

Automating the Configuration Management and Assessment of Practical Outcomes in Computer Networking Laboratories

Neville Palmer

School of Media Arts and Technology
Southampton Solent University
Southampton, UK
neville.palmer@solent.ac.uk

Warren Earle

School of Media Arts and Technology
Southampton Solent University
Southampton, UK
warren.earle@solent.ac.uk

Dr. Jomo Batola

School of Media Arts and Technology
Southampton Solent University
Southampton, UK
jomo.batola@solent.ac.uk

Abstract— Two dedicated networking laboratories are used by students on a computer networking degree program to undertake practical work using a range of networking equipment and operating systems. Various challenges are faced in the running of the laboratories, particularly in configuration management, sharing the facilities between a number of modules with diverse requirements and the need to assess the practical learning outcomes of a large number of students using them. Some of the configuration management processes have been automated to alleviate some of the challenges within the laboratory environment. An application has been developed to assist with generic assessment that can generate feedback for students. A simulation system with features of assessment automation is also available. This paper will focus on the issues faced in running the laboratories, in particular using this environment to assess practical outcomes. It will evaluate some of the work done to alleviate some of the issues and analyze areas for improvement. The need for a mechanism that can be used to automate the assessment of practical outcomes using real network systems or simulators has been identified. It is proposed that the mechanism will be implemented as a software application whose requirements will be discussed.

Keywords— *Computer Networking Laboratory; Automation; Configuration Management; Assessment; Feedback*

I. INTRODUCTION

Employability is an important objective of higher education pedagogy [1]. In the UK apprenticeship degrees are part of a new initiative to improve links between the needs of employers and the aspirations of students. In Higher Education students accrue substantial debts whilst studying for a degree, whereas as an apprentice they can pursue their studies whilst in paid employment. In the subject of computer networking employers expect students to possess procedural knowledge, theoretical understanding of systems as well as analytical and problem solving skills [2]. Therefore, computing laboratories play a key role in the teaching and learning strategy. These are shared

resources and therefore a number of challenges are faced in the design of a laboratory and how they are managed, including the need to deploy and maintain a standard laboratory build [3], [4]. In terms of assessment it may be desirable to assess both procedural knowledge in the form of practical learning outcomes and theoretical underpinning in the form of preparation activities. A question arises as to whether one should assess practical outcomes by requiring students to write about them or whether one should assess completed tasks by direct observation. If direct observation is chosen as the means of assessment one issue that arises is how evidence of practical outcomes can be recorded for the purposes of moderation and feedback. This report, based upon the results of our previous research in this area [5], [6], will look at the techniques involved in managing the laboratory and also methods for assessing practical learning outcomes. It will discuss how methods for automating laboratory management and assessment are complimentary in developing a means of automating the assessment of practical learning outcomes.

The Computer Networking Degree Program and other computer courses benefit from two dedicated computer networking laboratories that utilize hardware that mainly comprises business networking equipment from Cisco and industrial networking switches supplied by Westermo Data Communications. Dedicated laboratory PCs are available hosting various Windows and Linux based virtual machines. These facilities are primarily used for teaching computer networking modules on the degree programs and professional qualifications from Cisco, Westermo and EC Council. Both the physical aspects of networks, for example in the form of routing and switching, and logical aspects, in the form of operating systems and server services, are taught within the laboratory. Each laboratory can accommodate up to 25 students. Various challenges arise from operation of the laboratories. The

facilities are utilized by a number of courses and modules often with complex and differing hardware and software requirements. This means that various virtual machines are necessary with a number of software packages requiring various pre-requisites to be installed. There is often the need for pre-configuration of both hardware and software prior to each session. However the laboratories are often fully utilized so configuration management in preparation for each session can be very challenging. The needs of sharing hardware between users in different sessions can be a particular problem. Nevertheless the configuration management of operating systems can be automated using appropriate software and often simulation tools are available as an alternative teaching tool to alleviate pressures on shared hardware. Another challenge arises from assessment of practical outcomes. This can be done directly by manually assessing student competencies or indirectly by assessing written laboratory reports. Both methods have advantages and disadvantages. It would be beneficial to automate some of these processes. Some work has been done on this, but further work is required. The issues and the work done so far will be discussed and recommendations for future work can be identified.

II. CONFIGURATION MANAGEMENT

A. Management

In most computer networks hosting large numbers of computers there is the need for a standard organizational configuration for workstations, servers and network devices. A standard build is necessary, especially for workstations in the organization for reasons that include security and maintainability. Some larger organizations utilize a service management system, such as Information Technology Infrastructure Library (ITIL), to manage changes to their IT infrastructure [7]. However where innovation is important this process may hinder development of new systems and a more fluid system might be necessary. For example the Ordnance Survey has found that an Agile oriented approach is more appropriate for innovation [8]. To this end DevOps is a practice that can be used to unify software development with operations within an organization [9]. This also requires the management of a development environment so that software can be reliably deployed. In an educational environment the general IT infrastructure of an institution, consisting of staff and student computers and servers hosting systems such as email, virtual learning environment, timetabling, etc., might be less fluid and therefore it is appropriate to employ a service management approach, perhaps utilizing ITIL. However in an experimental laboratory based environment this approach may not be suitable. The laboratory facilities may be used for teaching or research purposes. In IT laboratory teaching it is important to keep up to date with the latest technologies and practices and in the case of research new systems may be developed and tested. Laboratory management practices need to be more fluid. Changes to the laboratories are unlikely to affect the whole organization and the laboratory based team's requirements might be more insular. In which case more formal approaches may seem over burdensome. In the case of the authors'

institution the dedicated networking laboratories comprise two rooms involving the work of a handful of academic and support staff. Nevertheless some form of management and documentation are necessary to manage change.

B. Configuration

Organizations need some means of deploying and managing IT systems and software. Often they need to maintain a standard build to ensure compliance with agreed specifications and security within the organization. In business organizations changes to IT requirements can be centrally automated by using directory services such as Active Directory [10]. More fluid development environments may require different management systems. It may be more appropriate to use scripts to manage systems. Windows PowerShell or Linux scripts could be used to perform certain tasks. However as scripts become more complex some management of their deployment is likely to be necessary. Applications like Chef, Puppet and Ansible allow scripts to be better organized and managed [11],[12],[13]. Configurations for an entire IT infrastructure are configured and are managed centrally. These systems are used to manage a production environment, though they require some expertise and time to use effectively on a daily basis and this may be available within a DevOps team in a larger organization.

On the other hand in an educational laboratory environment staff time can be limited, even if expertise is available. In this case it has been found necessary within the authors' university to develop a customized management application involving a graphical user interface to manage and automate the deployment and configuration of the laboratory environment as shown in Fig. 1. The application is based on Visual C# and calls PowerShell scripts to perform the management tasks [5]. Common tasks, such as reconfiguring or rebuilding computers can be performed easily by a point and click interface in which individual computers can be targeted. The problems that are faced by those supporting an educational laboratory are in some ways different to those faced by those supporting a wider organization. For example students will often need to work on virtual machines with administrative permissions. They might be making changes to the standard build that need to be undone soon after a session has taken place in the laboratory. Support

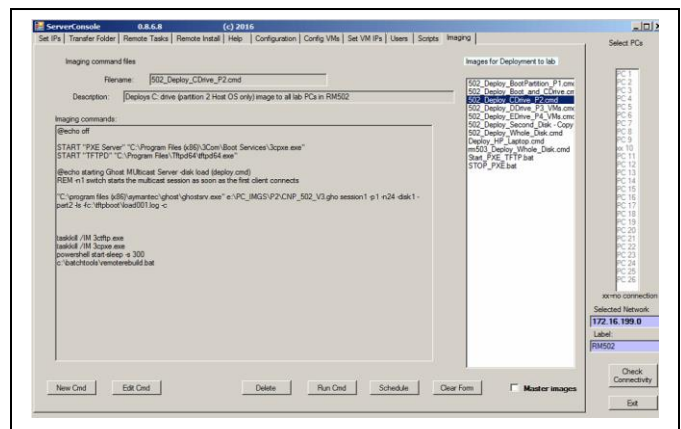


Fig. 1 Server Console configuration management application

staff need to quickly recover the systems back to a standard configuration. Using the application virtual machines can be cloned from a common master which can be quickly re-cloned using the customized application, and these tasks can be scheduled to be performed automatically at certain times.

One challenge of teaching IT is that it is a constantly changing subject. A master build is maintained and using Symantec Ghost this can easily be deployed to all computers when major changes are necessary [14]. In a production environment this might be an issue as existing data may be overwritten, but in the laboratory to return computers to a standard state may be desirable. The customized application can easily deploy the new build to certain computers or all within the laboratory by selecting the PCs and then the operation to perform on those selected. The application can call remote agents to perform certain tasks. For example as a result of the rebuilding process it has been found that all virtual machines might have the same physical address allocated to them. A remote agent, again written in C#, can be called from the server to parse the virtual machine configuration files to look for network interfaces so that their physical address can be randomized.

III. ASSESSMENT PRACTICES

A. Assessment Feedback

When assessing students there are a number of issues to address regardless of the assessment method in use. Assessment should be seen to be fair and feedback should be meaningful and timely. In order to be fair the assessment rubric must be clearly written so that it is not seen as ambiguous in nature, although it should still leave room for academic judgement [15]. Feedback to the student should show where marks have been awarded according to the rubric, yet also provide meaningful academic feedback on judgements made. However academics are often constrained by the need to provide feedback within a prescribed time period and often for large numbers of students. This might tend to limit the quality of feedback. Yet feedback is often cited as a major concern by students in student surveys, such as the National Student Survey (NSS) [16]. It would be beneficial to improve the quality of both summative and formative feedback, perhaps with the aid of some form of automation in order to alleviate time pressures on academics.

B. Assessment Automation

A range of computer and web based applications are available to make it easier to assess student work and annotate feedback. Some features are available in the Moodle and Canvas [17],[18] virtual learning environments which allow various assessments to be configured easily. Multiple choice assessment can be marked automatically and feedback available to the student upon completion. Assessment rubrics can be formulated and they allow online marking and comments can be inserted by academics.

Whilst most assessment practices, apart from perhaps multiple choice tests, involve the academic in making

judgments based on a given rubric this process does not normally involve artificial intelligence. However some work has been done on methods of automating assessment that is available in mainstream teaching in education. These methods lend themselves well to some aspects of practical work. Within the authors' institution an application, E-PHP, has been developed for students who are learning to program in PHP. The application provides feedback to learners on the syntax of their code and provides suggestions for improvement [19]. Whilst it is not being used for summative assessment it can provide formative feedback to students to improve their learning and at the same time their perception of feedback, which may in the past have involved tutor intervention and verbal comments. This may relieve time pressure on academics during some sessions and enable them to concentrate on students that require more support. This system might also encourage students to learn more independently of tutor intervention.

Often a number of students make the same mistakes in a particular assessment. It is therefore helpful if assessment systems allow standard comments to be included in feedback appropriate to each grade. Further comments highlighting common issues can be drawn from a standard palette of generic comments. The authors' have developed an application that assists in the automation of assessment practice, called Ultramarker, as shown in Fig. 2. A wide range of assessment methods are supported by the application ranging from reports, laboratory work, to presentations. A standard grading schema can be configured and then individual rubrics for each form of assessment. For criteria within the rubric standard comments can be entered for each grade within the schema and recommendations for feedback can be added to each grade. Additional comments can be added in freeform format with the assistance of standard phrases from a palette. The application can produce a detailed report for each student. Informal staff feedback has indicated that the application has been found to improve the consistency and timeliness of marking, leaving

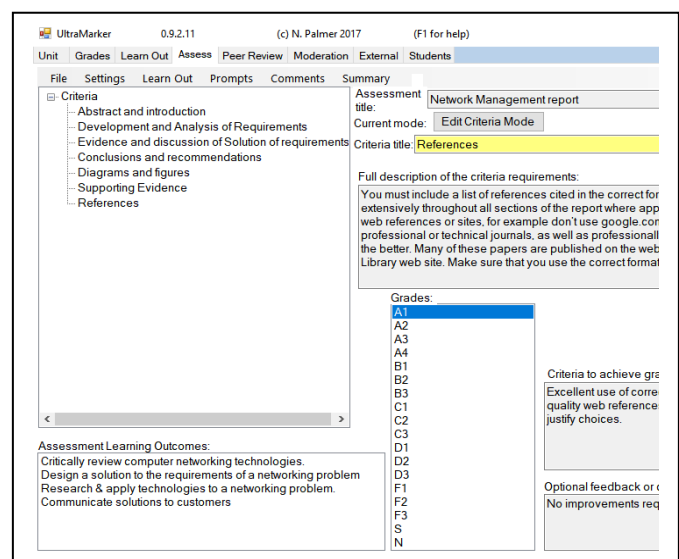


Fig. 2 Ultramarker assessment application

academics more time to concentrate on other issues. It has also been praised as an exemplar by external examiners for the detail of its feedback. Students have responded positively in a questionnaire conducted as part of previous research that indicates that they perceive feedback to have improved from use of the application [6].

These applications don't currently automate the process of marking student work and this is still left to academic judgement. However other applications have attempted to do this. Cisco have developed automated marking in their Packet Tracer application, which can be linked to Canvas [20]. This is an educational simulator that enables students to implement and test networks in software. An assessment rubric can be produced that requires students to conduct a number of tasks in implementing a network according to particular requirements. Students have to enter Cisco Command Line Interface (CLI) commands in order to complete the assessment in a time constrained situation. This has been used to mark a number of time constrained practical assignments at the authors' University. However it has been found that manual intervention may be required because the software cannot interpret whether minor errors have been made. It determines whether specific commands have been used with the correct syntax. There may be for example be 10 commands that must be entered, line by line, to complete a particular practical task in the assessment. If a student makes a mistake on one line, perhaps through a spelling error or incorrect network address, they will lose all marks for that task. It could be argued that as the objectives for the task were not fully completed the system wouldn't work in a real world situation, therefore students should test this and engage in problem solving in order to complete the task successfully. On the other hand it could be argued that a student may perhaps have made a simple mistake and eventually they would have corrected it once they have realized their error, in which case they should only lose one mark from 10 within the whole task rather than all 10 marks. This situation can be likened to a mathematical problem where we could consider marking the process as well as the final answer to provide a score for the question. Any improvements to assessment automation might need to accommodate these considerations. It may be easier to mark the output answer by testing a particular function than to assess the accuracy of the whole process used to achieve the result.

Whilst Packet Tracer is able to assess practical work within a simulation environment it is not a generic assessment mechanism and it doesn't assess the output of work within an environment using real equipment or alternative systems.

IV. ASSESSMENT OF PRACTICAL OUTCOMES

Some laboratory work in year 1 and 2 of the degree program are assessed by means of Personal Learning Records that are designed to record student engagement in their learning of practical outcomes. Students might demonstrate preparation work or practical competencies during laboratory sessions and then support this with evidence uploaded to a virtual learning environment. We might place greater emphasis on the

demonstration of practical outcomes during the laboratory sessions or on the supporting evidence in the personal learning record [6]. Problems associated with evidence of moderation of work may arise if practical outcomes are demonstrated, whereas if we accept written evidence of engagement from students who haven't engaged in the sessions it may be difficult to trust the originality of submitted work and reconcile this to reflect an accurate assessment of practical outcomes.

Cisco modules are an integral part of the networking degrees. They make use of practical time constrained assignments to assess practical outcomes and multi-choice tests to measure theoretical outcomes. These are mainly set at the end of the unit, though smaller theory tests can be used to provide feedback regularly each week. The practical time constrained assignments (TCA) can make use of real networking equipment. Though with large class sizes this can be time consuming to configure, manage and assess, since manual intervention is required to read configurations from networking devices. However the Packet Tracer network simulator can be used instead, which means that the assessment and feedback are more scalable and timely.

What if we want to use non Cisco equipment or servers in our assessments? For example an assessment may involve the configuration of a Windows or Linux server or we may wish students to configure Westermo industrial networking systems. Without a simulator we would need to use real equipment. Although we could use Graphical Network Simulator (GNS3), which is a network simulator that allows equipment from other manufactures to be configured in software that can be connected to real server systems [21]. If we wish to assess the students' ability to configure servers and network devices we would need to check the level of progress in configuration of each system. This means viewing the configuration commands or files from each system. This can be very time consuming for the assessor, especially if more than one device or computer system are involved. It might be possible to arrange for a small group conducting an end of unit TCA, but if we wished to monitor the practical outcomes from each session on a weekly basis it would be more difficult. In the case of some server systems it is possible to judge at a glance whether a system has been configured correctly, for example by viewing a configuration file on a server. but for other systems this isn't as easy.

V. PROPOSED ASSESSMENT SYSTEM

A. Requirements

A new automated laboratory assessment mechanism is proposed and some requirements for a working system have been identified from the characteristics of laboratory assessment practices and the observations made from the current processes. The new system must be capable of assessing both practical TCAs and the outcomes of practical work on a regular basis. An application is required to automate the assessment process. In the order to do this the requirements of the proposed system have been identified as follows.

The prototype application:

- must be capable of assessing both practical TCAs and the outcomes of practical work.
- must be capable of assessing the practical outcomes of students work on computers and networking devices.
- must be capable of operating with real equipment, virtual machines or real world capable applications like GNS3.
- must allow pre-configuration of the laboratory environment.
- must be able to read configuration information remotely from computers and networking devices in order to facilitate assessment.
- must allow the assessor to configure a template to allow retrieved information to be compared against practical outcomes defined in assessment criteria
- must allow the assessor to define assessment criteria and feedback comments for each practical task
- must be capable of automatically assessing the outcomes of practical work by parsing the configuration information.
- must generate feedback for students based on their performance in each part of the assessment

B. Discussion

Similar tasks are being performed during practical TCAs as those during laboratory sessions. If this were not automated the level of manual intervention required would render assessment of individual laboratory sessions very time consuming. Therefore a major objective of the new system should be to increase the timeliness of the assessment process.

Some networking laboratories require students to work on server or workstation operating systems, for example the configuration of a server, or on networking equipment, for example a router. The operating systems may differ, for example it may be Linux or Windows or it may involve an operating system specific to networking devices such as the Cisco operating system. It is therefore desirable that the system should handle multivariate systems.

Any system capable of interacting with the real world, whether GNS3, virtual machines or real equipment, should be able to communicate with the proposed assessment application that will be running on a server connected to the laboratory network.

An application capable of reading configuration information should also be able to manage the configuration of

the laboratory network. This could be part of a separate configuration management system or part of the new system.

The new system would need to operate on a central server and retrieve configuration information from computer and network devices remotely. On most computer systems and networking devices configuration information can be retrieved in human readable text form. This information could be retrieved using scripts, for example PowerShell scripts in Windows or Bash scripts in Linux.

The configuration information retrieved from the laboratory devices or systems would need to be assessed against defined outcomes. The configuration information could be parsed and compared to an assessment definition template. The extent of successful matches against the template would define the quantitative level of the assessment for each student.

Standard feedback could be defined for each criteria within the assessment template. A different statement of feedback could be generated depending on the level of success within each part of the assessment. As we have found some automated assessment may require manual intervention, depending on the viewpoint taken as to whether the assessment of a particular task should be wholly or just partially correct in order to allocate a mark to that task. This may lead to further work on the use of artificial intelligence to mitigate some of the failings of simplistic automation of assessment based on a comparative template. Nevertheless the proposed system will be a prototype and will not be constructed with serious artificial intelligence in mind. It is a system for proof of concept at this stage.

VI. CONCLUSIONS AND RECOMMENDATIONS

As we have seen tools are already available for the development of part of the proposed system. The customized laboratory configuration management system in use at our institution could be used to form the basis of the new assessment system. It is capable of configuring servers and workstations remotely from a central server, though at the moment it does not have the facility to configure routers remotely. More than one system could be used to support the new assessment application. It is intended that other configuration management systems will be investigated. A final year student project has been proposed that will involve a student in investigating the capabilities of Ansible in configuring computer and network devices and also retrieving configuration information for backup purposes. This will determine whether Ansible can be used as the configuration management and information retrieval mechanism for the proposed application.

The core assessment system would allow a grading schema to be defined. For each assessment criteria should be defined with feedback comments appropriate to each grade. To actually assess the students practical work this would need to be mapped to a template that the assessor should define. This template would define practical configuration tasks in a format native to each computer or network device against which real system or device configurations can be compared. This configuration

information may be provided by another application like Ansible. The prototype would parse the retrieved information and compare it to the template file, looking for successful matches. The new system would have to allow the assessor to define distinctive sections in the template that correspond to individual parts of the assessment. The application would need to determine the extent to which practical learning outcomes or competences have been met. The output might just be in the form of a figure like a percentage of success, but there would be the facility to map the output to assessment criteria and generate standard feedback phrases. The Ultramarker assessment application, developed in Visual C#, already allows academics to define grading schema and assessment criteria. The functionality of this could be extended to allow the definition of a practical assessment template. The assessment comparison feature could also be included by further development of the application. An efficient text parser already exists for use in the laboratory, developed in C#. As mentioned previously this currently parses a VMware configuration file looking for ethernet adapters in order to randomize their physical address. This parser can be used in similar way to compare actual configuration information with a pre-defined template. Further developments would see a less prescriptive and more human intelligence oriented approach to the comparison task, but for the prototype this will not be necessary.

It is intended to develop a prototype for use on a Network Security unit. This unit has used real equipment and the GNS3 simulator in the past for assessment of a practical TCA, but as the course grew the level of manual intervention required became very time consuming as it involved the tutor in analyzing the work of over 20 students with no automation. Packet Tracer offered a solution, but it doesn't allow students the flexibility of using the graphical configuration interfaces available for real systems and it also seemed to encourage students to remember sequences of commands rather than involving them in understanding the purpose of the overall task. On the next iteration of the TCA the tutor would like to use the GNS3 simulator or real equipment, but would also like to automate the assessment process. The proposed application will allow this, therefore the Network Security unit can be used as an experimental environment for a prototype system.

ACKNOWLEDGEMENTS

Thanks to Dr. Nick Whitelegg for technical assistance in the production of this paper.

REFERENCES

- [1] J. Artess, T. Hooley, R. Mellors-Bourne, *Employability: A Review of the Literature 2012 to 2016*. UK: Higher Education Academy, 2017.
- [2] BCS, *Qualification Specification for the Knowledge Modules that Form Part of the BCS Level 4 Network Engineer Apprenticeship*. UK: British Computer Society, 2016.
- [3] C. E. Caicedo and W. Cerroni, "Design of a computer networking laboratory for efficient manageability and effective teaching," in *Frontiers in Education Conference, 2009. FIE'09. 39th IEEE*, 2009.
- [4] J. F. Cigas, "A computer networking laboratory for administration and networking," in *Frontiers in Education, 2002. FIE 2002. 32nd Annual Conference*, 2002.

- [5] N. Palmer, "Work in progress - automation of a computer networking laboratory," in the *2015 IEEE Global Engineering Education Conference (EDUCON), 18-20th March 2015, Tallinn, Estonia* [Online]. pp. 348-353. Available: IEEE Xplore, <http://www.ieee.org>. [Accessed: 2nd Oct. 2017].
- [6] N. Palmer, "Portable tool for assessing practical learning outcomes," in *2017 IEEE Global Engineering Education Conference (EDUCON), 2017, Athens, Greece* [Online]. pp. 688-692. Available: IEEE Xplore, <http://www.ieee.org>. [Accessed: 2nd Oct. 2017]
- [7] G. C. Office and A. T. Orr, *Introduction to the ITIL Service Lifecycle*. (3rd ed.) Norwich: TSO, 2011.
- [8] J. Reeves, "Keeping up with the start-ups," British Computer Society presentation, Southampton, 23rd Feb. 2016.
- [9] K. Taylor, P. Waterhouse and A. Ravichandran. *DevOps for Digital Leaders*. Apress, 2016.
- [10] Microsoft, 2017. *About Active Directory Domain Services*. [online]. Available: [https://msdn.microsoft.com/en-us/library/aa772142\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/aa772142(v=vs.85).aspx). [Accessed: 9th Oct. 2017]
- [11] Puppet, 2017. *Why Puppet*. [online] Available: <https://puppet.com/products/why-puppet>. [Accessed: 9th Oct. 2017]
- [12] Chef, n/d. *An overview of Chef*. [online]. Available: https://docs.chef.io/chef_overview.html. [Accessed 9th Oct. 2017]
- [13] Red Hat. *Ansible in depth*, 2017. [online]. Available: http://cdn2.hubspot.net/hub/330046/file-480366556-pdf/pdf_content/Ansible_in_Depth.pdf?t=1410536005388. [Accessed: 9th Oct. 2017]
- [14] Symantec. *Symantec Ghost implementation guide*. USA: Symantec, 2008. [online] Available: ftp://ftp.symantec.com/public/english_us_canada/products/symantec_ghost_solution_suite/2.5/manuals/Ghost_imp_guide.pdf. [Accessed: 10th Oct. 2017]
- [15] S. Ball, C. Bew, S. Bloxham, et al. *A Marked Improvement - Transforming Assessment in Higher Education*. UK: Higher Education Academy, 2012.
- [16] HEFCE. *National Student Survey*, 2012. [online] Available: <http://www.hefce.ac.uk/lt/nss/>. [Accessed: 9th Oct. 2017]
- [17] Moodle, 2017. *Features* [online]. Available: <https://docs.moodle.org/33/en/Features> [Accessed 25th Oct. 2017]
- [18] Canvas, 2017. *Canvas Instructor Guide* [online]. Available: <https://s3.amazonaws.com/tr-learncanvas/files/pdf-guide/CanvasInstructorGuide.pdf>. [Accessed: 25th Oct. 2017]
- [19] N. Whitelegg and O. Isiaq. "EPHP - a visualisation tool for learning server-side web development: Initial work and pilot study," in *International Conference on Information, Communication Technologies in Education 2017*, 2017, Rhodes, Greece [Online]. Available: <http://www.icicte.org/ICICTE17Proceedings.htm>. [Accessed: 16th Oct. 2017]
- [20] Cisco, 2010. *Cisco Packet Tracer Skills Assessments*. [online]. Available: https://www.cisco.com/c/dam/en_us/training-events/netacad/course_catalog/docs/PT_AAG.pdf. [Accessed: 16th Oct. 2017]
- [21] GNS3, 2017. *Connect GNS3 to the Internet (local server)*. [online]. Available: <https://docs.gns3.com/1vFs-KENh2uUFfb47Q2oeSersmEK4WahzWX-HrMIMd00/index.html>. [Accessed: 16th Oct. 2017]