Article

Assessment of Students in Higher Education—Information and Communication Technology Tools and Tips

Leila Goosen
https://orcid.org/0000-0003-4948-2699
University of South Africa
goosel@unisa.ac.za

Dalize van Heerden
https://orcid.org/0000-0002-7539-8995
University of South Africa
vheerme1@unisa.ac.za

Abstract

The aim of this research project relates to assessment approaches at the University of South Africa (UNISA). The Learning Management System is used to address students' first-year experience in an Information and Communication Technology Open Distance e-Learning (ODeL) context. In this study a socio-critical model was suggested as a framework for improving student success in ODeL at UNISA, and constructivist instructional strategies to e-learning were applied in a case study of a web development course. A mixed-method research approach was adopted, which involved the collection and analysis of quantitative and qualitative data. The main findings of the study provide insight into the uptake of and assessment results for self-assessments, multiple-choice, blog and project-based assessments. The implications of the results are that there is a clear need in higher education for the transformation of teaching that is in line with emerging technologies.

Keywords: student assessment; higher education; information and communication; technology (ICT)

Introduction

Context of the Study

Introduction to Programming (ICT1511), Introduction to Interactive Programming (ICT1512) and Introduction to Web Design (ICT1513) are three of the 10 first-year modules in the Diploma: Information Technology that are presented per semester. ICT1512 and ICT1513 are also included in different engineering diploma streams.

Progressio
https://upjournals.co.za/index.php/Progressio
Volume 40 | Number 1–2 | 2018 | #4704 | 23 pages

https://doi.org/10.25159/0256-8853/4704
ISSN 0256-8853 (Print)

Introduction to Programming (ICT1511) concentrates on teaching the principles of programming using a programming language. The outcomes of this module are to teach students how to use mathematical concepts and principles to analyse problems and provide logical solutions by writing sequential steps in the form of algorithms. Students are also taught how to design functions, procedures and sub procedures, and how to use arrays. Error handling and debugging techniques form a further part of the outcomes, as well as an introduction to object-oriented programming. Students are taught how to apply selection control structures and iterations. Important to note is that students are taught how to use a computer to investigate and solve scientific subject-related real-life problems. Lastly, students are taught to communicate in English, which is the language of business and academic reporting in the field of computing. On completion of this module, students should therefore be able to

- demonstrate a clear understanding of problems and their statements, as stated by computer users in various industries;
- demonstrate understanding of general computer programming concepts through the use of algorithms and operators;
- demonstrate a basic understanding of object-oriented concepts; and
- design a solution from problem statements by writing algorithms and programs in sequential steps.

On the other hand, Introduction to Interactive Programming (ICT1512) allows the students to implement the skills they acquired in ICT1511 using the JavaScript programming language. The outcomes of this module allow students to show that they understand problem statements provided by users in various industries. Students should be able to apply fundamental programming principles in the development of a working program. The students are also able to use web design tools to develop a specific solution to the satisfaction of the client and they are able to use JavaScript to develop a program. On completion of this module, students should be able to

- develop a working computer program, with the knowledge, skills and values needed to add interactive functionality to the program through object-oriented programming;
- use logical programming skills to develop an introductory program with JavaScript;
- extend their knowledge by adding interactivity to websites through object-oriented programming;

- design and develop programs using industry process systems and organisational information systems that conform to specific standards that are user friendly and robust, solution specific and to the satisfaction of the client; and

- think conceptually to recognise the design rules, techniques and components that are required to develop a solution.

Furthermore, Introduction to Web Design (ICT1513) provides students with the skills to design and develop a website according to the rules of the World Wide Web Consortium (W3C). The skills developed in this module are also applied in ICT1512, where interactivity is added to the site through JavaScript coding. The outcomes of the module allow students to demonstrate an understanding of the concepts of interface design in view of its importance for the success of a computer-based product, as well as a clear understanding of problem statements as stated by computer users in various industries. The module teaches students to identify and apply user interface design principles that allow them to design a user interface and that take visual, cognitive and physical considerations of humans, as well as environmental and social issues. On completion of the module, the students should be able to

- design, develop and maintain graphical user interfaces;

- critically evaluate a graphical user interface based on values that conform to internationally accepted, ethical and aesthetical standards and design values;

- increase the quality standards of a graphical user interface;

- analyse the client's needs and design and develop a graphical user interface to specific standards of being well-designed, portable, accessible, clearly presented, cost-effective, re-usable, reliable, timeous and robust; and

- analyse a given situation or scenario, for instance, to determine the client's needs, provide design solutions, and develop, evaluate and improve websites.

None of the modules require any specialised software. For ICT1512 and ICT1513, students need access to a text editor and a browser. This software is usually loaded with the operating system. If students are using a standard computer, the text editor would be Notepad and the browser would be Internet Explorer.

Students are not allowed to use software such as Dreamweaver to write their code, since this software adds a lot of codes they do not need, and it is set up to automatically write specified pieces of code. This defies the object of the students learning how to write code themselves. Both ICT1512 and ICT1513 require students to test their code in different browsers—students need to download and install these browsers, so they can see how different browsers render the code differently. The links to locate the browsers are included in their online material.

As part of their formative assessment, students are required to submit screen captures of the output of their code. Students may download a program to assist them in making screen captures for their assessment, or they may use the Print Screen facility of the computer. They also need to purchase a prescribed book for each of the modules. These books are selected specifically for their clear theoretical explanations of the concepts that are then followed by a practical, step-by-step implementation of the concepts, including explanations. The prescribed book for ICT1511 is Pretorius and Erasmus (2012); ICT1512 uses Vodnik and Gosselin (2015), and ICT1513 uses Sklar (2015).

Problem Statement

The process of learning how to program, regardless of the programming language, requires both theoretical and practical knowledge (Matthews, Hin, and Choo 2015). Programming students must understand the theory behind each concept in order to apply the concept practically. Due to the practical nature of programming students often tend to know how to implement practical concepts in a given context, without understanding why the concept works. Thus, when the concept is changed, the student can no longer apply it (Brito and De Sá-Soares 2014). Students should therefore be assessed on both their theoretical and practical knowledge of the programming language.

The formal multiple-choice assessment system at the University of South Africa (UNISA) has a number of weaknesses, of which the two most crucial are the turnaround time for marking formative assessments and the quality of feedback provided. In the official multiple-choice system, formative assessments are marked on a given date after the formative assessment due date, which means students must wait for their feedback. Moreover, this feedback is given only in the format of a correct answer. Although the system allows students to resubmit their formative assessment, they can only re-do the same set of questions. Very little, if any, learning takes place through assessment, because of the limitations of the system.

One of the outcomes of the programming modules is that students should be able to develop a specific solution that satisfies the client. This implies that the student must be able communicate with a client and determine their needs and requirements. The institutional database shows that only 15 per cent of registered students use English as their first language of communication. Information and Communication Technology (ICT) undergraduate studies normally concentrate on teaching students about the various technologies and programming languages, but they provide little to no skills with regard to communicating in the discipline-specific language. ICT students should be able to write reports, technical materials, user manuals and in general communicate their ideas using the disciplinary language (He et al. 2015). Improving a student’s ability to communicate in the disciplinary language, especially if it is not their first language of communication, is important to the student's overall success.

The aim of this research project focused mainly on assessment approaches at UNISA, and specifically on how the Learning Management System (LMS) is used to address students' first-year experience in an ICT Open Distance e-Learning (ODeL) context. In order to be relevant to both the theme at issue in this journal, and to the fields of ODeL and higher education (beyond the case presented), this study therefore intended to achieve objectives related to showing how

- students' theoretical, as well as practical, knowledge of the programming language is assessed;

- learning takes place through assessment, despite the limitations of the system; and

- students' ability to communicate using discipline-specific language is assessed in an ODeL context.

The research questions, which framed this project, therefore included the following:

- How can students' theoretical, as well as practical, knowledge of the programming language be assessed?

- How can learning take place through assessment, despite the limitations of the system?

- How can students' ability to communicate using discipline-specific language be assessed in an ODeL context?

Literature Review

The conceptual framework of this study is influenced by the work of Subotzky and Prinsloo (2011) who applied a socio-critical model as a framework to improve student success in ODeL at UNISA. The work of Wang (2014), who applied constructivist instructional strategies to e-learning in a case study of a web development course, also became instrumental. In this literature review, the following studies were noteworthy:

- Halabi, Essop, Carmichael, and Steyn (2014), who found preliminary evidence of a relationship between the use of e-learning and academic performance in the first-year experience of a South African accounting course.

- Govender (2010), who examined students' attitudes towards the use of an LMS in a face-to-face instruction mode.

- Greenland and Moore (2014), who used a case study to explore patterns of student enrolment and attrition in an Australian ODeL context.

- Emelyanova and Voronina (2014), who investigated students' and lecturers' perceptions of introducing an LMS at a Russian university.

However, the notion of the use of technology for the assessment of students has not yet been explored sufficiently.

Formative Assessment

Different types of assessments are applied differently in a higher education context. This section discusses the importance of exploring the role of formative and summative assessment in higher education through the following types of information and communication technologies: self-assessments, multiple-choice tests and a blog, as well as project-based assessments.

Self-Assessments

Short online quizzes consisting of multiple-choice, true/false and short answer questions are used to assist students in determining their understanding of basic concepts (Bälter, Enström, and Klingenberg 2013). Bennett et al. (2017, 675) argue that the choice of online quizzes as a form of assessment in an online environment is driven by time and money constraints, as well as by the "ease of setting up and administering" these.

In an attempt to get students to study the theoretical concepts, a database of multiple-choice, fill-in-the-missing-word and true/false questions was created on the *my*UNISA Learning Management System. The Samigo tool was used for each of the chapters in the books prescribed for the three modules. The self-assessments were set up to create 15 new questions from the database each time a student attempts a self-assessment for a particular chapter. The students can therefore do the self-assessment for each chapter as many times as they prefer and feedback for each self-assessment is provided immediately upon submission. The feedback provided in the self-assessments for incorrect answers directs the student to the page number in the prescribed book where the correct answer can be located. This forces the student to find the correct answer by having to re-read the relevant section of work, instead of merely giving the correct answer as feedback. Learning through assessment is ensured in this way.

The self-assessments were created to provide students with the opportunity to prepare for their formative and summative assessments. Seeing that the same question pool is used in both cases, the formative and summative assessments are integrated. A further purpose of the self-assessments is to provide lecturers and tutors the opportunity to identify students who are not performing as well as they should (Antle and Wise 2013). The statistical analysis provided by self-assessment tools is also used to identify specific questions with which students may experience problems.

Multiple-Choice

The use of multiple-choice questions as a formative assessment tool is successfully implemented because it allows for immediate feedback. As with the other self-assessments, the feedback is a reference to the page in the prescribed book where the student needs to find the correct answer, rather than passively receiving the correct answer. Students are given two opportunities to complete the multiple-choice formative assessments, with the second opportunity containing a totally new set of questions from the question pool. By adopting this strategy, the knowledge content that is tested is expanded significantly. The highest score between the two attempts is captured as the student's result for the formative assessment.

Scoring the multiple-choice formative assessments by means of one of the self-assessment tools provides a level of validity and reliability to the formative assessments that cannot be obtained through the formal UNISA multiple-choice assessment system. This is because each student's formative assessment is individually created from the database of questions and students can therefore not copy answers from each other. The immediate feedback that students receive greatly reduces turnaround time in the ODeL environment, and consequently more assessments can be done—which helps the student to keep to the suggested study schedule.

Blog

Blogging is rarely considered as a means of assessment in programming modules. However, blogs as an assessment tool address a number of the topics identified by Blanco et al. (2011), for instance, supporting students in their self-reflection, providing instructional guidance, encouraging collaborative learning and strengthening students' self-efficacy. Ramasamy, Valloo, and Nadan (2010) used the number of threads and replies to gauge a student's initiative in gaining new knowledge, while Safran (2008) measured learning performance through blog activity, practical experience and theoretical knowledge. Both studies recognised the effectiveness of blogging in improving student understanding. Van Heerden and Van der Merwe (2014, 185) compared the blog results with the final results of students and found that "the implementation of knowledge blogging in an ODeL context is particularly well suited to introductory programming modules when such blogging demands reflective activities and continued engagement with the module work." They specifically suggested that "knowledge blogging [is] a constructive learning tool in a programming environment since it promotes metacognition and differentiated instruction by nurturing multiple learning skills."

Using the blog as formative assessment allows students the opportunity to share, in their own words, what they have studied. It is an ongoing assessment, which starts in the first week of their studies and continues until the end of the semester. For the purposes of the assessment, students need to write a blog about the work they studied in certain chapters, reflecting on what they have learnt. Each reflection must be a minimum of 600 words (200 words per chapter), which is approximately the same length as a typed A4 page. Students are required to submit a single blog for every three chapters they covered.

Students are also required to comment on a minimum of three other students' blogs—this provides them with the opportunity to experience the content from someone else's point of view, and to engage with other students.

Lecturers and e-tutors use the blogs to gauge the students' understanding of particular pieces of work and they comment accordingly on the blogs. A rubric that looks inter alia at students' insight, language usage, layout, and comments made on other students' blogs is used to mark the blogs. Because no two students can have similar reflections on what they studied, these formative assessments have a high validity and reliability, which is extremely important in the ODeL environment.

Project-Based Assessment

Project-based learning is ideally suited to assessment for learning in programming modules, since students actively participate, learn by doing, implement their learning and solve real or simulated problems (Doppelt 2003). This type of assessment is more than a mere evaluation of the students' knowledge: it allows them to show in practical ways that they have mastered the theory and are able to apply it in a real-world scenario (Rand 1999). Project-based teaching, learning and assessment have been and are currently being used by numerous residential universities as the preferred method of teaching and assessing programming modules (Todorova et al. 2010; Vega, Jiménez, and Villalobos 2013). Evidence from research indicates that there is an improvement in the performance of students taking programming modules when project-based learning and assessment tools are implemented (Bubaš, Ćorić, and Orehovački 2012; Wilson and Ferreira 2011).

Combining the assessments of the three modules (i.e. Introduction to Programming [ICT1511], Introduction to Interactive Programming [ICT1512] and Introduction to Web Design [ICT1513]) shows the students that their modules are not "islands" standing on their own, but interlinked components that build on each other. The students are shown how what they learn in one module should and can be applied in another module. The combination of the three modules' formative assessments also allows the lecturer to align the three major components of instruction: learning objectives, instructional activities and assessments.

For the module Introduction to Web Design (ICT1513), the students must contact a small business in their community and obtain its permission to design a website for the business. The site should consist of a minimum of four pages, for example a home page, products page, order form, and contact page. The practical formative assessment for the module Introduction to Programming (ICT1511) requires students to design the logic for the interactivity they must add to the website. This logic design includes a feature that uses at least one function to perform a mathematical calculation based on user input by using if, if/else, else if or switch statements, exception handling of code and validation of user input. Using the logic design of the Introduction to Programming formative assessment, the Introduction to Interactive Programming students are required to add functionality and interactivity to the site designed in Introduction to Web Design through JavaScript coding. The business owners are required to sign a letter of consent, which must accompany the formative assessment, and the students must sign a declaration that the work in the formative assessment is their own, providing a certain level of validity and reliability for this formative assessment.

Students are also exposed to real industry experience through their contact with the businesses—they need to interact with the user during various stages of the development of the site. The biggest advantage of this formative assessment is that it allows the students to build an evidence-based CV/application. Statements saying what you can do are limited, unless backed up with good examples/evidence when applying for a position or a promotion.

The lecturers and external markers use a rubric to mark these formative assessments and provide feedback to students. To ensure that students know exactly what it is they will be marked on, a column is provided in the rubric where students should give themselves a mark. The rubric therefore serves as a checklist for the student to ensure all work required has been completed.

To further ensure the validity and reliability of the practical formative assessment, the requirement is changed every so often from designing for a business, to designing for a school. The basic requirements, however, stay the same; students just need to apply them to a different environment.

Summative Assessment

Students write one examination paper for each of the modules, covering the entire syllabus, at the end of the semester. All the work in the syllabus is covered in the examination, not only the work focused on in the assessments. In order to integrate the formative and summative assessments, the first section of the examination paper comes from the same pool of questions that is available in the self-assessments; therefore, the more students practise these questions, the better they should perform.

In another attempt to create cohesion between the three modules, the layout of the examination papers is quite similar. The papers have a fill-in design, where space is provided on the paper for students to complete specific questions in the required format. This saves them the time for having to draw tables, etc., and reduces confusion on how questions are expected to be answered. The first part of each paper consists of 15 multiple-choice questions, 15 true/false questions and 15 fill-in-the-missing-word questions. These are followed by additional questions to test whether the student grasped the theoretical concepts covered in the module.

The examination papers of all three modules conclude with an “apply your knowledge” question. ICT1511 students are provided with a case study taken from their prescribed book and requested to complete the planning and Input-Processing-Output tables provided for them, as well as write the algorithm to solve the problem. Students are provided with a complete JavaScript file, based on an example from their prescribed book, which contains syntax errors to be identified and fixed. For ICT1513, we again provide the students with a case study from their prescribed book and request them to write the Cascading Style Sheet according to the instructions provided.

Research Methodology

Regarding educational research methodologies, a mixed-method approach was used as a research design, involving a collection process for both quantitative and qualitative data (Babatunde and Low 2015; Johnson 2014). In terms of discussing the sample and/or sampling technique, it should be noted that the entire population of students per module is applicable in each case. Specialised queries, constructed by the second author according to the requirements of this study, were used as instruments for collection from the institutional database. The data analysis processes applied consisted of percentage comparisons across the three applicable modules, as well as between-assignment comparisons across the three modules, for various formative and summative assessment aspects as indicated. Content validation “refers to the extent to which the instrument covers the complete content of the particular construct that it is set out to measure” (Pietersen and Maree 2007, 217). To ensure the content validity of the data collection, the second author presented a provisional version thereof to the first author, who is more experienced in the field, for her comments, before finalisation.

Results: Trends in Formative Assessment

Based on the process of data analysis, this section of the study discusses the emerging trends in formative and summative assessment.

Uptake of Self-Assessments

The submission of self-assessments is significantly lower than recommended across all modules. This can be attributed to students’ perception that they do not have to complete self-assessments, as these do not contribute towards their results. Another factor that influences students’ uptake of self-assessments is that there is not enough time during the shortened semesters to practise self-assessments.

The same database of questions is used in the self-assessments, the multiple-choice formative assessment, as well as the examination. It is recommended that students complete the self-assessments for each chapter a minimum of five times before they attempt the multiple-choice formative assessments. It is further recommended that students complete the examination preparation self-assessment at least five times per day, every day, before they write their examination, to assist them with their preparation. On average, a student should have a minimum of at least 80 self-assessment submissions.

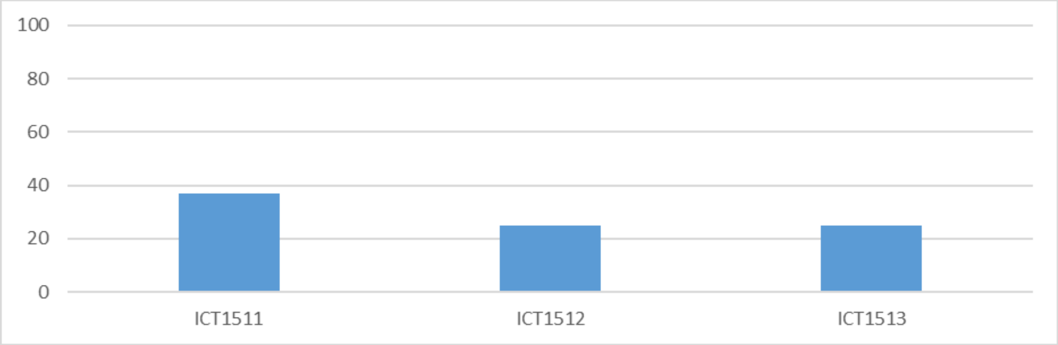
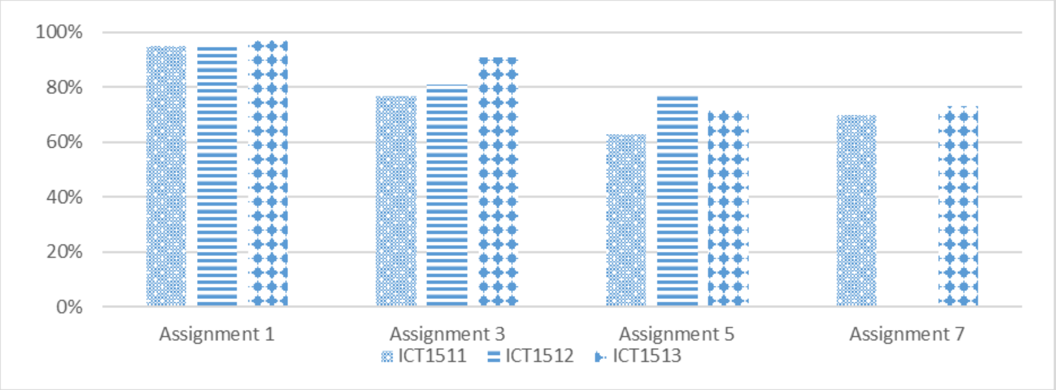


Figure 1: Submission rate per multiple-choice assessment

Submission rates per multiple-choice assessment are indicated in Figure 1.

Uptake of the Multiple-Choice Formative Assessments

Figure 2: Uptake of multiple-choice assessment



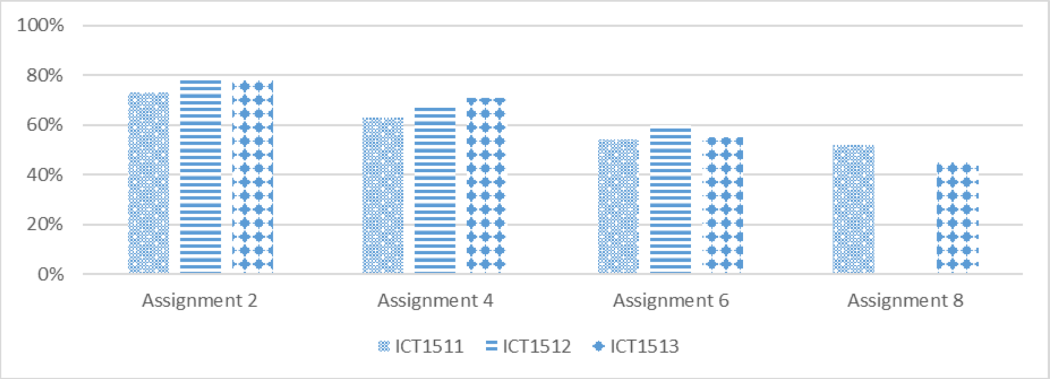
Three of the four multiple-choice assessments contribute six per cent towards the students' year mark, and the last contributes seven per cent for ICT1511 and ICT1513. For ICT1512, two of the three multiple-choice assessments contribute eight per cent towards a student's year mark and the last contributes nine per cent. Students need to complete the multiple-choice assessments using the self-assessment tool. These assessments consist of 25 multiple-choice questions, covering three chapters that students had to study within a given time frame. Students are allowed two submissions per formative assessment, and the formative assessment with the best result is captured as part of the official formative results. Submission rates per multiple-choice assessment are indicated in Figure 2.

Submission rates for the multiple-choice assessments show a steady decline of assessments as the semester progresses for all three modules.

The contribution of the compulsory blog formative assessments to the year mark is similar to that of the multiple-choice assessment, with the first three of the four multiple-choice assessments contributing six per cent towards the students' year mark, and the last contributing seven per cent for ICT1511 and ICT1513. Similarly, for ICT1512, the first two contribute eight per cent and the last contributes nine per cent. The students submit their reflections covering three chapters per blog post in the blog tool on *my*UNISA. They are also required to comment on three other students' blog posts. Figure 3 shows the submission rate of the blog formative assessments.

Uptake of the Blog Assessment

Figure 3: Uptake of blog assessment



Similar to the submission rates of the multiple-choice assessment, the submission rates for the blog assessments show a steady decline of assessments as the semester progresses for all three modules. Submission rates of the blog assessments are also lower than those of the multiple-choice assessments.

Uptake of Project-Based Assessment

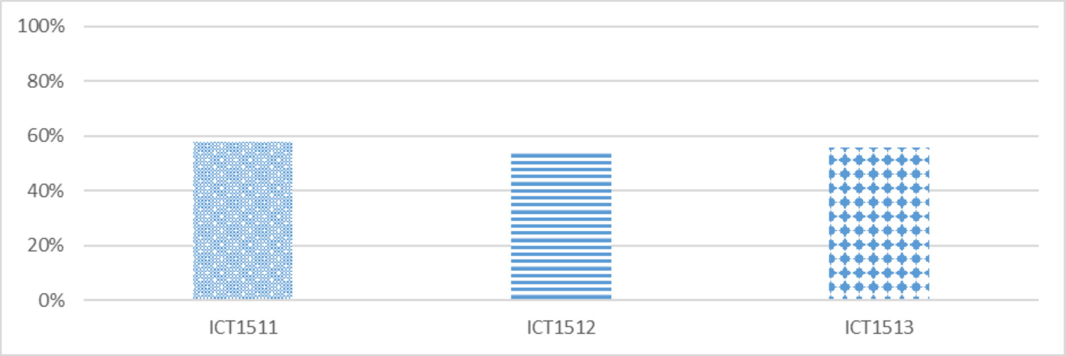


Figure 4: Uptake of project-based assessment

The project-based assessment contributes 50 per cent towards the students' year mark. In this assessment, students need to demonstrate their ability to practically implement the theoretical concepts they have studied, based on a specific real-world scenario. Figure 4 shows the submission rates of the project-based formative assessments.

Uptake of the project-based assessment is consistently low, below 60 per cent, for all three modules. Non-submission of the project-based assessment has a negative effect on students' year marks, which in turn has a negative effect on the throughput rate.

The decline in the submission of assessments is attributed to a number of factors. The first of these is that in most other modules, students are required to submit only two or three assessments throughout the semester, whereas the modules ICT1511, ICT1512 and ICT1513 respectively have seven and nine assessments. The university's requirement that students need only submit a single assessment to obtain examination admission is another factor that contributes to the decline in submissions. This requirement results in students seeing assessments as a way into the examination and not as part of their learning. The assessment plan for these modules is designed specifically to accommodate a 15-week semester system and to ensure students keep to the study schedule and complete all the work required for the module. Due to the extension of registration periods, which results in extended assessment due dates, students are not able to manage their time efficiently and are thus not able to complete the assessments required in the time available.

Multiple-Choice Assessment Results

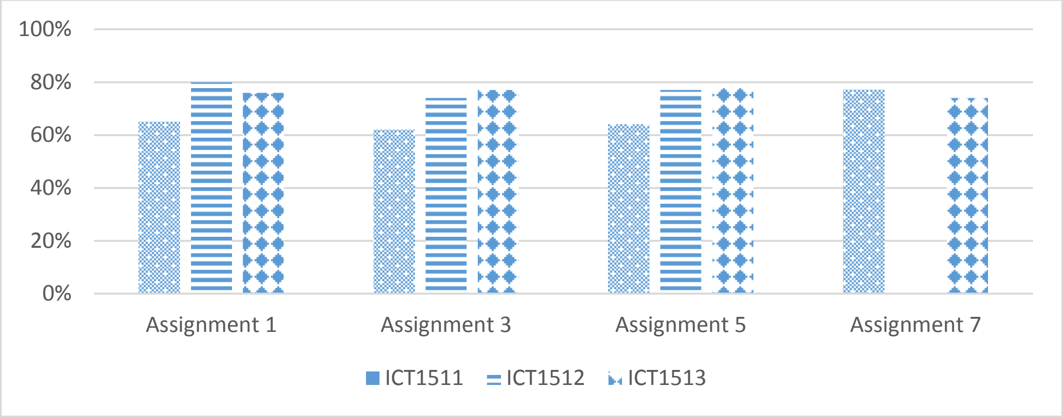


Figure 5: Multiple-choice assessment results

The average result for multiple-choice assessments remains consistent throughout the semester for the modules ICT1512 and ICT1513, as these modules are theoretically based and answers can be obtained from the prescribed book. Results for assessment 1, 3 and 5 for ICT1511 are somewhat lower, although still consistent throughout, as some of the questions require insight and application. Results for assessment 7 are somewhat higher and in line with that of ICT1513, as they are purely theoretical. Figure 5 shows the results of the multiple-choice formative assessments.

Blog Assessment Results

The blog assessment results for ICT1512 and ICT1513 show a slight increase throughout the semester. This can be attributed to the students becoming more familiar with the requirements of the blog assessment. The fact that ICT1511 does not show the same trend will have to be investigated. Figure 6 shows the results of the blog formative assessments.

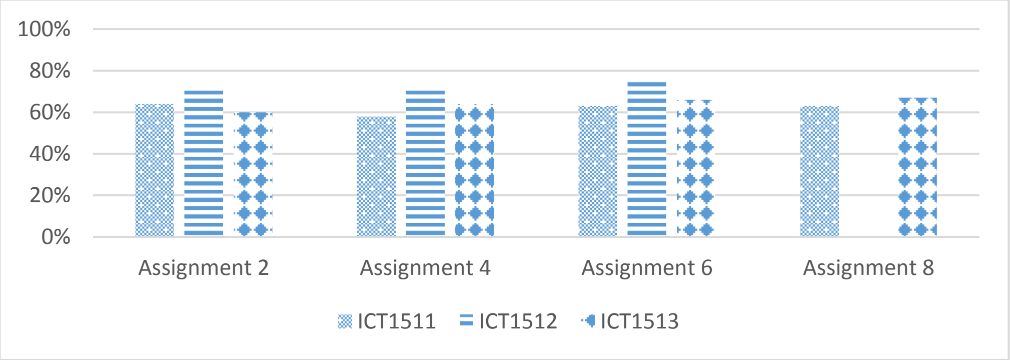


Figure 6: Blog assessment results

Project-Based Assessment Results

The project-based assessment results are very low, especially in comparison to the multiple-choice and blog assessment results. Two factors influence the results of this assessment: first, students do not spend adequate time in the shortened semester to complete the requirements of the assessment. Second, students are unable to apply the theoretical concepts they have studied and the practical exercises they are required to complete in a context that differs from their prescribed material. The latter factor will be investigated and addressed in future assessments, as it appears students are not completing the practical exercises.

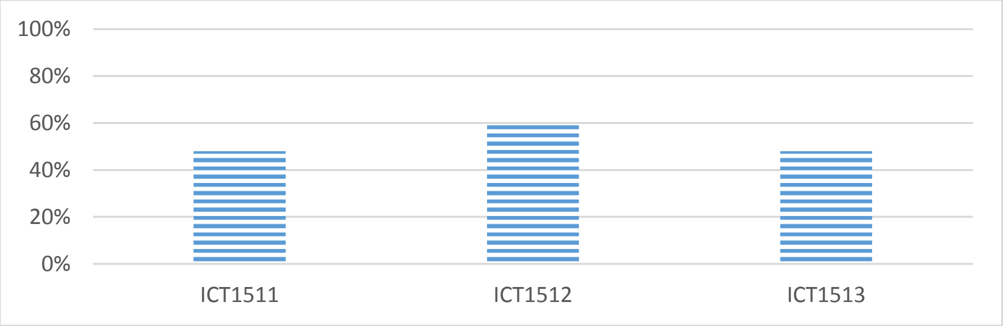


Figure 7: Project-based assessment results

The slightly higher results for the module ICT1512 project-based assessment, which includes application of certain aspects of the ICT1511 and ICT1513 modules, show that combining elements from different modules in formative assessments improves results. Figure 7 shows the results of the project-based formative assessments.

The low results of the project-based assessment have a negative effect on students' year marks, which in turn negatively affect the throughput rate. This, however, ensures that only those students who are considered competent in programming move on to the next level.

Results: Trends in Summative Assessment

True/False, Multiple-Choice and Completion Questions

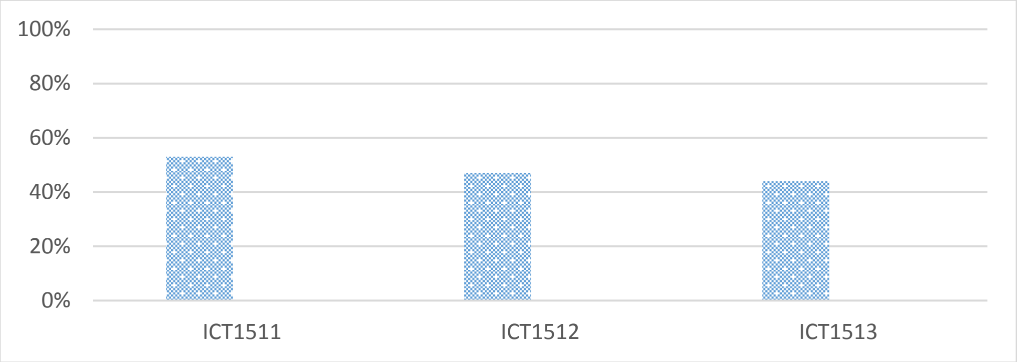


Figure 8: Results of the first three questions for the three modules

The first three questions in the examination for ICT1511, ICT1512 and ICT1513 are all based on the self-assessment questions available to the students, thus combining both formative and summative assessment design. Students are requested to complete the examination preparation self-assessment as many times as possible, in order to assist them in preparing for the examination. As noted in Figure 1, uptake of the self-assessment tool is below the expected level. The results of the first three questions for the three modules reflect students' poor participation, as can be seen in Figure 8.

Apply Your Knowledge

All three modules contain a question in which students are required to show that they are able to apply what they have studied in a practical way. In each module, the students are presented with a case that requires them to perform specific tasks.

In ICT1511, the student needs to complete the planning and Input-Processing-Output tables provided, write the algorithm to solve the problem and use the provided test data to show the output. This is similar to the requirements of their project-based assessment. As ICT1512 is a programming module, one cannot expect students to write code by hand; students are thus provided with existing code that contains syntax errors to be identified and corrected. The syntax errors are quite similar to those a student would make throughout the semester when entering code, either in the exercises or in their project-based assessment. For ICT1513, students are provided with a before and after image of a website and a list of instructions to create the Cascading Style Sheet to obtain the after image.

Again, the requirements are similar to the exercises in the prescribed book, as well as the requirements of their project-based assessment.

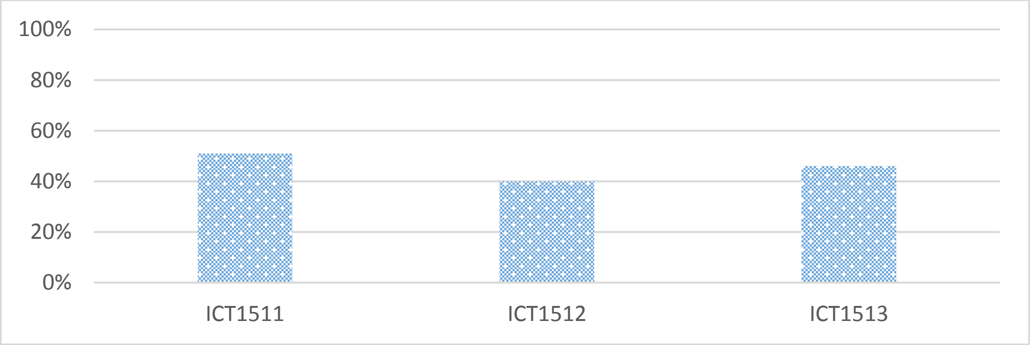


Figure 9: Results of examination question as a specific example from the prescribed book for the three modules

In each case, the examination question is a specific example from the prescribed book—it is thus based on a known context, rather than a new context. Despite this fact, the results of this question in all three modules are below expectations, as shown in Figure 9.

Final Summative Assessment Results for Each of the Three Modules

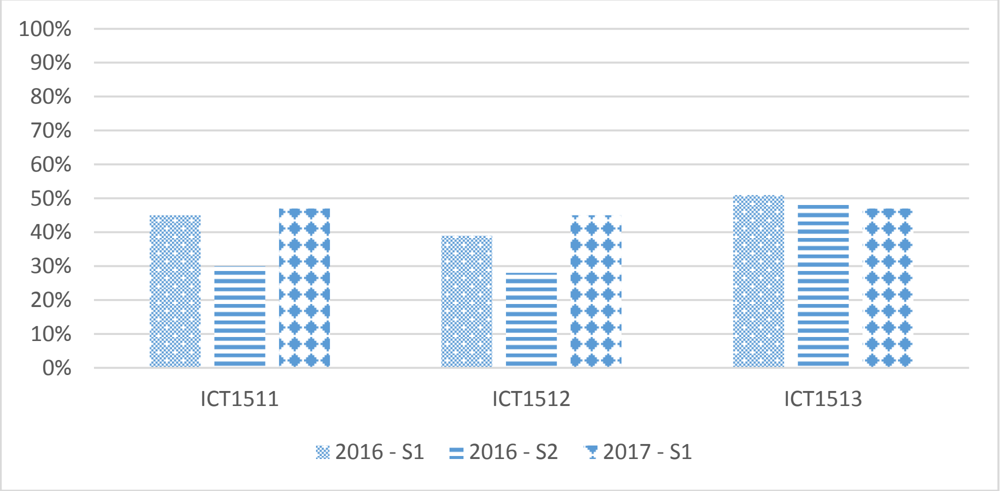


Figure 10: Final results for the three modules over three semesters

Because of the high rate of validity and reliability of the formative assessments, the latter contribute 49 per cent towards the students' final results. Students who complete all their assessments and receive good results are thus more likely to successfully complete the module. The combination of both the formative and summative assessment design should also have a positive impact on the final results for the modules. In line with what was seen with the uptake and results of the formative assessments, however, the final results for the modules are still below the 50 per cent pass rate, which is being aimed for (see Figure 10 for the final results of the three modules over the previous three semesters).

Discussion of Results: Assessment Support

Vodcasts

The homepage of each module on *my*UNISA contains a vodcast, or video podcast, that introduces students to the module. In this introduction, the importance of completing the formative assessments and their contribution to the final results are highlighted and stressed. Students are also shown where to find detailed information regarding the requirements for each assessment in the learning units. This is in line with suggestions by Van Heerden and Goosen (2012) that vodcasts should be used to teach programming in an ODeL environment.

Additional Resources

The additional resources contain a folder for the project-based assessment as well as the examination. In the project-based assessment folder, students are provided with a Microsoft Word template of the assessment that they can complete, thus ensuring all required information is included and formatted correctly. Students are also provided with sections of a previous semester assessment that achieved a high score. The rubric used to mark the project-based assessment is included in the template and students are requested to give themselves marks in the rubric. This enables them to see where they might be lacking information. Examples from the study by Goosen and Van Heerden (2015) are also provided to show how e-learning management system technologies such as additional resources technology can be used for teaching programming at a distance.

The examination folder contains a short answer document, with a database of questions and answers, for both ICT1512 and ICT1513, which is included in their examinations. The “apply your knowledge” questions from previous examination papers, including their answers, are also provided in this folder. Students are advised to make use of these documents (rather than previous examination papers) to prepare for their examinations, as questions are never repeated.

E-mail and SMS

As an intervention strategy to improve learning, students who do not participate in the self-assessments or who are under performing in the self-assessments are contacted and offered individual assistance via personalised e-mails and SMSs. The study by Goosen and Van Heerden (2016) specifically shows that email was one of the e-learning tools that students used to address challenges in the online and open distance education context, while Goosen and Van Heerden (2015) provide an example of an email sent to assist students.

Each formative assessment has a grace period of one week after the due date, during which assessments can still be submitted. During this week, the lecturer checks on the Monday, Wednesday and Friday for assessments that have been submitted. Students with outstanding assessments are contacted on these days via SMS and reminded that their assessment is still outstanding, when exactly the grace period ends, and the importance of submitting the assessment.

Once the assessment results have been finalised, students who did not submit or who performed poorly in the assessment are identified and contacted via e-mail. The e-mails are individualised for each student and provide the student with the contact details of his/her specific e-tutor. The students are encouraged to contact their e-tutors directly for assistance, so they may improve their performance in their next assessment.

At the end of the semester, each student receives an individualised e-mail indicating their formative assessment results, their year mark, as well the estimated examination mark they will need to achieve in order to either obtain a distinction or complete the module successfully. Students are then also provided with an opportunity to query any assessment results, to ensure everything is reflected correctly on the system.

Meetings

Approximately two weeks before the due date of the project-based assessment, online meetings are arranged with students to discuss any questions they may have before completion and submission of the assessment. Goosen and Van Heerden (2017) describe how such meeting educational technologies can be used to take learning programming beyond the horizon of distance limitations.

For each module, a weekday morning, weekday evening and weekend session is scheduled—thus three meetings at different times for each module. A short presentation is made, during which the requirements of the assessment are once again set out and students are given the opportunity to ask questions as the meeting progresses. The meetings are recorded and can be viewed by students who are not able to attend any of the sessions. The presentation is also made available in the project-based assessment folder under "Additional Resources."

The examination preparation meeting is arranged similarly to the project-based assessment meeting, with three sessions scheduled at different times for each of the three modules. During the session, students are informed that they will be examined on their entire prescribed book and that there are no sections that can be "spotted" for the examination. They are provided with the layout of the examination and informed how to study for each section of the examination: for example, they have to complete the examination self-assessment as many times as possible to prepare for the first three questions in the examination paper. The recorded meeting is again available to those who were not able to attend, and the presentation is made available in the examination folder under additional resources. Despite the effectiveness of the meeting tool, attendance of online meetings is extremely low, with an eight to 12 per cent attendance rate.

Conclusion

Even if the best-planned, supported and implemented assessment strategies are in place, but they are not supported by the institutional systems and students, they will not be effective in addressing high drop-out and failure rates. In conclusion, and in line with Bozalek, Ng'ambi, and Gachago (2013), the implications that the transformation of teaching through the introduction of emerging technologies has for higher education institutions should be considered. The significance of this article lies in presenting original work that contributes to debate by closing gaps regarding quality research, and by showing the interest, value and potential usefulness of the results of this ongoing research project and content for the intended audience.

References

Antle, A. N., and A. F. Wise. 2013. "Getting Down to Details: Using Theories of Cognition and Learning to Inform Tangible User Interface Design." *Interacting with Computers* 25 (1): 1–20. https://doi.org/10.1093/iwc/iws007.

Babatunde, Y., and S. P. Low. 2015. "Research Design and Methodology." In *Cross-Cultural Management and Quality Performance*, by Y. Babatunde and S. P. Low, 93–113. Singapore: Springer. https://doi.org/10.1007/978-981-287-362-0_7.

Bälter, O., E. Enström, and B. Klingenberg. 2013. "The Effect of Short Formative Diagnostic Web Quizzes with Minimal Feedback." *Computers and Education* 60 (1): 234–42. https://doi.org/10.1016/j.compedu.2012.08.014.

Bennett, S., P. Dawson, M. Bearman, E. Molloy, and D. Boud. 2017. "How Technology Shapes Assessment Design: Findings from a Study of University Teachers." *British Journal of Educational Technology* 48 (2): 672–82. https://doi.org/10.1111/bjet.12439.

Blanco, M. M., G. Van der Veer, L. Benvenuti, and P. A. Kirschner. 2011. "Design Guidelines for Self-Assessment Support for Adult Academic Distance Learning." In *Constructing Self-Discovery Learning Spaces Online: Scaffolding and Decision Making Technologies*, edited by S. Hai-Jew, 169–98. Hershey, PA: IGI Global.

Bozalek, V., D. Ng'ambi, and D. Gachago. 2013. "Transforming Teaching with Emerging Technologies: Implications for Higher Education Institutions." *South African Journal of Higher Education* 27 (2): 419–36.

Brito, M. A., and F. De Sá-Soares. 2014. "Assessment Frequency in Introductory Computer Programming Disciplines." *Computers in Human Behavior* 30: 623–28. https://doi.org/10.1016/j.chb.2013.07.044.

Bubaš, G., A. Ćorić, and T. Orehovački. 2012. "The Integration and Assessment of Students' Artefacts Created with Diverse Web 2.0 Applications." *International Journal of Knowledge Engineering and Soft Data Paradigms* 3 (3/4): 261–79. https://doi.org/10.1504/IJKESDP.2012.050724.

Doppelt, Y. 2003. "Implementation and Assessment of Project-Based Learning in a Flexible Environment." *International Journal of Technology and Design Education* 13 (3): 255–72. https://doi.org/10.1023/A:1026125427344.

Emelyanova, N., and E. Voronina. 2014. "Introducing a Learning Management System at a Russian University: Students' and Teachers' Perceptions." *International Review of Research in Open and Distance Learning* 15 (1): 272–89. https://doi.org/10.19173/irrodl.v15i1.1701.

Goosen, L., and D. Van Heerden. 2015. "e-Learning Management System Technologies for Teaching Programming at a Distance." In *Proceedings of the 10th International Conference on e-Learning, ICEL 2015*, edited by C. Watson, 116–26. Nassua: Academic Conferences and Publishing International. Accessed September 25, 2018. http://scholar.google.co.za/scholar?oi=bibs&cluster=16255326938393691479&btnI=1&hl=en.

Goosen, L., and D. Van Heerden. 2016. "E-Learning Environment Tools to Address Online and Open Distance Education Context Challenges." In *Proceedings of the 11th International Conference on e-Learning,* edited by R. M. Idrus and N. Zainuddin, 275–84. Kuala Lumpur: Academic Conferences and Publishing International.

Goosen, L., and D. Van Heerden. 2017. "Beyond the Horizon of Learning Programming with Educational Technologies." In *Beyond the Horizon, Proceedings of the South Africa International Conference on Educational Technologies,* edited by U. I. Ogbonnaya and S. Simelane-Mnisi, 78–90 Pretoria: African Academic Research Forum.

Govender, D. W. 2010. "Attitudes of Students Towards the Use of a Learning Management System (LMS) in a Face-to-Face Learning Mode of Instruction." *Africa Education Review* 7 (2): 244–62. https://doi.org/10.1080/18146627.2010.515394.

Greenland, S. J., and C. Moore. 2014. "Patterns of Online Student Enrolment and Attrition in Australian Open Access Online Education: A Preliminary Case Study." *Open Praxis* 6 (1): 45–54. https://doi.org/10.5944/openpraxis.6.1.95.

Halabi, A. K., A. Essop, T. Carmichael, and B. Steyn. 2014. "Preliminary Evidence of a Relationship between the Use of Online Learning and Academic Performance in a South African First-Year University Accounting Course." *Africa Education Review* 11 (3): 405–23. https://doi.org/10.1080/18146627.2014.934995.

He, W., K. Maly, H. Wu, and L. Xu. 2015. "Using a Scaffolding-Based Blogging Approach to Improve IT and CS Undergraduate Students' Disciplinary Writing Skills." *Proceedings of the Twenty-First Americas Conference on Information Systems (AMCIS)*, 1–6. Puerto Rico: Association for Information Systems (AIS). Accessed September 9, 2018. http://aisel.aisnet.org/amcis2015/ISEdu/GeneralPresentations/4/.

Johnson, B. 2014. *Educational Research: Quantitative, Qualitative, and Mixed Approaches*. 5th ed. Thousand Oaks, CA: Sage.

Matthews, R., H. S. Hin, and K. A. Choo. 2015. "Comparative Study of Self-Test Questions and Self-Assessment Object for Introductory Programming Lessons." *Procedia: Social and Behavioral Sciences* 176: 236–42. https://doi.org/10.1016/j.sbspro.2015.01.466.

Pietersen, J., and K. Maree. 2007. "Standardisation of a Questionnaire." In *First Steps in Research*, edited by K. Maree, 214–23. Pretoria: Van Schaik.

Pretorius, C. M., and H. G. Erasmus. 2012. *Basic Programming Principles*. 2nd. London: Pearson.

Ramasamy, J., S. Valloo, and J. M. P. Nadan. 2010. "Effectiveness of Blog for Programming Course in Supporting Engineering Students." In *Proceedings of the 2010 International Symposium in Information Technology (ITSim)*, edited by A. K. Mahmood, H. B. Zaman, P. Robinson, S. Elliot, P. Haddawy, S. Olariu, and Z. Awang, 1347–350. Kuala Lumpur: IEEE. https://doi.org/10.1109/ITSIM.2010.5561591.

Rand, M. K. 1999. "Supporting Constructivism through Alternative Assessment in Early Childhood Teacher Education." *Journal of Early Childhood Teacher* 20 (2): 125–35. https://doi.org/10.1080/0163638990200209.

Safran, C. 2008. "Blogging in Higher Education Programming Lectures: An Empirical Study." In *MindTrek '08: Proceedings of the 12th International Conference on Entertainment and Media in the Ubiquitous Era*, 131–35. New York, NY: Association for Computing Machinery (ACM). Accessed September 6, 2018. http://dl.acm.org/citation.cfm?id=1457228.

Sklar, J. 2015. *Principles of Web Design: The Web Warrior Series*. 6th ed. Boston, MA: Cengage Learning.

Subotzky, G., and P. Prinsloo. 2011. "Turning the Tide: A Socio-Critical Model and Framework for Improving Student Success in Open Distance Learning at the University of South Africa." *Distance Education* 32 (2): 177–93. https://doi.org/10.1080/01587919.2011.584846.

Todorova, M., H. Hristov, N. Stefanova, and E. Kovatcheva. 2010. "Innovative Experience in Undergraduate Education of Software Professionals—Project-Based Learning in Data Structure and Programming." In *Proceedings of the 3rd International Conference of Education, Research and Innovation (ICERI)*, edited by L. G. Chova, D. M. Belenguer and I. C. Torres, 5141–150. Madrid: International Association of Technology, Education and Development (IATED). Accessed April 12, 2013. https://library.iated.org/publications/ICERI2010.

Van Heerden, D., and L. Goosen. 2012. "Using Vodcasts to Teach Programming in an ODL Environment." *Progressio* 34 (3): 144–60.

Van Heerden, D., and M. Van der Merwe. 2014. "Employing Objective Measures in Search of a Relationship between Knowledge Blogs and Introductory Programming Performance Outcome." In *Proceedings of the 9th International Conference on e-Learning (ICEL 2014)*, edited by T. A. Yáñez, O. S. Rodriguez, and P. Griffiths, 185–89. Valparaiso: Academic Conferences and Publishing. Accessed September 6, 2018. http://tinyurl.com/icel2014.

Vega, C., C. Jiménez, and J. Villalobos. 2013. "A Scalable and Incremental Project-Based Learning Approach for CS1/CS2 Courses." *Education and Information Technologies* 18 (2): 309–29. https://doi.org/10.1007/s10639-012-9242-8.

Vodnik, S., and D. Gosselin. 2015. *JavaScript: The Web Warrior Series*. 6th ed. Boston, MA: Cengage Learning.

Wang, Y. D. 2014. "Applying Constructivist Instructional Strategies to E-Learning: A Case Study of a Web Development Course." *International Journal on E-Learning* 13 (3): 375–406.

Wilson, T., and G. Ferreira. 2011. "e-Learning and Support Tools for Information and Computer Sciences." Paper presented at the 7th China–Europe International Symposium on Software Industry Orientated Education, University of Northampton, May 23–24.