

VM4LMS - Virtual machines on education

Victor Jesus Farinha Agrela
Instituto Superior Técnico / UL, Lisbon, Portugal

ABSTRACT

E-learning appeared as a new pedagogical technology supporting the methods of learning. It became a preferable learning method that suits the modern requirements and is tailored to the needs of current students. At the moment, there are many softwares, such as Learning Management Systems (LMSs), that can be implemented in the academic community to serve various purposes (such as administration, documentation, evaluation etc) of courses online. Similarly, virtual laboratories allow for the substitution of the physical labs, using software for the virtualization of environments and physical equipment, therefore making them a very valuable tool. A virtual machine (VM) can create a virtual hardware in the host computers to be used by a new operative system (OS). Up to date, there are no programs that allow the use of VMs through the LMS. Therefore, in this work, I developed a program that connects the virtual laboratories (that use VMs) with the LMSs. In the most common LMSs, it is possible to create external applications that connect and interact with the LMS. In this thesis I present the construction of a program that creates virtual laboratories on which the VMs are used. The labs are created by the teacher and accessed by the students through the LMS.

Keywords: Virtual Laboratories, LMS, LTI, Machine virtualization, OpenStack.

I. INTRODUCTION

The technological revolution has brought the society new ways of simplifying different aspects of life. One of such facilitation was the creation of computer's applications and platforms suitable for teaching and learning. The usage of technology in pedagogy has created a novel concept used around the world known as e-learning (electronic learning). The e-learning applications are now standardised and implemented into the university's systems, being used as a method of teaching and tracking the progress of students [32]. The use of computing programs as a method of teaching has been studied by different institutions and the results obtained are promising [6] [1].

In the past years, e-learning has been developed in different areas using an array of different techniques. One of the most common applications used by the educational institutions are the Learning Management System (LMS) or Course Management System (CMS). LMS is a web-based platform that functions by acquiring practical and theoretical components through the participation between students and teachers. LMS has the capacity to provide varied types of content such as videos, documents and programs and can also be used to conduct tests, quizzes or as a submission portal [18] [13].

LMS is built with the anticipation to be user friendly; it possesses self-intuitive navigation and interface that catches the attention of the content uploaded or created by the teacher [5]. LMS allows the student to work anytime and from any-place, supporting distance learning and allowing for the ease of accessing files. Technology as Learning Tools Interoperability (LTI) applications, provides tools that are created to interact with the LMS [24].

Another e-learning concept is the virtual laboratory. This is a type of lab are made to simulate real environments, where the student can practice remotely and with virtual simulations. One of the biggest use of these labs is in the field of networking. Where the networks are created virtually and composed of virtual equipment such as routers, switches, hubs, etc. On these networks are usually connected VMs. A VM emulates one computing system providing the same functionality as one physical computer. The emulation is run as one isolated system from the real machine, hence providing high security and isolation from outside. The software inside the VM can be previously installed and configured, and this allows the possibility of the laboratory to be easily customized. Another characteristic of the VM is the property of being multiplied, i.e., the same VM can be replicated and distributed in multiple users [12]. It is important to note that most of the virtualization softwares support an application programming interface (API). The API allows for the complete manipulation of the VM from an external or internal application.

The virtualization of these VM's can be performed through the usage of clouds and local virtualization technologies. It is a scalable system that allows a versatile manipulation over the VM, not only for the execution but also for the replication and maintenance. The usage of the clouds has many benefits, for example, the efficiency and effectiveness in the use of the computing equipment and the reduction of power consumption through adjustment of the equipment characteristic for each VM [4] [2].

The use of virtual laboratories as a method of teaching has been applied in different fields such as computer science, engineering, natural sciences and others [1], obtained promising results. In engineering, Caminero et al. [4] and Damiani et al. [8] provided a system based on templates that can be reused for the creation of virtual laboratories in the field of network services in operating systems. Guler et al. [11] described a project of virtual laboratories that has been tested in courses like computer networking, operating systems, database, programming and security. There are other types of laboratories, where the virtual and factual laboratories are combined, in the case of Liu and Orban [15], who provides a system in which the virtual machines interact with physical components.

A. Problem statement

Nowadays the resolution of exercises in virtual laboratories are seen as a good method for learning and improving skills. Not only can they save money on buying equipment and are easier accessed, but can simulate a range of possible problems [7].

The existing virtual laboratories were created for specific cases, working as an isolated software. The flexibility and adaptation of these programs are limited when used in other types of laboratories. In the field of the LMSs, there are no applications providing the connection between the LMS and the VM and the use of VMs lacks in the e-learning applications. Up to date, a software in which a user could create and manage VMs for laboratories through LMS does not exist.

B. Objectives

The objective of this thesis is, hence, the development of a tool that would allow the use of VMs through a LMS. This tool would allow the teachers for the creation of virtual laboratories composed by VMs. The laboratory should be then accessed and solved by the students. The application needs to create, erase and manage the VMs uploaded by the professors and delivered to the students. The evaluation system needs to obtain files from the VMs after the conclusion of the exercises. This tool should allow the students to execute virtual laboratories and work in different areas. The application needs to work in parallel to the LMS and to the cloud. The VMs are preserved by the system with the possibility to be downloaded or executed in the cloud. Each student needs to be considered as an individual student where every of them can access the VMs through their logging from the LMS.

C. Results

As a result, the final application obtained in this project should satisfy all the objectives discussed before. This program has been named VM4LMS. VM4LMS can be connected with any LMS that supports the LTI connection. The application is able to run the VMs locally (on the student PC) or in the cloud. The VMs are connected into a virtual network with access to the internet. For the evaluation system a timer was added for the execution of the labs. This timer is optional and configured by the professors when programming the laboratory. The current implementation allows the execution of local VM using VirtualBox or on the cloud supported by OpenStack and it was integrated into Canvas.

II. BACKGROUND

A. Virtual laboratories

There are many projects in which virtual laboratories are used, primarily by institutions such as universities, with the purpose of supporting the students with the learning process. Most of these institutions have implemented the labs in their system or as an external application. The virtual labs are mainly focused on programs that create experience simulations and can be used in different disciplines such as sciences,

engineering, computing and others [1]. The research on the existing virtual laboratories are going to be divided into two types of laboratories. The first type is the one that does not use virtual machines and the laboratory is based in a software. The second type involves labs that use virtual machines:

1) *Virtual laboratories without virtual machines.*: Woodfield et al. [33], present a study about a project called Virtual ChemLab. It is a simulated laboratory based on the reproduction of various chemistry experiments. Another chemistry lab is presented by Tetko et al. [28]. They have developed the Virtual Computational Chemistry Laboratory (VCCLAB), with the main purpose of data analysis and calculation. In physics Ding and Fang [9] presented a simulation program for diffraction grating where the students are able to configure the experiment parameters. Tlaczala et al. [29], presents an European project calls VccSse (Virtual Community Collaborating Space for Science Education). The program is focused on instruments virtualization and has an educational purpose in different areas of science. Another project was the TEAL-sim [10] developed by Massachusetts Institute of Technology (MIT). It is an open source software and has an objective to represent the electromagnetic phenomena.

Vargas et al. [30], Tawfik et al. [26], Sancristobal et al. [22], Terkowsky et al. [27] and Barrios et al. [3] have created their programs with a LMS connection. These connections are executed through a modification of the LMS or a simple application that can access the LMS files for information.

2) *Virtual laboratories with virtual machines.*: Caminero et al. [4] provide a study carried out over a system called TUTORES that has been developed by Universidad Nacional de Educación a Distancia (UNED). It aims to create and manage virtual laboratories. The system is focused in the cloud computing and virtualization principles. Another virtual laboratory was built by Damiani et al. [8]. It is based only in open source technologies and composed by a XEN platform and a Gentoo Linux distribution made by University of Milan. Each student has their own Linux VM with all the administrator privileges. The access to the lab is provided on a browser, with an accessing link that opens the ssh shell of the VM. Guler et al. [11] introduces a virtual laboratory that has been used for courses in computer networking, operating systems, database, programming and security in the University of Michigan - Flint. Li and Mohammed [14] and Stewart et al. [25] present virtual laboratories with the VMs installed in the student's computer. Both of them are focused in the virtual networks area.

B. Learning Management System

The learning management system (LMS) is a software application that provides with all the tools required by the teachers to support their teaching process. It allows for the existence of distance learning courses and home courses [18]. LMSs are used by many educational institutions around the world like schools and universities [21] [20] [31] [23]. The choice for using LMS is due to several advantages that this application offers. These tools are made for different objectives, usually their main purpose being to give the students

mechanisms of learning on the way to prepare and perform their learning activities [31]. These tools can provide the students with different learning resources like PowerPoint slides from lectures, documents, programs or videos [23]. LMS offers the possibility to create groups, enabling their users for the formation of discussion forums or chats. The teachers can also track the progress of students by looking at the automatically generated statistics or through the reports from laboratories, quizzes, projects, assignments or other kind of exams completed by the student and submitted on the platform[20]. It is also possible for the teachers to receive the feedback on their performance (which is completed by the students) [21] [31]. There are various LMSs on the market such as Moodle, Blackboard Learn, Schoology, Canvas.

C. Learning Tools Interoperability (LTI)

With the high utilization of LMSs in different areas by many institutions, the development of the LMS through the years became more innovated and flexible. This changes in the service are turning the system more manipulable with higher personalization by their users. One important feature on them is the capability to interact with external applications [24]. To keep the security of the LMS and to allow the use of external tools, the creation of the Learning Tools Interoperability(LTI) was essential. LTI is a tool that connects the LMS and the external tool when this one fulfils all protocols required [24].

D. Virtualization

Virtualization is a software technique that through the sharing of one hardware, allow the execution of many virtual machines in only one machine. The virtualization can be done in the cloud or in a local computer [16].

1) *Local virtualization*: A VM works as an independent computer, it uses the hardware components(memory, cpu, network interface, etc.) of its host. This connection between the VM and the real computer is executed by the VMM. There are different techniques of virtualization, according to the needs of each application virtualization done by each VMM. The techniques of virtualization are used according to the needs of each application, and each environment, there are mainly three techniques (full virtualization, paravirtualization and hardware assisted virtualization.) that are supported by the most of the VMM[17]. There are many available options in the market for a software that creates the virtual machine. The two most used VMMs are VirtualBox and VMware.

2) *Cloud Virtualization*: Recently, the cloud computing has been developed and is looked as the technology for the next generation. It has been used in academic and industrial purposes. There are mainly three levels of services offered for cloud computing [19], the software as a service (SaaS), platform as a service (PaaS) and infrastructure as a service (IaaS).

III. VM4LMS

VM4LMS is a system that integrates LMS and VM. It allow the professors to modify the VMs before these are publish

in the LMS. These VMs can be reusable in another virtual laboratories. The execution of the VMs is done in the local student's computer or in the cloud, depending on the teacher preferences. The students will access these labs through the LMS, when entering in the lab and with the help of the LTI, the LMS will subsequently send information such as user, discipline, etc. to the application which will redirect the student to the virtual laboratory. This application is able to create, launch, terminate and erase the VMS. It has the capability to extract files from the VMs, this files are selected by the teacher when doing the registration of the laboratory.

A. Users

VM4LMS requires three different types of users. Each user has a set of functionalities available:

- Admin - Has the responsibility to configure the LMS for the creation of user (students, professors) and to authorize the use of the VM4LMS a LTI application. The admin needs to configure the cloud authentication in the VM4LMS.
- Professors - Have the ability to create virtual laboratories, configures the VMs and have the access to all the files from the students resolutions. The professor has the control through the LMS on which students can access the laboratory. VM4LMS saves individually the VMs that is uploaded by each professor.
- Student - Are the one that solve the labs, they have the access to the VMs that have been configured by the professors. Each student is handled individually by VM4LMS.

B. VM4LMS Requirements

The following requirements are needed for the system to be capable of running virtual laboratories:

- 1) Compatible with the highest LMSs.
- 2) Allow the use of VM as tool for teaching and evaluation.
- 3) Allow the professor to fully configure the VM.
- 4) Allow students to run provided VMs on the local computer.
- 5) Allow students to launch VMs on cloud platforms.
- 6) Allow the exxtraction of files from the VM before they are terminated by the students.
- 7) allow the professor to define a deadline for the execution of the VM.
- 8) Guarantee that only students enrolled in the course can access such VM.
- 9) Reduce the burden necessary for students to launch the VMs.

C. Program architecture

The program's architecture can be divided into two scenarios for a sake of simplicity. One for the case where the VMs runs over the student's local computer (fig.3) and another one when the VMs runs in the cloud (fig.2). In both cases VM4LMS receives information from the LMS's LTI, and with that data redirects the student for the specific lab selected.

VM4LMS verifies if the person accessing the application is a student, professor or another entity and redirects to the specific laboratory.

When the VM runs in the local computer the VM4LMS takes the control over the VMM through the VMM controller. This VMM controller is another web framework that runs at the student's computer. It is connected with the VMM and with the VM4LMS, these connections are done through the APIs from all the sides. The VMM controller receives instructions from the VM4LMS, this instructions, such as installing a new image, switching off the VM or take files from the VM, are applied over the VMM. All the VM images for the laboratories and the files obtained from them are saved in the VM4LMS database. The VMM controller is the same for all the labs.

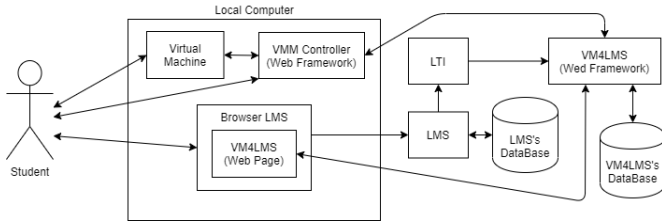


Figure 1. Local's laboratory

Using the cloud to run the VMs for the virtual laboratory makes the VM4LMS to work differently compared with the previous method. In this case the VM4LMS takes control over the VM with the help of the cloud's API. Most of all the clouds computing have their own database, these databases store all VM images needed for the virtual laboratories. For the files obtained from the VMs, these are stored in the VM4LMS database. When this laboratory is uploaded by the teacher, the VM4LMS connects to the cloud to save the VMs. In the case where the teacher wants to configure the VM when configuring the lab, VM4LMS uploads the VM, runs it and delivers it to the teacher through the web page.

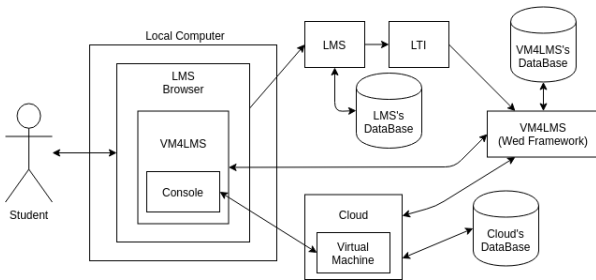


Figure 2. Cloud's laboratory

D. Execution steps

For the execution of a virtual laboratory, VM4LMS needs to be previously configured with the LMS and cloud. When configured the professor can use the VM4LMS for the creation of laboratories.

When a local laboratory is executed by the student (fig. 3), the student needs to run the VMM controller at his computer and introduce a code provide on the LMS browser. After that the student needs to download the VMs and introduce them on the VMM controller. With all theses steps the student is able to start the laboratory by pressing "Start laboratory" on the LMS browser. When finished the lab VMM controller takes the files and erases the VMs from the VirtualBox.

For the cloud laboratory (fig. 2), the student just need to run the machine at the LMS web page. VM4LMS launches the VMs in the cloud and delivered it to the student.

IV. VM4LMS IMPLEMENTATION

In this chapter it is explained how and which technologies are used in the development of the VM4LMS. Is taken into detail all the communications between programs and how they interact.

A. Support Technologies

In this section it is explained how every program and method was chosen for the construction of all the VM4LMS environment.

1) *Virtualization*: For the right operation of VM4LMS, the requirements for the VMM are focused on its ability to be controlled by the local web framework. One of the possible connections between them is through the use of the API that is integrated in the VMM. This VMM API needs to be easy to install in the student's computer and capable to be used in many OS..

For these reasons the option of using the VirtualBox was taken, because it is a simple program and not as much complex as the VMware. To use the Virtualbox's API, the student needs to install the VirtualBox Software Development Kit(SDK) in the computer. In order to get the files from the VMs, the VirtualBox Guest Additions needs to be installed in the guest OS, this can be done by the teacher when configuring the VM or by the student when solving the problem.

On the Cloud software virtualization, Openstack was selected due to its big amount of hypervisors and its big community compared to the OpenNebula. Important components such as Nova, Neutron and Keystone are used in VM4LMS for the implementation of the VMs. Their SDK are used for different purposes:

- To upload, erase, launch and stop VMs.
- To create networks and assign floating IP to the virtual machines. The floating IP turn the VM accessible from the public network, this association is important for the ssh connection between the VM4LMS and the VMs.
- To give the students a remote console access of the VMs.
- To store the VMs for the virtual laboratories.

The remote console access is obtained through the Nova's SDK. Nova enables the use of Virtual Network Computer(VNC), this VNC permit the sharing of a remote console or a remote desktop of the VMs. The access is obtained through a URL, this URL is open in the browser.

2) *VM4LMS*: VM4LMS is the main tool in this project, the program needs to be capable of:

- Managing the users individually (students and professors) that access the system and blocking the intruders.
- Supporting a big amount of student's requirements for their virtual laboratories.
- Supporting the LTI, this will help in the connection with the LMS.
- Interacting with the cloud operating system, in order to upload, deploy, manage and erase VMs from the cloud.
- Sending instructions and receiving information from the Web Framework that is running on the student local computer.
- Managing a Database, for storing files, VMs and others information for the labs.
- Getting files from the VMs when the lab is finished.

Therefore, the idea of building a RESTful API came up as a good solution to fulfill the requirements. There are many Web Frameworks available that can be used as solution, such as Bottle, Django and Flask. Django shows up as a large framework with many features built in it to help in the creation of complex web applications, is supported by a big community and it comes with an admin panel for the system administration. Flask is a small framework, very customizable, solid to scale, simple to use and has a big community. Bottle is another small framework, it is composed of only a single file module, its community is very small and it is used for simple cases. After this analyze, Flask was the choice taken for being simple to use and can perfectly satisfy the requirements.

For the Database used in VM4LMS, there are many options such as SQLAlchemy, peewee, Tortoise ORM, etc. In this project SQLAlchemy is used because it stands out due to its big community. SQLAlchemy is a Python SQL toolkit and a Object Relational Mapper(ORM). It gives the developers the full ability of the SQL. The VM4LMS's database is used for:

- Storing the material needed for the virtual laboratory, such as VMs and others.
- Storing the files obtained from the resolutions given by the students.
- Checking and registering new students or professors in the system.

The library Paramiko is used for the interaction of VM4LMS with the VMs running in the cloud. Paramiko allows the connections between two entities through the SSH protocol. This connection is encrypted and authenticated. The transaction of files are executed through another library called SCP. SCP uses the paramiko authentication to transmit the files under the scp protocol.

The Web Framework that is executed in the student's local computer needs to accomplish some requirements for the right execution of the virtual laboratory:

- Simple to use, small and can be use in any OS.
- Be able to communicate with the VM4LMS, receiving instructions and sending data.
- It needs to control the VMs that are running on the VMM through its API.

With all these parameters and with the previous discussion about the Web Frameworks. Bottle was the option since it is a single file which can be easily downloaded and installed by the students. It is easy to configure and is very customizable.

The interaction with VirtualBox is made with the library pyvbox2. With this library the local Web Framework connects with the Virtualbox's API. The API is used to install, run, stop, delete VMS and take files from it.

3) *LMS*: The LMS in this project is used for the creation of disciplines, registration of students/professors and the integration and configuration of the external LTI application VM4LMS. From the LMSs studied (Moodle, Blackboard Learn, Canvas, Schoology), Canvas was the option taken. It was chosen not only for its abilities on the course administration but also for its big development in the API and its interconnection capacity.

B. LMS integration technology

For the VM4LMS the integration with LMS is used for for different purposes. One of them is for the creation of users, courses and laboratories. The second one is the security, only registered users in the LMS can access the Lab, this access is verified by the LTI.

Every time the users access the laboratory, the LMS through the LTI sends information to the VM4LMS. From this information it is used three parameters:

- *user_id* - It is an individual code that differs the users.
- *lis_person_name_full* - It is the name of the user.
- *roles* - It describes the type of user that enters in the virtual lab.
- *resource_link_id* - This is an individual code that differs with the links. Each link is considered as an individual lab.

One of the biggest advantages of the LTI is the protection on the data exchange between the LMS and the application.

C. VM4LMS Backend

This section describes the functionalities of VM4LMS and how were implemented. The structure of the program can be divided into two parts, the database and the end points.

1) *VM4LMS Database*: The database of VM4LMS was built to satisfy all the requirements of the program and it is divided into different tables:

- *Student* - Saves the name and the user id as primary key(PK) of students, both of them provides by the LTI. This table store all the students that enters in VM4LMS.
- *Laboratory* - Is used to store lab parameters, such as the introduction text, time, professor_ID and the link_id provided by the LTI.
- *Local_Machine, Cloud_Machine* - Save all the machines needed for the labs (in the case of the cloud, it only is store the ID of the VM in the OpenStack). It also saves the users and passwords for every machine.
- *Laboratory, Local_Lab, Cloud_Lab* - Is used to store the labs, the PK of this table is the link_id provided by the LTI. Local_Lab and Cloud_Lab are attributes inheritance from the table Laboratory.

- CVM_Designated, LVM_Designated - Save which VMs are used for each lab.
- Cloud_Exercise, Local_Exercise - Are used to store the students that have done the labs.
- Active_CVM, Active_LVM - Store the VMs that are being used by the students.
- Files - Stores the paths of the files that are taken from each lab.
- File_Output - Stores the files obtained from the students.

2) *VM4LMS end points*: The end points are build to satisfy all the requirements of the system. Each end point has its own objective.

- /launch - The first end point used when accessing the lab. It returns the lab according to link accessed
- /downloadfile - This end point returns the files obtained from the resolution of the labs.
- /upload - It makes the Upload of the laboratory, saves the VM and the extra information.
- /getcode - Check the secret key written by the student and sent by the local Web Framework. This code is given when accessed the lab and it verifies if it is the student the one who is doing the lab.
- /install - It install and launch the VM image in the student's computer.
- /end - Makes a request to the local web framework to takes the files from the VM, turn it off and remove the VM.
- /launch_server - It launches the VM in the cloud and return the VNC console to the student.
- /test_server - This end point is used when the teacher runs the VM in the cloud when configuring the Laboratory
- /end_server - It takes the files and shut down the VM in the cloud.
- /clean_lab - It cleans everything that is related to that lab.

For the local Web Framework the end points are implemented for the communication with VM4LMS.

- /Install - Installs the VM image in the local computer.
- /VMon - Turns on the VM
- /VMoff - Turns off the VM
- /Copyfile - Copies the file from the VM
- /Remove - Removes the VM image from the VMM

D. Local execution Runtime

The execution of the local laboratory can be explained in different steps:

- 1) The students log in the LMS and access the lab through the web page of the course.
- 2) After that the LMS through the LTI send a request to the VM4LMS for the lab, the request returns a web page that gives the access to the local web framework, a secret code for the lab and the VM image with its username and password.
- 3) The student downloads the local web framework and the VM image.
- 4) The student inserts the secret code in the local web framework and the path of the VM image that was downloaded.

- 5) The local web framework sends a request to the VM4LMS with the code to verify if it is the student that runs the lab, if it matches it is showed a message in the console.
- 6) The student presses the button to start the laboratory which sends a request to the VM4LMS.
- 7) The VM4LMS sends a request to the local web framework to install the VM and to launch it after that, if any error happens it will be shown in the web page.
- 8) The local web framework installs and runs the VM. The steps of installation can be tracked through the console.
- 9) The student logs in the VM and does the lab.
- 10) When the student finishes or when the time ends, a request is sent to the VM4LMS to finish the laboratory.
- 11) The VM4LMS sends requests for every file from the VM.
- 12) The local web framework through the VMM's API gets the file and sends it to VM4LMS.
- 13) VM4LMS, after getting the files, sends a request to the local web framework to stop and delete the VMM and after that, it is shown on the web page that the lab is finished.

E. Cloud execution Runtime

For the cloud laboratory execution the procedure is:

- 1) The student logs in the LMS and access the lab through the web page of the course.
- 2) After that the LMS through the LTI sent a request to the VM4LMS for the lab, the request returns a web page that gives the access to the the lab web page.
- 3) The student presses the button to start the laboratory which sends a request to the VM4LMS.
- 4) The VM4LMS sends through the SDK a request to the cloud to launch the VM. After launching it gets the VNC console and returns the URL.
- 5) The student logs in the VM and do the lab.
- 6) When the student finishes or when the time ends, it is sent a request to the VM4LMS to finish the laboratory.
- 7) VM4LMS assigns a floating ip to the VM to make possible the connection through SSH and SCP.
- 8) With the SCP VM4LMS takes every file from the VM.
- 9) After getting the files, VM4LMS stops the VM and deletes it. After that it is shown on the web page that the labs its finish.

V. EVALUATION

In this chapter, it is discussed how the program accomplished the requirements previously described in the section III-B. Continuing with a demonstration of possibles laboratories applications and its limitations. Concluding with a obtained result of surveys answered by teachers and students.

A. Functional requirement Validation

In this section it is explained how VM4LMS satisfies every requirement previously described in the section III-B. Each requirement is explained individually about its integration in the system and its interface:

1) *Compatibility with the LMSs.*: VM4LMS is developed under the LTI standard technology. This technology allows VM4LMS to be used in all the LMSs that support it. After configuring the LTI in the LMS, every teacher is able to use the application. The configuration is done through three parameters: consumer key, shared key and the xml configuration file. All these parameters need to match either in the VM4LMS and in the LMS.

2) *VM as method of teaching/evaluation.*: The VMs are available through the laboratories. When creating a laboratory the teacher needs to file a group of parameters that complements the exercise. These parameter are divided in two sections, exercise and virtual machine. The section exercise is the same regardless to the type of lab:

- Text - Place that describe the lab, there can be written notes or instructions about the lab.
- Timer - Optional timer for the laboratories.

The section virtual machine allows the professor to choose between the local and cloud laboratory. The parameters for each lab differ depending on the option. For the local laboratory the parameters are:

- VM option - Possibility to upload a VM or to choose one from the database.
- VM name - VM's name to be save on the database.
- username - Username that is showed to the student to access the VM.
- password - Passwodt that is showed to the student to access the VM.

For the cloud laboratory it is needed:

- VM option - Possibility to upload a VM or to choose one from the database.
- Flavor - It defines the memory Ram, number of cpus and memory capacity that each machine will use.
- The disk format - It is needed for the selection of the hypervisor.
- VM name - VM's name to be save on the database.
- username - Username that is showed to the student to access the VM.
- password - Passwodt that is showed to the student to access the VM.
- Cloud-init(optional) - Code that allows the configuration of the VM when it is booted in the cloud.
- Output files path - Files that are taken from the VM after finishing the laboratory

3) *Total configuration of the VMs for the laboratory.*: In both types of the labs the teacher has the opportunity to upload a VM image previously configured. In the case of the cloud laboratory, the teacher has the opportunity to run it in the cloud and customize it in the web page.

4) *Capable to take files from the VM in the cloud or from the VMM*: Every time that a laboratory is finished, the VM4LMS takes the files that have been configured by the teacher on every machine. This method allows a better evaluation by the teachers over the students.

It is also important to referee that the students are also able to see download their files.

5) *Possibility to use a timer in the execution of the virtual laboratories.*: The set of the timer on the virtual laboratories are programmed when configuring the laboratories, this parameter is optional. The counter start every time that the machines are running.

6) *Executing the VMs in the cloud.*: Running the VMs in the cloud do not only overcharge the student's computer but also makes the laboratory more practical and easier to be executed. The cloud brings the capacity to run several machines for each student at the same time, allowing this, laboratories more personalized by the teachers.

7) *Using the student's local computer for the execution of the VMs.*: The ability to execute the virtual machines in the student's computer allows them to work on the VMs in places where there is a low access to the internet or in laboratories where it is needed to connect the VM with real equipment.

8) *Only the student registered on the course can enter the Virtual Laboratory*: The VM4LMS takes into the consideration the users who have accessed through the LMS. When accessed the lab VM4LMS takes the information from the LTI and returns the respective laboratory. Considering that only the students can access the page course in the LMS, then only they will be able to enter the lab.

VI. SURVEYS

In order to understand to understand the utility of the VM4LMS a set of questionnaires were made to students and professors.

A. Populations

After the conclusion of VM4LMS, questionnaires were answered by students and teachers in order to obtain an opinion about the concept of the thesis and the program. It was taken 25 answers, 23 from students and from 2 teachers. All surveys were obtained at the Instituto Superior Técnico.

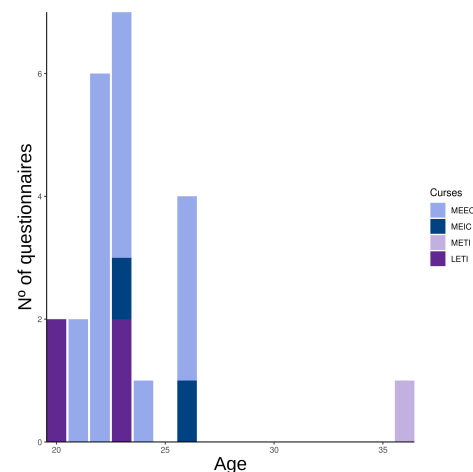


Figure 3. Students population

From these students, 19 have not finish the course and 4 are currently doing the PhD.

B. Questionnaires

The surveys were based on two standardized satisfaction questionnaires, the System Usability Scale (SUS) and the After Scenario Questionnaire (ASQ). The SUS is used to evaluate the usability of the program while the ASQ to assess how difficult a user perceived a task in a usability test.

The student's questionnaire is divided in four parts. The first three parts are composed by three tasks. For each task the student needed to answer one ASQ questionnaire. After concluding all three task with their respective ASQ questions a SUS questionnaire was answered. For the first task the student needed to turn on one VM through the LMS. For that it is necessary to logging in the LMS, access the course page, open the laboratory and then launch the machine. The second task was interrupting the exercise. The way to interrupt was up to the student, it could be done by multiple ways such as refreshing the web page, logging out and in the lms, closing the web browser, etc. For the third task the student needed to write a program, the classic hello world in c and conclude the exercise by submitting the lab.

The teacher questionnaire is divided in three parts, two tasks with a respective ASQ and the SUS at the end. The first task was creating a laboratory and launching a VM. While the second one was verifying the submitted files from the students on the hello world laboratory.

C. results

For the first task, the students got a good impression on how easy was opening the laboratory and how user-friendly was the interface. In some situations when launching the VM, it took more time than normal. The variation of this time was caused due to the OpenStack, Canvas LMS and the VM4LMS be running in one computer with low resources.

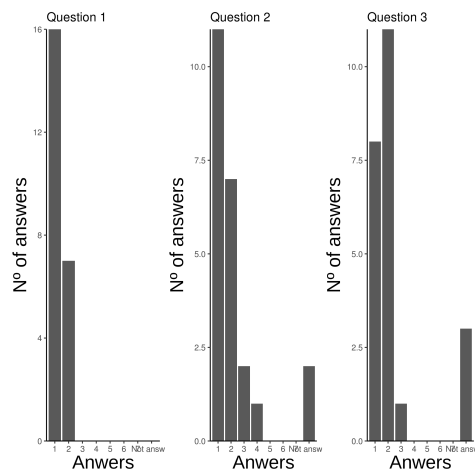


Figure 4. ASQ answers for the first task.

When interrupting the exercise many students found interesting the ability of the program for being able to keep the VM in the same condition as they had before the interruption. For other students the program should suspend the VMs even though knowing that the laboratory has an option for it.

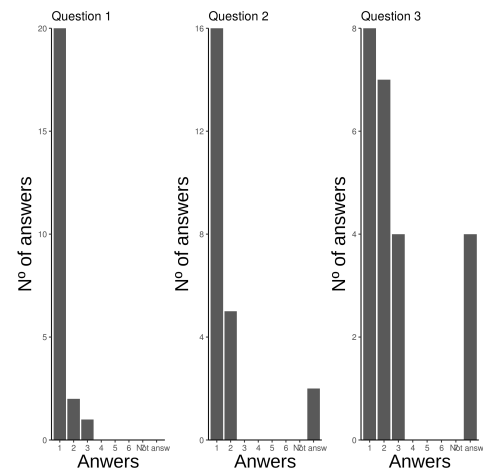


Figure 5. ASQ answers for the second task.

On the third part the student needed to write a program and submit the laboratory. The students did like the VM4LMS shows up which are files taken from the VM and gives the possibility to download the files.

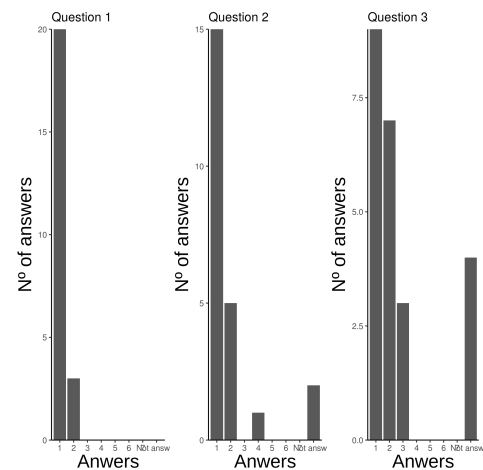


Figure 6. ASQ answers for the third task.

On all the three ASQ's questionnaires, some of the students answered Not applicable especially on the third task. Most of the time this answer was given due to lack of documentation/messages about how to use the program.

On the SUS questionnaires the students gave a punctuation about the program. They did like the idea, gave different opinions about how the VM4LMS could improve. Two students gave the idea that VM4LMS should give the ssh access to the VMs so the students could access them through another program instead of the browser. It is important to report that all the students knew what was a VM and have work with them. Overall the students liked the idea, giving the opinion that it could be used in different subjects where specific software or OS is required in certain labs. Another good opinion was that it would bring equality for all the students because everyone would use the same resources.

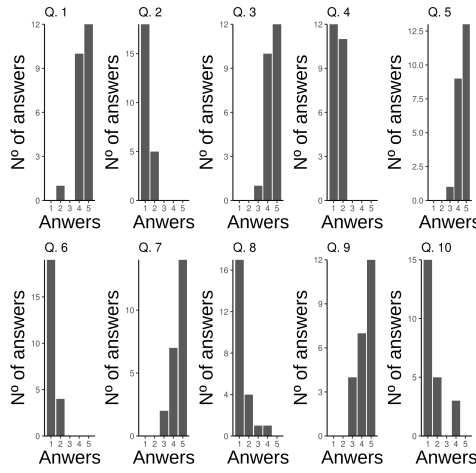


Figure 7. SUS answers

For the teachers surveys, their answers were positive. The tasks were done easily with the only bad point on the lack of help/information about how to fill the parameters. Both of the teachers had ideas about how they would use this labs on their subjects, giving different opinions about what type of exercises could be made with this system. Some of the ideas were configuring VMs in a network, analyzing a network traffic, diagnosis of network problems and creating client-server programs. Both of them would work with multiples VMs and would like to have possibility to configure the network.

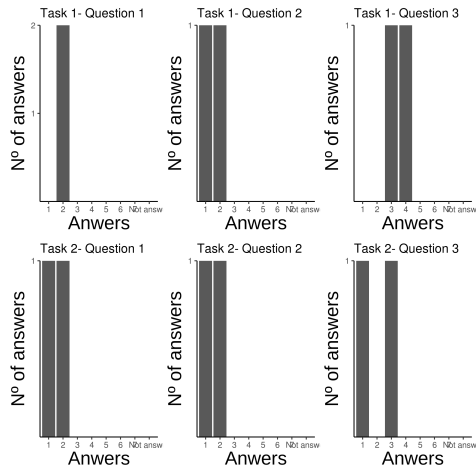


Figure 8. Teachers ASQ answers for first and second task.

VII. CONCLUSIONS

The use of VMs as a learning method is limited and one of the reasons is the lack of applications that would enable it access through the LMS. Therefore, there is a great interest in developing a program that could solve this problem by creating an application that could manage virtual laboratories (that uses

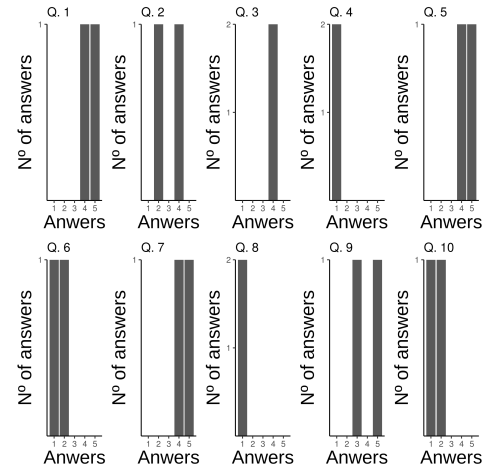


Figure 9. SUS answers

VMs), accessible through LMS. To do that, different technologies and methods were studied. In addition, different LMSs were analysed in terms of administration and compatibility with external applications. The research in the LMSs has found that the LTI is used in most of them and that it allows for the interaction with external entities. Other virtual laboratories that have already been made were also taken into consideration and their objectives, problems and the limitations were analyzed. Considering data gathered, and selecting the most appropriate technologies, VM4LMS was developed as a solution to the above mentioned problems.

VM4LMS is a web framework(Flask application) that can be easily connected to the LMSs that supports LTI. It allows the students, who are registered for the LMS's courses, for execution of virtual laboratories. It permits for execution of multiples VMs per laboratory and can take files from the VMs at the end of each laboratory. The execution of the VMs are taken either in the cloud or in the local student's computer. In the cloud virtualization, OpenStack is used as a cloud software. For the local computers method, the VirtualBox runs the VMs images meanwhile another web framework(Bottle application) is running in parallel taking the control over the VMs. In both types of labs the VMs are fully customized by the professor before publishing the lab. VM4LMS did satisfy with its principal objectives, the delivery of VMs to the students through the LMS. VM4LMS can be used for areas such as networking, programming, database, security and software.

A. Future Work

In order to make VM4LMS more powerful and practical, additional options of configuration need to be implemented in the system.

• Configuration of networks:

Openstack allows the use of different types of networks configuration, it allows the creation of subnets and routers. VirtualBox is also very customize in this area. These configurations are important for the networks laboratories. In the actual version of VM4LMS , for the local laboratory it is created a NAT network and in the

cloud a network with a subnet and a router connecting the network to the public network.

- **Methods of evaluation:**

Different methods of evaluation are possible to be added in the VM4LMS. Methods such as running commands on the VMs, uploading files or even saving clones of the VMs. All this methods are easy to be implemented since most of the connections between the VM4LMS and the VMs are already programmed.

- **Better user interface:**

The user interface could be improved, turning the lab more symple and understandable. With more helpful information about about how is the exercise done and what is the porpoise.

REFERENCES

- [1] Tareq Alkhaldi, Ilung Pranata, and Rukshan I Athauda. A review of contemporary virtual and remote laboratory implementations: observations and findings. *Journal of Computers in Education*, 3(3):329–351, 2016.
- [2] Michael Armbrust, Armando Fox, Rean Griffith, Anthony D Joseph, Randy Katz, Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica, et al. A view of cloud computing. *Communications of the ACM*, 53(4):50–58, 2010.
- [3] Arquimedes Barrios, Stifen Panche, Mauricio Duque, Victor H Grisales, Flavio Prieto, José L Villa, Philippe Chevrel, and Michael Canu. A multi-user remote academic laboratory system. *Computers & Education*, 62:111–122, 2013.
- [4] AC Caminero, R Hernandez, S Ros, LI Tobarra, A Robles-Gomez, E San Cristobal, M Tawfik, and M Castro. Obtaining university practical competences in engineering by means of virtualization and cloud computing technologies. In *Teaching, Assessment and Learning for Engineering (TALE), 2013 IEEE International Conference on*, pages 301–306. IEEE, 2013.
- [5] Miaoting Cheng and Allan Hoi Kau Yuen. Student continuance of learning management system use: A longitudinal exploration. *Computers & Education*, 120:241–253, 2018.
- [6] Rhonda Christensen. Effects of technology integration education on the attitudes of teachers and students. *Journal of Research on technology in Education*, 34(4):411–433, 2002.
- [7] Yevgeniya Daineko, Viktor Dmitriyev, and Madina Ipalakova. Using virtual laboratories in teaching natural sciences: An example of physics courses in university. *Computer Applications in Engineering Education*, 25(1):39–47, 2017.
- [8] Ernesto Damiani, Fulvio Frati, and Davide Rebecani. The open source virtual lab: a case study. In *Proceedings of the Workshop on Free and Open Source Learning Environments and Tools, FOSLET, Italy*, pages 5–12, 2006.
- [9] Yimin Ding and Hao Fang. Using a simulation laboratory to improve physics learning: a case exploratory learning of diffraction grating. In *2009 First International Workshop on Education Technology and Computer Science*, volume 3, pages 3–6. IEEE, 2009.
- [10] Fabio Ricardo dos Santos, Christian Guetl, Philip H Bailey, and V Judson Harward. Dynamic virtual environment for multiple physics experiments in higher education. In *IEEE EDUCON 2010 Conference*, pages 731–736. IEEE, 2010.
- [11] Evrim Guler, Suleyman Uludag, Murat Karakus, and Stephen W Turner. Virtualized lab infrastructure on a budget for various computing and engineering courses. In *Information Technology Based Higher Education and Training (ITHET), 2012 International Conference on*, pages 1–7. IEEE, 2012.
- [12] Wei Huang, Jiuxing Liu, Bulent Abali, and Dhabaleswar K Panda. A case for high performance computing with virtual machines. In *Proceedings of the 20th annual international conference on Supercomputing*, pages 125–134. ACM, 2006.
- [13] Gorgi Kakasevski, Martin Mihajlov, Sime Arsenovski, and Slavcho Chungurski. Evaluating usability in learning management system moodle. In *ITI*, volume 2008, page 30th, 2008.
- [14] Peng Li and Tijjani Mohammed. Integration of virtualization technology into network security laboratory. In *2008 38th Annual Frontiers in Education Conference*, pages S2A–7. IEEE, 2008.
- [15] Huan Liu and Dan Orban. Remote network labs: An on-demand network cloud for configuration testing. *ACM SIGCOMM Computer Communication Review*, 40(1):83–91, 2010.
- [16] Meriam Mahjoub, Afef Mdhaffar, Riadh Ben Halima, and Mohamed Jmaiel. A comparative study of the current cloud computing technologies and offers. In *2011 First International Symposium on Network Cloud Computing and Applications*, pages 131–134. IEEE, 2011.
- [17] David Marshall. Understanding full virtualization, paravirtualization, and hardware assist. *VMWare White Paper*, page 17, 2007.
- [18] Tanya J McGill and Jane E Klobas. A task–technology fit view of learning management system impact. *Computers & Education*, 52(2):496–508, 2009.
- [19] Bashir Mohammed and Mariam Kiran. Analysis of cloud test beds using opensource solutions. In *2015 3rd International Conference on Future Internet of Things and Cloud*, pages 195–203. IEEE, 2015.
- [20] Gali Naveh, Dorit Tubin, and Nava Pliskin. Student lms use and satisfaction in academic institutions: The organizational perspective. *The Internet and Higher Education*, 13(3):127–133, 2010.
- [21] Beth Rubin, Ron Fernandes, Maria D Avgerinou, and James Moore. The effect of learning management systems on student and faculty outcomes. *The Internet and Higher Education*, 13(1-2):82–83, 2010.
- [22] Elio Sancristobal, Manuel Castro, Sergio Martin, Mohamed Tawfik, Alberto Pesquera, Rosario Gil, Gabriel Díaz, and Juan Peire. Remote labs as learning services in the educational arena. In *2011 IEEE Global Engineering Education Conference (EDUCON)*, pages 1189–1194. IEEE, 2011.
- [23] Judith Schoonenboom. Using an adapted, task-level technology acceptance model to explain why instructors in higher education intend to use some learning management system tools more than others. *Computers & Education*, 71:247–256, 2014.
- [24] Charles Severance, Ted Hanss, and Joseph Hardin. Lms learning tools interoperability: Enabling a mash-up approach to teaching and learning tools. *Technology, Instruction, Cognition and Learning*, 7(3-4):245–262, 2010.
- [25] Kyle E Stewart, Jeffrey W Humphries, and Todd R Andel. Developing a virtualization platform for courses in networking, systems administration and cyber security education. In *Proceedings of the 2009 Spring Simulation Multiconference*, page 65. Society for Computer Simulation International, 2009.
- [26] Mohamed Tawfik, Elio Sancristobal, Sergio Martin, Rosario Gil, Gabriel Diaz, Antonio Colmenar, Juan Peire, Manuel Castro, Kristian Nilsson, Johan Zackrisson, et al. Virtual instrument systems in reality (visir) for remote wiring and measurement of electronic circuits on breadboard. *IEEE Transactions on learning technologies*, 6(1):60–72, 2012.
- [27] Claudius Terkowsky, Christian Pleul, Isa Jahnke, and A Erman Tekkaya. Tele-operated laboratories for online production engineering education-platform for e-learning and telemetric experimentation (petex). *International Journal of Online and Biomedical Engineering (iJOE)*, 7(S1):37–43, 2011.
- [28] Igor V Tetko, Johann Gasteiger, Roberto Todeschini, Andrea Mauri, David Livingstone, Peter Ertl, Vladimir A Palyulin, Eugene V Radchenko, Nikolay S Zefirov, Alexander S Makarenko, et al. Virtual computational chemistry laboratory–design and description. *Journal of computer-aided molecular design*, 19(6):453–463, 2005.
- [29] W Tlaczala, M Zaremba, A Zagorski, and G Gorghiu. Virtual physics laboratory for distance learning developed in the frame of the vccsse european project. In *Proceedings of the Fifth International Conference on Multimedia & ICT's in Education, Lisbon, Portugal*, volume 1, pages 467–71, 2009.
- [30] Hector Vargas, José Sánchez, Carlos Alberto Jara, Francisco A Candelas, Fernando Torres, and Sebastian Dormido. A network of automatic control web-based laboratories. *IEEE Transactions on learning technologies*, 4(3):197–208, 2010.
- [31] Jianfeng Wang, William J Doll, Xiaodong Deng, Kihyun Park, and Ma Ga Mark Yang. The impact of faculty perceived reconfigurability of learning management systems on effective teaching practices. *Computers & Education*, 61:146–157, 2013.
- [32] Elizabeth T Welsh, Connie R Wanberg, Kenneth G Brown, and Marcia J Simmering. E-learning: emerging uses, empirical results and future directions. *international Journal of Training and Development*, 7(4):245–258, 2003.
- [33] Brian F Woodfield, Merritt B Andrus, Tricia Andersen, Jordan Miller, Bryon Simmons, Richard Stanger, Gregory L Waddoups, Melissa S Moore, Richard Swan, Rob Allen, et al. The virtual chemlab project: A realistic and sophisticated simulation of organic synthesis and organic qualitative analysis. *Journal of Chemical Education*, 82(11):1728, 2005.