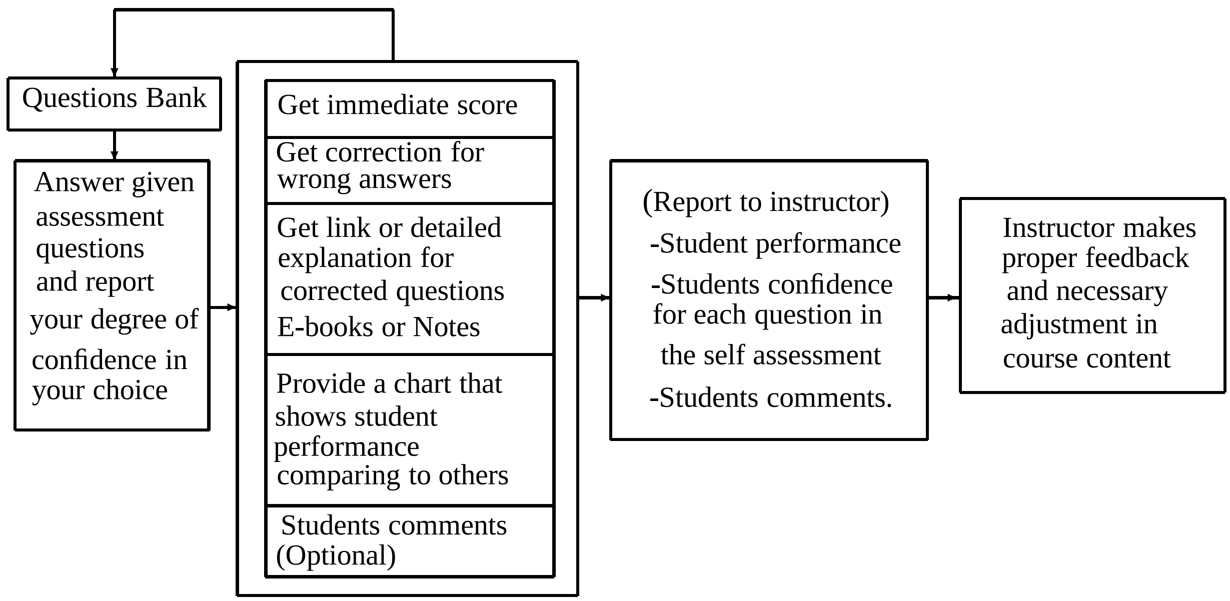


Figure 3.3: Proposed Design for Advanced General CAA, Adapted from (Iahad et al., 2004, p.4)

The self assessment tool will be designed to incorporate the following features:

- Easy to use.
- Immediate feedback will be provided using the students' scores and charts that relate the students' performance with others.
- Feedback will be provided for the wrong answers.
- Links or detailed explanations will be provided for the wrong answers. These could be linked to e-books or the original lesson, to identify the specific related material.
- The recording of students' confidence in their multiple choice test answers will be

provided to improve the quality of the feedback, as reported by Fone (2002). He suggested that students should choose an answer and at the same time state their level of confidence that they have in their selection, by choosing: very confident/confident/uncertain/very uncertain. In Table 3.1, we have summarised the confidence reported result and links their level of feedback. This method could be used to support distance learners and to help the instructor to eliminate any misconceptions identified by the student.

- Provide the students with the opportunity to give verbal feedback during their use of self assessment.

Table 3.1: Reporting Students' Confidence and its Implications

Answer + Degree of certainty		Result
Correct answer + very confident	⇒	Students understand the question. No feedback required.
Wrong answer + uncertain	⇒	Students need some help to understand. Feedback required from the instructor.
Wrong answer + very confident	⇒	Misconception identified Student needs help. Instructor has to give more feedback.
Correct answer + low confidence	⇒	Misconception identified. Feedback can be offered to raise confidence in the students selection.

3.3.5 Supporting Applet Library as a Feedback Tool for Students

One of the major challenges facing distance learning instructors is the ability to clearly describe certain algorithms and procedures to students, to link the output with the running algorithms. In normal taught classes, it is easier to follow and describe a computer program output with the respective running code and it is easier to describe the network queuing delay steps to students. To overcome these problems, distance learning environments need well designed supporting applet libraries that are classified according to the subject matter. These applets should be ready to be linked by the instructor to the students' interface. If the applet library was classified by the subject matter, then: the Data Structure Library could contain all applets that describe any subject relating to them, such as Sorting, Searching or Hashing algorithms.

The running applet should be designed to contain all the necessary elements that will guide and help the student to understand. For example, Figure 3.4 illustrates the necessary elements an applet must contain; however, the design of any applet template is likely to vary and be dependent on the functionality of the algorithm.

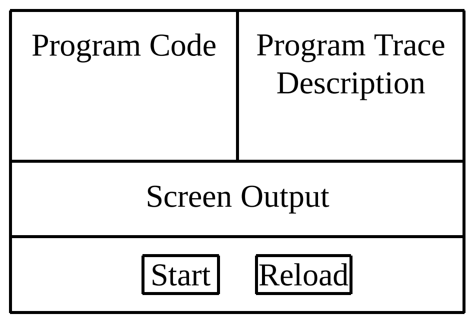


Figure 3.4: Design of Library Data Structure Template

The experiment has two types of assessment: formative and summative. Each week the students will be provided with class handouts, clear objectives, readings, normal lab sessions, examples and resources. The experiment will add two elements to help the student to improve their learning; from the concept of supporting students in distance learning:

- Formative assessments: include things to be practised before taking the summative assessments.
- Summative assessments: evaluate student performance, goals achieved, and monitor the course objectives.

3.3.6 Formative Assessment Model

Figure 3.5 illustrates the formative assessment to be utilised. The student can work on this open assessment in their own time; they will have unlimited access to it, meaning that they can do it anytime and as often as they like.

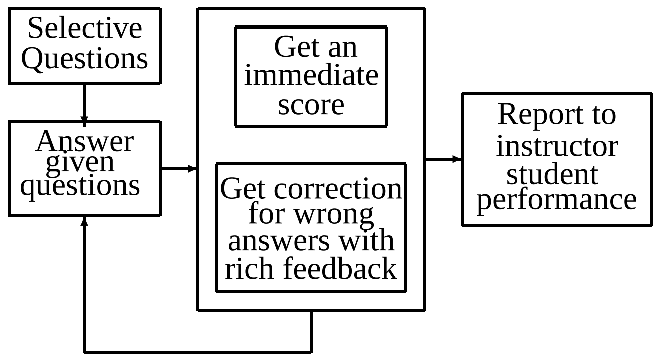


Figure 3.5: Design of Formative Assessment Used

3.3.7 Summative Assessment Model

Summative assessments evaluate the students' overall achievements. They are designed to evaluate student performance and report any unachieved course goals. They help the instructor to make any course adjustments needed to support the students' learning by giving additional group or individual instruction: based on the specific needs. Figure 3.6 illustrates the summative assessment that will be used with these students.

3.3.8 Assessment Elements

1. Question bank: a collection of varied questions relating to the introduced topic.

2. Question types: in Bloom's taxonomy of knowledge, comprehension and application are considered to be the lower levels of the cognitive domain and the ones most

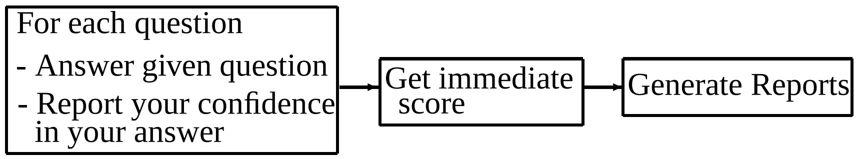


Figure 3.6: Design of Summative Assessment Used

frequently tested using multiple choice questions (Carter et al., 2003). Higher levels of the cognitive domain, including analysis, synthesis and evaluation, cannot be assessed by multiple choice or short answers.

Programming involves dynamic input and a variety of solutions to the same problem; therefore in this experiment, many different types of questions will be provided to the students, including:

- Multiple choice (true/false, yes/no, one from many, multiple response).
- Numeric and text outputs.
- Ordering.
- Matching.
- Problem solving.

3. Immediate feedback will be provided to students, including:

    - Their grades, they will be able to view their grades compared to the class average.
    - Viewing the corrected questions, with detailed explanations of:
        - the correct running applets;
        - referenced explanations using the class notes and any books;

    - why it is the correct answer.

4. Feedback to the teacher, using:

    - The students' grades.
    - The students' confidence in answering each question, this will lead the teacher to report any student misconceptions about the course material.
    - Students' comments on the assessments.

5. Student comments.

3.3.9 Assessment Questions

Assessment questions will be formulated to reinforce the learning outcomes. Bloom's taxonomy will be applied because it is the most commonly utilised theory applied to support cognitive attainment (Carter et al., 2003); Bloom identified six levels within the cognitive domain. The first and lowest level is knowledge, this level is also called simple recall or recognition of facts, and it is used for testing the memory of previously learned materials. The second level is comprehension and is concerned with explaining and summarising. The third level is application, which is concerned with applying information to produce results or to solve problems.

Knowledge, comprehension and application are considered to be from the lower levels and can be tested using multiple choice questions. Analysis, synthesis and evaluation are the upper levels of the cognitive domain; these levels cannot be assessed using multiple choice or short answer questions.

The assessment questions will follow the objectives of the class notes and will cover the three lower levels of Bloom's cognitive domain. Here are some examples of the given formative assessments; the feedback, the level of the cognitive domain and the question objective will also be identified for each example:

Example (1):

What is the name of the java file containing this program?

```
import myLibrary.*;
public class ShowSomeClass
{
        / code for the class...
}
```

a. ShowSomeClass

b. ShowSomeClass.java

c. ShowSomeClass.class

d. Any file name with the java suffix will do.

Feedback:

In Java programming, the source file name should be the same as the class name proceeded by the '.java' extension. This means that the file name should be **ShowSomeClass.java**

Level: Knowledge.

Question Objective: Naming the Java source file.

Example (2):

$$x = \frac{a + b - c}{2d}$$

Which one of the following Java expressions represents the above equation?

a. x = a+b-c/2*d;

b. x = (a+b-c)/(2*d);

c. x = (a+b-c)/2*d;

d. x = a+b-c/(2*d);

Feedback:

If the last option is selected, the feedback will be as follows: incorrect, because the selected expression would follow the flow in the diagram below, see Figure 3.7.

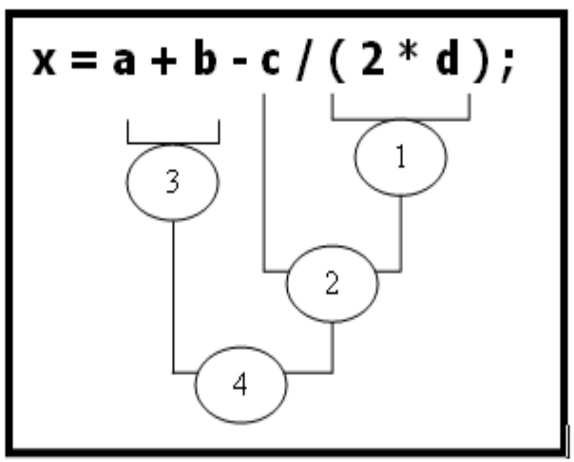


Figure 3.7: A Snap Shot of Feedback Given for the Selected Expression

Level: Understanding.

Question Objective: Evaluating expressions.

Example (3):

What is the output of the following code?

```
int i, j;
    int [] a = new int[5];
    for ( i = 0;i< a.length; i++)
        a[i] = 2 * i;
    for ( j = 0; j< a.length; j++);
        System.out.print(a[j] +" ");
```

a. 2 4 6 8 10

b. 0 2 4 6 8

c. 2 2 2 2 2

d. 0 1 2 3 4

Feedback:

This code prints 0 2 4 6 8. **Click here** to trace this code and understand how it works. Figure 3.8 is a snap shot of the provided applet.

Level: Application.

Question Objective: Tracing the array elements.

3.3.10 Analytical Results

Once the assessment has been submitted, all student data will be stored in the database. This data includes:

- Question and confidence answers.
- Scores for each assessment.

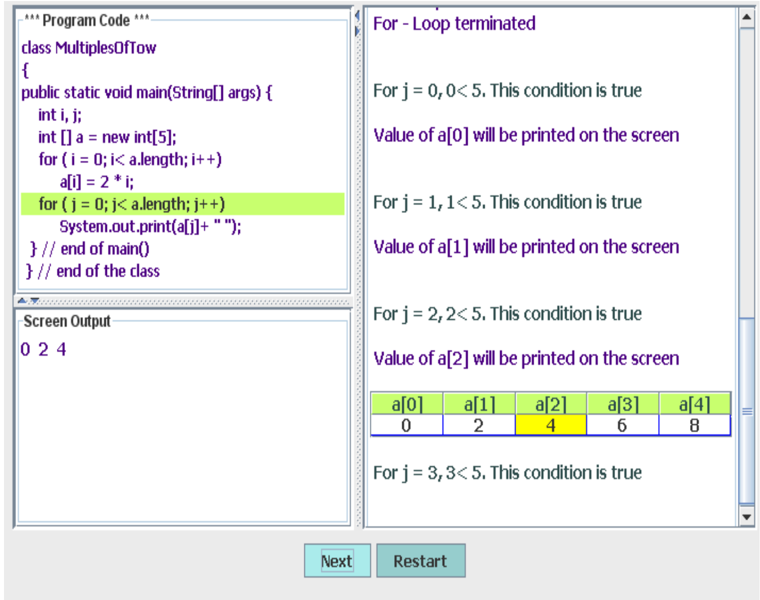


Figure 3.8: The Tracing Applet, as Feedback for the Student

- Date and time spent on each question and assessment.
- When the assessment has been started and ended.
- Status of the completion of the assessment.

The data will be reviewed and analysed to assess the students' performance, any weaknesses will then be reported with regards to the specific course objectives. The initial findings from the formative assessments indicated that the students tried to practise more than once. Furthermore there appeared to be a correlation between the students' scores and the order of the same formative assessments taken. The students utilised the feedback provided from the assessments to influence their future learning. As an example, Figure 3.9 shows the results obtained for a student who completed the same assessment seven times.

The same results have been shown for some other students. Most students repeated the assessments to review and solve the questions that they had answered incorrectly. A possible conclusion is that it would be much more convenient for the student to be able to customise their assessment questions, so they could choose to practice only those that

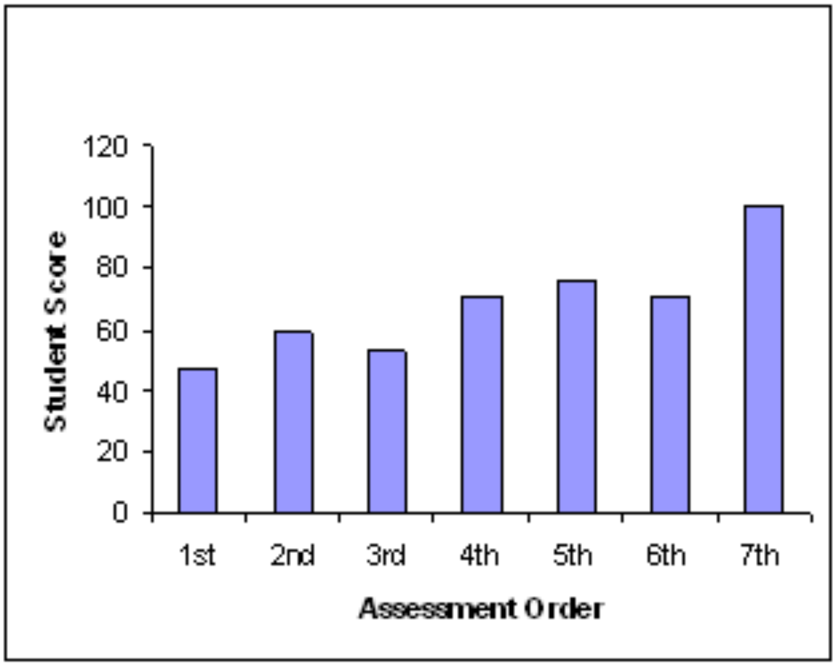


Figure 3.9: Relation between Student Score and Assessment Order

they desired to gain further knowledge from and leave aside those they already knew.

The data was analysed according to the declared objectives of the questions. For simplicity, the objectives in Tables 3.2 and 3.3 have been given numbers. Table 3.2 shows data analysis from the first summative assessment; this assessment was taken after the first two formative assessments and each of the listed questions covered certain objectives in the module. Although the data was gathered from a small group of students, the following results were concluded as evident. Questions three, five, six, eight and ten had low average scores with high percentages of confidence. These yield the identification of misconceptions, appropriate action needs to be taken with regards to these questions' objectives. At the same time, personalised feedback could be provided for each student, according to their specific areas of weakness.

Table 3.3 identifies the students' names along with the weaknesses detected for the specific objectives. Early and individual attention could be given, for each student, to help them overcome any difficulties encountered during the course.

Table 3.2: Assessment 1: Question Analysis

Question No.	Max Score	Mean Score	Confident%	Unconfident%
Q1	1	0.86	86%	14%
Q2	1	0.86	100%	0%
Q3	1	0.14	57%	43%
Q4	1	0.71	71%	14%
Q5	1	0.0	14%	86%
Q6	1	0.29	100%	0%
Q7	1	0.71	71%	14%
Q8	1	0.43	100%	0%
Q9	3	1.29	86%	14%
Q10	1	0.0	71%	29%
Q11	3	1.57	14%	86%
Q12	1	0.71	86%	14%

Table 3.3: Assessment 1: Students Analysis

Student Name	W/C	W/U	C/U	C/C	Total Score
Student1	5,6,8,9,10,12	none	11	1,2,3,4,7	9/16 56%
Student2	3,8,9,11,12	5,10	4	1,2,6,7	6/16 38%
Student3	6,10	3,5,11	none	1,2,4,7,8,9,12	10/16 63%
Student4	6,7,10	3,5	11	1,2,4,8,9,12	9/16 56%
Student5	6	3,4,5,9,10,11	7,12	1,2,8	7/16 44%
Student6	1,2,3,6,8,9,10	5,7	11	4,12	5/16 31%
Student7	3,4,8,9,10	5,11	1	2,6,7,12	7/16 44%

- W/C Wrong answers + Confident
- W/U Wrong answers + Unconfident
- C/U Correct answers + Unconfident
- C/C Correct answers + Confident

The second summative assessment occurred after two further formative assessments. Tables 3.4 and 3.5 identify the overall weaknesses observed for the students. The tables show the average scores for each question together with a percentage representing the students' confidence in their answers. This analysis clearly identifies where students have misconceptions; for example, questions one, two, 11, and 13 have low averages and very high percentages of confidence. These misconceptions need to be considered and reported to the instructor, to ensure that appropriate action to improve the students' understanding occurs, perhaps by further exercises and improved feedback. The table analysis provides valuable information for the instructor to identify, monitor and follow student progress.

Table 3.4: Assessment 2: Question Analysis

Question No.	Max Score	Mean Score	Confident%	Unconfident%
Q1	1	0.2	80%	20%
Q2	1	0.2	80%	20%
Q3	1	0.2	40%	60%
Q4	1	0.75	60%	20%
Q5	1	0.6	100%	0%
Q6	1	0.6	80%	20%
Q7	1	0.2	40%	60%
Q8	1	0.6	100%	0%
Q9	1	0.2	40%	60%
Q10	1	0.8	60%	40%
Q11	1	0.0	80%	20%
Q12	1	0.8	80%	20%
Q13	1	0.4	60%	40%
Q14	1	0.0	40%	60%
Q15	1	0.2	60%	40%

To analyse the effectiveness of the use of applets as feedback tools, the scores for 12 questions were gathered (see **Appendix A**). For each question in the formative assessment where applets were used, there was a question in the summative assessment where no applet was used. The same sets of questions were given to each student, but with different parameters. Table 3.6 presents the grades for the final formative assessment results and

Table 3.5: Assessment 2: Students Analysis

Student Name	W/C	W/U	C/U	C/C	Total Score
Student1	1,2,6 8,11,13	3,7,9 14,15	none	5,10,12	3/15 20%
Student2	2,3,5 11	1,7,9,10 12,13,14,15	6	4,8	3/15 20%
Student3	1,2,3 5,9,11,15	7,14	none	4,6,8 10,12,13	6/15 40%
Student4	4,6,7 8,4	2,11,13	3,9,10	1,5,12,15	7/15 47%
Student5	1,9,11 14,15	3	none	2,4,5,6 7,8,10,12,13	9/15 60%

- W/C Wrong answers + Confident
- W/U Wrong answers + Unconfident
- C/U Correct answers + Unconfident
- C/C Correct answers + Confident

the summative assessment results for the questions with supported applets. To determine whether the difference between the two means was statistically significant (Urdan, 2005), a two sample paired t_test was applied to the formative and summative assessment results. The paired t-test is used to compare the values of means from formative and summative assessments to test whether the students benefit form learning with applets as a formative and immediate feedback or not.

The statistical analysis of paired data is performed on the differences between the pairs, and for this data the mean difference (Summative Assessment _ Formative Assessment) between the scores is 0.06. Suggested null and alternative hypotheses are:

H0: On average there is no difference between students scores in the two assessments.
H1: On average there is difference between students scores in the two assessments.

Table 3.7 shows the results from performing a paired samples t-test on the students for-

Table 3.6: Students Scores

Formative Assessment Score Supported with Applets	**Summative Assessment Score**
4	6
5	7
7	8
9	8
10	9
11	11
11	11
11	12
12	12
12	12
12	12
12	12
12	13
12	13
12	13
12	13
13	14
13	14
13	14
14	14
14	14
14	15
14	15
14	15
14	15

mative and summative assessment results giving a p-value of 0.14. Thus the probability of getting a difference of 0.06 between the mean scores is 0.14 or 14.0%. This is not sufficiently low to conclude that use of applets affects their performance. Therefore, we fail to reject the null hypothesis with this data, and conclude that there is insufficient evidence to suggest a difference between the two tests, on average, in the two assessments.

Table 3.7: t-Test: Paired Two-Sample for Means Using Applets

	Formative Assessment	**Summative Assessment**
Mean	11.48	12.08
Variance	7.26	6.83
Observations	25	25
Pearson Correlation	0.71	
Hypothesised Mean Difference	0	
df	24	
t Stat	-1.53	
$P(T \leq t) one - tail$	0.06	
$tCritical one - tail$	1.71	
$P(T \leq t) two - tail$	0.14	
$tCritical two - tail$	2.06	

3.4 Experiment II: The Use of the JT Application

3.4.1 Introduction

Java applets are easy to use and very informative, however they are not without their limitations; their functionality is restricted because they only provide descriptions of the codes that they are linked to. Students are able to understand the code that is given because different variable values in the code are identified. Figure 3.10 shows an example of a Java applet utilised to provide feedback in a formative assessment. As Java programs get larger and more complex, students need more support and better interactive mechanisms to motivate and engage them in learning. The idea of the Java Translator (JT) application developed from the need to support students to learn Java whilst being isolated from their instructor and peers. Formative Java assessments are known to improve student learning. The quality of feedback provided is also important to support learning. The JT is a simple and basic application used to support the learning of programming (Alansari et al., 2008*a*,*b*); it can be used as a stand alone application or as part of a related learning environment. It is similar to Java applets that support the learning of programming by giving relevant and descriptive feedback. However it is more powerful in nature, because learners

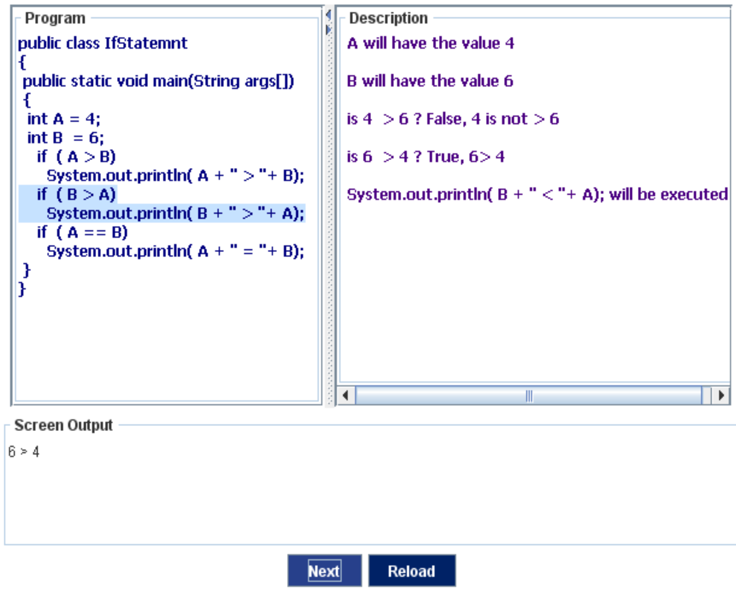


Figure 3.10: Tracing Applet as Feedback for the Students

can choose the code they want to study and JT application provides prompt and relevant feedback based on the learner's choice. The JT application provides a synchronised description for each line in the program code. Figure 3.11 illustrates the JT application. The following sections will present and discuss a number of experiments which identify

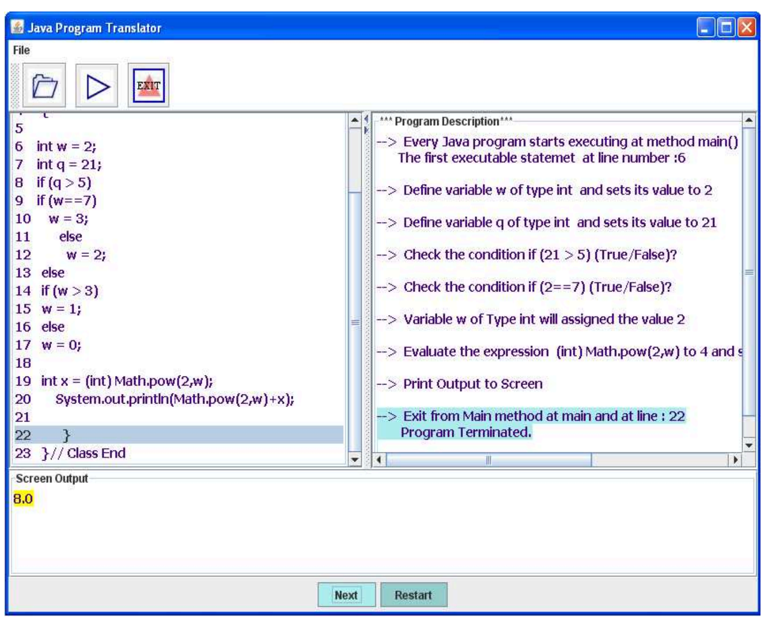


Figure 3.11: A Snap Shot of the JT Application

different tools to support learning.

3.4.2 Application Design for JT Application

Java Translator is a type of interface design software that allows learners to learn Java programming; it takes Java program code and translates it into readable and understandable descriptions for the learner. JT has two main strengths: generalisation and instance; providing learners with instant translations of any Java program code, as they need them. The JT is built on a Java Debug Interface (JDI) to ensure all variables, methods and running sequence are incorporated within the JT application.

3.4.3 The Java Debug Interface (JDI)

The JDI, in Figure 3.12, forms part of the Java Platform Debugger Architecture (JPDA) (*Java Debug Interface*, n.d.). It defines the mirrors for values, the types, and the targeted virtual machine itself. The JPDA consists of three parts, these are identified as follows:

- Java Virtual Machine Debugger Interface (JVMDI): is the native interface that each Java virtual machine has to provide to be debugged. It includes requests for information, actions and notifications.
- Java Debug Wire Protocol (JDWP): is the protocol that defines the format of information, and requests the transfer between the debugged process and the debugger at the front-end.
- Java Debug Interface (JDI): is implemented at the front-end and it defines the information and requests at a user code level. This interface facilitates the integration of the debugging capabilities into the development environment.

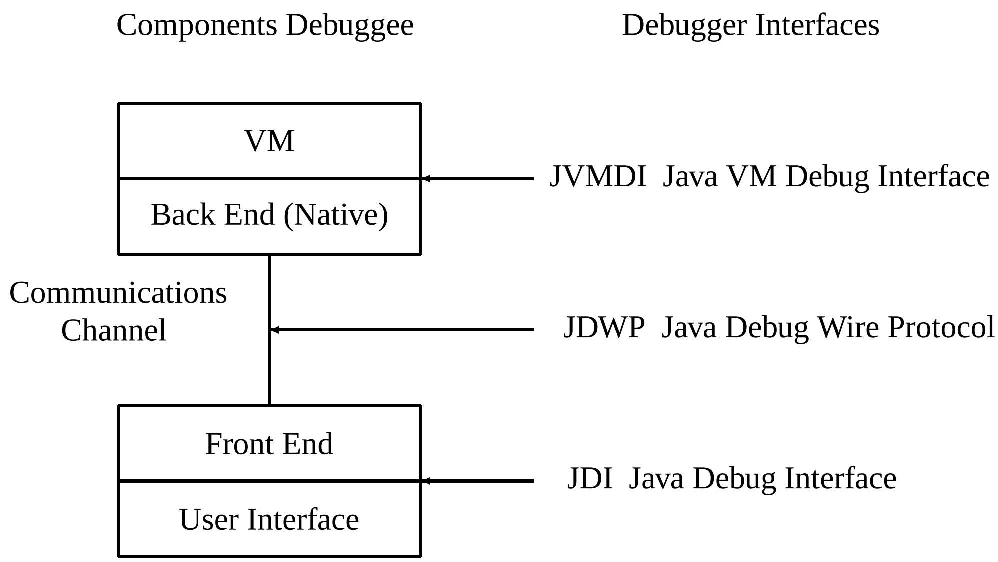


Figure 3.12: The Java Platform Debugger Architecture (JDPA)

3.4.4 Procedure

Once the JT application is developed, another experiment will be conducted to determine the effectiveness of the application. The experiment will be applied to 25 students who are, in their first year, studying Java at King Abdulaziz University. All the students will be volunteers in the experiment and will receive extra course credit for participating. The students will receive assessments to help support them to learn Java programming. The group will be given five different formative assessments that will cover the course content. Feedback will be supported via the interactive JT application and textual answers will be provided for any questions answered incorrectly. Their final scores will also be provided. If the score is below 75%, the student will be asked to repeat the assessment and utilise any feedback given. The assessments will have no time constraints and the student will be able to complete them in their own time and at their own pace.

The JT application will be provided prior to the assessments, to ensure that the students are familiar with the system and understand and learn how the code works. For the formative assessment, the students will be given a code as a feedback for the worng answers, to help them to understand correctly. They will be asked to run the JT application and trace the code, they will be able to change the code for their needs and use the JT application to test and learn, step by step, how the code is running. The students will also be given two summative tests to check their understanding. No feedback will be given at the test stage and the students will only obtain their final scores after their submission. The first summative test will be taken after the first two formative assessments and the second will be taken after the last three formative assessments; both tests will address the same learning goals.

3.4.5 Results

The effectiveness of the JT application will be analysed based on the scores gathered from 12 questions where the JT application was utilised as a feedback tool. Table 3.8 shows the student grades for the final assessment results and the formative test results. Table 3.9 represents the results from the two sample paired t-test, which was also applied to the formative and summative assessments results. Suggested null and alternative hypotheses are:

H0: On average there is no difference between students scores in the two assessments.

H1: On average there is difference between students scores in the two assessments.

Table 3.9 shows the results from performing a paired samples t-test on the students formative and summative assessment results giving a p-value of 0.14 with the mean difference (Summative Assessment _ Formative Assessment) between the scores is 4.24. Thus the probability of getting a difference of 4.24 between the mean scores is 6.95E-09. This is sufficiently low to conclude that use of JT application affects their performance. Therefore, we fail to reject **H1** hypothesis with this data, and conclude that there is sufficient evidence to suggest a difference between the two tests, on average, in the two assessments.

Table 3.8: Supported with JT Application and Text Scores

Formative Assessment Score Supported with JT Application	**Summative Assessment Score**
1	7
5	12
6	12
6	13
7	13
7	13
8	13
8	13
9	14
10	14
10	14
10	14
10	14
10	14
10	14
11	14
11	15
11	15
11	15
12	15
12	15
12	15
13	15
13	15
14	15

Table 3.9: t-Test: Paired Two-Sample for Means Using JT Application

	Group A	**Group B**
Mean	9.48	13.72
Variance	8.68	2.88
Observations	25	25
Pearson Correlation	0.56	
Hypothesise Mean Difference	0	
df	24	
t Stat	-8.70	
$P(T \leq t) one - tail$	3.48E-09	
$tCriticalone - tail$	1.71	
$P(T \leq t) two - tail$	6.95E-09	
$tCriticaltwo - tail$	2.06	

3.4.6 SUS Questionnaire for JT Application

It is not only important to measure the students' performance, but also their experience of using the JT application. The system usability scale (SUS) questionnaire was designed to measure effectiveness, efficiency and satisfaction (Brooke, 1996; Jordan, 1996). Effectiveness refers to the ability of users to complete the tasks using the system, and focuses on the quality of the output from these tasks. Efficiency identifies the level of resources consumed in performing the tasks, and satisfaction in terms of the users' subjective reactions to using the system. Table 3.10 shows the SUS questionnaire format. The SUS

Table 3.10: System Usability Scale Questionnaire

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

questionnaire was introduced by Brooke (1996); it is a 10-item questionnaire that provides an overview in terms of satisfaction of the software. The questionnaire uses a five point Likert scale to indicate the degree of agreement or disagreement; five identifies strongly agree and one refers to strongly disagree. The questionnaire was given to all students who used the JT application; Question Mark Perception software was used to deliver the questionnaire.

Data was gathered to score the JT in terms of usability and satisfaction. Each student's score was recorded by calculating an overall answer for each student, because the scores for individual items alone were not meaningful (Brooke, 1996). The mean score (50.9 out of 100) is at the SUS scale.s mid-point means that the usability of the product is satisfactory and no improvements are needed (Bangor et al., 2009).

The following figure represents the scores from the SUS questionnaires for the JT application. The scores, shown in Figure 3.13, indicate that 11 out of 30 users had scores below 60; the remainder had scores over the mid-point of 50.9 (out of 100) . The highest score obtained was 100 (out of 100) and the lowest was 35. The mean SUS score was 68.1 which means that JT application has perceived usability.

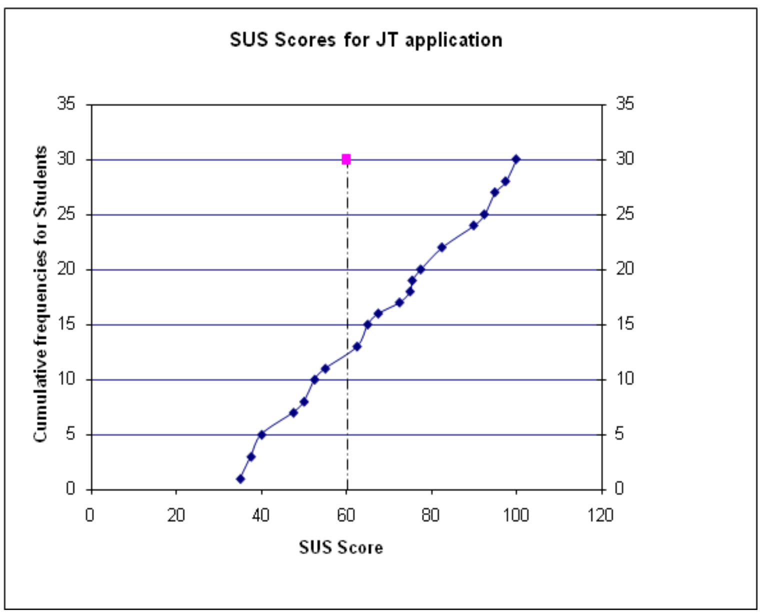


Figure 3.13: System Usability Scale Score for JT Application with 30 Students

3.4.7 Limitation of JT Application and Future Work

This version of the JT covers most of the Java programming fundamentals, such as assignment statements, if-statements, while loops, for-loops, switch statements, and method calling, but it is limited from an input perspective. In the future, this application could be made more powerful by synchronising the JVM with the program executing, this would enable user input whilst the program was executing.

3.5 Experiments I and II Comparisons

In order to compare the effectiveness of the use of the JT (Java Translator) application, data was gathered from two groups, 25 students in each. Both groups were studying the same course, an Introduction to Java Programming. The students were all at the same level of learning, second year university students, at King Abdulaziz University, girls section in Saudi Arabia, Jeddah. The students were all volunteers and will have received extra course credit for participating. Both groups utilised assessments to support their learning of Java programming.

Questions (for formative and summative assessmnts) were the same for both groups and were chosen carefully to ensure the objectives of the module were covered. The students had their own user names and passwords so that they could practise the assessments in their own time, and at their own pace.

Feedback was different for each group; the first group's feedback was supported by interactive applets and textual answers, whereas the second group received feedback, as a learning tool, from the JT application. For the JT application, the feedback to the students was sample code that directly related to wrong answers made in the assessment.

To ensure that both groups were relatively equal in terms of their learning background, the answers were scored before any feedback was given. At the end of the semester data was collected for questions that related to the use of the JT application and then compared to those supported by the applets and textual answers. In both experiments, Question Mark Perception software was used as the delivery mechanism and for recording student answers in the database.

3.5.1 Comparison of Results

The purpose of the experiment was to identify whether the JT application had an effect on the student's learning performance. As expected, prior to any feedback being given, the initial answers for both groups were similar, indicating that both groups were almost equivalent in their pre-knowledge. The average, standard deviation and variance for both groups are illustrated in Table 3.11. Group A was exposed to feedback supported by

Table 3.11: Pre-Test Evaluation for Both Groups

Group A	**Group B**
Average = 11.48	Average = 9.48
SD = 2.69	SD = 2.95
VAR = 7.26	VAR = 8.68

applets and textual information, whereas, group B was involved in using the JT application feedback as a learning tool.

3.5.2 Hypothesis

H0: A = B (Students performance in both groups remains the same).

H1: B $>$ A (Students who used the JT application in group B performed better than group A).

A two sample t-test was applied to the final grades, post-test. It was assumed that the variance would be different in both groups. Table 3.12 presents the final grade results, for both groups, using a t-test with unequal variance.

Table 3.12: t-Test: Two-Sample Assuming Unequal Variances

	Group A	**Group B**
Mean	12.08	13.72
Variance	6.83	2.88
Observations	25	25
Hypothesised Mean Difference	0	
df	41	
t Stat	-2.632409061	
$P(T \leq t) one - tail$	0.01	
$tCritical one - tail$	1.68	
$P(T \leq t) two - tail$	0.01	
$tCritical two - tail$	2.10	

In the post-test result, group B was found to have a higher average total score, in the final test (mean $= 13.72$, SD $= 1.7$), than group A (mean = 12.08, SD = 2.6). The t-test showed a significant difference in learning performance for group B with $p = 0.01$ which is less than 0.05. Since $p < 0.05$ the null hypothesis is rejected.

3.5.3 Data Analysis

Two different experiments were conducted on participants from the same academic course, but from different semesters. The first experiment had five different assessments containing questions which were supported by applets. These were embedded within the assessments and were accessed only when a question relating to programming concepts was answered incorrectly.

The second experiment was carried out on the second group. They also took five assessments and were tested on the same questions (**Appendix A**), but they were supported by the Java Translator application. The students were asked when they used the JT application and to rate how helpful they had found the application based on a five point scale. The following table shows the percentage of questions answered correctly and the ratings from the students. From Table 3.13, the question difficulty can be concluded based on the percentage of correct answers at each question level, furthermore, the ratings in terms of how useful the students found the JT application are also apparent.

Table 3.13: Test Score Result

Question No.	Correct Answers Percentage	5	4	3	2	1	N/A
(1)	98%	29/63	19/63	4/63	5/63	2/63	4/63
(2)	62%	15/63	21/63	9/63	7/63	5/63	6/63
(3)	49%	25/64	18/64	8/64	4/64	4/64	5/64
(4)	90%	30/64	13/64	9/64	5/64	2/64	5/64
(5)	92%	15/59	22/59	5/59	5/59	5/59	4/59
(6)	90%	17/59	16/59	10/59	9/59	2/59	5/59
(7)	81%	39/88	23/88	10/88	4/88	2/88	10/88
(8)	36%	20/93	35/93	8/93	13/93	5/93	12/93
(9)	62%	39/87	21/87	7/87	6/87	3/87	11/87
(10)	79%	30/84	25/84	11/84	6/84	1/84	11/84
(11)	74%	26/82	18/82	13/82	6/82	7/82	12/82
(12)	86%	37/86	20/86	8/86	8/86	2/86	11/86

3.6 Experiment III: Assessment in Studying Law

In order to support students with different discipline and investigate how computer assisted assessment and feedback can be used to support learning with completely different branch of knowledge, a third experiment will be conducted to support students who are studying a Computer Law course.

3.6.1 Assessment for Non-Scientific Subjects

A Computer Law course covers Copyright Law, Trademark Law, Patent Law, etc. The students need an assessment to help them apply the Computer Law to real problems; they will be given problems and will be asked to select which rules apply to it. The following sections will analyse the experiment and provide further suggestions for improving learning.

3.6.2 Assessment Analysis

The assessment was optional for students; out of 17 students, only nine participated. To simulate distance learning, the students were given the assessment to be completed at their own pace and in their own time.

The students were given 16 questions, and students were asked to rate the difficulty of each question on a 5point a Likert scale: five represented strongly helpful and one indicated strongly unhelpful. The students were asked to practise and to take into account the feedback provided for any questions answered incorrectly. Table 3.14 shows the analysis of the data gathered from the assessments.

3.6.3 Results Analysis

Each question was designed to target specific areas of law application knowledge. The results indicated that some of the questions had higher rates of correctness (11 of 13) meaning that they were generally easier for the students to answer, other questions had mid ranges in terms of correctness (5/10), and some had very lower rates of correctness (2/11). This type of evaluation has two purposes: firstly, they help the instructor to determine the level of difficulty of a question to the student; and secondly, they show the instructor which computer law problems can be targeted.

Table 3.14: Law Assessment Results

Question No.	Times Answered Correctly	Rank 1 Strongly Unhelpful	Rank 2 Unhelpful	Rank 3 Neither/Nor	Rank 4 Helpful	Rank 5 Strongly Helpful
1	6/13	1	0	3	7	2
2	10/13	1	1	1	7	3
3	6/13	1	0	3	8	0
4	4/12	1	0	3	7	1
5	11/13	1	2	2	7	1
6	9/12	1	2	2	7	0
7	5/7	1	3	2	5	0
8	2/11	1	0	2	6	2
9	5/10	1	3	1	6	0
10	8/11	1	0	3	7	0
11	7/12	1	1	4	6	0
12	8/11	1	2	1	7	0
13	4/11	1	0	4	6	0
14	7/11	1	2	0	6	2
15	6/11	1	1	1	7	1
16	2/11	1	2	1	6	1

Another issue causing concern for the instructor could be that some of the questions were not answered. For example, question seven was answered correctly five times, incorrectly twice and not answered six times. This could indicate that the question was too difficult and the students had to guess the answer or leave it blank.

With reference to the rating side for the questions, feedback rated as unhelpful correlated with fewer correct answers. Instructors could use both these results to evaluate where the difficulties occurred; the low percentage of correctness combined with higher ratings of unhelpfulness could indicate that the instructor needs to focus more effort on the question objective and the clarity of the question.

3.6.4 Student's Behaviour

Students act differently depending upon their needs. The following tables represent the behaviour of two students who practised the assessment on three separate occasions. Student one, in Table 3.15, started to do the assessments and improved on the second trial, while taking less time. However, in the third trial, the student criticised the helpfulness of the questions and gave less credit to some of the questions previously rated as having higher usefulness. For this student, eight questions (5, 6, 9, 10, 11, 12, 14 and 15) were answered correctly in all three trials, therefore the goals for these questions were adequately met. In Table 3.16 student two acted differently; the feedback was used more as the scores increased. Student two spent less time on assessment two, this could have influenced the lower grade that was achieved, when compared to the first one. The third trial was much better; the learner took more time to understand the question and used the feedback to answer the questions properly.

Table 3.15: Student 1 Behaviour

Student 1	First Trial 00:14:17		Second Trial 00:9:41		Third Trial 00:25:24	
Question No.	Answer	Degree of Help	Answer	Degree of Help	Answer	Degree of Help
Q1	Incorrect	4	Correct	4	Incorrect	4
Q2	Correct	4	Correct	4	Incorrect	4
Q3	Incorrect	4	Correct	4	Correct	4
Q4	Correct	4	Correct	4	Incorrect	4
Q5	Correct	4	Correct	4	Correct	4
Q6	Correct	4	Correct	4	Correct	4
Q7	Incorrect	4	Correct	4	Incorrect	3
Q8	Incorrect	4	Correct	4	Incorrect	4
Q9	Correct	4	Correct	4	Correct	4
Q10	Correct	4	Correct	4	Correct	3
Q11	Correct	4	Correct	4	Correct	4
Q12	Correct	4	Correct	4	Correct	3
Q13	Correct	4	Incorrect	4	Incorrect	4
Q14	Correct	4	Correct	4	Correct	4
Q15	Correct	4	Correct	4	Correct	4
Q16	Incorrect	4	Incorrect	4	Incorrect	4
Total	11/16		14/16		9/16	

Many e-learning courses need to be supported to improve student learning. Computer Law is one of the courses that requires direct assessment, to test and help students to improve their knowledge; they have a set of rules that must be applied for certain conditions. In real life situations, students need to be able to analyse and apply these rules rather than simply

Table 3.16: Student 2 Behaviour

Student 2	First Trial 00: 08:25		Second Trial 00:05:13		Third Trial 01:01:38	
Question no.	Answer	Degree of help	Answer	Degree of help	Answer	Degree of help
Q1	Correct	4	Correct	4	Correct	4
Q2	Incorrect	4	Correct	4	Correct	4
Q3	Incorrect	4	Incorrect	4	Incorrect	4
Q4	Incorrect	4	Correct	4	Incorrect	4
Q5	Incorrect	4	Correct	4	Correct	4
Q6	Incorrect	4	Correct	4	Correct	4
Q7	Correct	4	Incorrect	4	Correct	4
Q8	Incorrect	4	Incorrect	4	Correct	4
Q9	Correct	4	Incorrect	4	Correct	4
Q10	Correct	4	Incorrect	4	Correct	4
Q11	Incorrect	4	Incorrect	4	Correct	4
Q12	Correct	4	Correct	4	Incorrect	4
Q13	Correct	4	Incorrect	4	Incorrect	4
Q14	Correct	4	Incorrect	4	Incorrect	4
Q15	Incorrect	4	Incorrect	4	Correct	4
Q16	Incorrect	5	Incorrect	4	Incorrect	4
Total	7/16		6/16		10/16	

memorising them. Students therefore benefit from the assessments because it allows them to apply their knowledge and gain feedback, guiding them on the correct way to apply a rule for an otherwise incorrect answer.

According to the time spent on assessments, students might get bored from reading large amounts of text on certain problems. Students could be focused and motivated to use the assessments if the problems were presented using multimedia formats, such as short movies, the problems could then be understood quicker and appropriate solutions to the problems could be suggested. Instructors could benefit from ready formatted reports used to monitor the assessment progress and determine the difficulties that might be facing the learners. Furthermore, individual student progress could be monitored based on the answers and difficulties identified by the individuals learning on their own in isolated environments.

3.7 Conclusion

There are no doubts that assessments are effective tools that can play a major role in student learning. It is important to find general assessment tools that could be used for the majority of distance learning classes and designed to help students' understanding and to provide immediate feedback on the quality of their work. Furthermore, it could regularly inform the instructor regarding the students' progress. This type of self assessment environment permits the student to have intensive engagement with the subject matter; it provides a student-friendly tool that supports the student as they progress through the course.

Using prompt and relevant feedback is very important in student learning. Using Applets as a feedback to students is useful, but limited to its restriction for given codes. In contrast, Java Translator has proved to be a valuable Java programming learning tool, it is robust and reliable; its strengths come from it being easy to use and it providing relevant and instant feedback. Students who are learning Java can use the JT application at any time to translate any Java program with an easy and understandable description of the specified code. The Java Translator helps learners to understand the code, line-by-line, and follow the program sequences as they are executed. The JT application is not limited solely for feedback usage; it could also be used as a standalone application for the purpose of learning.

All three experiments identified useful results that could be incorporated to support future learners. If the student is a distance learner, or the tutor is depending on e-learning environment as a communication channel to learners, it is essential to use assessments to engage students in learning and to link the assessments to offer prompt and relevant feedback. It is also important to monitor student confidence, to ensure that individuals are supported and weaknesses are targeting and resolved to ensure that learning is not inhibited.

Chapter 4

A Proposed Educational E-Learning Model

4.1 Introduction

The modern and typical virtual learning environments (VLEs) provide repositories of ‘resources’ for students; these resources can be accessed as and when the student wishes. The VLE does not facilitate a strategy for learning or teaching, nor does it guide students through the resources, and it does not aid tutors in constructing their courses.

Although there have been many new innovations in educational technology, they have failed to alter the main features of education (**??**). The use of technology does not change face-to-face learning unless it is integrated within the curriculum **?** and then provides an active learning environment (KeenGwe et al., 2009). Researchers and educators are working towards defining appropriate models where technology and learning are integrated **??**.

In contrast to the typical VLE, the proposed model of e-learning provides a framework with tools to support both students in learning and tutors in teaching. It aims to facilitate and encourage tutors to apply the established and reliable theories of learning (examined in Chapter 2) informed by the results of the experiments described in this thesis (see

Chapter 3).

The key point is that the focus here should not be on the technology itself, instead it should be on what is done with the technology (**?**). Therefore, the pedagogical concepts used to develop e-learning environments, which utilise strategies for supporting learners to practise and use educational technologies, should be considered (KeenGwe et al., 2009). Furthermore, instructors need to go beyond any technical hurdles to provide the best of their teaching to the learners; teachers could encourage students to learn if they could ‘draw on their own experiences in learning with technology’ (**?**, p. 4).

Initially, any obstacles that face the teachers to use the technology need to be considered and eliminated. In order to develop successful learning, teachers need e-learning environments that help and guide them to integrate their technological skills into their subject areas; in the long term this will help to activate learning.

To explain the model, the duality of student and tutor views, as presented by Biggs (2003, p. 141) and his stages of learning, should be explained for both a complete course and for an individual lesson composed within a well constructed course.

The following section presents the features and processes by which the tutor constructs a course and each component lesson within it; this process encourages ‘constructive alignment’ (Biggs, 1996). The section 4.2 concludes with two practical examples detailing the application of the process for two very different subjects.

4.2 A Model for E-Learning Environment

The proposed model, presented in Figure 4.1, shows both views, one from the students' perspective and one from the tutors' perspective, as well as the general concept and flow of the learning system design. As Biggs describes, for teachers the assessment comes at the end of the teaching and learning sequence, but for students it comes at the beginning.

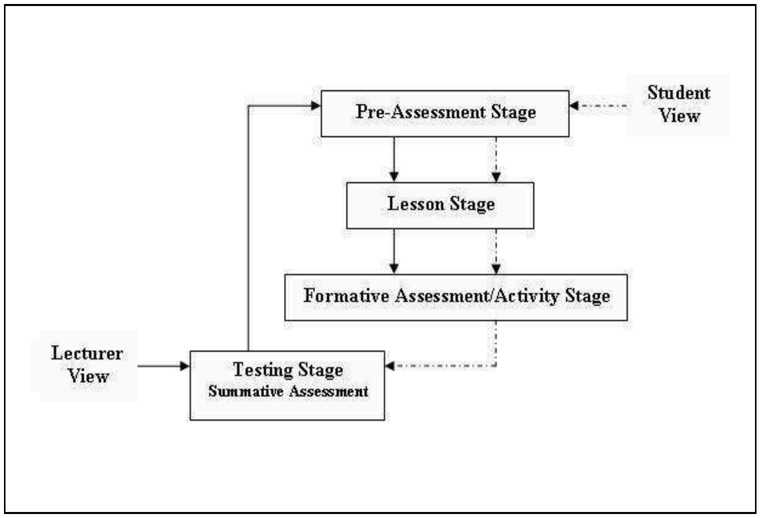


Figure 4.1: Conceptual Model Design

Therefore, if the objectives are reflected on the assessment, as shown in Figure 4.2, the teaching and learning activities are all directed towards the goals (Biggs, 2003, p. 141). To simulate learning, tools should be provided for the instructor to best present the learn-

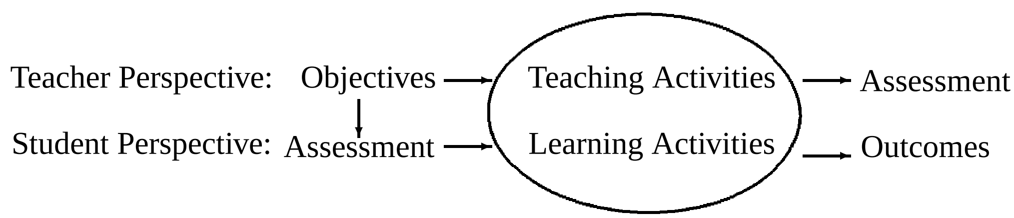


Figure 4.2: Teacher's and Student's Perspectives on Assessments *(John Biggs)* (Biggs, 2003, p. 141)

ing activities and class notes; this could include text information, multimedia, simulation games, applets or interactive questions. Once the teacher has specified the course objectives, the proposed system should enable them to select the required components as tools. The main aim of the system is to utilise the four stages to deliver the course objectives and deliver deep approaches to learning. Figure 4.3 represents the structured design of the proposed model using the Jackson Structured Programming (JSP) design. The following two sections describe the flow and stages of both teaching and learning from the respective perspectives.

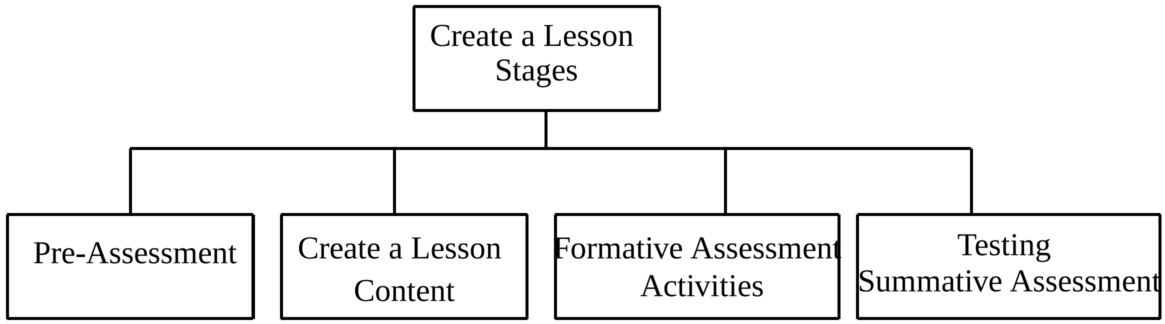


Figure 4.3: Structured Design for the Proposed Model

4.2.1 Application Design from the Lecturer's Perspective

There is a growing demand for lecturers to design effective holistic learning for their learners. The lecturers require frameworks to support them to develop smooth and supportive learning environments that can be integrated within the course content; consequently, these frameworks need to help the lecturer throughout the process. An environment of dialog and interactive tools will help the lecturer to smoothly develop their course content in a focused way to support the learning process. The design will support the learning environment by providing readily available templates for educators to complete or the lecturer will be guided to develop their own. Before describing the system design in detail, the general flow of learning a subject will be considered, from the lecturer's perspective, see Figure 4.4.

Testing Stage (Summative Assessment):

This is the final stage for the learner, however it is the first stage for the lecturer and it is critical for ensuring that the target goals are addressed and met. The lecturer needs to clearly identify what the learner should achieve by the end of the specified

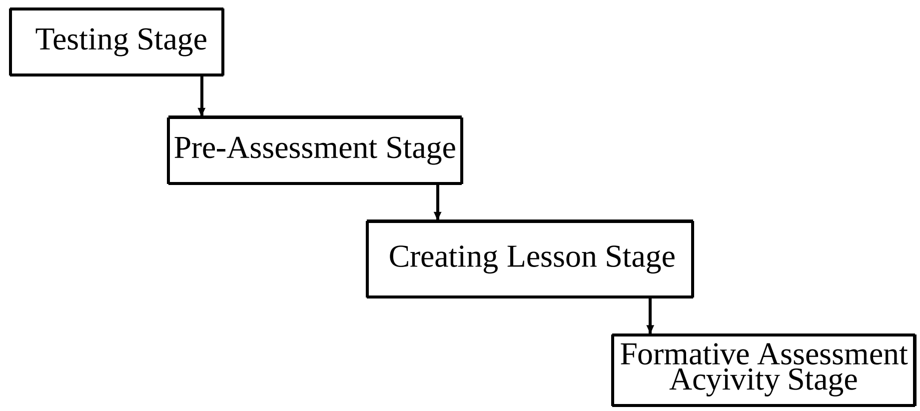


Figure 4.4: Stages Flow from Lecturers' Perspective

lessons and then build the rest of the stages around these goals. Furthermore, at the end of this stage, the lecturer will know whether the learner has met the specified goals for that lesson. This is a critical stage for the lecturer because it determines the next stage for the learner and will therefore direct the learning process to either progress to the next advanced level or to a simpler one. Figure 4.5 presents the structure of the testing stage.

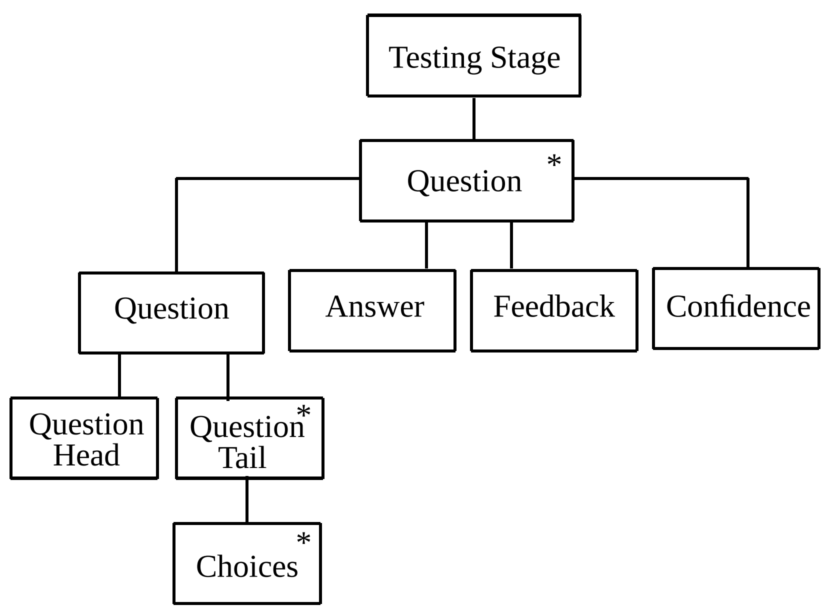


Figure 4.5: Structured Design of the Testing Stage

Pre-Assessment Stage:

This stage ensures that previous knowledge is checked and specific objectives of a particular lesson are targeted. Pre-instructional activities are rare in e-learning environments, but they play a significant role in the success of learning interventions (**?**). To guarantee the success of a learning environment, lecturers must consider the preparation needed prior to the actual lesson taking place; different objectives require different preparations. At this stage, lecturers could utilise introductions to ensure previous knowledge or they could use quizzes to highlight what should be known by the end of the lesson. Figure 4.6 identifies the structure of this stage.

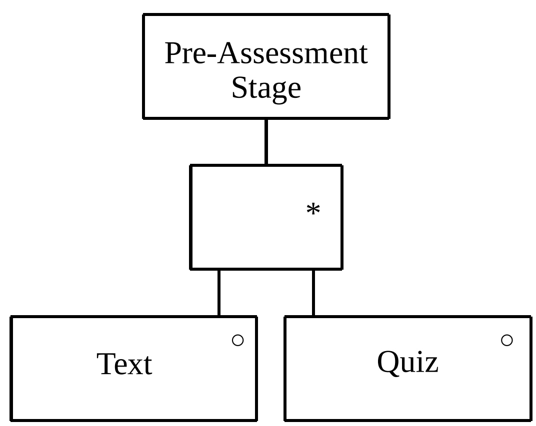


Figure 4.6: Structured Design for the PreAssessment Stage

Creating Lesson stage:

At this stage, the actual content of the lecture is developed. The lecturer develops the course content with supportive interactive multimedia tools such as images, videos, etc. The actual lesson contains a title, body and summary; the body refers to the lesson content itself and could simply be a collection of paragraphs. Figure 4.7 shows the structure of the content of an individual lesson.

Formative Assessment / Activity Stage:

This stage incorporates the assessment of understanding; the lecturer needs to be

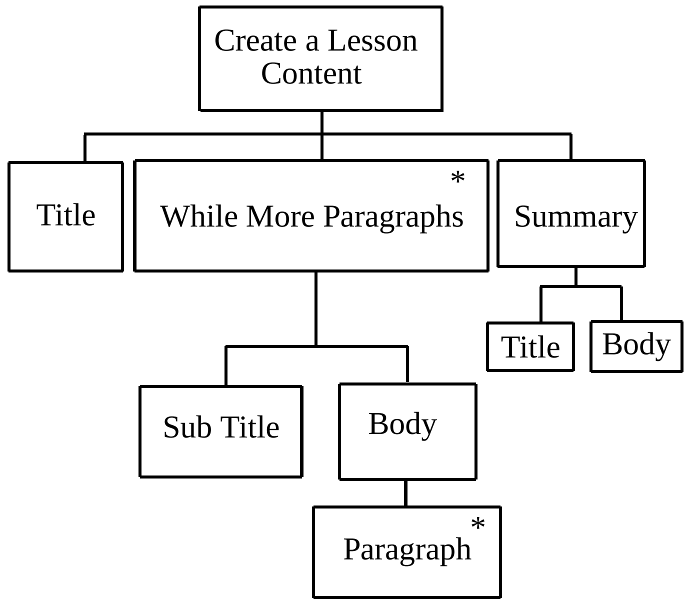


Figure 4.7: Structured Design for Creating a Lesson Content Stage

able to assess learners to determine their understanding, different activities should be provided which are adaptable to the different student needs. The lecturer needs to ensure that the students are engaged in the learning process; different lessons and different types of assessments are therefore required. Activities could include a number of exercises that ask the student to evaluate, practise, or criticise, or it could be in the form of an interactive self assessment providing relevant and prompt feedback. Figure 4.8 identifies the structure of the formative assessment/activity stage.

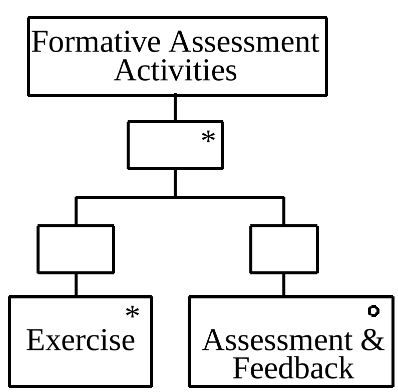


Figure 4.8: Structured Design for Formative Assessment / Activity Stage

The design considers a list of objects that are required to be integrated, to form the different individual lessons. The design focuses the tutor to target the goal by creating the flow of learning. The lecturer's design starts at the testing stage; therefore, if the lecturer is supported to focus on what is needed for a particular lesson, the instructor is likely to remain focused as they develop their course content. If the lecturer needs to test a specific goal, then they need to ensure that the entire learning environment focuses on this particular goal. This environment will therefore ensure that the learner focuses on what the lecturer has identified as the goals for the lesson and course.

4.2.2 Application Design from the Learner's Perspective

Isolated learners need to be guided in their self paced learning; they need to be provided with the tools to engage them to learn on their own. The following stages help learners to be prepared, motivated, involved and experienced in learning. Figure 4.9 represents the flow of the learning environment from the learner's perspective using the Jackson Structured Programming (JSP) design.

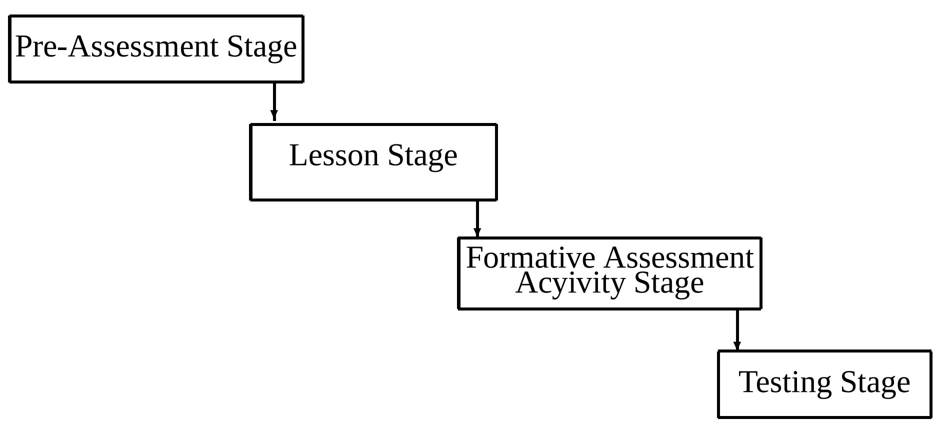


Figure 4.9: Stages Flow from Learner' Perspective

Stage 1-Pre-Assessment

A simple introductory discussion should focus on the main subject in context, thus identifying the overall goal for the lecture. This will relate what they need to learn with the subject matter as a whole. This could be done by introducing a real life situation and then asking the student what they would do about it; thus encouraging them to think, and help them to build new knowledge based on connections with what they already know.

Stage 2-Lesson

The lesson should be supported by interactive tools which provide the student with practical interactive experiments to simulate what they are going to learn.

Stage 3-Formative Assessment/Activities

An interactive formative self-assessment should be provided to test the student's understanding, interactive feedback should then be provided to encourage understanding. The environment needs to monitor student activity, such as performance; it is also important to know how the student feels about experiencing the assessment. The use of assessments could also identify any misconceptions discovered. Students also can be given an interactive application as an activity to work with that help them to practise and develop skills on their learning.

Stage 4-Testing (Summative Assessment)

This stage monitors understanding; the student's performance should be evaluated against the learning outcomes, to determine whether they were achieved.

4.3 Proposed Model Applied Examples

The following two sections provide applied examples of how the proposed model could be used. Two examples will be looked at to identify how the model could be applied in different subjects.

4.3.1 Case Study I: Chemistry Lesson - Balancing Chemical Equations

The first case study involves testing the proposed design on the development of lecture material for the scientific discipline of chemistry. Two questions need to be focused on: how can the proposed learning system help the tutor to develop their tutorial, to gain the optimal objectives from their lesson; and how can the system help the learner to be engaged in the learning process? The proposed design follows Biggs' theory of constructive alignment to align the teaching/learning activities, assessment tasks and curriculum objectives, with each other. The proposed system involves four stages: pre-assessment; lesson; activity; and test stage; the stages are aligned to target the objectives of the lesson. The instructor's view for the system is: testing, pre-assessment; lesson content; and activity, respectively. In practice, the student will follow the normal flow of learning: pre-assessment; lesson content; activity and testing, respectively. The following simulates the flow from the instructor's view of designing and implementing the student's flow of learning.

Testing Stage (Summative Assessment):

Firstly, the tutor focuses on the testing stage, this enables the tutor to focus on what the objectives are that need to be targeted in this particular lesson. This is the final stage of supporting student learning, a lot of decisions can be made on this level. The test stage tests whether the learners have learned what they needed to and therefore determines whether the learner can move onto the next learning level or whether they need extra support to enable understanding. The test, at this level, is focused on testing the goal objective: in this case balancing chemical equations. Appendix C-page 155 contains an example of the chemistry test.

Pre-Assessment Stage:

Preparation is required to ensure that the learner develops enough knowledge from engaging in the learning process. This stage could involve: a simple lesson to introduce the learner to the new knowledge by linking it to their existing knowledge; or it could be a small quiz to test their knowledge prior to the new lesson, to help them prepare for what they are going to learn. Tutors could use both types of pre-assessment.

In the following chemistry lesson, the tutor could use an introduction for the basic requirements for developing chemical equations, as well as a small quiz to test their knowledge. At this stage, the tutor does not expect the learner to answer correctly; however this method is advantageous because it prepares the learner and provides expectations regarding the lesson content and what they are likely to learn. For this particular lesson, which focuses on teaching the student how to balance chemical equations, the pre-assessment stage assumes that students have an awareness of chemical reactions and chemical equations. The tutor could design a brief description of what chemical reactions are (Appendix C-page 156) and a quiz (Appendix C-page 158) to determine whether students know how chemical reactions work; these would focus on the elements and components, and the need for balancing them. Figure 4.10 shows what is needed for this particular lesson.

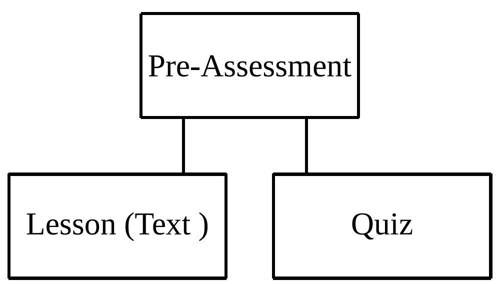


Figure 4.10: Pre-Assessment Stage

1. **Lesson (Text)**: This is the first part of the pre-assessment stage; it introduces some of the main concepts about chemical reactions and element definitions. The format of the lesson could simply be in text format and could include a general title, a collection of paragraphs that describe the lesson content, and

a summary. The focus is on the learning content rather than on the platform itself. Creating a lesson should help tutors to focus on what needs to be learned rather than what platform should be used, what should be done next, or how the lesson content should be organised. The diagram (4.7) describes the flow of an organised lesson and should help the tutor to develop the lesson content. At this particular stage, it should represent the pre-assessment stage lesson; Figure 4.7 represents the structure of this lesson content.

2. **Quiz**: : This is the second part of the pre-assessment stage; it helps the learner to develop knowledge about the learning objective of the particular lesson and ensures that any prior knowledge needed by the learner is known by them. This engagement process is important because it helps the learner to develop new knowledge based on existing knowledge. From the design aspect, the quiz is a form of test and the structure, represented in Figure 4.5, is needed and used for the pre-assessment lesson content. This should be in the form of an interactive quiz which provides students with their score and simple feedback. Students should not worry about their scores at this stage, because the quiz is used to focus the student to identify what they need to know before they have the lesson.

Create Lesson Content Stage:

Tutors usually require some form of template to add their lesson content to; sometimes they can find it difficult to develop lecture materials that are easy to follow for the learner. The proposed design can help tutors in many ways to: present, organise, and summarise their lesson content. The proposed design eliminates the need for support from trainers or from any other software to develop and design the lesson content. The structure, presented in the diagram 4.7, helps tutors to add the course content, including: the initial lesson title, the individual content, and the lesson summary, (Appendix C-page 160 details the actual lesson content).

Formative Assessment/Activities Stage:

Usually learners need some assurance that they have understood and met the required lesson objective. For this lesson, students need to practise and gain immediate feedback for their answers. Different subjects and different lessons require different activities; for this particular lesson, the tutor needs to ensure that the students are able to balance a chemical equation to illustrate that they can apply what they have learned. There are a number of online interactive applications which can help students to practise balancing chemical equations. For example, the link [1] below utilises flash software that presents an unbalanced chemical equation, the learner is required to balance the equation. The software supports the learner by identifying hints and it allows them to check for the correct answer, see Figure 4.11.

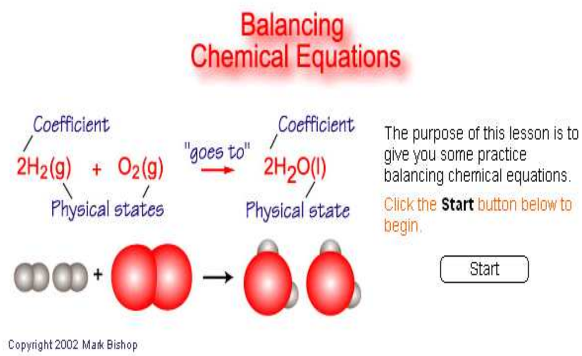


Figure 4.11: A Snap Shot for the Interactive Flash Activity

4.3.2 Case Study II: History Lesson - Hijra of Prophet Mohammad

The second case study involves testing the proposed design on developing lecture materials for the art and humanities discipline of history. Two questions will be focused on: how can the proposed learning system help the tutor to develop their tutorial so as to gain the optimal objectives of the history lesson; and how can the lesson help the learner to be more engaged in the learning process?

[1] http://www.mpcfaculty.net/mark_bishop/balancing_equations_tutorial.htm

History lessons are often difficult to teach because they usually contain large amounts of detailed information that are easily forgotten. The tutor needs to align the teaching/learning activities, the assessments tasks and the curriculum objectives with each other to target the objectives; this should help the learner to follow the lesson and remember the events if they can be made to feel closer to the course content. The learners need to think about why and when the events occurred to help them remember them.

The proposed system involves four stages: pre-assessment, lesson, activity, and test stages; the stages are aligned to target the objectives of the lesson. Firstly, the tutor will focus on the testing stage which will help them develop the learning environment. Next, the pre-assessment stage will prepare the learner to develop knowledge and engage them with the learning process. A simple lesson could be utilised to introduce the learner to the new knowledge, by attempting to link this to the learner's existing knowledge. Alternatively, a quiz could introduce some of the new concepts from the lesson to prepare the learner for the new learning. Tutors could use both types of pre-assessment.

In the following history lesson, the tutor could use an introductory list of events that occurred during the Hijra of Prophet Mohammad. This list of events could help target the main objectives of the lesson, focusing on the great event in Islam. In addition, a quiz could be used to test their existing knowledge about Hijra, this could encourage the learner to target the event once the lesson begins. At this stage, the tutor does not expect the learner to answer correctly; however, this could still be advantageous because learners might identify expectations, in terms of what they should learn from the lesson content.

Testing Stage:

This stage supports student learning, numerous decisions can be made from this level. The stage needs to test whether the learner thoroughly understood the story of Hijra and whether they have learned the events. The test, at this level, is focused on testing the goal objective which is the Prophet Mohammad's Hijra, the **causes**, the **flight** and the **effects** (refer to Appendix C - page 166).

Pre-Assessment Stage

History lessons usually contain events, to ensure the learner learns the most about these events the pre-assessment could be used as the preparation stage, exposing the learner to the actual lesson. Firstly, the preparation lesson could contain a list of events that Prophet Mohammad went through during his Hijra to Medina. Once these events are introduced, the student could be quizzed to highlight the role of some of the events and people that were significant during the journey. This particular pre-assessment uses an introductory lesson (Appendix C - page 169) and a small quiz (Appendix C - page 170) prior to the actual lesson on Hijra.

Create Lesson Content Stage

Tutors usually require some form of template to add their lesson content to; sometimes they can find it difficult to develop lecture materials that are easy to follow. The proposed design can help the tutor in many ways to: present, organise, and summarise their lesson content. This design eliminates the need for support from trainers or from any other software to help in developing or designing the lesson content. The structure, presented in the diagram 4.7 helps the tutor to add their course content, including the lesson title, the individual content, and the lesson summary (please see Appendix C -page 171).

Formative Assessment/Activities Stage

Learners usually need some assurance that they have understood and met the required lesson objective. For this particular lesson, learners need to understand the story of Prophet Mohammad's Hijra. As an activity, learners could watch the movie 'Alresalah' and then describe the story of Al-Hijra based on the events in the movie. They could be asked to identify the causes of Al-Hijra, the colonies involved in Al-Hijra and what happened as an effect of Al-Hijra.

4.4 Conclusion

A proposed educational model for e-learning has been discussed throughout this chapter; the proposed model facilitates a strategy for learning and teaching. The model could help teachers to manage and construct their course content and guide their students through learning. It comes with four stages: the pre-assessment stage, creating lesson content stage, the formative assessment/activity stage and the testing (summative assessment) stage . All four stages are viewed differently by the teacher and the learner. The proposed model guides the teacher to use resources to develop lecture materials in a strategic way and according to theories of learning. To guide the tutor to focus on learning outcomes, the design will start at the testing stage which is the final stage for learners then the rest of the learning stages. This will help the tutor to focus on what should be made to guide learners and engage them in learning form the pre-assessment stage until the testing stage. The design gives the flexibility for the tutor to construct a learning environment on the needs. On the other hand, the design will automatically creates the learner view constructed by the tutor in normal flow for learning.

The two examples prove that the proposed system could work within different settings; it worked well in a history lesson and in a chemistry lesson. The proposed model therefore provides a practical learning environment that could be applied to different subjects.

Chapter 5

Proposed Educational Model Structure and Design

5.1 Introduction

This chapter will introduce the ways in which the proposed design will be physically implemented; furthermore the proposed model will be reviewed to determine how it will work as a software design. Two main issues affecting the implementation will be considered, focusing on the structure and the components of the model. The tutor is in control of the structure and is responsible for: considering how to control the learning process; and at the lower level, identifying the components needed to make it happen. Technically, the model proposed is based on two main components: the **lesson**, and the **assessment**. In the following sections, the model components, and how they can be used, will be described.

5.2 A Lesson

A lesson is generally an organised collection of presentations and activities, the learner progresses through these to accomplish the sub-goals of a course (Horton, 2000). Each lesson should have its own necessary components, including: defining leaning outcomes, lesson content, assessments with feedback, and tests. For the model provided in this thesis, the lesson is a sequence of four stages, these are presented in Figure 5.1.

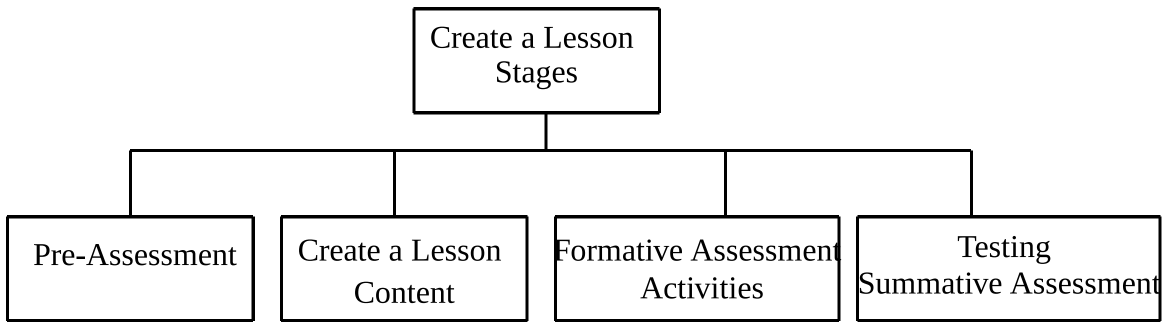


Figure 5.1: Structured Design for the Proposed Model

5.2.1 Forming Lesson Structure and Control

The learning outcomes are defined first, the tutor can then start to construct the lesson; starting with the testing stage and then proceeding from the pre-assessment stage to, and through, the remaining stages. The tutor views the lesson on a computer screen where there is a list of four components. For a new lesson, the tutor is guided firstly to record the intended learning outcomes, secondly to construct the test(summative assessment) and then the pre-assessment before proceeding to lesson content and objectives. For each component, the tutor can choose from a list of further components, e.g. at the pre-assessment stage there are two components which can be chosen: lesson(text) and quiz. One or both can be added to the lesson. The tutor therefore has the ability to control the content and can then prepare for what is needed at this stage by the students.

At the formative assessment/activities stage the tutor can construct one or more activities; they can select from: an exercise or an assessment with formative feedback, or they can choose both. At the testing(summative assessment) stage, which is the final stage for the learners, the tutor can design a test to evaluate the targeted learning goals.

In terms of control, the tutor is able to control the sequence of learning and how this appears to the learner. The structure could be: a fixed sequence that the student needs to follow; a conditional sequence that is aligned with student progress; or free browsing of the whole course content. For example, the tutor can control the students' access to the test, until a certain date or until an assessment is completed. They can also control access and viewing of the whole course content.

The instructor should be able to develop an interface to report any weaknesses in the course content and follow up on each student's individual achievements, based on the student's formative assessments. This could be a very useful tool for the instructor to determine and resolve any weaknesses observed in terms of the student or the course content.

From students' perspective questions should be able to be selected and practised within the assessments. If the students decide to practise on any given self assessment, they should have the option to choose to answer only certain questions that they aim to practice on. It also should be possible to search the assessment bank for certain questions (search tool) and be able to use the search results to create an assessment of their choice.

As a result of practicing the assessment, the student should be provided with the ability to view a table that contains their assessment results, thus identifying their strengths and weaknesses and able to view their progress during the course. As each question answered has a specific level of understanding according to the five levels in SOLO taxonomy, learners can be provided by their progress against meeting learning outcomes.

5.2.2 A Lesson Content

The lesson content is structured and utilises components to present the content. To develop the lesson content, different stages of the design are progressed through. The developer identifies the lesson structure as a title, a collection of paragraphs and a summary; Figure 5.2 represents this structure. Within software design, the paragraph component can include necessary objects to form the lesson; these could include: text, applets (as used in the experiments within this thesis), videos and other media required to develop the lesson effectively and efficiently. The tutor can control the construction of the lesson

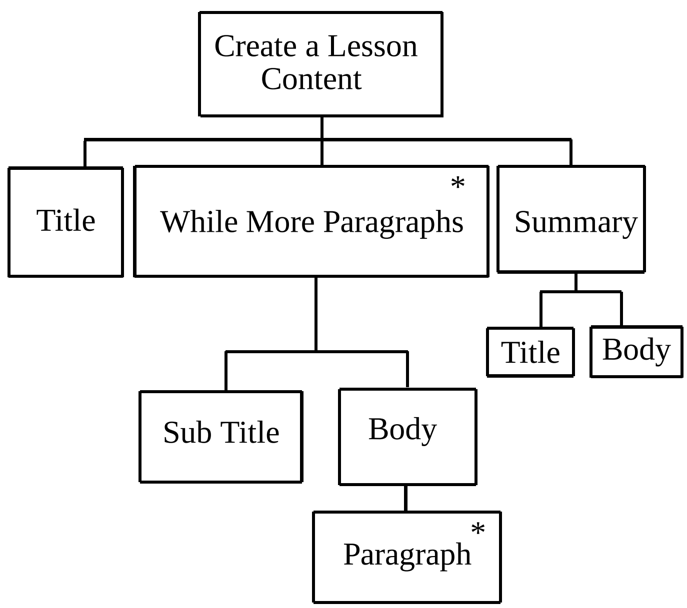


Figure 5.2: Structured Design for Creating Lesson Content

content. Figure 5.3 represents a visual presentation of some of the elements that can be controlled. At terms of software design, the lesson content is composed of a collection of objects. Lesson class and paragraph class are two main classes that are used at this point.

Lesson Class:

The lesson class is the class that represents the lesson content. The lesson is composed of a collection of titles, paragraphs, summaries, and objects. Figure 5.4 rep-

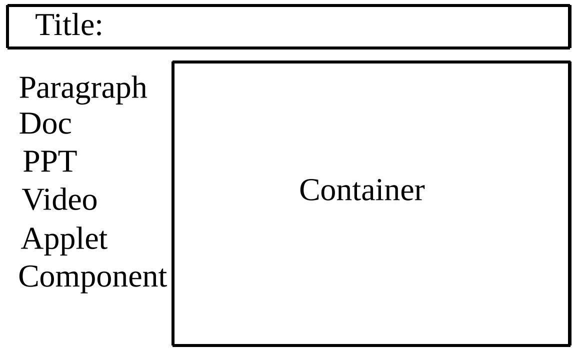


Figure 5.3: A Structure of the Screen for Developing Lesson Content

resents the lesson class CRC[1] card.

Lesson Class	
-Title -Add -Show -Edit -Insert	-Paragraph -Summary

Figure 5.4: Lesson Class CRC Card

Paragraph Class:

The paragraph class is the class that represents the paragraphs included in the content. A paragraph is composed of a collection of titles and information body, or both. It also allows to add components to the contents. Those components could be a selection from applet library, flash, video, or any component that can be used as add-ons. Figure 5.5 represents the paragraph class CRC card

[1]Class Responsibility Collaborator (CRC) can be used to simplify the model structure. A CRC Card is utilised in developing object-oriented models, by providing a simple and easy to use method for exploring the objects to be utilised in the new system . The CRC focuses on three dimensions which identify the real world object design: class name, responsibilities, and collaborators. A class represents a collection of similar objects and the responsibility represents what the class knows or does. The collaborator refers to a collaboration between this class and another class that is required to ensure that the responsibility of the

Paragraph Class	
-Title -Body -Add -Show -Edit	-Lesson -Test

Figure 5.5: Paragraph Class CRC Card

5.2.3 An Assessment

Assessments are needed in the pre-assessment, activities and testing stages, therefore as the lesson is being designed, they should be considered. Figure 5.6 describes the assessment design; a collection of questions can be implemented by the tutor to design the formative assessment or test. The aim is to provide supportive tools which enable the design and construction of informative assessments; tools like embedded applets, videos or other media as feedback, or in the questions themselves. Figure 5.7 identifies the options available in constructing assessments; the tutor could drag a question into the container and then edit it. In terms of software objects the the assessment is composed of two main classes; test class and question class:

Test Class:

The test class is the class that represents a test. It can be in any form and could include: formative assessment, summative assessment, quiz, or pre-assessment. Figure Fig 5.8 identifies the test class CRC card.

Question Class:

The question class is the class that represents the individual question in the test, assessment, or quiz. The question is composed of a collection of question body, choices or both. The question can be extended to different types of questions like

required task is met. Collaboration can be a request of information or a request to do something

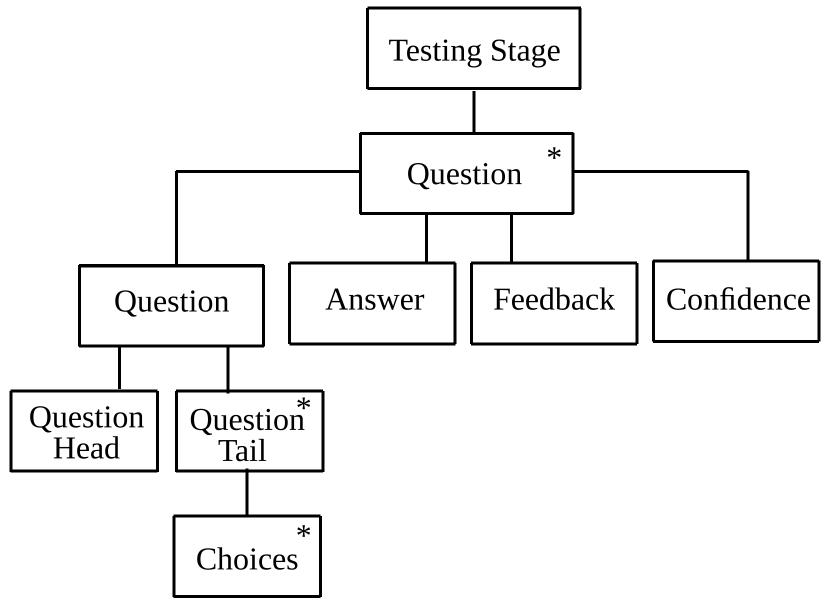


Figure 5.6: Assessment Design

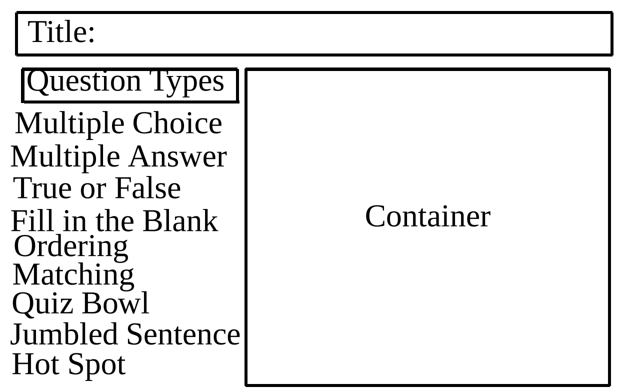


Figure 5.7: A Structure of the Screen for Developing an Assessment/Test

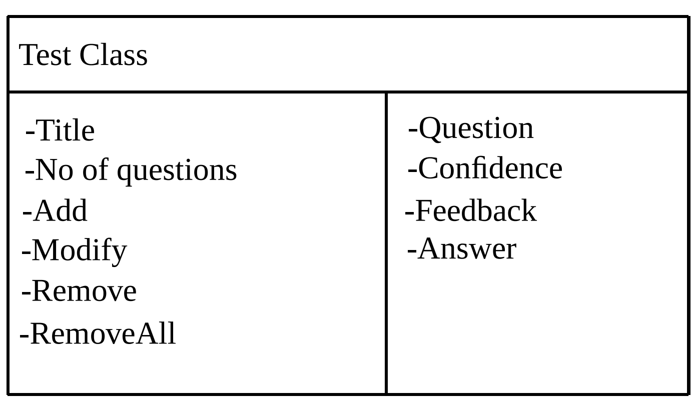


Figure 5.8: Test Class CRC Card

true/false questions; multiple choice questions, or fill in the blanks. The choices class is part of the question class and contains the answers for the respective questions. Question class provides the ability to relate the question to certain topics, so customised formative assessment could be created based on the student's needs, for example if the student wanted to study While-Loop, they should be able to choose from the questions relating to this subject.

Based on the student's learning approach (deep/surface), the student could choose their question feedback (brief, detailed or linked to the class notes). Regular and short feedback is provided, but the student could have the option to choose detailed feedback and a link to the class notes, where the correct answer could be found. This would help students to adopt deep learning approaches by linking their knowledge to the actual course content. Furthermore, the students' scores, after each assessment, are reported in terms of their degree of excellence (need to practice, good, excellent). Figure 5.9 represents the question class CRC card.

Question Class	
-ID -Body -Type -Add -Show -Edit -Answer -Feedback -Confidence -Related topics	-Choices

Figure 5.9: Question Class CRC Card

The following classes, can be inherited from the previously constructed classes above. The following are a list of the inherited classes:

- **Formative assessment/Activity Class:** An activity could be an exercise asking the student to do a certain activity (story telling) or it could be a formative assessment inherited from the test class.
- **Confidence Class:** A confidence class could be inherited from a question with choices that ask the student about their confidence or confidence level.
- **Pre-Assessment Class:** Refers to the activities required before accessing a specific lesson. This could involve some pre-reading which could be either a class text which could be inherited from a class lesson or a class quiz or from them both.
- **Summary Class:** Summary class could be inherited from the paragraph class that contains the title and body.

The relationship between the classes in the system that are responsible of helping tutors to organise and prepare their lessons, based on educational concepts. The tutor, who is also the course administrator, will be able to do the following functions:

- create pre-assessments;
- create lessons;
- create formative assessment/activities;
- create tests.

To create a lesson system, four major classes need to be composed: pre-assessment, lesson, formative assessment/activitiy, and test; all are major parts for student learning. The classes functionality can be described as follows:

- **Pre-Assessment Class:** The pre-assessment class is responsible for delivering a background for the student, prior to them actually starting the lesson. The concept behind this class is to help students to integrate and link any previous knowledge to the new one. The class also ensures that learners are aware of, and understand, the learning intentions and lesson goals. This class can be composed of any number of quizzes, texts (lessons), or both. Quiz is inherited from the class test and text is inherited from the lesson class. It is the tutor's responsibility to use what is needed at this stage.

- **Lesson Content Class:** The lesson class is responsible of the lesson content; this class consists of a collection of paragraphs ending with a summary, this is inherited from the paragraph class.

- **Formative Assessment/Activity Class:** The activity class is responsible for ensuring that engagement occurs with the learning content. The tutor needs to provide activities to ensure that students practise what they have learned. The activity class could be an exercise such as: writing an essay, or discussing a problem, this would get the learner to apply the knowledge that they have just learned. It could also be an assessment, in the form of multiple choice questions (MCQ); this would identify whether learning has occurred and whether they have understood.

- **Test(Summative Assessment) Class:** The test class is the final stage of the lesson; it is responsible for targeting and evaluating whether the student has met the learning goals or achieved the lesson outcomes. The test class contains four main classes: question, confidence, feedback and answer class. The question class contains choices that are shown to the student, from which they choose from.

5.3 Conclusion

This chapter provides a description of how the proposed design can be utilised and implemented. It gives an overview of how the system can be visualised and constructed for lower level implementation. The description of the classes that can be used to implement the design has been described. There are four major classes for constructing a lesson.

Pre-assessment class, lesson content class, formative assessment/activties class, and test class. All other described classes are used by four major classes. The system classes are expandable as it could be linked to new components that can added as a learning tool.

Chapter 6

Conclusions

The aim of this research was to investigate what was required to make a better e-learning environment. The contribution is the design of an e-learning system based on integrating technology with pedagogy. Unlike other e-learning environments, the design guides the tutor to construct lessons and help learners to use effective learning environment. The model of e-learning proposed, provides a framework with tools to support both students in learning and tutors in teaching. It facilitates and indeed encourages tutors to apply established and reliable theories of learning and the results of the experiments have been described in this thesis.

The investigation started by designing self assessment and feedback as an efficient and functional learning tool for distance learning students. This was implemented and analysed to determine how the design influenced their learning when they were isolated from their peers and the instructor. The research identified engagement as being a major concern for isolated students; the theories of learning emphasised the use of assessment to support and engage the student in learning. The conclusions of this thesis can be summrised as follows:

- A detailed study of recent learning theories (**Chapter** 2), with particular focus on the role of assessment and feedback in e-learning environments. The study incorporate the theories of learning to design practical assessment and feedback tools, to support student learning.

- Experimental work was conducted (**Chapter** 3) to develop tools to support self assessment and feedback for students. The experiment tested the use of applets to support students learning Java programming; and also focused on the development of the JT application software. Both interventions are powerful tools which can support learners.

- The theoretical and practical models for an e-learning application (**Chapters** 4 and 5) describe how to construct an effective e-learning environment that engages students and leads them to develop deep approaches to learning. The proposed model provides a practical educational guide to instructors on how to generate an effective organised and controlled learning environment that directs and focuses the learner. The greater advantage of the proposed model is its versatility as case studies showed how it can be applied to any discipline.

- The system is extensible. It is designed deliberately to facilitate addition of further software components.

6.1 Suggestions for Future Work

Further to the work reported in this thesis, several advances for further development are suggested. The proposed model should be implemented into a real application. Since the system is extensible, collaborative work needed to make this work open source and add components that may serve all types of education. For example, link the software to tools needed by Math, Geography, History, Art, etc to be used and made avaliable when needed. This means that any tutor doesn't need to look for tools when use this software application.

Another suggested future work, is continuing the development for JT application used in the experiments. The JT application should be extended to cover more advanced application usage as covering all the commands needed in Java language. Another suggestion is to make the JT application to be used as a translator for other languages as well.

References

Alansari, I., Konstadopoulou, A. and Fretwell, R. (2008*a*), Assessment and Instant Feedback in Learning Java Programming, *in* 'PGNET:The 9th Annual Postgraduate Symposium,The Convergence of Telecommunications, Networking and Broadcasting', Liverpool John Moores University.

Alansari, I., Konstadopoulou, A. and Fretwell, R. (2008*b*), Prompt and Relevant Feedback in e-Learning, *in* 'International Conference on Excellence in Education 2008: Future Minds and Creativity', Proceedings of the Annual Conference of the International Center for Innovation in Education (ICIE) held in Paris-France.

Allen, M. (2007), *Michael Allen's E-Learning Library: Designing Successful E-Learning*, Wiley.

Atkinson, R. and Shiffrin, R. (1968), "Human memory: A proposed system and its control processes", *The psychology of learning and motivation* , Vol. 2, pp. 89–195.

Augar, N., Raitman, R. and Zhou, W. (2004), Teaching and learning online with wikis, *in* 'Beyond the comfort zone: Proceedings of the 21st ASCILITE Conference', Citeseer, pp. 95–104.

Bangor, A., Kortum, P. and Miller, J. (2009), "Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale", *Journal of Usability Studies* , Vol. 4, pp. 114–123.

Bertolo, E. and Lambert, G. (2007), “Implementing CAA in chemistry: a case study”, © Loughborough University, pp. 73–84.

Biggs, J. (1987), *Student Approaches to Learning and Studying. Research Monograph.*, Australian Council for Educational Research Ltd., Radford House, Frederick St., Hawthorn 3122, Australia.

Biggs, J. (1996), “Enhancing teaching through constructive alignment”, *Higher Education* , Vol. 32, Springer, pp. 347–364.

Biggs, J. (1999), *Teaching for quality learning at university*, Open University Press Philadelphia.

Biggs, J. (2001), “The reflective institution: Assuring and enhancing the quality of teaching and learning”, *Higher Education* , Vol. 41, Springer, pp. 221–238.

Biggs, J. (2003), *Teaching for Quality Learning at University: What the Student Does (Society for Research into Higher Education)*, 2nd edn.

Biggs, J. and Collis, K. (1982), *Evaluating the quality of learning: The SOLO taxonomy (Structure of the Observed Learning Outcome)*, Academic Pr.

Blogs (n.d.), ‘Blogger’, http://www.blogger.com. Last accessed: 31 August 2009.

Broadbent, B. (2002), *ABC’s of E-Learning: Reaping the Benefits and Avoiding the Pitfalls*, John Wiley & Sons, Inc., New York, NY, USA.

Brooke, J. (1996), “SUS-A quick and dirty usability scale”, *Usability evaluation in industry* , pp. 189–194.

Brusilovsky, P. and Sosnovsky, S. (2005), Engaging students to work with selfassessment questions: A study of two approaches, *in* ‘Proceedings of the 10th annual SIGCSE conference on Innovation and technology in computer science education’, ACM New York, NY, USA, pp. 251–255.

Bryan, C. and Clegg, K. (2006), *Innovative assessment in higher education*, Routledge.

Carter, J., Ala-Mutka, K., Fuller, U., Dick, M., English, J., Fone, W. and Sheard, J. (2003), How shall we assess this?, *in* 'Annual Joint Conference Integrating Technology into Computer Science Education', Vol. 35, ACM New York, NY, USA, pp. 107–123.

Cuban, L. (2000), "Computers oversold and underutilized: Computers in the classroom", *New York: Harvard University Press* .

Dalziel, J. (2000), Integrating computer assisted assessment with textbooks and question banks: Options for enhancing learning, *in* 'Fourth International CAA Conference, June, Loughborough University, Loughborough, UK', Citeseer.

Dennis, A., Wixom, B., Tegarden, D., Ghattas, R. and McKee, S. (2005), *System Analysis and Design with UML Version 2.0*, Prentice Hall.

Dermo, J. (2007), "Benefits and obstacles: factors affecting the uptake of CAA in undergraduate courses", © Loughborough University, pp. 155–159.

Dougiamas, M. (n.d.), 'Moodle', http://moodle.org. Last accessed: 31 August 2009.

Duck (n.d.), http://bcrc.bio.umass.edu/projects. Last accessed: 31 August 2009.

Entwistle, N. (1988), "Styles of Learning and Teaching", London, David Fulton, p. 106.

Entwistle, N. (2000), Promoting deep learning through teaching and assessment: conceptual frameworks and educational contexts, *in* 'TLRP Conference, Leicester, November'.

Folkerts, A. (2008), "Blogs for Learning", *Literacies, Learning & Libraries* , p. 27.

Fone, W. (2002), Improving feedback from multiple choice test, *in* 'Proceedings of the 7th annual conference on Innovation and technology in computer science education', ACM New York, NY, USA, pp. 196–196.

Forcier, R. and Descy, D. (2002), *The computer as an educational tool: Productivity and problem solving*, Prentice-Hall, Inc. Upper Saddle River, NJ, USA.

Gayo-Avello, D. and Fernandez-Cuervo, H. (2003), Online self-assessment as a learning method, *in* 'Advanced Learning Technologies, 2003. Proceedings. The 3rd IEEE International Conference on Advanced Learning Technologies', pp. 254–255.

Geisert, P. and Futrell, M. (2000), *Teachers, computers, and curriculum: Microcomputers in the classroom*, Allyn & Bacon.

Grabe, M. and Grabe, C. (2008), *Integrating technology for meaningful learning*, Houghton Mifflin.

Horton, W. (2000), *Designing web-based training: How to teach anyone anything anywhere anytime*, New York.

Horton, W. (2006), *E-learning by design*, Pfeiffer & Co.

Hot Potatoes (n.d.), http://hotpot.uvic.ca. Last accessed: 31 August 2009.

Huba, M. and Freed, J. (2000), *Learner-Centered Assessment on College Campuses: Shifting the Focus from Teaching to Learning.*, Allyn & Bacon, 160 Gould St., Needham Heights, MA 02494.

Iahad, N., Dafoulas, G., Kalaitzakis, E. and Macaulay, L. (2004), Evaluation of online assessment: The role of feedback in learner-centered e-learning, *in* 'Proceedings of HICSS', Vol. 1, Citeseer, p. 10006.1.

Inc, B. (2007), "Blackboard", *URL: http://www. blackboard. com* , Citeseer.

Janossy, J. (2007), Student reaction to podcast learning materials: preliminary results, *in* 'Instructional Technology Conference', p. 98.

Java Debug Interface (n.d.), http://java.sun.com/j2se/1.5.0/docs/guide/jpda/architecture.html. Last accessed: 31 August 2009.

Jordan, P. (1996), *Usability evaluation in industry*, Taylor & Francis, pp. 191–196.

Keegan, D. (1990), *Foundations of distance education*, 2nd edn, Routledge.

KeenGwe, J., OnChwari, G. and OnChwari, J. (2009), "Technology and Student Learning: Toward a Learner-Centered Teaching Model", *AACE Journal* , Vol. 17, pp. 11–22.

King, T. and Duke-Williams, E. (2001), Using computer aided assessment to test higher level learning outcomes, *in* 'Proceedings of 5th International Computer Assisted Assessment Conference', Citeseer, pp. 177–187.

Kleinman, J. and Entin, E. (2002), "Comparison of in-class and distance-learning students' performance and attitudes in an introductory computer science course", *Journal of Computing Sciences in Colleges* , Vol. 17, Consortium for Computing Sciences in Colleges, USA, pp. 206–219.

Kybartaite, A., Nousiainen, J., Marozas, V. and Jegelavicius, D. (2007), "Review of e-teachinge-learning Practices and Technologies", Biomedical Engineering Institute, Kaunas University of Technology (KUT),Ragnar Granit Institute (RGI), pp. 1–25.

Lance, D. (2000), "Venture capital viewpoints and elearning futures", *The Business of E-learning: Bringing your Organization in the knowledge E-conomy, University of Technology, Sydney* .

Lipkewich, A. (2008), "Introducing Web 2.0 to Staff", *Literacies, Learning & Libraries* , p. 13.

Macdonald, J. (2003), "Assessing online collaborative learning: process and product", *Computers & Education* , Vol. 40, Elsevier, pp. 377–391.

Marton, F. and Saljo, R. (1976), "On Qualitative Differences in Learning:1- Outcome and Process", *British Journal of Educational Psychology* , Vol. 46, pp. 4–11.

Marton, F. and Saljo, R. (1984), "Approaches to learning", *The experience of learning* , pp. 36–55.

Oliver, R., Omari, A. and Australia, W. (2001), "Exploring student responses to collaborating and learning in a web-based environment", *Journal of Computer Assisted Learning* , Vol. 17, Citeseer, pp. 34–47.

Pask, G. (1988), “Learning strategies, teaching strategies, and conceptual or learning style”, *Learning strategies and learning styles* , Plenum Pub Corp, pp. 83–99.

Question Mark Perception (n.d.), http://www.qmark.com. Last accessed: 31 August 2009.

QuizGuide (n.d.), http://www2.sis.pitt.edu/ taler/QuizGuide.html. Last accessed: 31 August 2009.

QuizPACK (n.d.), http://www2.sis.pitt.edu/ taler/QuizPACK.html. Last accessed: 31 August 2009.

Ramsden, P. (2003), *Learning to teach in Higher Education*, 2nd edn, RoutledgeFalmer.

Reiser, R. (2001), “A history of instructional design and technology: Part I: A history of instructional media”, *Educational Technology Research and Development* , Vol. 49, Springer, pp. 53–64.

Rogers, A. (1999), “The origins of a global learning network”, *Global SchoolNet* .

Schrader, P. (2008), “Learning In Technology: Reconceptualizing Immersive Environments”, *AACE* , Vol. 16, pp. 457–475.

Sitthiworachart, J. and Joy, M. (2004), “Effective peer assessment for learning computer programming”, *ACM SIGCSE Bulletin* , Vol. 36, ACM, New York, NY, USA, pp. 122–126.

Sommerville, I. (2004), *Software Engineering*, Addison Wesley.

Steinberg, E. (1991), *Computer-assisted instruction: A synthesis of theory, practice, and technology*, Lawrence Erlbaum.

Taylor, R. (2002), “Pros and cons of online learning-a faculty perspective”, *Journal of European Industrial Training* , Vol. 26, Bradford, West Yorkshire, England: MCB University Press, 1977-, pp. 24–37.

Tschalar, R. (n.d.), ‘JXXX Compiler Service’, http://www.innovation.ch/java/java_compile.html. Last accessed: 28 August 2009.

Turoff, M. (2000), “An end to student segregation: No more separation between distance learning and regular courses”, *On the Horizon* , Vol. 8, MCB UP Ltd, pp. 1–7.

Urdan, T. (2005), *Statistics in plain English*, Lawrence Erlbaum.

WebMCQ (n.d.), http://www.webmcq.com.au/. Last accessed: 31 August 2009.

Wikipedia (n.d.), http://en.wikipedia.org. Last accessed: 31 August 2009.

Appendices

Appendix A

Assessment Questions

A.1 Formative Assessment

Question 1:

What is the value *a* of after the following code is executed?

```
int a = 7;
int b = 4;
a = b;
a = a+1;
```

a. 4

b. 7

c. 5

d. 8

Question 2:

If x is an int and y is a float, all of the following are legal except which assignment statement?

a. x = y;

b. y = x;

c. y = (float) x;

d. x = (int) y;

Question 3:

What is the output produced by this code?

```
double result;
result= (2/3)*3;
System.out.println("(2/3)*3 equals "+result);
```

a. (2/3)*3 equals 2.0

b. (2/3)*3 equals 0

c. (2/3)*3 equals 0.0

d. (2/3)*3 equals 2

Question 4:

Evaluate the following expressions and fill in the blanks:

What is the values of x, y, z, and w?

```
int x = 2 * 5 * 5 + 3 * 5 + 7;
int y = 7 + 3 * 6 / 2 - 1;
int z = ( 3 * 9 *( 3 + 9 * 3 / 3) );
int w = 2 % 2 + 2 * 2 - 2 / 2;
```

---

x=[....], y=[....], z=[....], w=[....]

Question 5:

$$x = \frac{5 + 10 - 3}{2 * 3}$$

Which of the following represents the above equation?

a. x = (5 + 10 - 3)/(2 * 3);

b. x = (5 + 10 - 3)/ 2 * 3;

c. x = 5 + 10 - 3/(2 * 3);

d. x = 5 + 10 - 3 / 2 * 3;

Question 6:

Since you can not take the square root of a negative number, you might use which of the following instructions to find the square root of the variable x (x can be either positive or negative)?

a. Math.sqrt(-x)

b. Math.sqrt(x*x)

c. Math.sqrt((int)x)

d. Math.sqrt(Math.abs(x))

Question 7:

What will be the value of w after the following code executed?

```
public class ComputeW
{
 public static void main(String[] args)
  {
    int w = 4;
    int q = 6;
    if ( q > 5 )
        if ( w == 7 )
                 w = 3;
             else
                 w = 2;
        else
           if ( w > 3 )
                 w = 1;
             else
                 w = 0;
    System.out.println(w);
  }
}
```

a. 3

b. 2

c. 1

d. 0

Question 8:

Consider the following code that will assign a letter grade of ‘A’, ‘B’, ‘C’, ‘D’, or ‘F’ depending on student’s test score:

```
if ( score >= 90 )
          grade = ‘A’;
if ( score >= 80 )
          grade = ‘B’;
if ( score >= 70 )
          grade = ‘C’;
if ( score >= 60 )
          grade = ‘D’;
else
          grade = ‘F’;
```

a. This code will work correctly in all cases

b. This code will work correctly only if grade >= 60

c. This code will work correctly only if grade < 70

d. This code will not work correctly under any circumstances

Question 9:

Suppose x = 5, What is the value of x after the following switch statement executes?

```
switch (x)
{
    case 3: x+=1;
    case 4: x+=2;
    case 5: x+=3;
    case 6: x++;
    case 7: x+=2;
    case 8: x--;
    case 9: x++;
}
```

a. 5

b. 11

c. 8

d. 14

Question 10:

What is the output of the following code?

```
for(int i=1; i<=10, i++)
{
  if (i%2) != 0)
   System.out.println(i);
}
```

a. Prime numbers from 1 to 9

b. Prime numbers from 2 to 10

c. Odd numbers from 1 to 9

d. Even numbers from 2 to 10

Question 11:

What does the following code do?

```
int p = 1;
int n = 1;
while(n<=5)
{
  p = p * 2;
  n++;
}
System.out.println(p);
```

a. It computes 2*2*2*2*2

b. It computes 1*2+2*3+3*4+4*5

c. It computes 1*2*3*4*5

d. It computes 1+2+3+4+5

Question 12:

What does the following code do?

```
int product = 1;
int count = 1;
while(count<=5)
{
  product = product * count;
  count++;
}
System.out.println("product ="+product);
```

a. It computes 2^5

b. It computes 5^2

c. It computes 5!

d. It computes 2!

A.2 Summative Assessment

Question 1:

What is the value of *a* after the following code is executed?

```
int a = 10;
int b = 12 % 2 + 1;
a = b;
a = a+1;
```

a. 10

b. 11

c. 8

d. 2

Question 2:

An assignment statement to compute the gas mileage of a car where the int values $miles\_traveled$ **and** $gallons\_needed$ **have already been input. The variable** $gas\_mileage$ **needs to be declared and should be double.**

a. double gas_mileage = miles_travelled/gallons_needed ;

b. int gas_mileage = miles_travelled/gallons_needed ;

c. double gas_mileage = (double) miles_travelled/ gallons_needed ;

d. double gas_mileage = miles_travelled(int) gallons_needed ;

Question 3:

What is the output produced by this code?

```
double result;
result= (3.0/2.0)*3.0;
System.out.println("(3.0/2.0)*3.0 equals "+result);
```

a. (2/3)*3 equals 4.5

b. (2/3)*3 equals 3.0

c. (2/3)*3 equals 0.0

d. (2/3)*3 equals 4.0

Question 4:

Evaluate the following expressions and fill in the blanks:

What is the values of x, y, z, and w?

```
int x = 2 * 5 / 5 + 3 * 5 + 7;
int y = 8 - 3 * 6 / 2 + 1;
int z = ( 3 * 2 *( 3 + 3 * 6 / 3) );
int w = 8 / 2 % 3 + 2 * 2 - 2 / 2;
```

x=[....], y=[....], z=[....], w=[....]

Question 5:

$$A = \frac{12}{18 - 3x4}$$

The following code is written to compute and print the value of A. What type of error this code might have?

```
class ComputeA
{
  public static void main(String[] args)
  {
  int A = 12 / 18 - 3 * 4;
  System.out.println(A);
  }
}
```

a. Execution error

b. Logical error

c. Syntax error

d. Non of the above

Question 6:

Since you can not take the square root of a negative number, you might use which of the following instructions to find the square root of the variable x (x can be either positive or negative)?

a. Math.sqrt(-x)

b. Math.sqrt((Math.sqrt(x*x))

c. Math.sqrt((int)x)

d. Math.sqrt(x*x)

Question 7:

What will be the value of w after the following code executed?

```
public class ComputeW
{
 public static void main(String[] args)
  {
    int w = 2;
    int q = 3;
    if ( q > 5 )
        if ( w == 7 )
                 w = 3;
             else
                 w = 2;
        else
           if ( w > 3 )
                 w = 1;
             else
                 w = 0;
    System.out.println(w);
  }
}
```

---

w = [....]

Question 8:

Consider the following code that will assign a letter grade of ‘A’, ‘B’, ‘C’, ‘D’, or ‘F’ depending on student’s test score:

```
if ( score >= 90 )
          grade = ‘A’;
else if ( score >= 80 )
          grade = ‘B’;
else if ( score >= 70 )
          grade = ‘C’;
else if ( score >= 60 )
          grade = ‘D’;
else
          grade = ‘F’;
```

---

a. This code will work correctly in all cases

b. This code will work correctly only if grade >= 60

c. This code will work correctly only if grade < 70

d. This code will not work correctly under any circumstances

Question 9:

What is the output produced by the following code?

```
int x = 0;
for(int i = 3; i<=6; i++)
{
  switch (i)
  {
    case 3: x+=1;
    case 4: x+=2;break;
    case 5: x+=3;
    case 6: x--;
  }
  System.out.println(x+" ");
} // end of for
```

a. 1 2 3 2

b. 1 5 6 5

c. 3 5 8 7

d. 3 5 7 6

Question 10:

What is the output of the following code?

```
for(int i=1; i<=10, i++)
{
  if (i%2) == 0)
   System.out.println(i);
}
```

a. Prime numbers from 1 to 9

b. Prime numbers from 2 to 10

c. Odd numbers from 1 to 9

d. Even numbers from 2 to 10

Question 11:

What does the following code do?

```
int s = 0;
int n = 1;
while(n<=5)
{
  s = s + n;
  n++;
}
System.out.println(p);
```

a. It computes 2*2*2*2*2

b. It computes 1*2+2*3+3*4+4*5

c. It computes 1*2*3*4*5

d. It computes 1+2+3+4+5

Question 12:

What does the following code do?

```
int product = 1;
int count= 1;
while(count<=5)
{
  product = product * 5;
  count++;
}
System.out.println("product ="+product);
```

a. It computes 2^5

b. It computes 5^5

c. It computes 5!

d. It computes 2!

Appendix B

JT Application Description

B.1 Requirement Analysis for JT Application

A requirement is simply a statement of a service that the system should provide and its operational constrains (Sommerville, 2004). From different point of view, a requirement can be broadly divided into user requirement and system requirement.

B.1.1 User Requirement

User requirements, sometimes called business requirements, focus on business user needs. User requirements provides what we expect from the system services and the constrains of its operations (Sommerville, 2004).

B.1.2 JT User Requirement

Communication:

JT application is suitable application for all students who are familiar with simple application. It can be easily run on Windows operating environment.

Accuracy:

In JT application can load files specified java files in real time, wherever they are stored. So JT application ensures that students can load the java specified file and get immediate and accurate description for the code they need to learn.

Constancy:

JT application solves the trouble of choosing the right file to load, i.e. when a specific directory is being accessed by a user, only java files are shown to be selected. Also it ensures that the file chosen contains a main function to be analyzed and run by the application.

Error Indication:

If a user makes some mistakes on choosing file that does not contain a main method, or the file have any compilation errors, there will be a notice dialog with system message provided for leading user to correct his/her operation.

Friendly GUI:

Graphic User Interface of JT application design concentrates on two issues: readable and friendly. Students can easily deals with file loading, compiling and translating through an easy GUI component. The students can easily understand the basic components relatively to their functions which is easy to use.

Flexible Access:

JT application makes it easy to students to access to all java files and directories that are located on user local machine. It also compiles, run and get description for the program located on any directory on the user local machine.

Guidance:

JT application provides useful and helpful information to introduce a simple java integrated environment. It is used to help students understand the basic functions to compile and run simple java applications smoothly and comfortably.

B.1.3 JT System Requirements

From the developer's perspective, requirements are usually called system requirement. Actually, there is no obvious distinction dividing a user requirement and a system requirement. Requirements are the description of the services provided by the system and it operational constrain, which reflects the customers needs for solving a problem (Sommerville, 2004). System requirement gives functions, services and operational constrains in detail. can be break down into functional requirement and non-functional requirement in nature (Dennis et al., 2005).

B.1.4 Functional Requirement

A functional requirement describes the services that the system should provide, how it reacts to the user input, how it behaves a particular situations, and what the system should not do. JT application is a single user interface used to translate java programs. In JT application, the most of functional requirements focus on the java file operations and Java Virtual Machine Interface.

Open:

A file/directory operation can be executed on local machine. It is used for selecting a java file to be opened and translated for a verbal description on the program code. The Open operation allows the user to choose form any java program file that are located on the user local machine.

Compile:

A file operation is allowed on local machine. The user can compile the loaded java file on any location on the local machine, i.e. the Compile process will handle compiling the code as an integrated environment. It will insure that the selected file is free from errors and contains a java main method. It is also manages the connection to JVM to get the data flow and the variables memory content at each step of the program code.

Start/Next:

A step operation is executed at each line of the program and manages the description for that line. User is able to get synchronously the line of the program code with the verbal description of it as Next button is pressed.

Restart:

An operation that the user can use to restart the program description form the beginning.

Exit:

An operation that allows the user to exit the application at anytime.

B.1.5 Non-Functional Requirement

Non-functional requirements reflect the constrains on the system functions or services. It is generally applied to the system as a whole, and it can be divided into operational requirements, performance requirements, security requirements, cultural and political requirements (Dennis et al., 2005).

B.1.5.1 Operational Requirements

1. JT application will operate in Window, Linux.

2. JT application will translate only java program codes.

3. JT application will give prompt and relative translation to the program code.

B.1.6 Performance Requirements

Once the file is loaded and compiled with no errors, the user can start viewing the verbal descriptions for the program code. Users can step line by line on the code at their needs.

B.1.7 Security Requirements

There is no security pre-condition for using JT application. Users can run JT application by double click on the JT application icon where it is located.

B.1.8 Cultural and Political Requirements

No special cultural and political requirements are anticipated.

B.2 Functional Modeling for JT Application

Functional models describe the interaction of JT application with its environment Java. In objected-oriented system development, tow types of models are used to describe the functionality of JT application: use cases and activity diagrams.

B.2.1 Use Case Diagrams

A use case diagram illustrates the main functions of JT application, the user and JVM which process the data analysis for the JT application.

B.2.2 Actors/Roles in JT Application

An actor is not a specific user in design of information system, but a role that a user can play while interacting with the system. An actor can also represent another system in which current system interacts. In JT, the actor plays a specialized role: accessing and translation a java code, see Figure B.1.

User: is a system user who can loading any java program files operation and perform basic JT application operations such as load, compile and translate.

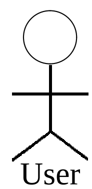


Figure B.1: An Actor in JT Application

B.2.3 JT Use Cases

The following diagram describes the user cases in using JT application. See fig B.2

B.2.4 Relationships

Association Relationship

Links an actor with the use case(s) with which it interacts

Include Relationship

Represents the inclusion of the functionality of one use case within another. The arrow is drawn from the base use case to the included use case.

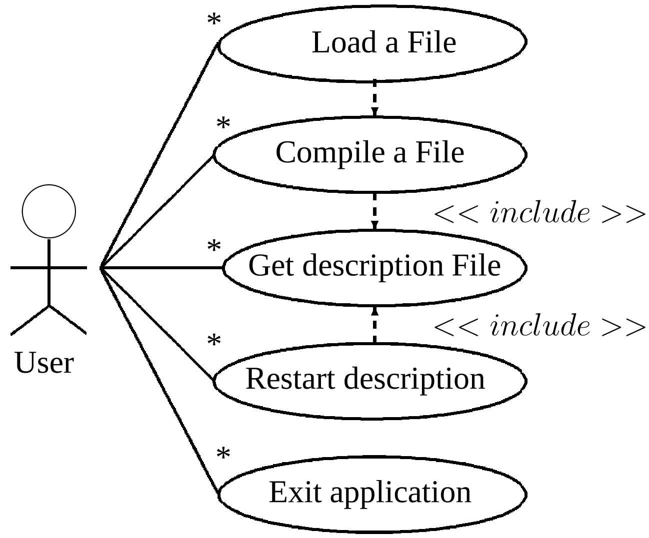


Figure B.2: JT Use Case Diagram

Extend Relationship

Represents the extension of the use case to include optional behavior. The arrow is drawn from the extension use case to the base use case.

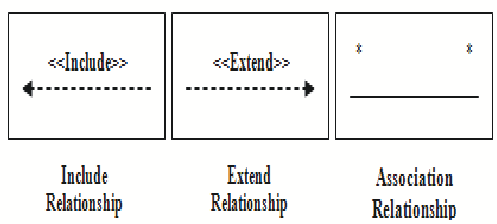


Figure B.3: User Cases Relationships

B.2.5 Use Cases Description

A use case description contains all the information needed to build the diagram. There are three basic parts in use case description: overview information, relationships, and the flow of events. There are five use cases in the JT application, Load, Compile, Start, Restart and

Exit.

1. Load File

   Use Case Name Load a file

   ID 1

   Primary Actor user (Student)

   Brief Description This use case describes how the student can load a java file on local machine.

   Preconditions User should run JT application in success.

   Relationships Association: user (Student)

   Normal Flow of Events

   (a) The student runs JT application successfully.

   (b) The student opens a file chooser window by clicking 'Load' button.

   (c) The student selects select the java file which he/she wants to get its detailed description.

2. Compile a Java File

   Use Case Name Compile a java file

   ID 2

   Primary Actor user (Student)

   Brief Description This use case describes how a user can compile a selected java file using JVM to obtain variables values, methods, and program flow.

   Preconditions

   - Student should load the file before compiling.
   - The file have to be of java program and contains main method.
   - The program is free from syntax errors.

Relationships Association: user (Student).

Normal Flow of Events

(a) The user have to load the java file.

(b) JVM complies the Java program to make sure that the program is successfully running and free form any syntax errors.

(c) The application also check if the compiled code contains a main method which make the class runable.

Alternative Flow

(a) If the compiled contains any compilation errors, JT application will report that to the student. It will display a dialog containing the errors founded in the compilation process.

(b) If the loaded code is free from errors, but does not contain a main method, there will be a message dialog indicating that this class does not contain a main method.

3. Start/Next

Use Case Name Get description.

ID 3

Primary Actor user (Student)

Brief Description This use case describes how the student can trace each line of the code and get the description for it.

Preconditions

- The java file must be loaded.
- The file must be compiled and free from any syntax errors.
- The java file must contains a main method.

Relationships Association: user (Student)

Normal Flow of Events

(a) The user loads the java file successfully.

(b) The user compiles the file successfully.

(c) The java program contains a main method.

(d) The data is collected from JVM, output such as variables values, called methods and program sequence is used with the application moderator to get the description for each line of the code.

Alternative Flow

(a) If the compiled contains any compilation errors, JT application will report that to the student. It will display a dialog containing the errors founded in the compilation process.

(b) If the loaded code is free from errors, but doesn't contain a main method, there will be a message dialog indicating that this class does not contain a main method.

4. Restart

Use Case Name Restart program description.

ID 4

Primary Actor User (Student).

Brief Description This use case describes how a user can restart the program description form the beginning.

Preconditions

- The java file must be loaded.
- The file must be compiled and free from any syntax errors.
- The java file must contains a main method.
- The program moderator contains all required variables, methods and program flow.

Relationships Association: user (Student).

Normal Flow of Events

(a) The user loads the java file successfully.

(b) The user compiles the file successfully.

(c) Ensure that the java program contains a main method to enable execution.

(d) The data is collected from JVM, output such as variables values, called methods and program sequence is used with the application moderator to get the description for each line of the code.

(e) Restart will initialize all the program sequence to start from the beginning as it runs from the first time

5. Exit

Use Case Name Exit JT application

ID 5

Primary Actor User (Student).

Brief Description This use case describes how the student exits the JT application at anytime.

Preconditions User can exit the application at anytime, there is no precondition unless the application is actually running.

Relationships Association: user (Student).

Normal Flow of Events

(a) The user runs JT application successfully.

(b) The user can exit the application by clicking Exit button on JT application menu.

(c) The user will be asked to confirm his choice before ending the application.

Alternative Flow There will be question dialog prompt for the user. If "Yes" is selected, the JT application ends successfully, else the application will be running.

B.2.6 Activity Diagrams

Activity diagram explains the primary activities and the relationships between the activities in the process. Fig B.4 shows the syntax and the activity diagram (Dennis et al., 2005).

Activity:	is used to represent a set of actions (behaviour).
Object Node:	is used to represent an object that is connected to a set of Object Flows.
Initial Node:	portrays the beginning of a set of actions or activities.
Final-Activity Node:	stop all control flows and object flows in an activity.
Control Flow:	shows the sequence of execution.
Object Flow:	shows the flow of an object from one activity/action to another activity/action.
Decision Node:	a test condition to decide that the control flow/object flows only goes down one path.
A Fork Node:	is used to split the behaviour into a set of parallel or concurrent flows of activities (or actions).
A join Node:	is used to bring back together a set of parallel or concurrent flows of activities (or actions).

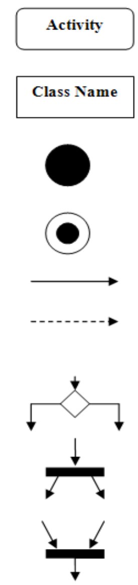


Figure B.4: Elements of Activity Diagram

Open File Activity Diagram Fig B.5 illustrates open file activity diagram.

- The user selects 'open file' menu form the program interface.
- An open dialog is showed to the user to browse the program file. Note that this file chooser is filtered to display only java programs and folders.
- If the user confirmed his choice for a certain program. It will load that file form the path specified by reading its file.
- If there is a reading error, the open file activity shows an error message and terminates, other wise it display the java program file (as a list) in the program area on the interface.

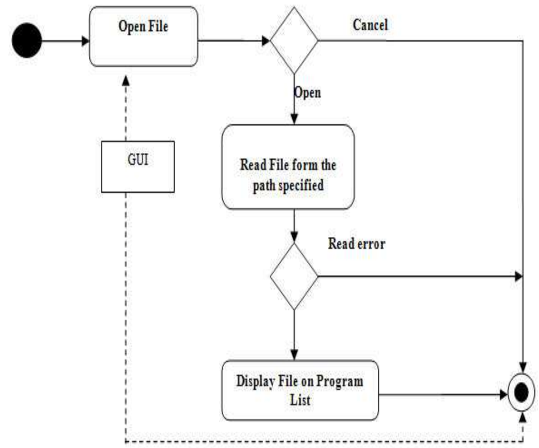


Figure B.5: Open File Activity Diagram

Compile java File Activity Diagram Fig B.6 illustrates compile file activity diagram.

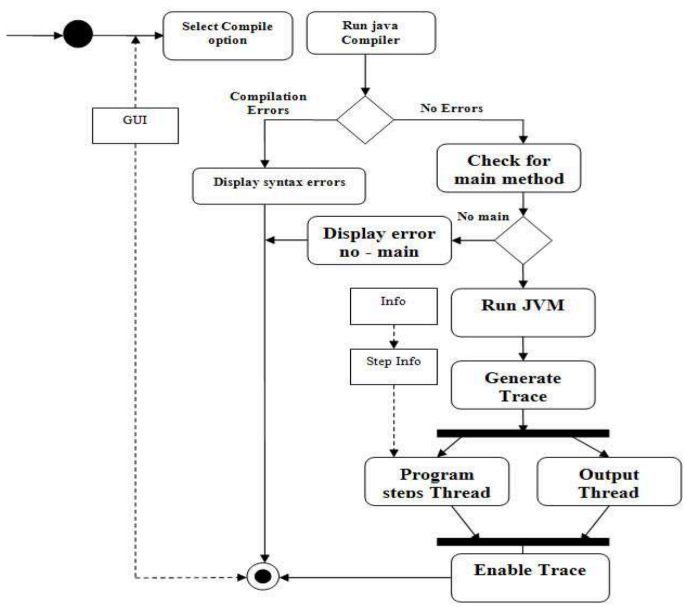


Figure B.6: Compile File Activity Diagram

- After the program has been loaded the user will be able to compile the loaded code by clicking on the Compile button on the menu.
- Compile will run JavaCompiler class to compile the code. If the compile is not successful the application will report the errors for the user and terminates.

- If the compilation successful, the application will check if the program contains main method to run. If the java program loaded doesn't contain main method, it will terminate.
- If the compilation successful and the program contain main method, the application will run JVM (java virtual machine to execute the program).
- Two threads will work for gathering the information form the trace process. One thread to collect the output of the program and the other will collect information for every step in the program. This information includes line number, variables, methods located at each step.
- The allocated information will kept in vectors to be used later for the application translation process.

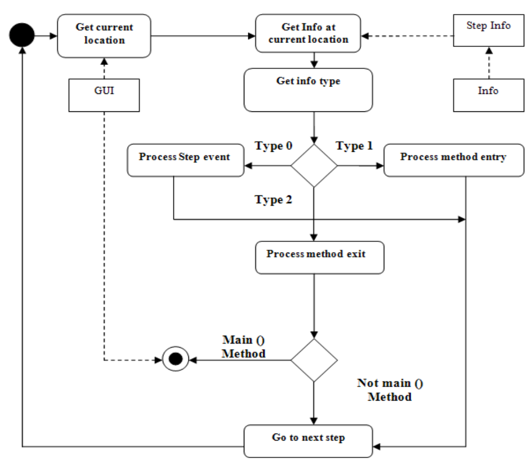


Figure B.7: Start Next Activity Diagram

Start/Next Activity Diagram Fig B.7 illustrates Start and Next activity diagram.

1. The application starts at the event step code and get the information at the current line.
2. There are three types of processing information inside each step:
   (a) If the information type is 0, this means it is a step event and the applica-

tion will start processing it. Step event can contains any java statements such as assignment statement, if statement, for or print statement. After executing this step and process it, the application step to the next step and go to step 1.

(b) If the information type is 1, this means it is a method entry event and the application will start processing it. Method entry event can contains any called java method inside the class. After executing this method entry event and processing it, the application step to the next step and go to step 1.

(c) If the information type is 2, this means it is a method exit event and the application will start processing it. Method event can contain any called java method. If the method called here is main, it means exiting the translated program, otherwise, the application will process the next step and go to step 1.

Restart Activity Diagram Fig B.8 illustrates Restart activity diagram.

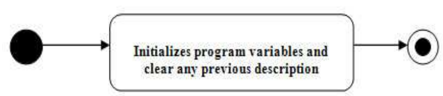


Figure B.8: Restart Activity Diagram

- The application will set all the variables to its initial values as after compilation of the program.
- The description area is cleared of any previous description.

Exit Activity Diagram Fig B.9 illustrates Exit activity diagram.

- User selects Exit button form the menu, and the application prompts the user for his choice to confirm his selection.

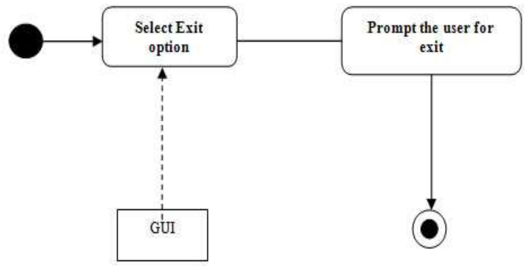


Figure B.9: Exit Activity Diagram

- If the user confirmed his selection, then the application exited.

B.3 Structural Modeling

Structural models illustrates people, things, places, or concepts, that is, objects contained in the domain of the problem and the relationships between them.

B.3.1 Class Diagram

A class diagram is used to show the classes and the relationships between them. Each class contains attributes and operations. There are different types of relationships used between classes (Dennis et al., 2005). Below highlights the items and relationships that have been used in creating JT application. Fig B.10 describes the elements of class diagram.

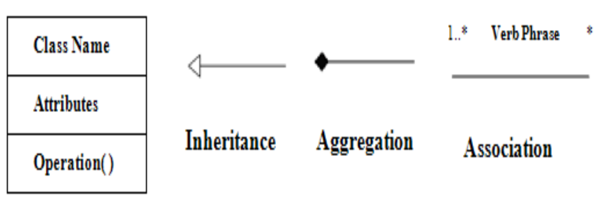


Figure B.10: Elements of Class Diagram

Class Is the main building block of class diagram that is used to store information about the system objects. Each class has three rectangles: class name, class attributes and class operations. Attribute represents properties used to describe the state of object and operation represents the actions or functions that a class can perform.

Inheritance A relationship that is used to inherit the attributes and operations of a particular class from another class. The triangle touches the parent class.

Aggregation An aggregation relationship has been used for the classes that comprise other classes. The diamond is placed near the class that represents the aggregation.

Association An association relationship has been used for relationship between multiple classes or a class and itself. It labeled using a verb phrase and contains a multiplicity symbols.

The class diagram of JT application includes five main classes: JT, Info, StepInfo, StreamRedirectThread, and EventThread. The figure shows these classes and the relationships between them :

B.3.2 Detailed Description

The following paragraphs give a brief description for some of the main classes in JT application and the relationships between them.

JT: It contains the Graphical User Interface (GUI) of the JT application. The user has to start any command from this class. It is responsible for loading, compiling, executing, and manipulating java program code. It uses EventThread to processes incoming JDI events and store them to be manipulated later by JT getDescription() method.

EventThread: This class processes incoming JDI events and stores it to a vector that is later processd by the JT application.

ThreadTrace: This class keeps context on events in one thread.

StreamRedirectThread: is a thread which copies it's input to a vector and terminates when it completes. It is basicaly takes the running java program output as its input and store it in a vector that is later processed in the JT application.

StepInfo: This class is responsible for storing the running events on the VM. It stores each event information such as type, location and any variable data located on that step.

Info: It will store all the information for each variable located on that step list. It stores variable type, name and value.

B.4 Behavioral Modeling

Behavioral modeling describes the underlaying collaboration between objects. It contains two types of diagrams, communication diagram and sequence diagram(Dennis et al., 2005). Fig B.11 is Sequence diagram to describe the interaction of JT application. It shows the order of the activity that take place among a set of objects depending on the time-based ordering.

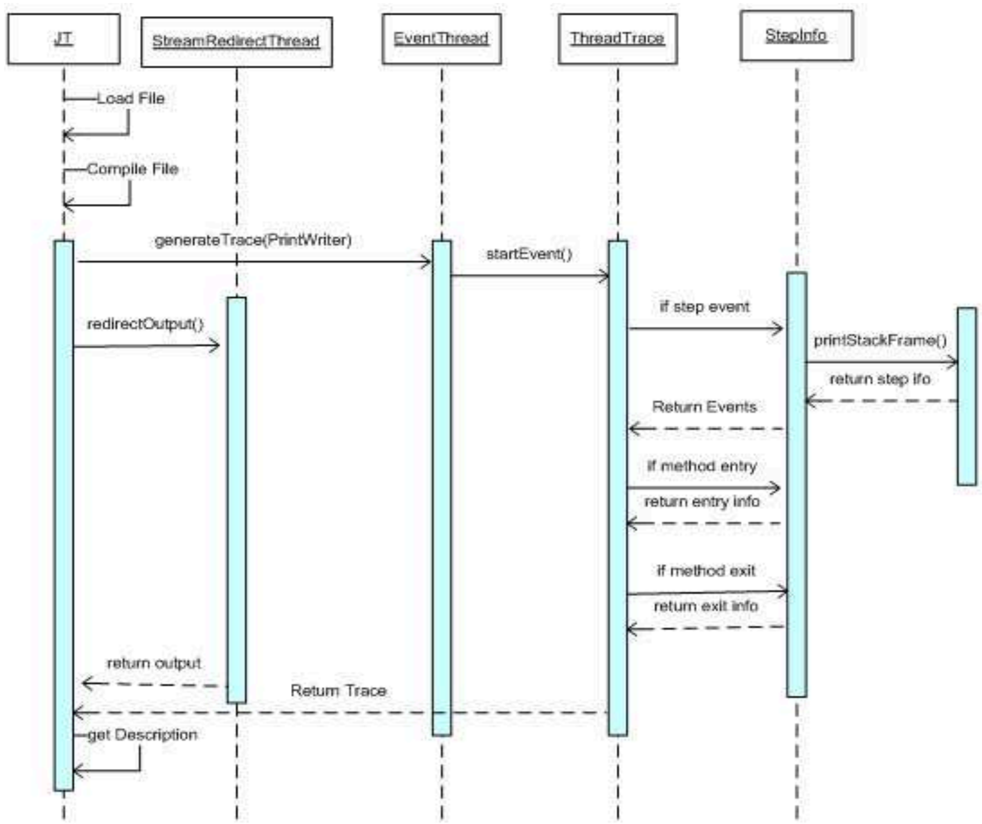


Figure B.11: JT Sequence Diagram

Appendix C

Case Studies for the Proposed Model

C.1 Chemistry Lesson (Balancing a chemical equation):

C.1.1 Balancing a chemical equation Test

1. A balanced equation represents the law of mass conservation, and as a result all of the following are true except .....

   a. the number of moles of products equals the number of moles of reactants (**Correct**).

   b. the total mass of the product equals the total mass of the reactants.

   c. the number of atoms in the products equals the number of atoms in the reactants.

   d. the states of matter of the products are independent of the states of reactants

2. **Gap-Fill Exercise:**

   Enter your answers in the gaps. Every space will require a coefficient. Unlike when we balance equations in class, you will have to include coefficients of “one” by typing in a value of "1." When you have entered all the answers, submit your answers.

   (a) $...H_2 + ...O_2 \rightarrow ...H_2O$

(b) $...H_2 + ...N_2 \rightarrow ...NH_3$

(c) $...Al_2O_3 \rightarrow ...Al + ...O_2$

(d) $...KClO3 \rightarrow ...KCl + ...O_2$

(e) $...S_8 + ...O_2 \rightarrow ...SO_2$

(f) $...C_2H_6 + ...O_2 \rightarrow ...CO_2 + ...H_2O$

(g) $...Al_2(SO_4)_3 + ...Ca(OH)_2 \rightarrow ...Al(OH)_3 + ...CaSO_4$

(h) $...P_4 + ...O_2 \rightarrow ...P_2O_5$

(i) $...Ag + ...S_8 \rightarrow ...Ag_2S$

(j) $...Al + ...Br_2 \rightarrow ...AlBr_3$

(k) $...Cr + ...O_2 \rightarrow ...Cr_2O_3$

(l) $...NaClO3 \rightarrow ...NaCl + ...O_2$

C.1.2 Balancing a chemical equation Pre-Assessment Lesson

Title: Chemicals Reactions

Sub-Title: Reaction Definition

- **Definition**: A chemical change which forms new substances.
- **Example**: The chemical reaction $H_2 + O_2 \rightarrow H_2O$ describes the formation of water from its elements.

Sub-Title: Element Definition

A substance that cannot be broken down by chemical means. Elements are defined by the number of protons they possess. Examples: **copper, cesium, iron, and neon**.

Sub-Title: Chemical Equations

When a chemical reaction occurs, it can be described by an equation. This shows the chemicals that react (called the reactants) on the left-hand side, and the chemicals that they produce (called the products) on the right-hand side. The chemicals can be represented by their names or by their chemical symbols. Unlike mathematical equations, the two sides are separated by an arrow that indicates that the reactants form the products and not the other way round.

Example 1:

This chemical word equation:

$$Aluminum + Oxygen \rightarrow AluminumOxide$$

This is the equation for the burning of aluminum in oxygen. If we convert each of the chemical names into the appropriate symbols, we get the following:

$$Al + O_2 \rightarrow Al_2O_3$$

Example 2:

$$Sodium + Oxygen \rightarrow SodiumOxide$$

Sodium metal is a soft silvery-white lustrous metal which can be easily cut with a knife. Sodium metal is highly reactive and tarnished readily in air due to the formation of a layer of Sodium Oxide on the surface of the metal.

$$Na + O_2 \rightarrow Na_2O$$

Summary

A chemical reaction is a transformation resulting from the interaction between two atoms or molecules. Reactions involve the breakage and reformation of the chemical bonds which hold atoms together and can therefore cause changes in the structure or composition of substances.

A reaction may involve two or more reactants that combine to form a reaction product; it may involve a compound that breaks down to create individual reactants. Chemical equation is a written representation of a chemical reaction, showing the reactants and products, their physical states, and the direction in which the reaction proceeds. A chemical equation provides quantitative information about a chemical reaction, but only if it is **balanced**. For a chemical equation to be balanced, the same number of each kind of atom must be present on both sides of the chemical equation.

C.1.3 Balancing a Chemical Equation Pre-Assessment Quiz

Please choose the right answer:

1. What is a chemical reaction?
   a. a process of bonding substances together.
   b. a process that leads to filled valence shells.
   c. a process that releases or consumes energy.
   d. all of the choices.
2. What is a chemical equation?
   a. a diatomic molecule
   b. a shorthand representation of a chemical reaction
   c. a substance
   d. all of the choices
3. What are substances on the left side of a chemical equation called?
   a. electrons
   b. groups
   c. products
   d. reactants
4. What are substances on the right side of a chemical equation called?

a. electrons

b. groups

c. products

d. reactants

5. What does the number in front of the molecular symbol mean?

a. It shows the number of molecules in the reaction

b. It shows that there is only one molecule present in a reaction

c. It shows the number of reactions that took place

d. All of the choices

6. Can the subscripts in a chemical reaction be changed?

a. Cannot be determined

b. Maybe, depending on the energy released

c. No, they define the molecules in the reaction

d. Yes, to balance the equation

7. 16.05 g of methane reacts completely with 64.00 g of oxygen to form carbon dioxide and water. What will be the total weight of the CO_2 andH_2O?

a. the weight cannot be determined.

b. 16.05

c. 64

d. 80.05

8. (8) When the equation below is balanced, the coefficient in front of carbon dioxide is:

$$C_3H_8 + O_2 \rightarrow CO_2 + H_2O$$

a. 1

b. 3

c. 6

d. 12

9. When the equation $H_2 + O_2 \rightarrow H_2O$is balanced?

a. $2H_2 + O_2 \rightarrow H_2O$

b. $H_2 + 2O_2 \rightarrow 2H_2O$

c. $2H2 + O_2 \rightarrow 2H_2O$

d. The equation is already balanced.

10. When the equation $Fe + CuCl_2 \rightarrow FeCl_2 + Cu$ is balanced?

a. $2Fe + CuCl_2 \rightarrow FeCl_2 + Cu$

b. $Fe + CuCl_2 \rightarrow FeCl_2 + 2Cu$

c. $2Fe + CuCl_2 \rightarrow FeCl_2 + 2Cu$

d. The equation is already balanced.

C.1.4 Balancing a chemical equation - Creating Lesson content

Lesson Title: Balancing a Chemical Equations

Sub-Title: Introduction

“**Reactants go to Product**”. To understand the concept for balancing a chemical equation, see the following example: In order to produce a bicycle we could say:

$$Wheel + Frame \rightarrow Bicycle$$

By looking to fig C.1 reactants are not balanced, because to have a bicycle we need two wheels and one frame and in the above equation we have one wheel and one frame on the left-hand side.

Figure C.1: Two Wheels and One Frame Will Make One Bicycle

According to Dalton's Atomic Theory , "A chemical reaction is a rearrangement of atoms" so we know that atoms can not be destroyed by chemical reactions. So in order to balance the above equation we need to write it in the form:

$$2Wheels + 1Frame \rightarrow 1Bicycle$$

So balancing chemical equations is very similar to the equation above. Note that if we want to produce 2 Bicycles, we can rewrite the equation in the following form:

$$4Wheels + 2Frame \rightarrow 2Bicycles$$

Basically, multiplying the whole balanced equation by 2 will do the job.

Sub-Title: Balancing a chemical equations

Balancing a chemical equation refers to establishing the mathematical relationship between the quantity of reactants and products. The quantities are expressed as grams or moles.

Sub-Title: Steps

It takes practice to be able to write balanced equations. There are essentially three steps to the process:

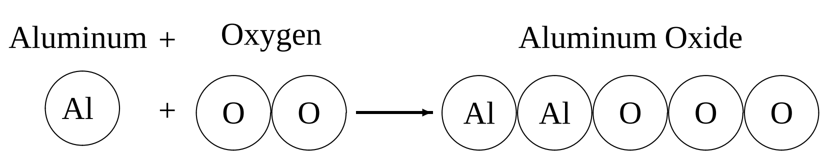
1. Write the unbalanced equation.
    - Chemical formulas of reactants are listed on the left-hand side of the equation.

- Products are listed on the right-hand side of the equation.
- Reactants and products are separated by putting an arrow between them to show the direction of the reaction. Reactions at equilibrium will have arrows facing both directions.

2. Balance the equation.
   - Apply the Law of Conservation of Mass to get the same number of atoms of every element on each side of the equation. Tip: Start by balancing an element that appears in only one reactant and product.
   - Once one element is balanced, proceed to balance another, and another, until all elements are balanced.
   - Balance chemical formulas by placing coefficients in front of them. Do not add subscripts, because this will change the formulas.

Sub-Title: Example 1

Consider the following chemical reaction : You can see that there is something



wrong with this equation. If you count the number of atoms of each type on each side, you will see that there is only one Aluminum atom on the left side whereas there are two on the right. There are two oxygen atoms on the left side, as compared to three on the right side. This clearly doesn't match.

- We can balance the equation by multiplying the different atoms and molecules on each side by different amounts. Firstly, multiply the aluminum atoms on the left side by 2:

$$2Al + O_2 \rightarrow Al_2O_3$$

- Now there are the same numbers of aluminum atoms on each side of the equation. We could also multiply the number of oxygen molecules on each side by one and a half (1.5), which would give three oxygen atoms on the left side (1.5 x 2 = 3) to match the three oxygen atoms on the right side:

$$2Al + 1.5O_2 \rightarrow Al_2O_3$$

- This is now balanced, but that 1.5 is a horrible thing to have in an equation - how can you have one and a half molecules? We can solve this problem by multiplying everything throughout by 2:

$$4Al + 3O_2 \rightarrow 2Al_2O_3$$

Sub-Title: Example 2

By looking at the table C.1 it shows that there is something wrong with this equa-

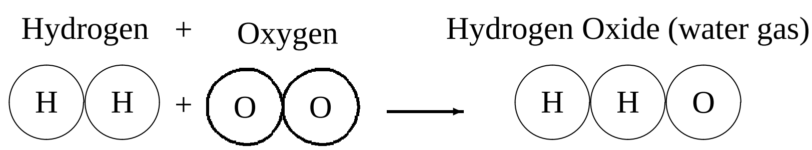


Table C.1: Number of Atoms in Hydrogen and Oxygen Reaction

	Before	After
H	2	2
O	2	1

tion. If you count the number of atoms of each type on each side, you will see that there are only two Oxygen atoms on the left side whereas there are two on the right. There are two Hydrogen atoms on the left side, as compared to two on the right side, so the Hydrogen is balanced. You can use the table to monitor the number of atoms in each element of the equation. It helps you to visualize and balance the equation.

- Firstly, multiply the water molecules on the right side by 2. This will make 2 oxygen atoms on the right hand side and two oxygen atoms on the left hand side, but 2 Hydrogen atoms on the left and four on the right. See table C.2 $H_2 + O_2 \rightarrow 2H_2O$

Table C.2: Number of Atoms in Hydrogen and Oxygen Reaction after Multiplying the Water Atoms on the Right Hand Side by 2

	Before	After
H	2	4
O	2	2

- Now there are the same numbers of Oxygen atoms on each side of the equation, but we need to balance the hydrogen. We could multiply the number of Hydrogen molecules on the left side by two, which would give four Hydrogen atoms on the left side (2 x 2 = 4) to match the four Hydrogen atoms on the right side. The equation is then balanced. $2H_2 + 1O_2 \rightarrow 2H_2O$

Example 3 (sub-title)

$$Ethanol + Oxygen \rightarrow HydrogenOxide(watergas) + Carbondioxide$$

$$C_2H_5OH + O_2 \rightarrow H_2O + CO_2$$

Table C.3: Number of Atoms in Ethanol and Oxygen Reaction

	Before	After
C	2	1
H	6	2
O	3	3

- First, we can start by balancing the Carbon by multiplying **Carbon dioxide** by 2. We will get the equation:

$$1C_2H_5OH + O_2 \rightarrow H_2O + 2CO_2$$

Table C.4: Number of Atoms in Carbon, Hydrogen and Oxygen after Multiplying the Carbon Dioxide on the Right Side by 2

	Before	After
C	2	2
H	6	2
O	3	3

- Second, we need to balance the Hydrogen. We have six hydrogen on the left and 2 in the right hand side. We can multiply the Hydrogen in the right hand side by 3 to make it six. The hydrogen is then balanced. We will get the equation:

$$1C_2H_5OH + O_2 \rightarrow 3H_2O + 2CO_2$$

Table C.5: Number of Atoms in Carbon, Hydrogen and Oxygen after Multiplying the Hydrogen in the Right Hand Side by 3

	Before	After
C	2	2
H	6	6
O	3	7

- Finally, we need to balance the Oxygen. We have 3 Oxygen on the left and 7 in the right hand side. We can multiply the Oxygen in the left hand side by 3 to make it six. The Oxygen is then balanced. We get the equation:

$$1C_2H_5OH + 3O_2 \rightarrow 3H_2O + 2CO_2$$

Table C.6: Number of Atoms in Carbon, Hydrogen and Oxygen after Multiplying the Oxygen in the Left Hand Side by 3

	Before	After
C	2	2
H	6	6
O	7	7

Balancing equations - a summary (Summary Title)

There is no exact method to balance an equation, but there are some strategies that you can follow.

1. If an element occurs only once on each side balance that first.
2. Balance free reactants (one element) last.
3. Don't worry if the numbers turn out to be fractions - you can always double or treble all the numbers at a later stage.
4. You may only put numbers in front of molecules, never altering the formula itself.

C.2 History Lesson (Hijra of Prophet Muhammad)

C.2.1 Hijra of Prophet Muhammad Test

1. Why did Prophet Mohammad decided for his flight to Madinah (Hijrah)?

   a. Because Makka city authorities decided to kill him.(Correct)

   b. Because his followers was there.

   c. Because his family went to Madinah.

2. The one who slept in the prophet's bed in the night of Hijra when Quraish conspired to murder him was...

   a. Ali (Correct)

   b. Umar

   c. Abubakr

3. Where did prophet Mohammed hide after he left Makkah

   a. cave of Thaur (Correct).

   b. In Quba.

   c. In Makkah

4. ...Was the one who tried to kill the prophet on his way of Hijra

   a. Suraqah Ibn Malik (Correct)

   b. Abu Jahl

   c. Abu Lahab

5. What did protect Prophet Mohammed at the cave entrance:

   a. a spider's web.

   b. a spider's web and pigeons with a nest at the entrance. (Correct)

   c. a pigeons with a nest at the entrance.

6. ...Was the reward for whom gives a correct information about Prophet Mohammad location.

   a. 1000 Camels.

   b. 100 Camels. (Correct)

   c. 110 Camels.

7. How long did Prophet Mohamed journey form Makkah to Quba in Madinah?

   a. 12 (Correct).

   b. 26

c. 36

8. How long did Prophet Mohamed stay in Quba before going in Madinah?

    a. 12

    b. 14 (Correct).

    c. 36

9. Who did accompany Prophet Mohammad on his way to Madinah?

    a. Umar.

    b. Abubakr (Correct).

    c. Ali.

10. What is the correct route for Prophet Mohammad's Hijra?

    a. Makkah- cave of Thaur- Madinah - Quba.

    b. Quba -Makkah- cave of Thaur - Madinah.

    c. Makkah- cave of Thaur- Quba - Madinah (Correct).

11. What did Prophet Mohammad do when he has arrived Quba?

    a. He laid the foundation stone for the first mosque of Islam. (Correct)

    b. He laid the foundation of Islamic brotherhood.

    c. He built Al Masjid-E-Nabawi

12. Hijra Calendar started relative to....

    a. The date when Prophet Mohammad migrated from Makkah to Madinah.

    b. The date when Prophet Mohammad has built Masjid-E-Nabawi.

    c. The date when Prophet Mohammad has built Quba Mosque.

C.2.2 Hijra of Prophet Muhammad Pre-Assessment Lesson

Title: Hijra of Prophet Muhammad

Sub-Title: Introduction

Prophet Mohammad Hijra form Makkah to Madinah has played an important role in Islamic community. You will be introduced his journey and follow the list of events that took a major part of his journey.

- By 615, Muhammad had gained about 100 converts in Mecca. City leaders became hostile to him, and in 619 his uncle Abu Talib died and was succeeded as head of the Hashim clan by another one of Muhammad's uncles, Abu Lahib. Abu Lahib refused to protect Muhammad, and persecution of the prophet and his Muslims increased. Prophet Mohammad encouraged his Meccan followers to make their way to Medina in small groups. He chose Madinah because people of Madinah took an oath to Muhammad to defend him as their own kin.
- When city authorities learned that the Muslims had begun an exodus, they plotted to have the prophet killed.
- Prophet Muhammad slipped away unnoticed with a chief disciple and made his way to Medina, using unfrequented paths.
- Imam Ali's sincere and significant self-sacrifice and slept in prophet Mohammad bed.
- Mohammad, the Prophet of Islam goes to cave of Thaur (Thawr cave).
- On the way to Yathrib, cave of Thaur (Thawr cave) and Quba.
- Yathrib eagerly awaiting Mohammad, the Prophet of Islam.
- Laying the Foundation for an Islamic Fraternity in Medinah

Sub-Title: Summary:

Hijra is a journey of Muhammad from Makkah to Madinah in 622 to escape persecution and found a community of believers in Madinah. After Quraish leaders decidied and planned to kill Prophet Mohammed he started his long journey with his friend Abubaker form Makkah, Thur Cave, Quba until they arrived Madinah. The following lesson will introduce you to his journey.

C.2.3 Hijra of Prophet Muhammad Pre-Assessment Quiz

The nature of the pre-assessment quiz is to take learners to focus on some important events and people who have an important part of Prophet Mohammad's Hijra.

1. The one who slept in the prophet's bed in the night of Hijra when Quraish conspired to murder him was...
   a. Ali (Correct)
   b. Umar
   c. Abubakr
2. ...was the one who tried to kill the prophet on his way of Hijra
   a. Suraqah Ibn Malik (Correct)
   b. Abu Jahl
   c. Abu Lahab
3. How long did Prophet Mohamad journey form Makkah to Quba in Madinah?
   a. 12 (Correct)
   b. 26
   c. 36
4. How long did Prophet Mohamad stay in Quba before going in Madinah?
   a. 12

   b. 14 (Correct)

   c. 36

5. Who did accompany Prophet Mohammad on his way to Madinah?

   a. Umar.

   b. Abubakr (Correct).

   c. Ali

6. How long did Prophet Mohammad stayed in cave of Thaur

   a. 2 days

   b. 3 days (Correct).

   c. 5 days

7. What is the correct route for Prophet Mohammad Hijra?

   a. Makkah- cave of Thaur- Madinah - Quba.

   b. Quba -Makkah- cave of Thaur - Madinah.

   c. Makkah- cave of Thaur- Quba - Madinah (Correct).

C.2.4 Hijra of Prophet Muhammad Lesson

Title: Hijra of Prophet Muhammad

Sub-Title) Introduction

After persecution of the prophet and his Muslims increased., he encouraged his Meccan followers to make their way to Medina in small groups. When city authorities learned that the Muslims had begun an exodus, they plotted to have the prophet killed. Under this threat, Muhammad slipped away unnoticed with a chief disciple and made his way to Medina, using unfrequented paths. When Prophet Muhammad undertook the physical migration from Makka to Madinah, it was a journey from danger to safety, from oppression to freedom. Prophet Muhammad could have remained in Makka, he could have sought an accommodation with the Quraish tribe, who offered him many incentives to give up his mission. They felt threatened by his teaching that spoke of One unseen God, rights for women and slaves, and justice for the poor. They even tried to bribe him by offering him wealth and status in the tribe. He could have enjoyed a peaceful and comfortable retirement in his mother city. Prophet Muhammad had left behind his tribe, his home, his family, his city, his financial and emotional security for something much more important. He had to build the foundations of an Ummah, the worldwide family of Believers. For the first time in the history of mankind, a nation was born, that had nothing to do with race, or language, or tribe or skin color. This was the first ideological community: the first nation to be defined by a simple belief that there is no god but Allah, and that Muhammad is the messenger of Allah.

1. **Imam Ali's sincere and significant self-sacrifice**

   When Mohammad, the Prophet of Islam was divinely commanded to migrate to Yathrib, he called Imam Ali, disclosed his secret to him, gave him the people's trusts to be returned to their owners and then said, 'I have to migrate, but you must lie in my bed'. Even though Imam Ali ,knew that the enemies of Islam wanted to murder Mohammad, the Prophet of Islam and that if he slept in Prophet Muhammad's bed his life would be at stake, he did not hesitate to risk his life to protect Mohammad, the Prophet of Islam and Islam. Imam Ali was very happy to do this for Mohammad, the Prophet of Islam. In fact Imam Ali said that the best sleep that he had ever had was on that night (Shab-E-Hijrat). Imam Ali's self-sacrifice was so sincere and significant that Allah praised it in Noble Qur'an as: "And among men is he who sells his Nafs (self) in exchange for the pleasure of Allah." (2:207)

2. **Mohammad, the Prophet of Islam goes to cave of Thaur (Thawr cave)**.

   At midnight the enemies of Islam surrounded the house of Mohammad, the

Prophet of Islam to carry out their satanic plot. When half the night was over, Mohammad, the Prophet of Islam left his house to begin his journey. As he came out of the house he threw some sand towards the men who were waiting to kill him and recited the following verse of Noble Qur'an: "And We have made before them a barrier and a barrier behind them, then We have covered them over so that they do not see." (36:9) While Imam Ali lay on his bed, Mohammad, the Prophet of Islam began his journey out of Makkah. Before he had left the city, he met Abu Bakr on the way and took him along with him. Mohammad, the Prophet of Islam knew that the Quraish would waste no time in pursuing him once they learnt of his departure, so he took refuge in the cave of Thaur (Thawr cave), which was to the south of Makkah on the way to Yathrib. The infidels rushed towards Mohammad, the Prophet of Islam's bed with drawn swords in their hands, but to their surprise, they found Imam Ali in his place. Upset and enraged, they asked, where has Muhammad gone? Imam Ali answered, 'Had you assigned me to watch him? Well, you intended to expel him and he has left the city. Realizing that all their plots were frustrated, the idol worshipping Quraish took serious measures but all in vain.

3. **On the way to Yathrib, cave of Thaur (Thawr cave) and Quba**.

   When the Quraish found out that Mohammad, the Prophet of Islam had left Makkah, they sent men to block all routes leading to Yathrib. They also hired some men who could trace the location of travelers by their footprints. It was declared that whoever gave correct information about the hiding place of Mohammad, the Prophet of Islam would be rewarded with 100 camels. One of the best trackers of the Quraish, a man named Abu Karz, traced the footprints of Mohammad, the Prophet of Islam to the cave of Thaur. However, when some men came near the mouth of the cave, they saw that its entrance was blocked by a spider's web and some wild pigeons had laid eggs in a nest at the entrance. The men knew that the spider and pigeons would not have made their homes there if there had been anyone in the cave. Also, if the web had been there from before, it would have been damaged if someone had entered the cave. They therefore returned without looking inside. By this miracle Allah, protected His beloved messenger. Mohammad, the Prophet of Islam remained in the cave for three days and nights. On one of these nights Imam Ali came to visit him. Mohammad, the Prophet of Islam told him to arrange for camels for Abubakr and himself. He also directed him to announce in Makkah the following day that if anybody had left something in trust with Mohammad, the Prophet of Islam, or had loaned him anything, he should claim it from Imam Ali. He further instructed Imam Ali to make arrangements for the Fawaatim (The three Fatimas - Fatima al-Zahra, Fatima bint Asad and Fatima bint Zubayr), as well as any other members of Bani Hashim who wished to leave Makkah. Imam Ali was to escort these people personally to Yathrib. After staying in the Thawr cave for three days, Mohammad, the Prophet of Islam proceeded towards Yathrib. One of the Makkans, Saraqa ibn Malik, attempted to pursue him, but his horse's hoof sank into the ground three times and threw him down, so he repented and returned to Makkah. On the 12th of Rabi al-Awwal, Mohammad, the Prophet of Islam reached a place called Quba, where he stayed for a few days. Abu Bakr insistently asked Mohammad, the Prophet of Islam to begin traveling towards Yathrib, but Mohammad, the Prophet of Islam refused to go without Imam Ali. He said to Abu Bakr, Ali has endangered his own life to save mine. He is my cousin, my brother, and the dearest among the family to me. I will not leave here until he joins me. After fulfilling the mission assigned to him and having arranged for the safe departure of Prophet Muhammad's family members to Yathrib, Imam Ali hastened forward on foot to Yathrib, traveling only in the night and hiding himself in the day, lest he should fall into the hands of the Quraish. He reached Quba three days after the arrival of Mohammad. The converts at Quba desired Mohammad, the Prophet of Islam to lay the foundation stone of a mosque for them. Mohammad, the Prophet of Islam marked the site and fixing the position of

Qibla, he laid the foundation stone for the first mosque of Islam. Mohammad, the Prophet of Islam used to pray Qasr while waiting for Imam Ali. It is mentioned in the Noble Qur'an as the mosque founded on piety and devoutness. Then Mohammad, the Prophet of Islam left Quba on the 16th of Rabi al-Awwal towards Yathrib.

4. **Yathrib eagerly awaiting Mohammad, the Prophet of Islam**.

   Yathrib had taken on an extraordinary air and intense excitement and eagerness had overtaken the whole city. In every alley and neighborhood people impatiently awaited Mohammad, the Prophet of Islam He entered Yathrib on Friday. People were overjoyed and could not stop looking at the resplendent countenance of Mohammad, the Prophet of Islam. Each tribe, which he passed through, desired the honor of his presence and requested him to take up his abode with them. Mohammad, the Prophet of Islam, refusing all these offers, said that the camel, which he rode on, was inspired and would take him to the proper quarter. The camel proceeded on to the eastern sector and knelt down in the open courtyard of the Banu Najjar, near the house of Khalid ibn Zayd, known in history as Abu Ayyub al-Ansari, the then head of the Banu Najjar family. He was delighted to be fortunate to have the honour of Prophet Muhammad's presence. Mohammad, the Prophet of Islam took up his temporary residence in the house of Abu Ayyub al-Ansari for seven months, until the Masjid-E-Nabawi, with proper quarters for himself, was built in the courtyard where the camel had stopped. Mohammad, the Prophet of Islam settled in Yathrib and there laid the foundations of Islam and a magnificent culture based on justice and faith. After the blissful entrance of Mohammad, the Prophet of Islam into Yathrib, its name was changed into Medinat ul-Nabi, meaning 'the City of Mohammad, the Prophet of Islam'. That year, the year Mohammad, the Prophet of Islam migrated (Hijra/Hijrat 622CE) at the age of 53 years from Makkah to Yathrib (Madinah), was recognized as the origin of history (beginning of the Islamic Era or the Hijra calendar), due to this significant historical event, the triumph of righteousness and justice. **Note:** When we say 1430 Hijra, it means that 1430 years have passed since the time of Prophet Muhammad's flight (Hijra) from Makkah to Medinah.

5. **Laying the Foundation for an Islamic Fraternity in Medinah**.

   After having settled in Medinah and after building a mosque (Masjid-E-Nabawi) that was indeed the military and constitutional base of the Muslims, Mohammad, the Prophet of Islam took an excellent initiative. He laid the foundation of Islamic brotherhood, individually between the people of Medina (known as Ansar/Helpers) and the people of Makkah (known as Muhajir/Emigrants), so that great unity and sincerity would be engendered in Muslim society and so that the emigrant Muslims would know that, though they had lost a number of their friends and relatives and had been forced to leave their homes, in return, they had gained brothers who were much more loyal and sympathetic from every point of view. Islamic brotherhood firmly holds all Muslims responsible toward each other and establishes an all-embracing responsibility so that Muslims cannot be heedless of each other's troubles and problems but every Muslim must, within his own abilities; endeavor to solve the problems of Muslims and to create possibilities for the advancement and promotion of Islam. Thus Islam began to flourish with its enormous luminosity.

Sub-Title: Chronology of the Hijra

- Day 1: Thursday 26 Safar AH 1, 9 September 622 CE. Left home in Mecca. Stayed three days in the Cave of Thawr near Mecca.
- Day 5: Monday 1 Rabi' I AH 1, 13 September 622 CE. Left the environs of Mecca. Traveled to the region of Yathrib.
- Day 12: Monday 8 Rabi' I AH 1, 20 September 622 CE. Arrived at Quba' near Medina.

- Day 16: Friday 12 Rabi' I AH 1, 24 September 622 CE. First visit to Medina for Friday prayers.
- Day 26: Monday 22 Rabi' I AH 1, 4 October 622 CE. Moved from Quba' to Medina.

Sub-Title: Summary

In the year 622, Makka city authorities learned that the Muslims had begun an exodus; they plotted to have the prophet killed. Under this threat, Muhammad slipped away unnoticed with a chief disciple and made his way to Yathrib. Later prophet Mohammad has changed its name to Medina, meaning city of the Prophet. This date was designated by later Muslims as the beginning of the Muslim calendar, year one of hegira (Arabic hijra, "immigration"). Only two years after Muhammad's arrival in Medina, the core community of Muslims started to expand.