International Journal of Research in E-learning
Vol. 6 (2), 2020, pp. 1–20

https://doi.org/10.31261/IJREL.2020.6.2.04

Pedro Ramos Brandao
Evora University (CIDHEUS) – Evora (Integrated Researcher)
https://orcid.org/0000-0001-6351-6272

Cloud Computing and Distance Learning in Computer Science

Abstract

This work demonstrates that the use of laboratories for the development of curricular work in the area of information technology exclusively supported by cloud computing technology does not decrease the level of learning and assessment objects on the part of students. This scenario arose from the need to interrupt face-to-face classes in physical laboratories due to the COVID-19 Pandemic.

Keywords: Cloud Computing, Azure, distance learning, WEB App, Virtual Labs, DevOps.

The Context

In everyday language Cloud Computing (CC) is a set of computational resources made available to a group of users, remotely, taking the form of services. And like any service today, electricity, water, gas, etc., they are always available for both individual and business use. This is possible because throughout the second half of the 20th century, the technical industry adopted a set of standard models from different sources and technological platforms. CN works as a universal, paid service, but is always available to users/customers who need it, in the same way that they have the electricity service. The main principle that supports this technology and this model is the provision of computing, storage and software as a service. Cloud is a distributed and parallel computing system, consisting of a collection of virtualized and interconnected computers that are presented as a set of dynamic and unified resources, based on Service Level Agreement (SLA) established between the service provider and the end customer. With the crisis caused by the pandemic of COVID-19, almost all universities had to change the model of classroom classes to a model of distance classes. The paradigm is radically different. In certain scientific and technical areas, the transition was more straightforward. However, in the case of laboratory classes the process became more complicated or even impossible. In the specific case of computer network laboratories which were based on computers and servers installed at universities' on-premise infrastructures, the process was almost impossible and these laboratory classes had to be postponed, or complex processes had to be created so that students could go to the labs. This work intends to develop a demonstration of a model that can allow to simulate the teaching of the development and operation of computer networks through a laboratory created by the teacher in a Cloud Computing system – in this case, in Azure.

The main objective of this work is to verify if the use of cloud computing laboratories (in this case in Azure) for curriculum development in substitution of physical laboratories on university campuses, due to the mandatory interruption of classroom classes in the context of COVID-19, had an impact on learning and what is the effect on school success, namely on assessments. The second objective of this work is to compare if the school objectives and the evaluation between two different curricular units obtained the same performance standard or, in case of divergences, this second objective helps to validate whether the method used based on laboratories in cloud computing is successful or not. This study is contextualized in the area of information technology and, specifically, in the area of a master's degree in computer science.

Main objectives may be encapsulated as follows:

- to compare the evaluation made in the pre-Covid-19 period and the evaluation during the first phase of Covid-19;
- to analyze whether the use of online laboratories had a negative impact on the evaluation.

At the beginning and during the first phase of the Covid-19 pandemic, more than 90% of teachers were not familiar with distance learning. In a record period of two months, they had to adapt to a new paradigm. The adaptation was made essentially with self-study by the teachers. In the case of students there was an identical situation, they were also not familiar with distance learning. They had always taken classes exclusively face-to-face.

History of Cloud Computing

The concept of CC arises from another idea that was called time-sharing, and which implied sharing, by several entities at different times, the same computing equipment. However, today, there are substantial differences between the concept of computational time-sharing and CC. At the time of computational time-sharing, the services or machines that held them could only be used by one operator at a time, that is, they were divided according to portions of time and not portions of accommodation space; today the question of time-sharing does not exist, it pays according to the service that is obtained or the space that the information occupies, but not according to the time that this resource is being used. But the basic idea is very similar – it is to provide a computing service remotely and not locally. Using daily and straightforward concepts by analogy, we can say that both the computational time-sharing system and the DC system can be seen as a service that is made available in the same way as electricity or water, that is, they are available when we need them, we pay for that availability and its use. For example, if we use Office 365 we pay for its use as when we consume electricity, but we can also pay for the accommodation of the documents produced by it, as we pay if we want to have a storage system for reserve energy. Initially, in the sixties of the 20th century, computer systems were in large rooms with powerful and expensive cooling systems and a high consumption of electricity. Besides, they could only be used locally. The next step was to change this situation through a system that would allow remote access to a central computer; it is from here that we can talk about time-sharing.

Organizations could buy time to use a computer system, without having to have it in their facilities, and without having to worry about its maintenance and administration. Microcomputers started to appear when the concept of time-sharing was already popular. Microcomputers were smaller than mainframes and were significantly cheaper. Microcomputers quickly became more sophisticated than the old mainframes, but more importantly, microcomputers had multitasking capabilities, allowing for time-sharing. The concept of time-sharing and microcomputers quickly led to the creation of a new idea, that of distributed systems.

Fundamentals of Cloud Computing

Definitions

The term cloud computing (CC) has become somewhat of a sensational term, almost everyone in the industry has its definition of CN, in this paper we adopted the definition of CC assumed by the National Institute of Technology and Standards (NIST, 2020):

> Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model promotes availability and is composed of five essential characteristics (On-demand self-service, Broad network access, Resource pooling, Rapid elasticity, Measured Service); three service models (Cloud Software as a Service (SaaS), Cloud Platform as a Service (PaaS), Cloud Infrastructure as a Service (IaaS)); and, four deployment models (Private cloud, Community cloud, Public cloud, Hybrid cloud). Key enabling technologies include fast wide-area networks, powerful, inexpensive server computers, and high-performance virtualization for commodity hardware. The Cloud Computing model offers the promise of massive cost savings combined with increased IT agility. It is considered critical that government and industry begin adoption of this technology in response to difficult economic constraints. However, cloud computing technology challenges many traditional approaches to datacenter and enterprise application design and management. Cloud computing is currently being used; however, security, interoperability, and portability are cited as major barriers to broader adoption. The long-term goal is to provide thought leadership and guidance around the cloud computing paradigm to catalyze its use within industry and government. NIST aims to shorten the adoption cycle, which will enable near-term cost savings and increased ability to quickly create and deploy enterprise applications. NIST aims to foster cloud computing systems and practices that support interoperability, portability, and security requirements that are appropriate and achievable for important usage scenarios. (Mell, 2011)

In everyday language, Cloud Computing (CC) is a set of computational resources made remotely available to a group of users, taking the form of services. And like any service today, electricity, water, gas, etc., are always available for both individual and business use. This is possible since throughout the second half of the 20th century, the technological industry adopted a set of standard models from different sources and technical platforms. CC works as a universal, paid service, but always available to users/customers who need it, in the same way that they have the electricity service. However, CC is a term that has many meanings, but an effort has been made in the last ten years to establish common denominators in these meanings, to create a more objective definition. This effort has mainly been made through the publications of the Information Technology Infrastructure Library (ITIL), which is a library of good practices to be applied in infrastructures, operations and maintenance of information technology services, having been developed by the Central Computer and Telecommunications Agency (CCTA), today under the Office for Government Commerce (OGC) in England. In the most generic form, and following the ITIL concept, a service is a relationship between a consumer and a provider, in which the provider makes available and delivers to the consumer a value (service) and the consumer avoids the risks and investments in providing it with this value (service) itself, currently the CC is also inserted in this context. For example, the simple storage and database management services of a retail store on the Web, if they are using the CC, are always available to the store owner, as well as to his customers, without the said owner having to worry about maintaining those services, and don't have to bear the investment risks of purchasing a complex platform. If the business does not go well, it merely cancels the services, having had no losses associated with the investment of fixed technological assets (CSIAC, 2020).

Main characteristics of cloud computing technology

- On-demand self-service services: a consumer can unilaterally provision computing resources, such as server time and storage network, as needed automatically, without requiring human interaction with each service provider.
- Broad network access: resources are available on the network and accessed through standard mechanisms that promote use by thin or thick heterogeneous client platforms (e.g. cell phones, tablets, laptops, and workstations).
- Pooled resources: the provider's computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to the consumer's requirements. There is a sense of location independence, as the customer generally has no control or knowledge about the exact location of the resources provided, but may be able to specify the location at a higher level of abstraction (for example, country, state or data-center). Examples of resources include storage, processing, memory and network bandwidth (Soundarjan, 2015).
- Rapid elasticity: resources can be provisioned elastically and made available, in some cases automatically, to scale out and in quickly, according to the needs. To the consumer, the resources available for provisioning generally appear to be unlimited and can be appropriated in any quantity and at any time.

- Services with metrics, cloud systems automatically control and optimize the use of resources, leveraging a metering resource at levels of abstraction appropriate to the types of services (for example, storage, processing, bandwidth and active user accounts). The use of resources can be monitored, controlled and reported, providing transparency to the provider and the consumer of the services used (Brandao, 2019).

Service Model

Currently, cloud computing technology has three service models (Figure 1):

- Infrastructure as a Service (IaaS): the user can implement and execute software arbitrarily, which can include operating systems and applications. The user does not administer or control the underlying cloud infrastructure, but has control over operating systems, storage, deployed applications and limited control over some network components, for example, host firewalls. The services offered by this delivery model include: hosting servers, Web servers, storage, computing hardware, operating systems, virtual instances, load balancing, Internet access and bandwidth provisioning.
- Platform as a Service (PaaS) allows a cloud user to implement a product created or purchased by the consumer as applications using programming languages and tools supported by the service provider. The user has control over the implemented applications and, possibly, the settings of the hosting environment. He/she does not administer or control the underlying cloud infrastructure, including network, servers, operating systems or storage. This model is not particularly useful when: the application must be portable; proprietary programming languages are used; hardware and software must be customized to improve application performance (Carlyle, 2016).
- The service provider provides software as a service only. The user does not administer or control the underlying cloud infrastructure or the resources of individual applications. Services offered include business services, such as workflow management, groupware and collaboration, supply chain, communications, digital signature, customer relationship management (CRM), desktop software, financial, geospatial and research management. Web 2.0 applications, such as: metadata management, social networks, blogs, Wiki services and portal services. It is not suitable for real-time applications or for those where data cannot be hosted externally. Examples: Office 365, Salesforce.com, Gmail (Voorsluys, 2011).

Implementation Models

There are four models for implementing cloud computing technology:

- Private Cloud: the cloud infrastructure is provisioned for exclusive use by a single organization, composed of several consumers (for example, business units). It can be owned, managed and operated by the organization, third parties or a combination of them, and it can exist on or off the premises (Armbust, 2009).

- Community Cloud: the cloud infrastructure is provisioned for exclusive use by a specific community of consumers from organizations that share concerns (for example, mission, security requirements, policy and compliance considerations). It can be owned, managed and operated by one or more community organizations, by third parties or a combination of them, and it can exist on or off the premises (Page, 2012).
- Public cloud: the cloud infrastructure is provisioned for open use by the general public. It may be owned, administered and operated by a company, academic or governmental organization, or some combination thereof. It exists on the premises of the cloud provider.
- Hybrid Cloud: the cloud infrastructure is a composition of two or more distinct cloud infrastructures (private, community or public) that remain unique entities, but are united by standardized or proprietary technology that allows the portability of data and applications (for example, cloud augmentation for load balancing between clouds).

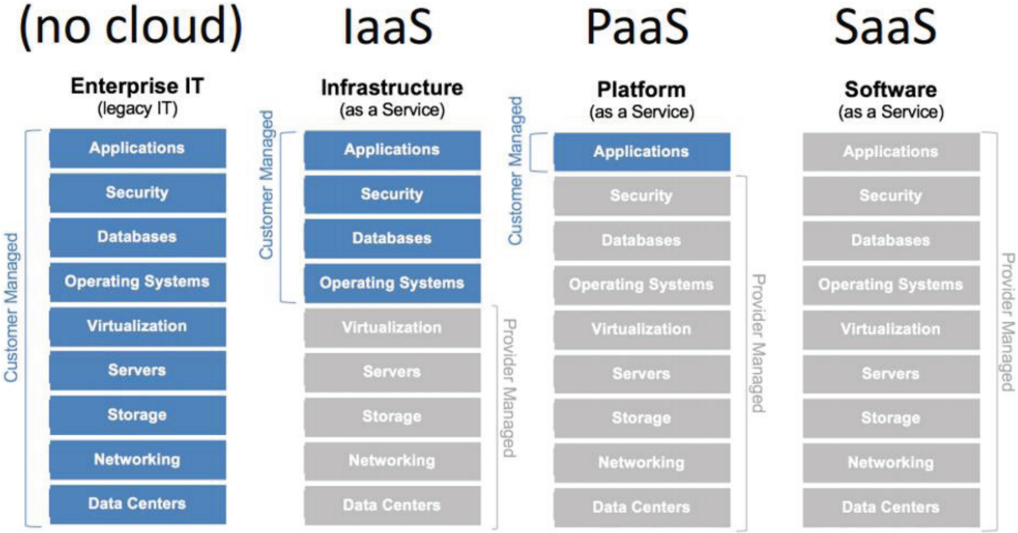

Figure 1. Models of Cloud Computing Services. *Source:* own work

DevOps

Azure DevOps consists of a mixture of development (Dev) and operations (Ops) – DevOps is the union of people, processes and technology to deliver applications quickly and safely. DevOps gives way to functions that previously worked in silos – development, IT operations, quality engineering and security coordinate and collaborate to produce better and more reliable products. By adopting a culture of DevOps and DevOps practices and tools, teams gain the ability to better respond to development needs and increase confidence in the applications they create as well as achieve their goals faster. The students who used DevOps did so in the context of a curricular unit for developing applications for mobile devices. The methodology followed in curriculum development and application development was as follows:

1. Plan. In the planning phase, the DevOps teams devise, define and describe the features and capabilities of the applications and systems they are going to create. They track progress through high and low levels of granularity, from tasks for individual products to cross-cutting tasks for multiple products. Some of the ways in which DevOps teams plan with agility and visibility include creating to-do records, monitoring errors, managing agile software development with Scrum, using Kanban boards and viewing progress with dashboards.
2. Develop. The development phase includes all aspects of programming – writing, testing, revising and integrating the code by the team members – as well as compiling that code into compilation artefacts which can be implemented in various environments. DevOps teams seek to innovate quickly without sacrificing quality, stability and productivity. To do this, they use highly productive tools, automate mundane and manual steps and iterate in small increments through automatic testing and continuous integration.
3. Distribute. The delivery consisted of the process of implementing applications in production environments consistently and reliably. The delivery phase also includes implementing and configuring the entirely governed base infrastructure that makes up these environments. At this stage, the teams define a launch management process with exact manual approval steps. They also define automatic limits that move applications between phases until they are made available to customers. The automation of these processes makes them scalable, repeatable and controlled. Thus, teams that follow DevOps can frequently deliver with ease, confidence and tranquility.
4. Operate. The operation phase involves the maintenance, monitoring and troubleshooting of applications in production environments. Teams, when adopting DevOps practices, work to ensure the reliability of systems and high

availability and aim for no downtime, while reinforcing security and governance. DevOps teams seek to identify problems before they affect the customer experience and mitigate them quickly before they occur. Maintaining this surveillance requires advanced telemetry, actionable alerts and full visibility to applications and the underlying system.

The research methods

The main techniques and tools used for gathering research data include the following quantitative methods:

- Observation. This involved counting the number of times that a specific event occurred or encoding observational data to translate them into numbers. In this case, the difficulties in completing the laboratories were requested for the evaluation.
- Screening of assessment documents – obtaining numerical data from the evaluation forms and counting of unrealized events that were considered mandatory objectives for evaluation.
- Comparative experimentation – testing hypotheses in laboratories, testing cause and effect relationships, through experience in the development of network laboratories, comparing the results in terms of performance between laboratories developed in person before the COVID-19 crisis and the performance in the development of network labs in a 100% cloud computing environment.
- Hypotheses. One of the leading hypotheses presented is that the transition from a classroom model to a fully online class model does not affect the students' performance and learning level, nor does it provide a lower level of evaluative success in the area of information technology information. As a second hypothesis, it is expected that precisely in the area of information technology, the results will be even more positive in the case of curriculum development done in online laboratories, in particular via cloud computing.

Laboratory to teach how to create a WEB App

The plan for creating a WEB App follows the exact indications advised by Microsoft in its teaching plan for the development of these solutions in Azure (Microsoft, 2020).

If you want to build a website for a new business, or you are running an existing web app on an old on-premises server, you can do it in 15 minutes. Setting up a new server can be challenging. You need appropriate hardware, likely a server-level operating system, and a web hosting stack. Hosting your web application using Azure App Service makes deploying and managing a web app much easier when compared to managing a physical server. In this module, we will implement and deploy a web app to App Service.

Learning objectives:

- use the Azure portal to create an Azure App Service web app;
- use developer tools to create the code for a starter web application;
- deploy your code to App Service.

Using the Azure portal, you can easily add deployment slots to an App Service web app. For instance, you can create a staging deployment slot where you can push your code to test on Azure. Once you are happy with your code, you can easily swap the staging deployment slot with the production slot. You do all this with a few simple mouse clicks in the Azure portal.

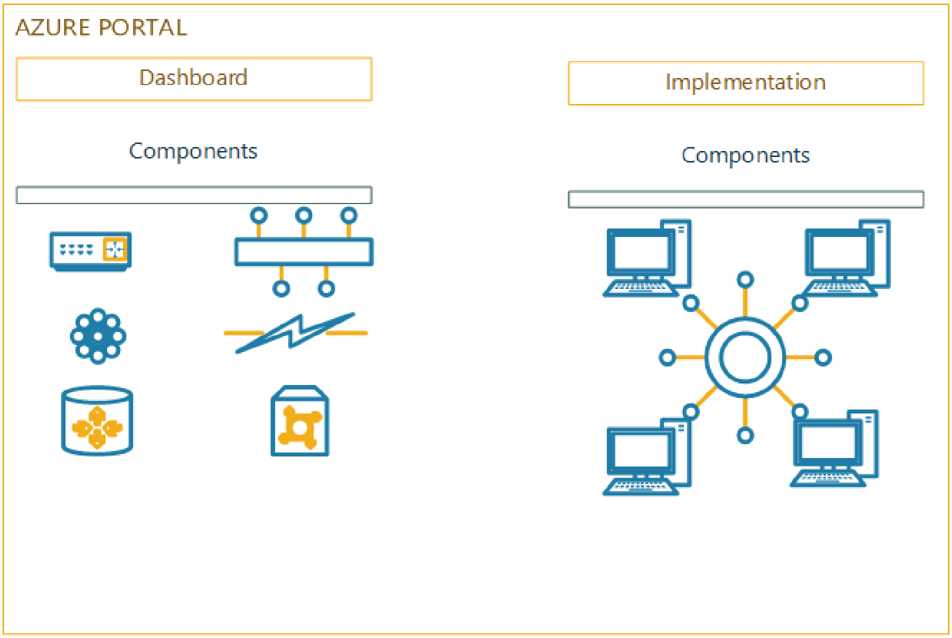

Figure 2. Azure Portal. Source: own work

When you are ready to run a web app on Azure, you visit the Azure portal and create a Web App resource. Creating a web app allocates a set of hosting resources in App Service, which you can use to host any web-based application that is supported by Azure, whether it be ASP.NET Core, Node.js, Java, Python, etc.

If you are deploying your app as code, many of the available runtime stacks are limited to one operating system or the other. After choosing a runtime stack, the toggle will indicate whether you have a choice of operating system. If your target runtime stack is available on both operating systems, select the one that you use to develop and test your application. If your application is packaged as a Docker image, choose the operating system on which your image is designed to run. Selecting Windows activates the Monitoring tab, where you have the option to enable Application Insights. Enabling this feature will configure your app to automatically send detailed performance telemetry to the Application Insights monitoring service without requiring any changes to your code. Application Insights can be used from Linux-hosted apps as well, but this turnkey, no-code option is only available on Windows. An App Service plan is a set of virtual server resources that run App Service apps.

A plan's size (sometimes referred to as its Sku or pricing tier) determines the performance characteristics of the virtual servers that run the apps assigned to the plan and the App Service features that those apps have access to. Every App Service web app you create must be set to a single App Service plan that runs it.

A single App Service plan can host multiple App Service web apps. In most cases, the number of apps you can run on a single plan will be limited by the performance characteristics of the apps and the resource limitations of the plan. App Service plans are the unit of billing for App Service. The size of each App Service plan in your subscription, in addition to the bandwidth resources used by the apps deployed to those plans, determines the price that you pay. The number of web apps deployed to your App Service plans does not affect your bill.

You can use any of the available Azure management tools to create an App Service plan. When you create a web app via the Azure portal, the wizard will help you to create a new plan at the same time if you do not already have one. The Azure Portal provides a wizard to create the solution with the following options: subscription, resource group, App name, publish, runtime stack, operating system, region and App service plan.

To create the WEB App, the steps below must be followed (Figure 2):
On the Azure portal menu or from the Home page, select Create a resource. Everything you create on Azure is a resource. The portal navigates you to the Marketplace page. From here, you can search for the resource you want to create or select one of the popular resources that people create in the Azure portal. Select Web > Web App to display the web app creation wizard. Fill out the wizard with the following values: Subscription (Concierge), Resource Group (Sandbox resource group), Name (enter a unique name), Publish (Code), Runtime stack (.NET Core 3.1 – LTS), Operating System (Linux), Region (chose the close to you), SKU and Size (F1). Then select Review and Create.

The Analysis

The investigation focused on the analysis of the results of school progress, including the final assessment of 21 students in a master's degree course in Information Technology.

We present two hypotheses to verify: first, whether or not there was a decrease in school results in the period of confinement caused by the Pandemic of COVID-19; second, check if there is any correlation between the evolution of the evaluation results between different curricular units, in the referred period. Both hypotheses considered referring to contexts of curriculum development through Cloud Computing systems.

The participants were the students of the first year of the master's degree. Both classes needed to use the same laboratories for curriculum development work.

One of the curricular units, Digital Systems Architecture, was run from the first semester, that is, from a pre-COVID-19 period. In this period the laboratory work was performed in person, that is in the laboratories of the university. The other curricular unit was Private Cloud Computing, already developed in the period of COVID-19, without face-to-face classes. It is important to consider that our university closed all face-to-face classes at the beginning of March – unlike many other countries, Portugal was one of the first countries to close schools and universities. The law of university autonomy in Portugal allows university directors to establish autonomous contingency plans, this is what happened at our institution.

In the confinement phase, students only took online classes, Google Classroom taught asynchronous classes, synchronous classes by Google Meet. In the case of this master's degree in computer science, the part of technological laboratories migrated entirely to Microsoft Azure. The laboratories developed by this class were all implemented in the area of computational infrastructure.

Analysing the evolution of the curricular unit of the first semester, Digital Systems Architecture, it was found that the average of the evaluations obtained was 16 values (on a scale from 0 to 20), it was also found that the standard deviation was 2.718 (Figure 3).

Laboratory work in the first semester was only completed in the last week of that semester. Not all students were able to finish the laboratories on time, mainly because they were working students and did not have the necessary time to dedicate to the face-to-face laboratories, as well as the fact that they had to miss classes for different reasons. The teacher had to extend classes for over a week for everyone to finish their work.

On the other hand, the Private Cloud Computing Course Unit, was taught in the confinement period of COVID-19, all took place through distance learning, that

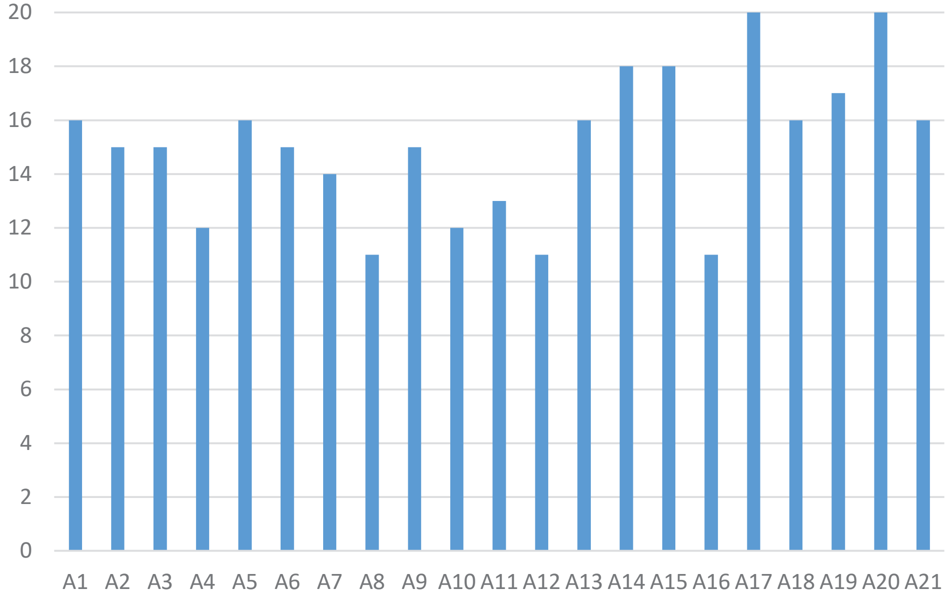

Figure 3. Result of the final evaluation of Private Cloud Computing – Pre-COVID-19. Source: own work

is, online, the curriculum development works in the laboratory were all installed in Microsoft Azure infrastructure.

The results obtained from the analysis to the evaluation and performance of the students were genuinely superior to what happened in the first semester, curricular unit.

The average of the final evaluation was 18 values, and the standard deviation was 1.590 (Figure 4). This is a significantly higher average – the fact that the general population and students were experiencing a period of enormous emotional stress, which did not occur in the first semester should be considered here) – yet in spite of this, they obtained higher ratings. Another aspect to note are the differences in standard deviation between the two scenarios. The standard deviation of the second-semester assessment is lower, which indicates that the general consolidation of objectives and learning, in addition to being more elevated, is more consistent and more students improved their performance.

The average difference between the results of the assessment obtained before confinement and the results obtained during the confinement period for similar curricular units (represented in Figures 3 and 4) is two values; in the confinement period, the students on average obtained a superior evaluation in two values, on an evaluative scale of zero to twenty values.

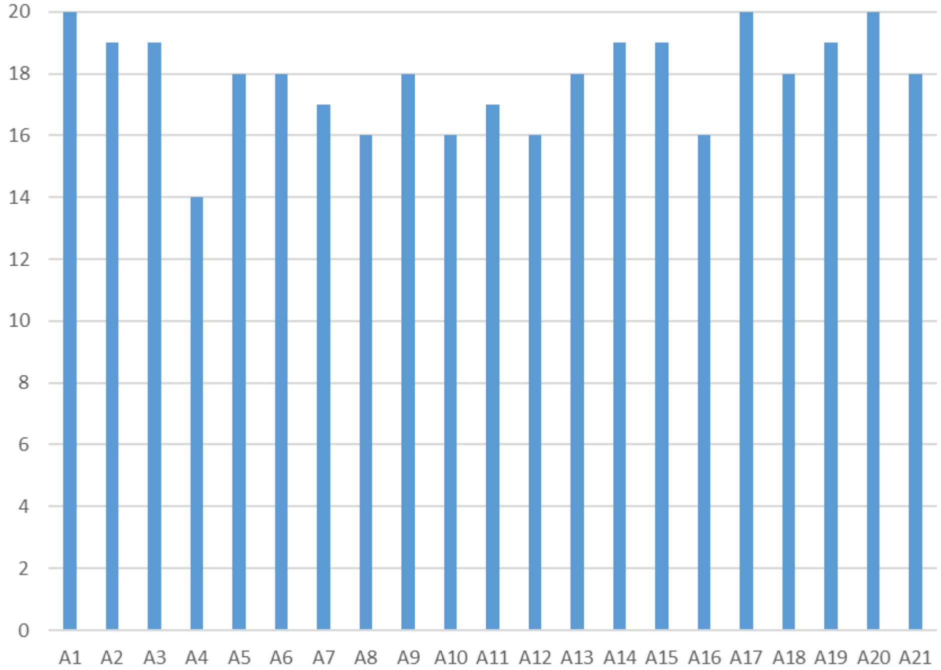

Figure 4. Result of the final evaluation of Digital Systems Architecture – confinement – COVID-19. Source: own work

It was also found that all students finished laboratory work before the deadlines with fewer doubts about the contents to be developed. This is not a quantitative aspect, but I consider it relevant. It is the opinion expressed by all students that the fact that the laboratories are in cloud computing allowed them to work at any time and repeat the tests and learning in a more straightforward way. They were able to review all the work easily. In technical terms, these labs also allowed students to reset any development errors as they were using virtual machines which, due to their characteristics, allow easy to create reverse images.

Advantages of Cloud-based E-Learning

We believe that there are many advantages in laboratory teaching based on Cloud Computing for some technical-scientific regions of Computer Sciences (Riahi, 2015), we also recognize that this model does not adapt so well to other areas.

However, some advantages of Cloud-based E-Learning should be summarized:

1. Low cost; e-learning computer users need not configure up for e-learning applications. They can cloud applications via PC, mobile phone, tablet with an Internet connection to run with minimal configuration (Maskarade, 2014).
2. Improved performance; since cloud-based applications for e-learning run with super-strength, super-source software is automatically updated. So, students always received updates (Viswanath, 2012).
3. Direct benefits for students; they can take online courses, take the exam online, received feedback about the coaches, and post projects and assignments online through their teachers (Patel, 2014).
4. Cybersecurity; cloud computing providers supply some major security benefits for individuals and companies that are using e-learning solutions (Mohammed, 2014). All the information is more secure.

We carried out the same type of study for two other curricular units in an area other than that of the Azure infrastructure. In this case, we compared two curricular units for the development of applications for mobile devices, one taught in the pre-COVID-19 period, the other taught at the peak of the COVID-19 pandemic in which students only had classes through online technologies, and in this case through Azure DevOps.

In the curricular unit Web and Multimedia Programming, taught before the COVID-19 period, always in face-to-face classes in the classroom and with the use of development software existing on students' computers or in the classroom, the results were average (Figure 5).

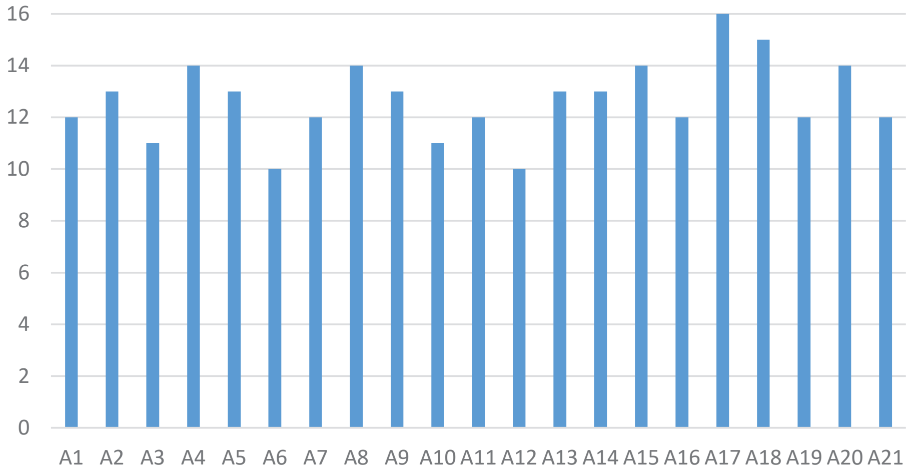

Figure 5. Result of the final evaluation of Web and Multimedia Programming – pre-COVID-19 (the vertical axis represents the assessments and the horizontal axis represents the students). Source: own work

On the other hand, in the curricular unit Development for Android Platforms, taught during the COVID-19 period, always in online classes, and still using the Azure DevOps ceiling, the students obtained, on average, much higher final evaluation results. We also received direct information from students who much appreciated this method of corrective development, as they could efficiently work at their own pace and easily correct errors without jeopardizing the work already done. We recognize that these curricular areas are easily adapted to the use of these technologies; however, the results in terms of the evaluation were outstanding in the generality of the pool of students analyzed. We can even consider the assessment as excellent (Figure 6).

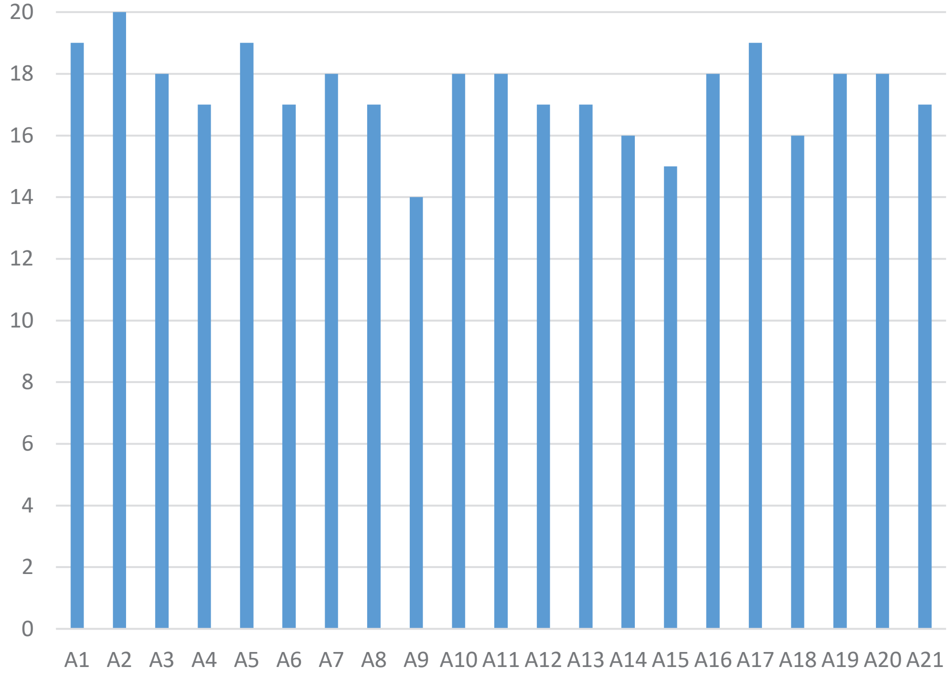

Figure 6. Result of the final evaluation of Development for Android Platforms – pre-COVID-19 (the vertical axis represents the assessments and the horizontal axis represents the students). Source: own work

In the analysis of the second hypothesis, that is, when comparing the results of well-differentiated curricular units, in which they used types of curriculum development tools other than the Azure infrastructure, the results obtained would also have a pattern during the period of confinement comparable to the pattern obtained with the results shown in Figure 1 and 2. In the teaching scenario which employed DevOps, the average difference between the results of the assessment obtained before confinement and the results obtained during the confinement period for similar curricular units and represented in Figures 5 and 6, is five values – in the period of the confinement, the students obtained on average a much higher evaluation of five more values. On an evaluative scale of zero to twenty values, the difference is even higher than that obtained in an identical scenario in the Azure infrastructure curriculum. On the other hand, the results of standard deviations demonstrate that students learned and had better evaluations.

The value of the standard deviation for the results of the assessment in the pre-confinement period and shown in Figure 5, is 1.5275; the standard deviation value for the evaluation results during the period of confinement, shown in Figure 6, is 1.3989. The latter value is lower, which means that the assessment obtained by students during the period of confinement, using Cloud Computing tools and taking classes exclusively online was much higher in the entire group of students studied.

Conclusion

In a new paradigm in which online education or combined teaching, both online and classroom, must be considered, having technologies which allow a correct teaching/learning process that achieves the objectives of obtaining skills through laboratories is a fundamental requirement.

The use of cloud computing technology allows today to give laboratory classes in the area of computer science entirely via online. As demonstrated, we can create an online laboratory on Azure and demonstrate/teach students how to develop a WEB application. On the other hand, students can use the same online infrastructure to build their project and put the acquired knowledge into practice. One can easily set up and provide on-demand access to preconfigured virtual machines (VMs) to support one’s scenarios. One can teach a class, train professionals, run a hackathon or a hands-on lab, and more. Simply define your needs, and the service will roll the lab out to your audience. Users access all their lab VMs from a single place. For this reason, cloud computing laboratories solve problems of not accessing laboratories where students must be in person. From the study carried out using a mainly quantitative method, comparing the school results of two curricular units that had the need to use computational laboratories, in which one had access to on-site laboratories and others had only access to laboratories in Cloud Computing, it was found that the results of evaluation, performance, and easiness in reaching the objectives was more positive in the use of the laboratories in Cloud Computing, showing that all this teaching activity was carried out under enormous social and psychological pressure, due to the context of COVID-19. Another conclusion that we can also draw is that the most favourable results obtained through the use of Cloud Computing platforms are identical in totally different curricular units. That is why there is a correlation between the good results obtained in the use of Cloud Computing for laboratories in Azure and the application development platform (DevOps). However, if a second wave of COVID-19 occurs, this study should be repeated this year to increase the validation of the results now obtained.

References

Armbrust, M. et al. (2009). Above the Clouds: A Berkeley View of Cloud Computing. Technical Report No. UCB/ EECS-2009-28, University of California, Berkley.

Brandao, P. (2019). Forensics and digital criminal investigation challenges in cloud computing and virtualization. *American Journal of Networks and Communications, 8*(1), 23–31.

Carlyle, H. (2016). HPC Cloud, em *Cloud Computing technology – IEEE*. CSIAC. Retrieved from https://www.csiac.org (accessed 30 July 2020).

Maskare, P.R., & Sulke, S.R. (2014). Review paper on e-learning sing cloud computing, *International Journal of Computer Science and Mobile Computing, 3*(5), 1281–1287.

Mell, T., & Gance, T. (2011). The NIST definition of Cloud Computing. NIST. Retrieved from https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf

Microsoft (2020). DevOps. Retrieved from https://docs.microsoft.com/en-us/learn/modules/host-a-web-app-with-azure-app-service/3-exercise-create-a-web-app-in-the-azure-portal?pivots=csharp

Mohammed, F. (2014). Cloud computing utilization for e-learning. *International Journal of Scientific & Technology Research, 3*(3), 385–390.

National Institute of Standards and Technology (2020). Cloud computing definition. Retrieved from https://www.nist.gov/publications/nist-definition-cloud-computing (accessed 30 July 2020)

Page, G. (2012). Cloud computing in *cloud computing*. Cisco Press. Retrieved from https://ptgmedia.pearsoncmg.com/images/9781587204340/samplepages/1587204347.pdf

Patel, M., & Chaube, A. R. (2014). Literature review of recent research on cloud computing in education. *International Journal of Research, 1*, 887–897.

Riahi, G. (2015). E-learning system based on cloud computing. *Procedia Computer Science, 62*, 352–359.

Soundarjan, G. (2015). Online data migration for automic provisioning of databases in dynamic content web servers." In *CASCON '05: Proceedings of the 2005 conference of the Centre for Advanced Studies on Collaborative research* (pp. 268–282). IBM Press. IBM.

Viswanath, D., Kusuma, S., & Gupta, S. K. (2012). Cloud computing issues and benefits modern education. *Global Journal of Computer Science and Technology & Ditributed, 12*(10). Retrieved from https://computerresearch.org/index.php/computer/article/view/508

Voorluys, J., Bromberg, J., & Buyya R. (2011). Introduction to cloud computing. In R. Buyya, J. Broberg, & A. Goscinski (Eds.), *Cloud Computing – Principles and Paradigms*. John Wiley & Dons, Hoboken. https://doi.org/10.1002/9780470940105.ch1

Pedro Ramos Brandao

Chmura obliczeniowa i kształcenie na odległość w informatyce

Streszczenie

Z niniejszej pracy wynika, że wykorzystanie laboratoriów do opracowywania zajęć dydaktycznych w zakresie technologii informatycznych wspomaganych wyłącznie technologią przetwarzania w chmurze nie obniża poziomu uczenia się i obiektów oceniania po stronie studentów. Ten scenariusz powstał z potrzeby przerwania bezpośrednich zajęć tradycyjnych w laboratoriach fizycznych z powodu pandemii COVID-19.

Słowa kluczowe: Cloud Computing, Azure, nauczanie na odległość, WEB App, Virtual Labs, DevOps.

Педро Рамос Брандао

Облачные вычисления и дистанционное обучение в информатике

Аннотация

Эта работа демонстрирует, что использование лабораторий для разработки учебных программ в области информационных технологий, исключительно поддерживаемых технологиями облачных вычислений, не снижает уровень объектов обучения и оценивания со стороны студентов. Этот сценарий возник из-за необходимости прервать традиционные занятия в физических лабораториях из-за пандемии COVID-19.

Ключевые слова: облачные вычисления, Azure, дистанционное обучение, веб-приложение, виртуальные лаборатории, DevOps.

Pedro Ramos Brandao

Computación en la nube y aprendizaje a distancia en informática

Resumen

Este trabajo demuestra que el uso de laboratorios para el desarrollo del trabajo curricular en el área de tecnología de la información apoyado exclusivamente en la tecnología de computación en la nube no disminuye el nivel de aprendizaje y objetos de evaluación por parte de los estudiantes. Este escenario surgió de la necesidad de interrumpir las clases presenciales en los laboratorios físicos debido a la Pandemia COVID-19.

Palabras clave: Computación en la nube, Azure, aprendizaje a distancia, Aplicación WEB, Laboratorios virtuales, DevOps.