Østfold University College

.

Arrowhead Remote Virtual Physical Lab

Designing and Deploying Networking & Computing Infrastructure.

Master's Thesis

Author: Haroon Mohammad Sarwar

Department of Computer Science and Communication
Østfold University College
Halden, Norway
May 2022

Abstract

This thesis aims to design and implement a robust remotely accessible virtual lab for students to get a learning experience on ThingML, running with Arrowhead Framework. The lab is meant to allow students to control certain properties, i.e., the luminance and temperature of the room, using various peripherals physically integrated into the lab setup. In such a case, it is crucial to make sure that the computing and network infrastructure is designed and implemented in a way that it would handle hosted frameworks and applications virtually in an effective manner. Initially, during a systematic literature search, we reviewed various designs and terminologies being used to offer remote learning to students, this lays the foundation for our research to answer: Effective Designing of Computer & Networking Infrastructure in Virtual Lab and Learning Influence it offers. To evaluate the developed solution, we performed two presentation sessions with participants who were master's students and asked them to perform certain tasks, while we also gave a demo presentation in a webinar to attendees and answering them the questions they had in their minds after being exposed to the developed solution. By having such a proposed infrastructural design, students would get a lab learning experience that is easy for them to understand and relate to while also limiting the time and effort required to perform expected tasks from them. The proposed design would also be applicable in situations where lab learning is concerning other smart home or IIoT based systems.

Keywords: Remote lab, Virtualization, VPN, IoT, Network & Computing, Remote Desktop Connection, Eclipse Arrowhead, Openhab, ThingML.

Acknowledgements

My heartiest gratitude to my advisor and Professor Øystein Haugen for guidance and enabling setting up the remote systems for the experiments and supporting throughout the master's thesis with sustained enthusiasm and motivation. I would also like to thank my family and close friends for their love and support through my years at Østfold University College.

Contents

List of Figures

Chapter 1 Introduction

The emergence of the contagious coronavirus, in late 2020, has brought a change in the normalization of certain aspects of our life. Education has been one of the hard-hit sectors that have suffered a momentous disruption and have contributed to the rise of learning loss. According to the report of the World Bank, during the pandemic, there was a large number of educational institutions had to be closed which affected approximately 185 million students (Tyzhnia, 2021).

To tackle such disruption, the administration of educational institutions had to come up with plans to implement learning recovery methods. Due to the nature of abruptness, teachers and administration were not ready for such a transition, and emergency remote learning systems were looked up to for dealing with the cause. Various methods were considered, including social media, email, and meeting applications, in a hunt for the provision of proficient remote educational experience.

Gradually, several study findings suggested that during COVID it became almost a necessity to use online and remote learning (Ali, 2020). However, the limitation of remote learning is the absence of personal interaction between students and teachers. This factor is more challenging in scenarios where students are required to learn the use the system and technologies through hands-on experience in a physical lab.

Traditionally, before the COVID era, students would be present in the lab in the form of groups where the instructor usually guides them through the session by exposing them to systems and letting them be experimental on the systems as required. Since this was not possible anymore due to restrictions, there was a need to provide the lab learning through, in-demand, remote learning education. There have been studies that implemented remote lab solutions using different technologies, methods, and platforms. Although one common challenge for such solutions has been to come up with designing a Computing and Network Infrastructure in a way that would handle hosted framework and applications virtually in a proficient manner.

Furthermore, the choice of technology to be used in the remote lab solution and the productivity of learning the remote lab solution offers to the student are major factors in evaluating the robustness of the solution.

In this thesis, I set out to explore robust designing and implementation criteria for setting up the remotely accessible lab for students to get a learning experience on smart home and IIoT lab systems.

We designed and implemented a lab that would allow students to get learning exposure to a smart home lab system “The Room” involving ThingML, Openhabian, and Eclipse Arrowhead, while also allowing them to control certain properties, i.e., the luminance and temperature of the lab, using various peripherals physically integrated into the lab setup.

1.1 Motivation

In the master course “Modeling Cyber-Physical Systems” at Østfold University College, students learn to model and develop a cyber-physical system accessible through Arrowhead-based cloud application. Due to restrictions imposed on physical attendance in the university, the motivation built up to propose a solution for accessing the CPS system remotely where students can get lab learning experience regardless of wherever they are physically present.

Such a solution would not only facilitate students, but also other parties who would want to access the lab for the sake of training and getting an overview of the developed CPS system.

1.2 Research Question

The thesis aims to explore and propose various criteria for designing a robust remotely accessible lab in which students can get a learning experience and perform experiments and tasks on it. Then, an attempt to evaluate the influence the developed lab has on the learning productivity of the students.

1.2.1 Research Question 1

How to design an effective computing & networking infrastructure for students to remotely access Eclipse Arrowhead Lab and carry out their tasks? In terms of recognizing the effectiveness, mainly three of the following factors are to be considered:

- Operability: Operability is a measure of how well a system works when operating. A software system with good operability works well and is operable. A highly operable software system minimizes the time and effort needed for unplanned interventions (whether manual or automated). Basically, a system that is easily understandable and straightforward to use in terms of tasks to be done.

- Flexibility: The ability of a system to respond to a potential internal or external change. For instance, doing changes on ThingML and how the system changes its state accordingly. (The function of motion sensor, luminance, etc.)

- Accessibility: How well users can perceive, understand, navigate, and interact with that system.

Furthermore, another factor that is vital to determine the effectiveness of the system is the choice of hosting platform for the software system which suits the needs in terms of smooth accessibility for students while also being secure at the same time from unwanted intrusions and attacks. By hosting a platform, it is meant to decide which one of the platform technology, Virtualization or Containers, fits better the desired needs.

1.2.2 Research Question 2

To what extent the learning of students is influenced by the use of remote virtual lab? Evaluating the influence, in terms of learning, the developed lab has on students.

1.3 Thesis Structure

This section briefly summarizes the contents of each remaining chapter in the thesis.

Chapter 2: Background

This chapter presents the theoretical foundation of this thesis. It starts by introducing Arrowhead Frameworks' structure and functionality along with CPS system for the "The Room" project, and the use of technologies in terms of remote connectivity.

Chapter 3: Literature Review

This chapter gives an overview of the few current existing remote solutions designed and developed for the sake of providing remote learning.

Chapter 4: Design

This chapter introduces a proposed design of the working terminology for students to access the lab remotely with the use of various technologies and tools involved. Two designs are considered and studied. One where the CPS system is to be hosted using Virtualized Environment as a separate Operating System than the management one; while another one is with the use of Docking technology where the whole solution is hosted on one Operating System.

Chapter 5: Implementation

This chapter discusses how the remote lab was implemented on the university's premises and how it could be assessed remotely. Furthermore, the chapter then presents an overview of Raspberry Pi and gadgets integrated which are to be controlled through the modeling language in the "The Room" project.

Chapter 6: Evaluation

This chapter summarizes the results from Chapter 4 and Chapter 5. Also, the approach to answering the research questions.

Chapter 7: Discussion

This chapter discusses the findings from Chapter 5 and the results in Chapter 6. Furthermore, suggesting ideas and directions for future work.

Chapter 8: Conclusion

This chapter summarizes the thesis and presents the answers to the research questions.

Chapter 2 Background

This chapter is dedicated to the elaboration of technical information about the building blocks of the technology on which the thesis is built, the cyber-physical system “TheRoom", the Arrowhead Framework, ThingML, Virtualization, and VPN.

2.1 The Room

The Room is the running example and lab for the course CPS. We used The Room lab for testing out the solution developed in this thesis. “TheRoom" is a thermostat meant to control the temperature of a room. The user decides which temperature sensors and actuators the program can use and sets the comfort temperature. Based on the input from the sensors, “TheRoom" turns on and off. Furthermore, the system offer motion detection where a motion sensor detects a motion, and accordingly certain action can be taken, such as turning on the light or a lamp.

The Room consists of a temperature control and lighting control program where students work to access the remote lab. (Gopalakrishnan, 2022)

In our developed solution, we will be offering students an exposure to experiment in the lab through ThingML modeling language added on Eclipse IDE.

2.2 ThingML

ThingML is being used as modeling language to interact and manipulate The Room lab system. ThingML is a modeling language for designing and implantation of distributed reactive systems. It possesses well-proven software modeling constructs for such purposes:

- Statecharts and components (aligned with the UML) communicating through asynchronous message passing
- An imperative platform-independent action language
- Specific constructs targeted at IoT applications.

The ThingML language is supported by a set of tools, which include editors, transformations (e.g. export to UML), and an advanced multi-platform code generation framework, which supports multiple target programming languages (C, Java, Javascript) (TelluIoT, 2022).

2.3 Eclipse

Eclipse is an integrated development environment (IDE) used in computer programming. It contains a base workspace and an extensible plug-in system for customizing the environment (Wikipedia, Eclipse (software), 2022)

In the proposed lab solution, ThingML is installed on Eclipse as a plug-in allowing one to model and manipulate the smart home or IIoT systems.

2.4 Eclipse Arrowhead

The Eclipse Arrowhead aims to provide automation based on IIoT. The approach taken is that IoT's are abstracted to services, this enables IoT interoperability in-between almost any IoT's (Arrowhead, 2022).

Automation can be achieved based on the idea of local automation clouds. A local Arrowhead Framework cloud is comparable with the global cloud concept with added improvements and guarantees regarding:

- Real-time data handling.
- Data and system security.
- Automation system engineering.
- Scalability of automation systems.

2.5 Virtualization

In the computing world, the term Virtualization refers to the terminology of having a virtualized instance of a computer system running on top of the actual physical computer system. Generally, Virtualization refers to operating multiple operating systems on a computer system simultaneously. For the applications running on top of the virtualized machine, it appears as if they are on their dedicated machine, where the operating system, libraries, and other programs are unique to the user's virtualized system and unconnected to the host operating system which sits below it.

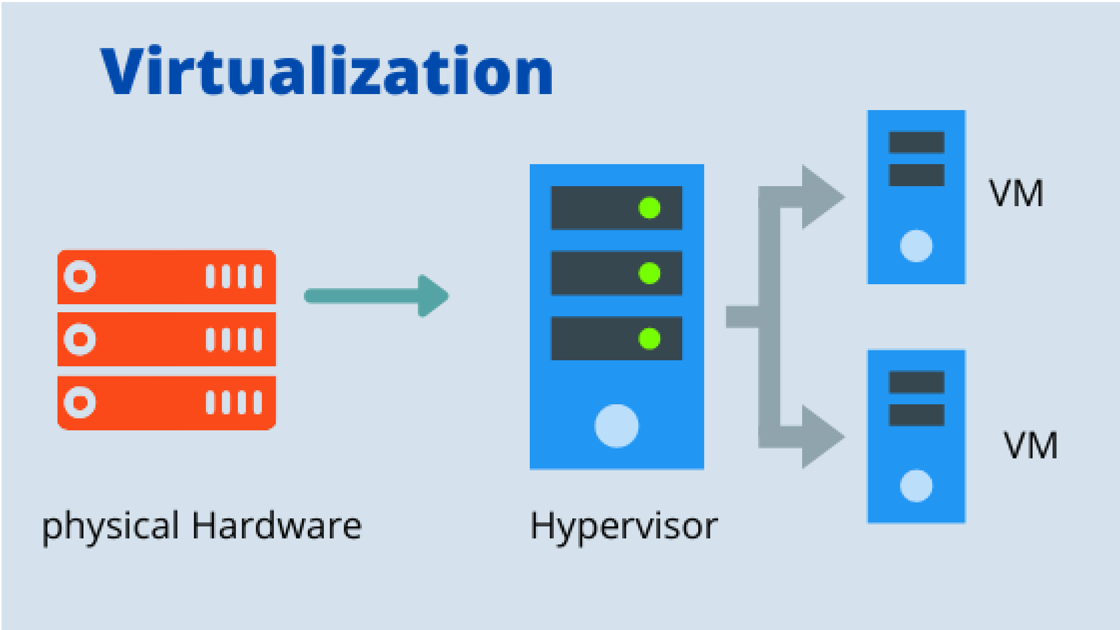


Figure 2.1 Overview of the Virtualization structure.

Figure 2.1 illustrates the concept of Virtualization where physical hardware is a physical computer machine, and a Hypervisor is a software or a tool on the machine's Operating System which allows the creation and operation of additional Operating Systems on it. The VMs are the virtual instances where software systems are installed on.

With the use of Virtualization, one can access an individual desktop by any device at any time and anywhere (Yan, 2011). It offers several advantages which are given below:

- *Simplifying the control and management process of software or a system hosted on Virtualization platform.* Since multiple and isolated operating systems are being operated on one physical machine, this makes it easier to troubleshoot and maintain the specific system software. For instance, if one system software requires the operating system to be rebooted, only that specific VM would be rebooted rather than rebooting the whole system running on the physical hardware machine which might have other software systems running too.

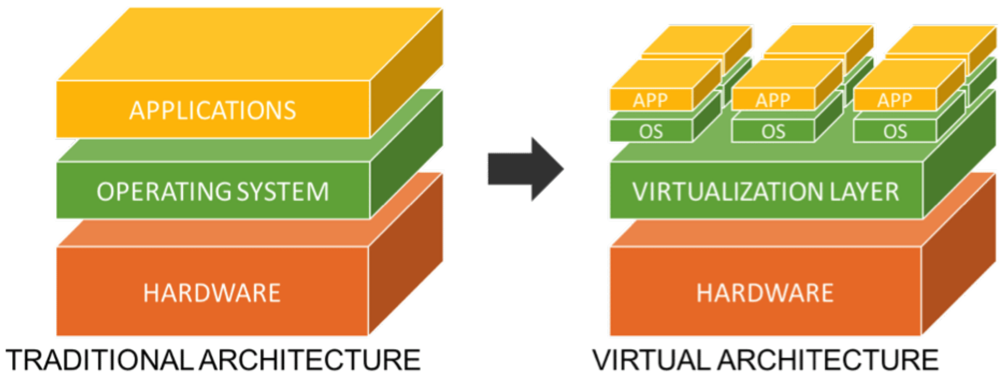


Figure 2.2 Traditional Computing Architecture vs Virtual Architecture

- *Saving the cost of more equipment and hardware.* In a virtualized environment, that single physical server transforms into many virtual machines. These virtual machines can have different and logically isolated operating systems and run different applications while still all being hosted on a single physical server. Such a scenario reduces the need of purchasing separate hardware for each isolated software system.

- *Minimizing downtime and greater disaster recovery response.* With a virtualized environment, it is easy to replicate or clone the virtual machine that's been affected. The recovery process would take mere minutes as opposed to the hours it would take to provision and set up a new physical server.

- *Faster provision of applications and resources.* The use of Virtualization allows organizations to respond faster to changing workplace needs and emerging opportunities. Virtualized desktops and applications can also be quickly and easily delivered to other branches of the organization.

The use of virtualization for the sake of student learning is not a new trend. A number of studies have been published in the domain of E-learning where such technology has been implemented to gain benefits as highlighted above. For instance, (Radhamani, 2014) in their paper compared groups on the usage of virtual labs against a control (traditional lab), results suggested improved performance in students using virtual labs. This is concluded after utilization of various features virtualization offers and using them to provide an enhanced guided educational method to the students.

While (Wazan A. S., 2021) pointed out the potential of virtualization in terms of its usage in labs. Much of the factors are the amount of time one can save in setting up the desired system for students with the ease of manageability and control. With time, an effort for the staff is accordingly reduced making it efficient. Furthermore, the cost is considerably reduced by deploying multiple systems on one physical machine and isolating them into multiple operating systems as required.

In traditional times, for a system to run on one operating system, there was supposed to be a separate hardware machine reserved for the purpose. For hardware functions such as input and output and memory allocation, that single operating system acted as an intermediary between programs and the computer hardware (Stallings, 2012)

However, with virtualization, one can host multiple operating systems which means, each isolated lab or a system can be operated on the same hardware machine as a separate VM. Furthermore, this terminology can be utilized to have various projects and task levels offered to students of the single system depending on the student's advancement in the course.

Considering the benefits and its uses in labs, the use of virtualization in our lab fits the need of deploying our CPS system on hardware with better utilization of it, while at the same time not compromising on user experience (UX) and range of learning experiences for students.

2.5.1 Oracle Virtualbox

Oracle Virtualbox is a virtualization software tool installed on a base operating system of the computer system to host virtual instances of a computer system on it (Oracle, 2022). It supports the creation and management of guest virtual machines running Windows or Linux. It offers a free virtual machine license for personal, educational, or evaluation use.

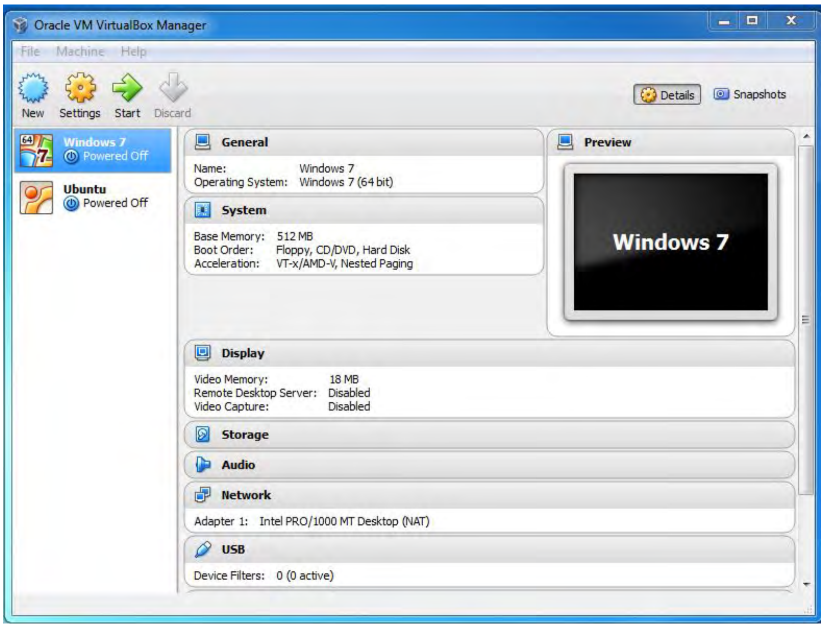


Figure 2.3 Screenshot of Virtual Box for deploying and managing the VMs

2.5.2 Hyper V

Just like Oracle Virtualbox, Hyper V is a virtualization role on Windows Server machines to deploy and manage virtual machines. Guest virtual machines with Windows or Linux OS can be hosted on Hyper V. A standalone Windows Hyper-V server is free to be sued with Command Line Interface rather than GUI based. (Wikipedia, Hyper V, 2022). Although the Hyper V can only be hosted on Windows platform, unlike Oracle VirtualBox which can be deployed on Windows as well as MAC OS.

2.6 Containers (Dockers)

A container is a standard unit of software that packages up code and all its dependencies, so the application runs quickly and reliably from one computing environment to another. A Docker container image is a lightweight, standalone, executable package of software that includes everything needed to run an application: code, runtime, system tools, system libraries, and settings. (Dockers, 2022)

2.7 Containers vs Virtual Machines

The operating system support of Virtual machines and Docker container is very different. From the figure given below, it is noticeable that each virtual machine has its guest operating system above the host operating system, which makes Docker containers lightweight in terms of resource utilization.

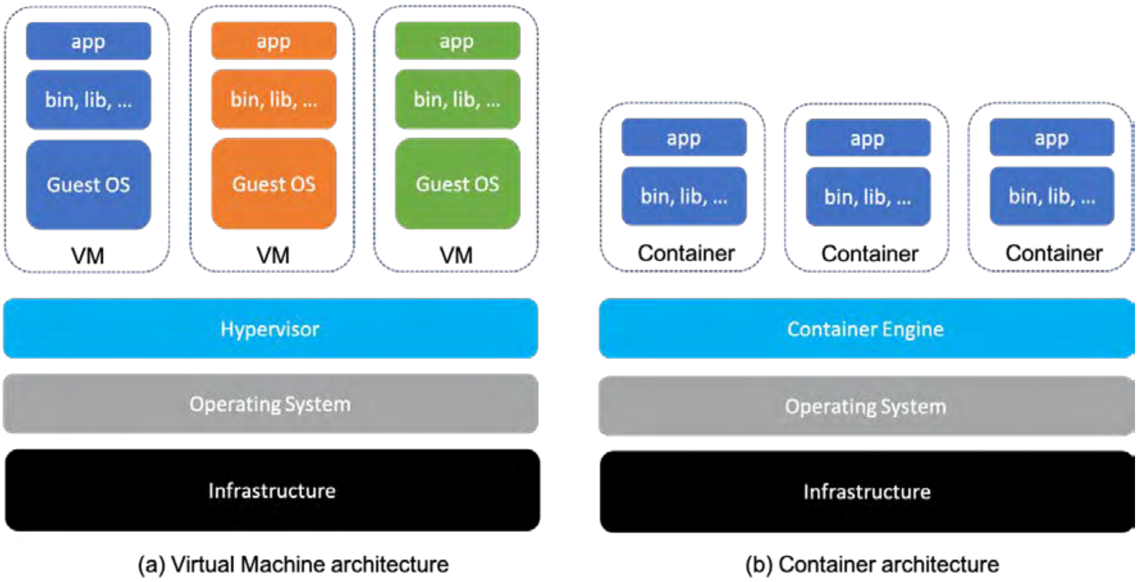


Figure 2.4 VM architecture vs Container architecture

However, having separate operating systems gives the flexibility in terms of security and isolating OSs where users can be offered their desktop with no access to the host physical machine's OS. This facilitates the Organization's administration team in terms of keeping the record of actions taken on certain OSs.

Although containers have benefits also and have been deployed as a core technology for the student's learning in the labs. The use of resources is further enhanced with containers, however, one noticeable difference where virtualization fits better with our needs is that it provides separate operating systems on the same hardware. The provision of a separate operating system allows ease of management for lab designers as well as gives a relatable experience to students as if they are using their personal computer or a Mac device. This makes the lab easy to use and access.

2.8 Virtual Private Network (VPN)

The purpose of a Virtual Private Network is to establish a secure, encrypted connection between computers on the internet as if they are residing on the same network. Such terminology allows users to communicate across public networks safely, hence eliminating the risks of intrusion and other network security hazards on the internet (Wikipedia, 2022).

2.8.1 Cisco AnyConnect Secure Mobility Client

Cisco AnyConnect Secure Mobility Client is a tool allowing remote users to have highly secure access to an organization's network from any device. With AnyConnect an organization can protect their network operations while giving users the flexibility of mobility while they access the services. (Cisco, 2022).

One added advantage of using this tool is the provision of a consistent experience to the users across different operating systems and hardware platforms. Users are offered the same experience whether they are using Windows or Mac OS.

2.8.2 OpenVPN Connect

OpenVPN Connect lets remote users sitting outside the organization's network premises connect with the network and access the network resources. It offers a quick and easy-to-use interface for users to do so (OpenVPN, 2022).

OpenVPN offers the Community Edition version which is a free and open-source version. Although OpenVPN is not being hosted as a VPN server on HIOF network premises, OpenVPN could be an alternative to Cisco AnyConnect Secure.

Chapter 3 Literature Review

This chapter presents an overview of the few current existing remote solutions designed and developed for the sake of providing remote learning. Additionally, presenting challenges found during the literature review. The aim here is to offer a reader an overview of existing remote solutions for learning purposes as well as the key points that are to be considered before designing and implementing such remote solutions.

The literature review was conducted as a mix of systematic literature review described in "*Guidelines for performing Systematic Literature Reviews in Software Engineering*" (Keele, 2007) and snowballing as described in "*Guidelines for snowballing in systematic literature studies and a replication in software engineering*" (Wohlin, 2014).

Following are the keywords which formed the basis of the search for a related literature study on what is to be explored. The keywords list is mostly targeting topics related to the use of technology in labs and various designs that exists to deploy remotely accessed labs.

- Virtual learning & education
- Virtualization & education sector
- Remotely accessed labs
- COVID & Pandemic & distance learning
- Containers/Dockers & Virtualization
- Security for remote labs

In a large part, the following mentioned sources were the target to shortlist the literature studies based on the techniques suggested in the (Keele, 2007) (Wohlin, 2014).

- Google Scholar (For the most part)
- IEEE Xplore
- Science Direct
- ACM Digital Library
- SpringerLink

3.1 Virtualization

The emergence of the contagious coronavirus has brought a change in the normalization of certain aspects of our life (Chakraborty I. & Maity, 2020). For the sake of the health security of students and professors, educational establishments are shifting their focus towards virtual learning for the provision of educational services in the scenarios of self-isolation and social distancing (Kurbakova, 2020).

The point worth noting here is that moving towards virtualization itself is not the desired goal, but the target is to offer a virtual platform, or a medium, which is efficient in terms of Operability, Flexibility, and Accessibility. Additionally, it is crucial to understand that careful planning is required before designing and implementing such virtualized learning platforms for students, as it is not obvious that following certain models or certain practices would turn out to be equally effective in one scene as the other one (Rashid S & Yadav, 2020).

3.2 Existing Solutions

In terms of relatable existing labs, (Celdrán, 2019) in his paper writes about the technologies of virtualization and dockers in the smart campus of the university which allow students to remotely control servo motor placed in the lab. The study also compares performance with three different types of setup. 1- Where VM is deployed in computer/server, 2- Deploying Docker on PC, 3- Deploying Docker on Raspberry Pi.

In (Broisin, 2017) , a remote laboratory Lab4CE is proposed, which uses virtualization along with additional technology tools to support students and instructors during practical activities. Lab4CE brings significant educational assets through a set of scaffolding tools and interfaces aimed at offering the best user experience possible.

3.3 Design Guidelines

Charles Border in his paper (Border, 2007) presents a design guideline for setting up a flexible virtual server environment for both distance and local students to conduct lab sessions for provisioning of similar experience to that of local labs. It is based on a cost-effective model including industry-standard technologies which could be used as a useful reference guide for a building design for our lab.

(Wazan, 2021) their paper analyzed different existing labs in terms of various criteria, such as setup time, feasibility, storage, and performance, their research suggests that it is the responsibility of the instructor and the lab architect to select the right virtualization design depending on the functional and technical requirements of the lab's activity and its expectation.

3.4 VPN and Remote Desktop Connection

In addition to the discussion of hosting platforms for labs, other necessary features shall be integrated into our lab to make sure of having reliable and secured lab (Deshmukh, 2017). Virtual Private Networks or VPNs allow organizations to extend access to their internal networks to external employees, or students in the case of a university, over standard Internet public networks (Sun, 2011).

Furthermore, Remote Desktop Protocol allows users to communicate with any of the remote servers. The experience of usage is like that of using a personal desktop. With a mix of VPN technology, one can avoid vulnerabilities and cyber-attacks. (Longzheng, 2004)

3.5 Summary

From the above-mentioned references, it can be assumed that the use of virtualization technology or container-based (docker) technology suits well with the desire of achieving proficient remote lab for students. Virtualization allows the separation of an operating system from the hardware on which it is working. It can allow a single physical resource to serve as multiple virtual resources and vice versa (Choudhary, 2016).

On the other hand, dockers allow virtualization within an operating system rather than on the hardware level. This makes them more lightweight and agile as a computing resource in comparison to virtual machines, who run on top of a hypervisor running on top of an operating system (Anderson, 2015). This means that in container-based virtualization users must use the same operating system to host users, which might not be a suitable scenario in our case (Singh, 2016).

Also, the use of VPN technology and RDP feature would allow users to connect seamlessly from anywhere in the world to the university's network premises while keeping the network safe from unintended access or usage.

Chapter 4 Design

To answer the research question presented in the section 1.2, we need to implement the cyber-physical system, The Room, on Eclipse Arrowhead which could be accessed remotely by students anywhere in the world and observe the changes in the lab after performing tasks and experiments on the system software.

This chapter presents the designing criteria for the proposed Arrowhead Remote Virtual Physical Lab and how it is to be accessible remotely. The physical lab is set up on the Halden campus of Østfold University College. The lab is then integrated with the university's existing network infrastructure and VPN service which sets up the foundation of remote access to the lab.

4.1 Conceptual Design

Initially, the single line diagram is presented to provide an overview of the whole solution, which sets up the base for achieving what is desired by the lab. Figure 4.1 illustrates how a student, with internet access, sitting at any remote place can get access to the lab using a VPN connection as a trustable user in the university's network. Once connected the students, after given an instruction, can access any of the desired Virtual Machines deployed for their access to the lab.

The Eclipse Arrowhead is hosted on Main Server, additionally, the Virtual Machines are deployed on Main Server which students access to manipulate The Room lab. The manipulation would be to control certain properties of the lab through ThingML language which is presented to students through Eclipse IDE installed on the Virtual Machine.

The Raspberry Pi, with OpenHabian hosted in it, serves the purpose of integration with peripherals in The Room lab. The peripherals in the lab are an actuator and sensor, which communicates via a Z-wave stick directly connected to Raspberry Pi. The Raspberry Pi is also connected with the Raspberry Pi Camera which streams the view of the lab to the students on the web dashboard presented to them upon logging in to the Virtual Machine. The SAN (Storage Area Network) could be additional storage for keeping the backup of Virtual Machines and offering logs for professors to observe the performance of students' work in the lab.

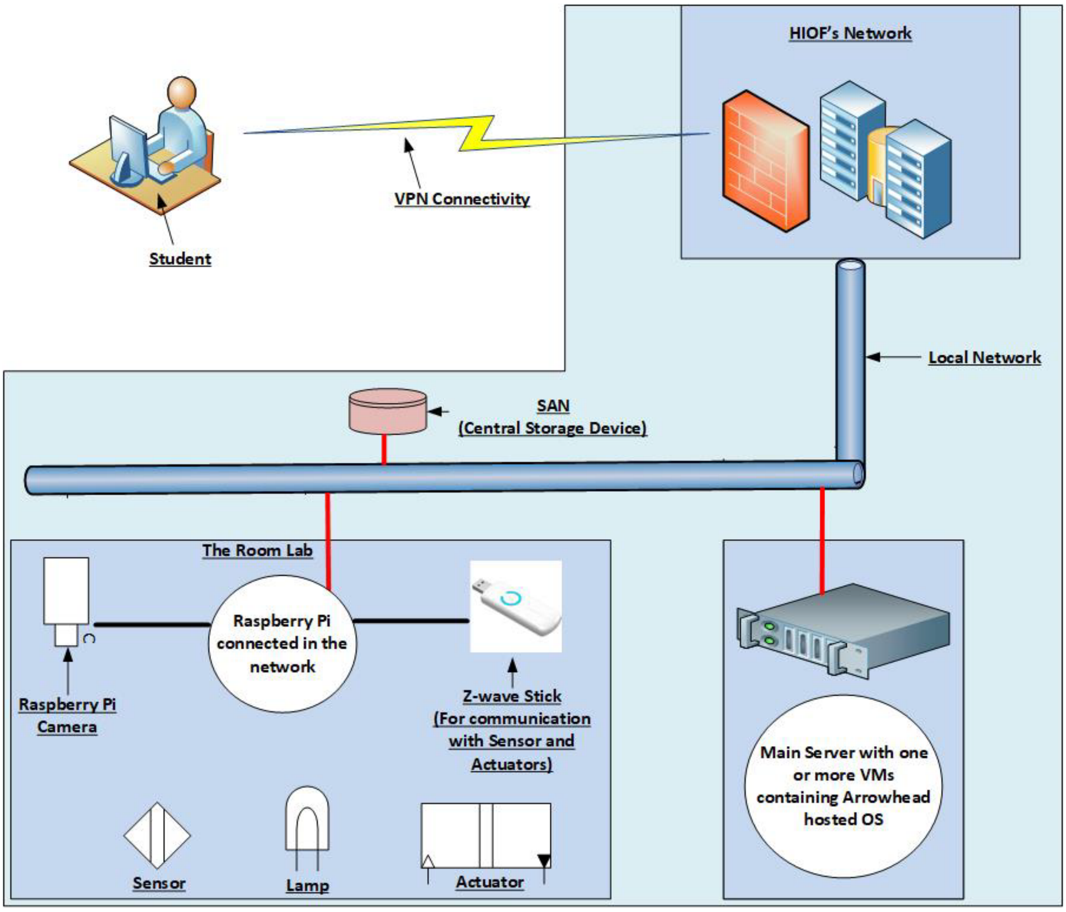


Figure 4.1 Single Line Diagram

One of the challenges recognized during the initial designing phase was the matter of figuring out a way to have secure access to the lab for students. Since the university already has an existing VPN service for students to access certain E-Learning resources, we considered utilizing the same for authorizing students to access the lab. We further utilized it to make use of it to offer remote lab learning education to the students.

As per figure 4.1, starting from the top left, the student is meant to be a user who might be present in any place of the world with their work machine and internet access and would be able to access the lab remotely. The VPN is a service functioned through Cisco AnyConnect to authorize the access of students to the lab. Once authorized on University's network, a student can access a specific VM which is hosting the CPS system.

On this base concept, a case scenario and storyboard were developed to take the conceptual design into the context of imagining how such computing and network infrastructure would allow a student to access the remote lab.

4.2 Scenario Case

In addition to conceptual design, we also came up with preparing a scenario case which assisted us in imagining the real-world scenario of how the students will be using the remote lab for learning purposes. Once the steps were defined, it helped us set a certain goal to achieve, and prepare ourselves for what must be done and what shall be expected.

4.2.1 Scenario Background

Tim Jhon is studying master's program at the University of Østfold, where he is enrolled in the course Cyber-Physical System. The course requires students to have a hands-on experience with Eclipse software for The Room project which involves UML coding to observe the state of the room having sensors and lamps in it.

Due to gathering restrictions imposed by the government, students are not allowed to be physically present at the university's lab to carry out above mentioned tasks. Therefore, the university is offering a Virtual Physical Lab for students to remotely login in that and get exposure to the working lab environment while sitting in their homes.

4.2.2 Scenario

For Tim to access the Virtual Lab, he requests a time slot from course professors to access the lab. Upon being granted access by the professor, he receives an email with the IP Address/DNS Hostname of the VM along with temporary credentials created for him. Tim then lands into HIOF's network through VPN using the credentials assigned to him by the university and connects to the provided VM through Remote Desktop Connection.

Once logged in, he is presented with opened windows of an Eclipse (Ready to code) and a real-time CCTV view of the room (with lamp, and sensors installed in it). The VM is designed and configured in a way where Tim is restricted to perform any action on the operating system apart from the actions allowed by the course professors. After Tim has finished working on the VM, the backup of the VM is saved/overwritten on the central network storage of the lab.

4.2.3 Storyboard

For further elaboration of the scenario, it was transformed into the storyboard as given in Appendix section.

4.3 The Lab

The Physical Lab was set up on the Halden campus of Østfold University College. Two primary reasons for this were that Lab would be physically secured from intrusion and well as utilize the existing network infrastructure of the university. Integrating the lab with the university's existing network meant we could use the authorizing terminology to authorize regulated access to the lab.

We set up the main server along with the peripheral components to meet the needs of the functioning The Room software system in a traditional lab environment way. Although, one addition to the peripheral devices is the camera which would be used as a monitoring device to provide a view of the lab.

An overview of the component's list, as well as the specifications of them, is given below:

4.3.1 Main Server

- HP Tower PC (4 cores processor, 1 CPU) with Windows Server OS.
- 16G RAM & 2X 500 GB HDD (Raid Configured)
- 1X NIC port
- Standard accessories along such as Mouse, Keyboard.

4.3.2 Components for "The Room"

- Lamp
- Motion Sensor
- Illumination Sensor
- Heater

- Actuator
- Camera
- Raspberry Pi

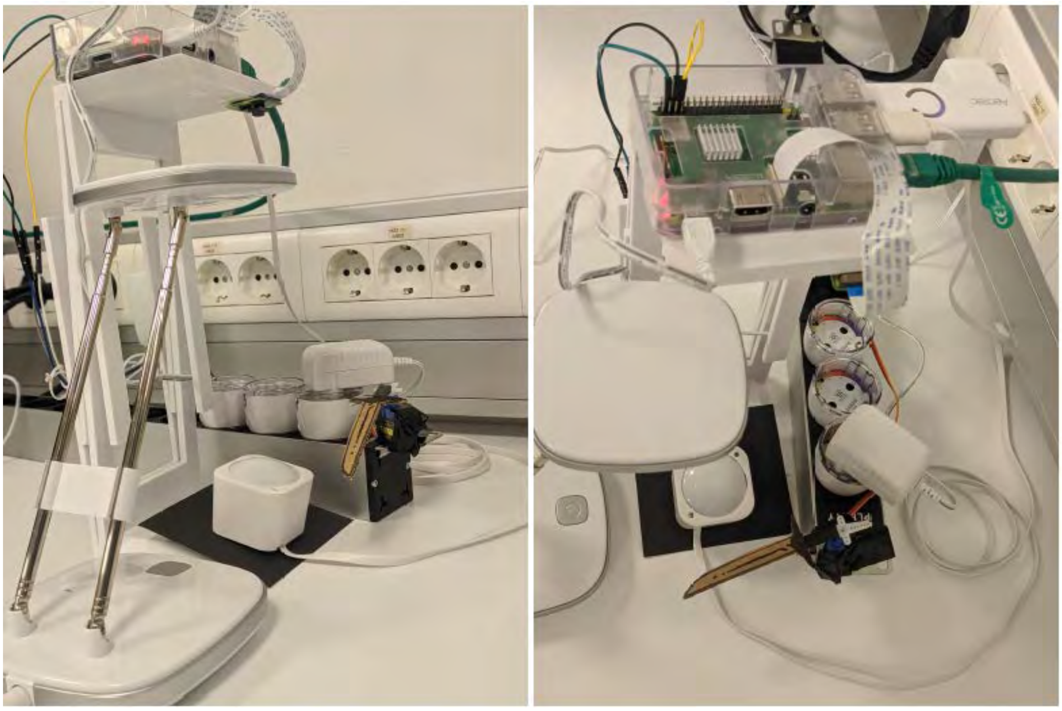


Figure 4.2 The Room Lab

4.3.3 Specifications of VM for "The Room" Lab

- VM deployed on Oracle Virtual Box (2 cores processor, 1 CPU) with Windows 10 OS.
- 8G RAM & 1X 300GB HDD
- 1X Virtual NIC port (For communication with network)

4.3.4 Software Tools

- Eclipse.
- Oracle Virtual Box
- Remote Desktop Connection
- Cisco AnyConnect

4.4 Virtual Machine

In the Main Server's OS, Oracle Virtual Box was installed for the creation of Virtual Machines. The Virtual Machines are meant to be hosting the Eclipse Arrowhead along with the web dashboard for students to observe the state of the room.

Furthermore, the decision of opting to use a container-based approach or virtualization environment seemed straightforward after doing the literature review and using that learning points in the conceptual design and imaginative scenarios. With the use of Virtual Machines, it would be possible to provide an isolated and easily operatable interface to work on. The factors mentioned in the research question 1 to achieve the goal of effectiveness seemed more possible with the creation of Virtual Machines. This is done by providing a student with a web dashboard and running Eclipse upon logging in to the VM, in this way the students don't have to wonder, set up, or open any desktop programs to do their tasks. Our goal is to let students have the ready-to-use interface to perform tasks they have been explained to perform, hence reducing their effort and time.

Also, an available dashboard to monitor the changes performed on Eclipse would allow them to instantly notice the change in the system's state. We implemented certain policies in the VM where distraction of different programs is avoided so that system is perceived and understood well in terms of tasks to be done. For instance, no other program would be operating on their desktop apart from the required ones which are Eclipse and web dashboard.

In terms of the creation of virtual machines, two alternatives were considered first. 1- To have separate Virtual Machines for each student, or 2- To have one virtual machine but separate user accounts for the students. Both options were feasible as per the need. However, in our case, we opted to try out both options for the sake of experimenting with the learning approach in different ways.

Chapter 5 Implementation

This chapter explains in more detail how Arrowhead Remote Virtual Physical Lab is implemented and is accessible remotely based on the designs presented in Chapter 4. First, the architecture of the network and applications are explained. Secondly, an overview of instructions to access the lab is presented to illustrate how the system is being used by a student.

5.1 Operating Systems

On the Main Server, we installed Windows Server operating system where the core services of Eclipse Arrowhead are deployed and are being executed. This operating system is strictly limited to access for developers and administrators.

The operating system on the main server is also installed with Oracle VirtualBox, as mentioned in the Background section, which allows the creation and development of the Virtual Machines. The reason for opting for Oracle VirtualBox is that it can be installed on Windows or MAC based OS. Although we are having Windows OS in our setup for the sake of flexibility in the future in case there is a need to migrate the created instances of VMs to a different server, then it would be compatible with different OS as well.

Initially, we deployed one Virtual Machine with an Operating System of Windows 10. This VM is meant to run Eclipse to communicate with Eclipse Arrowhead as a client. Since it is to be used by students, they must be presented with a desktop interface that allows them to work or access only the services required for the “The Room” lab. Therefore, restriction policies are implemented by us on the user profiles where they were restricted to do the following:

1- Access or/and login to the Main Server’s OS.
2- Change the Network Configuration of the Virtual Machine.
3- Accessing Control Panel, disables users to change any of the settings on their user profile existing on any of the Virtual Machine.
4- Accessing the file storage to modify the files needed to access Eclipse Arrowhead core services.

5- No other program or service would be running on the VM apart from the required ones. (In this case Eclipse, and a web dashboard for monitoring the state)

Furthermore, we had to make sure that users are only presented with the Eclipse software and web dashboard to limit them from the distraction of non-required services or programs.

5.2 Raspberry Pi

Raspberry Pi is a single-board computer made by the Raspberry Pi Foundation, a UK charity that aims to educate people in computing and create easier access to computing education. (Opensource, 2022)

The Raspberry Pi is generally renowned as a lightweight computer that runs Linux, but it also provides a set of GPIO (general purpose input/output) pins, allowing one to control electronic components for physical computing and explore the Internet of Things (IoT) or Industrial Internet of Things (IIoT).

The Raspberry PI is set up in our lab to integrate with peripheral devices using a wireless dongle (Z-wave stick connected to USB port) and NIC port for connection with the local network or main server on ethernet. The Raspberry Pi also has a slot of SD card on which Openhab (an open-source home automation software) is hosted.

The OpenHab allows the creation of a web dashboard that is to be visible to the users for monitoring the environment. The figure below shows how the web dashboard would appear to the user.

Figure 5.1 Raspberry Pi

Figure 5.2 Z-Wave stick connected on
Raspberry Pi for communication with peripherals

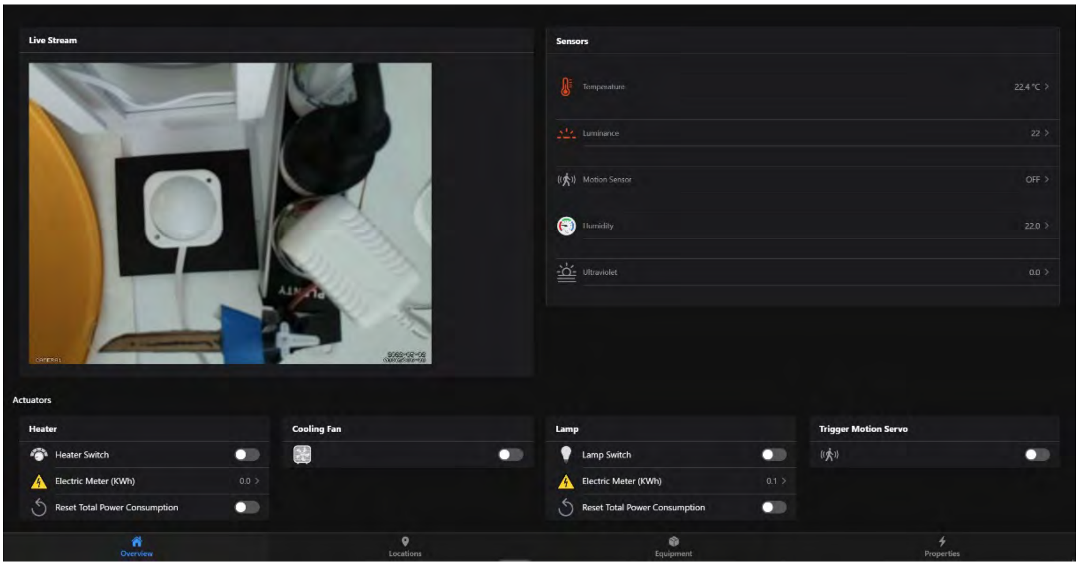


Figure 5.3 Openhabian Web Dashboard

5.3 Accessing the Lab

For a user to access a lab, a detailed user manual is prepared which provides an instruction for them to be able to connect and get familiar with the working terminology of the lab. This section presents an overview of the requirements for students to access the lab.

Firstly, only HIOF's student is allowed to access the lab, therefore they must have an active account that they use to access other E-Learning resources of the university. After that, a stable internet connection along with a PC (either MAC or Windows) with Remote Desktop Connection installed on it. For Windows PC, this is preinstalled, however, on MAC it has to be installed from the store and is offered free of charge.

After complying with initial requirements, the students must connect with University's network premises using the student account's username and password. Such authorization helps to reduce the risk of the lab being accessed by unauthorized users.

After a successful connection to HIOF's network. Users can remotely log in to the lab machine using the Remote Desktop Connection. From here on, after a while, the window for OpenHabian web dashboard and Eclipse (already set up with the base config of "The Room") would pop up on the screen.

We also prepared a demo video to illustrate the working terminology of the lab as seen below.

To reach the aim of having the use of lab as illustrated in Scenario section, we planned to distribute the user manuals to students on their email addresses before the lab session, or alternatively, it could be distributed by a professor on the course portal. The user manual would have required information on the IP Address/Hostname of the VM and credentials required for the access. The manual contains information on how to connect with the VM, and also what is to be expected to be shown on the desktop screen upon logging in, in our case that is Eclipse, and real-time view of the lab on the web dashboard through the camera connected on Raspberry Pi. The VM's OS is designed to restrict the distractions students might face by implementing policies mentioned in the Operating System section.

5.4 The Demo

We recorded a demo video showing how students would be accessing the developed Arrowhead Remote Virtual Physical Lab. Based on the user manual provided to them, they follow the instructions and guidelines mentioned to do so.

Upon logging in to the VM, they are presented with a web dashboard from monitoring The Room lab while Eclipse IDE for modelling it, if required.

Chapter 6 Evaluation

This chapter presents the results from the implementation explained in Chapter 5 of the designs proposed in Chapter 4. We performed three presentations of our system to participants with different aims for each presentation. For two of the presentation, the participants involved were 6 students who were currently enrolled in CPS course of the Master's program at Østfold University College. While one presentation was in Canada-Norway collaboration meetings for DevOps and SW Engineering.

6.1 Presentation 1

In the first presentation, the aim was to present our system to students and let them access the lab to get to know the look and feel of it. The target here was to get them familiar to have such a lab available, and what it does offer to students. Before the presentation, we prepared the user profile accounts for each student and prepared their desktop as mentioned in the section 5.2.

On the day of the presentation, a general brief was given to the students, and they were emailed out the individual user manuals before having a one-on-one session with them as they attempt to follow the required procedures to access the lab.

In this first presentation, it was expected for the users to log in to their assigned Virtual Machine account and witness what it offers. The expected behavior of the lab in this session upon logging on was to have OpenHabian web dashboard opened to them where they could get exposure to what it offers. The web dashboard was supposed to provide details on the current room temperature, and the toggle button to turn off and on the lamp. Also, toggling the motion of the sensor and monitoring the changes happening in the room through a camera view appearing on the dashboard. Additionally, they would also observe the Eclipse is automatically executed and ready to be coded, but they were not expected to do any changes to that in this session.

During the one-on-one session with the students, the students were offered support as they followed the user manual to successfully connect to the Virtual Machine without being physically present on the campus. The students accessed the web dashboard and used the features offered on it to observe the room's behavior as we desired. Approximately, on average one session with one student took 10 minutes.

6.1.1 Results of the Presentation 1

Although we didn't interview students or handed them over a questionnaire after the presentation to have their feedback on the usage of the system. The presentation itself meant a lot for us to observe how understandable it was for students to comprehend what was being elaborated to them and what the user manual was suggesting to them. Our goal was to let them do the required in the least amount of time without causing confusion and interruptions while following any of the steps given to them. We felt the presentation went as per our expectations and students were able to comprehend what we desired.

6.2 Presentation 2

During the second presentation, the aim was to present our system to participants involved in the Canada-Norway collaboration meeting for DevOps and SW engineering. The aim here was to give a brief overview of what we have developed and provide them the demo as we use the system and features it offers. The system acted as per the desired results during the demo where we showed the participants how peripherals can be controlled through the web dashboard and how the changes can be monitored at the same time. Although one minor unexpected behavior of the lab was observed during this presentation where the continuous toggling of the motion sensor caused the motion object to slip out from its original position.

6.2.1 Results of the Presentation 2

Apart from one unexpected behavior during the demo, the lab worked in a way that was expected from it. We learned that the components installed inside the lab may respond uncharacteristically upon rapid use of them, and to tackle such a situation we must perform more in-depth tests before a presentation or setting the lab for the operational use for learning sake.

6.3 Presentation 3

In this third presentation, the aim was to present our system to students again, but this time they were asked to import their projects to the Eclipse and perform the necessary coding to control and monitor The Room's behavior through a web dashboard. Before letting them have access to the virtual machines, a brief demo was given to them where changing certain parameters on the Eclipse project would result in the changes being reflected on the web dashboard.

Unfortunately, the presentation and experiment didn't go accordingly to the expectations. Firstly, we planned to assign 20 minutes for one group with two students in it to do the required, this turned out to be a short duration where students had to first import the project and then perform the required changes on the Eclipse itself. Apart from time limitation being a hurdle, we noticed students facing difficulty to execute what is expected from them on the first attempt.

6.3.1 Results of the Presentation 3

The foremost thing learned from this experiment was how important it is to spare enough time for students to perform the required operation. Apart from time, it would be wise to make students have hands-on experience with the programs they would be used to in the remote virtual lab environment. The reason for this is that they would be familiar with the basic use of the programs in advance prior to using certain features first time in lab class as time is usually limited during the sessions.

Also, it is important to be patient and give them enough exposure to the lab before expecting them to perform certain actions on the lab systems. The practice and exposure would give them the required confidence to perform given tasks in time-sensitive situations.

Chapter 7 Discussion

This chapter starts with a discussion on the design strategies to achieve a meaningful and relevant learning experience for students in the lab which is accessible remotely over the internet anywhere in the world based on the work done in Chapter 4 and Chapter 5. Followed by a discussion on how implementing such a remote lab solution influences the learning experience of students based on the evaluation done in Chapter 6.

This chapter is concluded with a section presenting future work.

7.1 Designing criteria

To tackle research question 1, we designed computing and network infrastructure aimed at providing a meaningful and relevant learning experience to students as they perform tasks in the lab by sitting anywhere in the world. Three main factors were considered for the cause which are further discussed below:

7.1.1 Operability

The use of Oracle Virtual Box tool on the main server allowed the creation of Virtual Machines serving the purpose of hosting system software of the remote lab. The Virtual Machine offers students with Windows Desktop experience which seems to offer a relatable experience in a sense of how to use and navigate the operating system.

Furthermore, by configuring the Virtual Machine's OS for students in a way where Eclipse and web dashboard is popped up upon logging in allowed minimal effort for students to look for and execute required programs related to the system software. Overall requiring minimal effort and reducing time for them to do their learning tasks, gaining easy operability in terms of the learning experience.

7.1.2 Flexibility

This factor is defined in terms of the ability of the system to provide a mechanism for students to monitor the changes in the state of the system after performing a variety of tasks on Eclipse. With the use Raspberry Pi, offering the connection to the system's gadgets and components over the network, which is accessible to Virtual Machine on the network offered a view of the lab environment on the web dashboard that could be seen and observed instantly as one perform manipulation on The Room project using Eclipse. The web dashboard also offered toggle buttons to manipulate the environment and observe the changes instantly.

Such instant observation of state change represents the ability of the remote lab to respond to an internal or external change in an agile way.

7.1.3 Accessibility

In an attempt to make the developed remote lab perceivable, easy to understand, and navigate, we opted to limit the distraction by removing or restricting irrelevant programs and services. Since the students were only offered two program windows on their desktop, which are web dashboard and Eclipse, this made their interaction and attention pointed at only what is desired to work with.

Such terminology also allowed control of misuse of developed remote solutions. Also, the features offered by Oracle Virtual Box enhanced the possibilities of a separate desktop experience for students on one Virtual Machine or separate Virtual Machines made for different system software.

7.1.4 Hosting Platform

The options pondered upon for suitable hosting platform options in this paper are Virtualization and Containers. In the design phase, the main benefits of both technologies are carefully studied and compared. The choice to opt for Virtualization is based on achieving the desired three factors mentioned in this section above.

The ability to have more than one Virtual Machine with its OS on one hardware machine allows for offering various isolated lab setups without the need of having a separate server for those labs. Also, this makes them independent from one system software to another on the same server, also independent for a student to another student one working on the same system software.

Such independency allows easier control and management of system software for the developers. And accordingly, new virtual machines can be created or updated as per the demand while the old ones can be removed as needed, making the utilization of hardware less costly and scalable.

Although in our case there is only one setup of the physical lab to work with, with the proposed design it is possible to have more than one setup of physical lab on the campus and let more than one student access and experiment on these labs at the same time. With the limitation of one setup of the lab meant that we had to restrict the access to the lab to one student or one group of students at a time.

7.2 Influence of learning

For research question 2, we presented the developed remote lab solution to participants three times, with each presentation aimed at a different purpose. Two of those presentations were to the master's students studying CPS course while one was in the Canada-Norway collaboration meetings for DevOps and SW Engineering.

In our first presentation to the students, where they were given a user manual with the instructions to access the lab and get an overview of what it offers. Although it wasn't expected from them to do any task on Eclipse in this specific presentation, we aimed to observe how comfortable the participants are with accessing the lab and seeing the working terminology of it.

Ultimately, witnessing them doing what is expected from them with minimal confusion and difficulty helped us to assume that it is being comprehended well enough. This presentation helped us to determine the positive influencing outcomes the lab could offer to the participants.

Although the second presentation is largely focused on providing the demo to the participants and having their feedback on the system, while at the same time using the opportunity to observe the functionality of the developed lab in a situation that might be like that of a real lab teaching situation. This presentation helped us realize the fact that components might act uncharacteristically upon rapid use and there might be a need for an operator to be present physically in the lab during the remote teaching session for support in case such unexpected instances occur. However, more in-depth operational tests before the session could be an alternative.

During our third presentation, we analyzed the importance of sparing enough time for students to perform what is desired from them as per the task assigned to them. However, that could be seen as a session planning limitation rather than a system limitation. The system allowed students to seamlessly access the Virtual Machines and they were set to start working on their tasks as soon as possible upon logging in without having to spend needless time or/and effort. It can be argued that the system provided the foundation of lab learning as we intended to however other factors contribute to influencing the remote learning experience, such as session time and practice on the system software being hosted in the remote lab.

7.3 Remote Lab for E-Learning

Although the motivation for the thesis is to design and implement a computing and networking infrastructure for CPS students to remotely access the physical lab on HIOF's campus, the proposed design can be utilized even to offer e-learning of other IoT or smart home systems by preparing a physical lab on university's premises and letting students access those resources remotely without being physically present on the campus or in the lab.

By setting up a lab environment and integrating it with one main server containing various VMs for different purpose systems would allow the university to use the resources effectively while at the same time offering students a learning experience that would be relatively easy to operate and access with minimal effort and time required to do the desired tasks.

Such terminology would also offer professors and course examiners to log and evaluate the student's performances in the lab for the sake of examining students' working knowledge of the systems as per the course requirements.

The sample user manual is attached in Appendix section.

7.4 Future Works

There are number of ways how the developed remote lab solution can be further worked on to provide a better learning experience to the students, which would also contribute to the influence it may have on the learning productivity during lab sessions.

Firstly, we can build a greater number of environments for The Room project by installing more than one Raspberry Pi along with number of components and gadgets in the lab which could be integrated with the Virtual Machines. This will allow the creation of more than system software on separate VMs, hence students can be given access to specific labs based on their knowledge of the system. For instance, one lab with that specific VM will require the basic or introductory tasks on Eclipse from students, while the other labs with their VMs will be meant for students with advanced knowledge of the system, therefore the tasks on Eclipse would be accordingly challenging. Such terminology would allow different students with different knowledge to work simultaneously during the remote session.

Secondly, since the Virtual Machines developed are hosted on Windows OS. However, having the feature of Virtual Machines on MAC OS as well will offer student's option to choose the type of desktop OS they would like to have access to. Such addition will further enrich the operability of the remote lab.

One benefit of such infrastructure is that the educational sectors can use the concept of the developed solution to offer different types of system learning to students. The infrastructure design proposed in the thesis is not limited to being used with "The Room" project only. These systems could be any of the IoT systems or smart home systems to control the environment, while also offering programming lessons to students on various platforms.

Chapter 8 Conclusion

In this thesis, we developed a remotely accessible lab following guidelines of computing and network infrastructure aimed at providing a meaningful and relevant learning experience to students as they perform tasks in the lab by sitting anywhere in the world. The remote labs work in a way where students would authorize themselves to use their student online account's login credentials to access the system software. Once they are authorized, they are presented with a desktop ready to perform tasks and observe the change in the state of the lab. Our target is to make the learning experience for students well understandable, agile in terms of observation, and straightforward to use with minimal confusion and effort needed to perform what is expected from them.

To evaluate the influence the remote lab could have on the learning experience of students, we did three presentations to the participants. We evaluated that although the system works well in terms of provisioning the learning experience as originally intended, however, there are other factors, such as time for a remote lab session with students and in-depth testing of components in the lab before the session, the key to utilizing the value it is meant to offer.

By expanding the lab setup with the possibility of having more than one instance of system software on the main server probably would provide a variety of learning experiences to the students. Also, the operability of the lab could be enhanced if students are offered a choice of working with the specific OS on the VM they would be experimenting on.

Bibliography

Ali, W. (2020). Online and Remote Learning in Higher Education Institutes: A Necessity in Light of COVID-19 Pandemic. *Ccsenet*, 16-25.

Anderson, C. (2015). *Docker [software engineering].* IEEE Software, 32(3).

Arrowhead. (2022, April 22). *This is Eclipse Arrowhead*. Retrieved from Arrowhead: https://arrowhead.eu/1492

Border, C. (2007). The development and deployment of a multi-user, remote access virtualization system for networking, security, and system administration classes. *ACM SIGCSE Bulletin*, 576-580.

Broisin, J. V. (2017). Lab4CE: a remote laboratory for computer education. *International Journal of Artificial Intelligence in Education, 27(1)*, 154-180.

Celdrán, A. H. (2019). Self-organized laboratories for smart campus. *IEEE Transactions on Learning Technologies*, 404-416.

Chakraborty I. & Maity, P. (2020). COVID-19 outbreak: Migration, effects on society, global environment and prevention. *Science of the Total Environment*.

Choudhary, J. N. (2016). Overview of virtualization in cloud computing. *Symposium on Colossal Data Analysis and Networking*, 1-4.

Cisco. (2022). *Cisco AnyConnect Secure Mobility Client*. Retrieved from Cisco.com: https://www.cisco.com/c/en/us/products/security/anyconnect-secure-mobility-client/index.html

Deshmukh, D. &. (2017). Design of IPSec virtual private network for remote access. *International Conference on Computing, Communication and Automation (ICCCA)*, 716-719.

Dockers. (2022, May 1). *Use containers to Build, Share and Run your applications.* Retrieved from Docker.com: https://www.docker.com/resources/what-container/

Gopalakrishnan, S. (2022). *Modeling an Arrowhead Remote Lab with an Image.* Halden: HIOF.

Keele, S. (2007). *Guidelines for performing systematic literature reviews in software engineering (Vol. 5).* EBSE.

Kurbakova, S. V. (2020). Virtual Learning and Educational Environment: New Opportunities and Challenges under the COVID-19 Pandemic. *The 4th International Conference on Education and Multimedia Technology*, (pp. 167-171).

Longzheng, C. S.-l. (2004). Research and implementation of remote desktop protocol service over SSL VPN. *IEEE International Conference onServices Computing* (pp. 502-505). IEEE.

Opensource. (2022, May 1). *What is Raspberry Pi?* Retrieved from opensource.com: https://opensource.com/resources/raspberry-pi

OpenVPN. (2022, April 30). *Secure access and network connectivity reimagined.* Retrieved from OPenVPN: https://openvpn.net/

Oracle. (2022). Retrieved from Oracle VM VirtualBox: https://www.oracle.com/virtualization/virtualbox/

Radhamani, R. S. (2014). Virtual labs improve student's performance in a classroom. *International Conference on E-Learning, E-Education, and Online Training*, 138-146.

Rashid S & Yadav, S. (2020). Impact of Covid-19 pandemic on higher education and research. *Indian Journal of Human Development*,, 340-343.

Singh, S. (2016). Containers & Docker: Emerging roles & future of Cloud technology. *2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)* (pp. 804-807). IEEE.

Stallings, W. &. (2012). *Operating systems: internals and design principles (Vol. 9).* New York: Pearson.

Sun, S. H. (2011). The advantages and the implementation of SSL VPN. *IEEE 2nd International Conference on Software Engineering and Service Science* (pp. 548-551). IEEE.

TelluIoT. (2022, April 18). *ThingML*. Retrieved from GitHub.com: https://github.com/TelluIoT/ThingML

Tyzhnia, D. (2021). *The Impact of COVID-19 on Education*. Retrieved from worldbank.org: https://www.worldbank.org/en/news/opinion/2021/04/02/the-impact-of-covid-19-on-education

Vermeer, B. (2022, April 18). *synk*. Retrieved from IntelliJ IDEA dominates the IDE market with 62% adoption among JVM developers: https://snyk.io/blog/intellij-idea-dominates-the-ide-market-with-62-adoption-among-jvm-developers/

Wazan, A. S. (2021). Which Virtualization Technology is Right for My Online IT Educational Labs? *IEEE Global Engineering Education Conference (EDUCON)* (pp. 1254-1261). IEEE.

Wazan, A. S. (2021). Which Virtualization Technology is Right for My Online IT Educational Labs? *IEEE Global Engineering Education Conference*, 1254-1261.

Wikipedia. (2022, April 18). *Eclipse (software)*. Retrieved from wikipedia.org: https://en.wikipedia.org/wiki/Eclipse_(software)

Wikipedia. (2022, April 30). *Hyper V*. Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Hyper-V

Wikipedia. (2022). *Virtual private network*. Retrieved from Wikipedia.org: https://en.wikipedia.org/wiki/Virtual_private_network

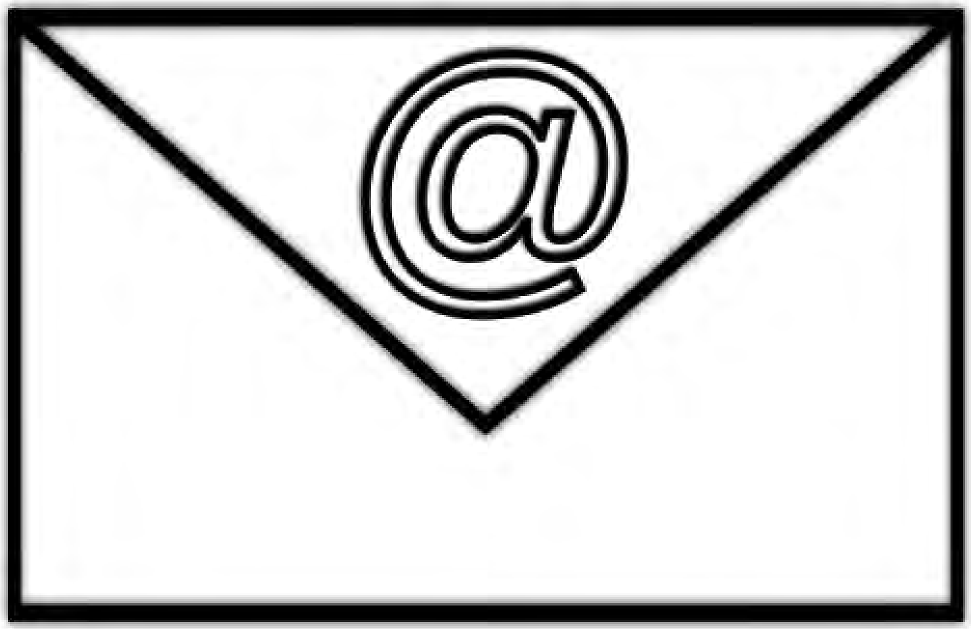
Wohlin, C. (2014). Guidelines for snowballing in systematic literature studies and a replication in software engineering. *Proceedings of the 18th international conference on evaluation and assessment in software engineering*, (pp. 1-10).

Yan, L. (2011). Development and application of desktop virtualization technology. *IEEE 3rd International Conference on Communication Software and Networks*, 326-329.

Appendix

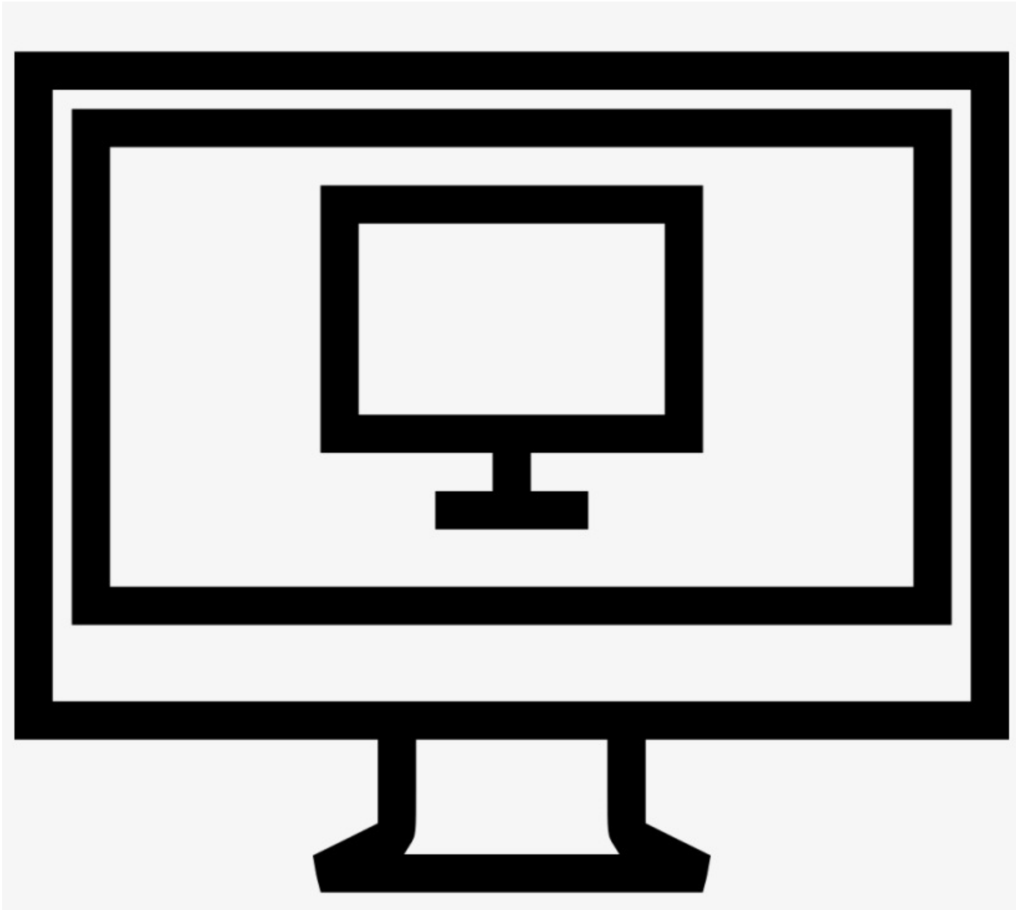
A.1 Storyboard

Tim Jhon is enrolled in the course Cyber Physical System. He requests for a time slot from course professors to access the CPS Virtual lab

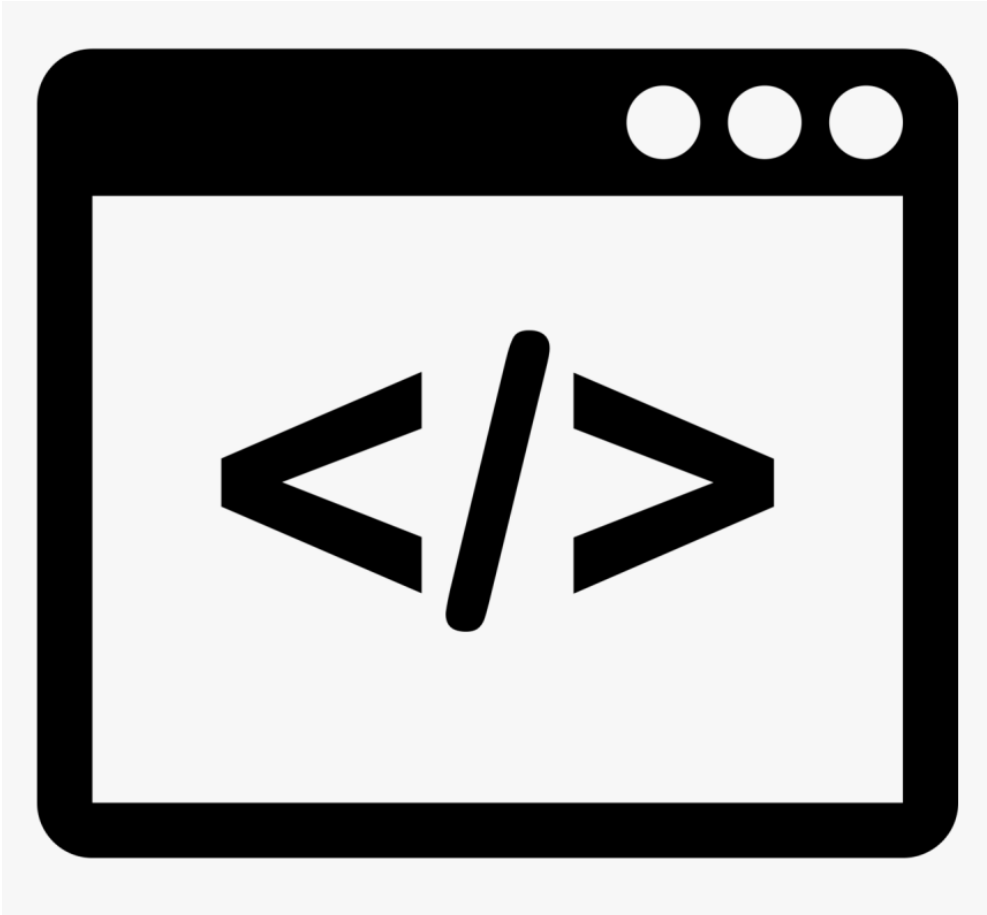


He receives an email with the IP Address/DNS Hostname of the VM along with temporary credentials created for him.

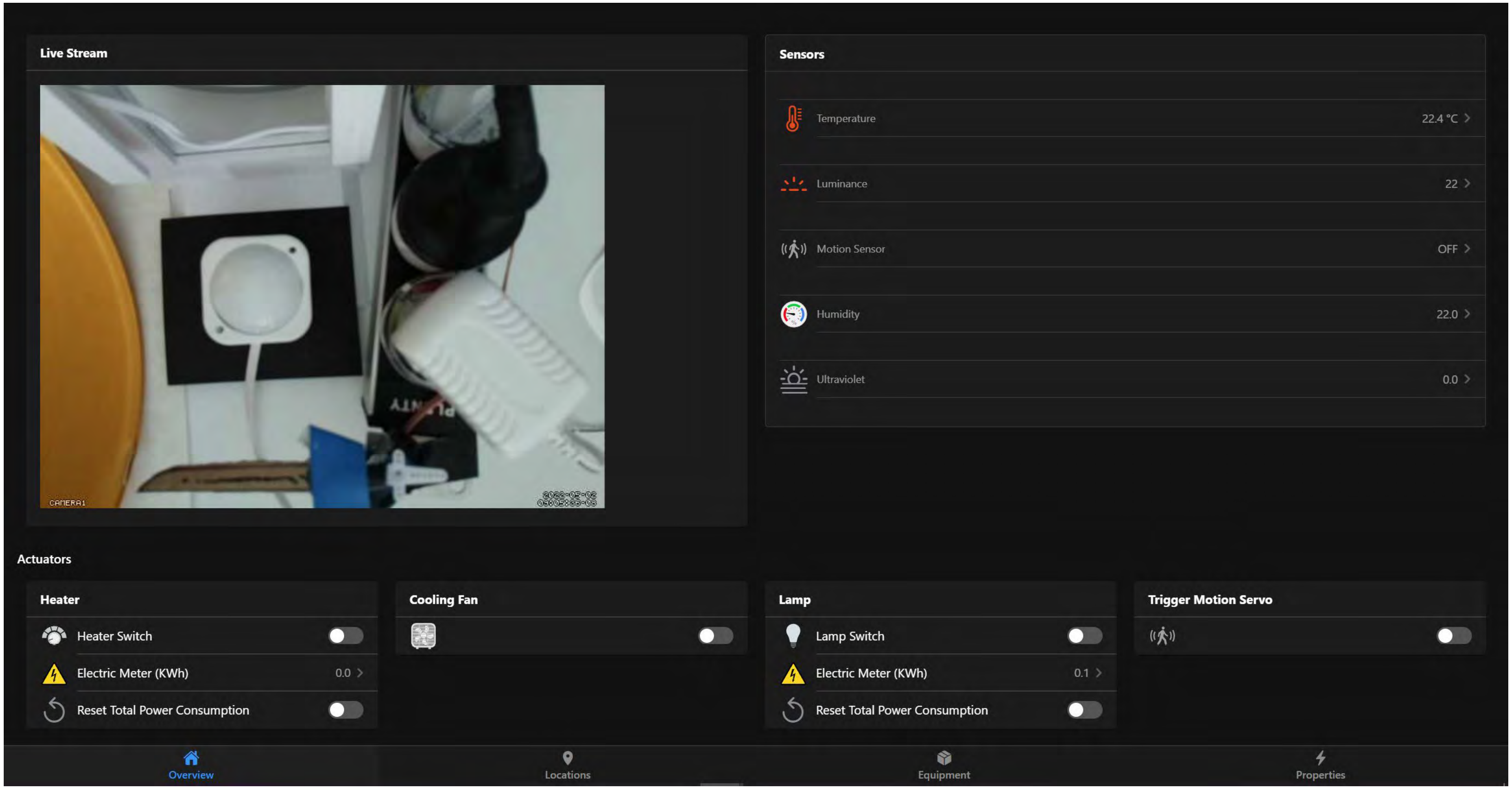
Tim connects to HIOF's network though VPN using the credentials assigned to him by the university



Tim connects to the assigned VM through Remote Desktop Connection using the details sent to him via Email.

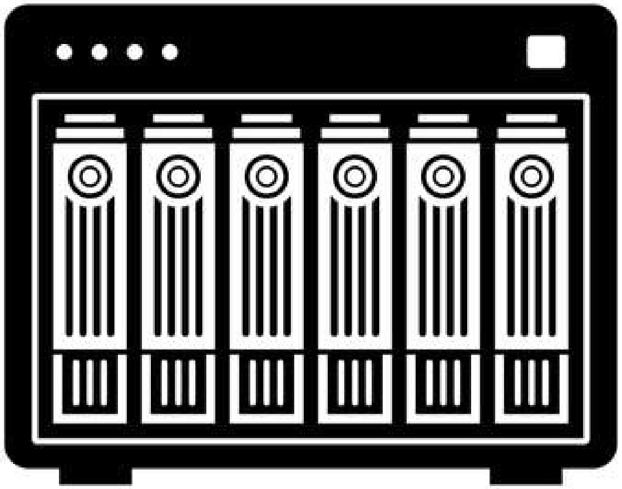


Once logged in, he is presented with opened window of an Eclipse on which he can do the needed coding for the The Room project.



He is also presented a real-time view of the lab recorded through a CCTV that has components (i.e. sensors, switches, lamps) installed in it.

The VM is designed and configured in a way where Tim is restricted to perform any action on the operating system apart from the actions allowed by the course administrators.



After Tim has finished working on the VM, the backup of the VM is saved/overwritten on the central network storage of the lab.

A.2 User Manual (Sample)

User Guide

January 2022

Version 1.10

Document Revisions

Date	Version Number	Document Changes
02/12/2021	1.00	Initial Draft
17/01/2022	1.10	User Manual with OpenHabian Dasboard

Table of Contents

1 Introduction

1.1 Overview:

The aim is to provide students remote access to physical lab for them to have hands-on experience on ThingML, with Arrowhead Framework. The lab is meant to allow students to control certain properties, such as luminance and temperature, of the room using various peripherals integrated in the lab setup.

Following are the requirements to be fulfilled to have access to the lab:

1- HIOF's valid student username and password.
2- PC with Remote Desktop Connection (Windows or MAC OS).
3- Internet Connection.

1.2 Process Overview

Following provides a sequential step-by-step guide for accessing and using the lab.

1- Connecting to HIOF's network
   a. Through EDUROAM network.
   b. Through VPN.
2- Accessing Arrowhead Virtual Lab
   c. RDP session
3- OpenHab / ThingML:
4- Eclipse
5- Monitoring

2 Connecting to HIOF's network:

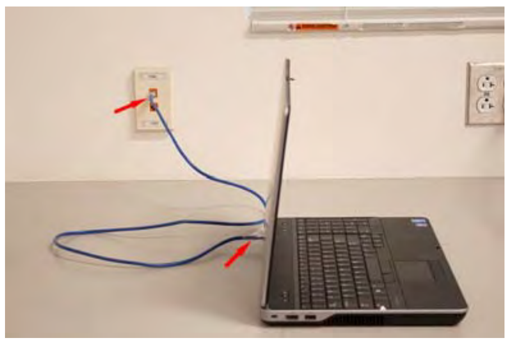
For users to reach the virtual lab. One must be logged in university's network through either one of the following method:

2.1 Through EDUROAM network:

If a user is in university's premises (campus or housing), user can connect his PC to get connected to the HIOF's network, there are two different procedures for that, ethernet cable connection and WiFi.

2.1.1 Procedures for Ethernet Connection

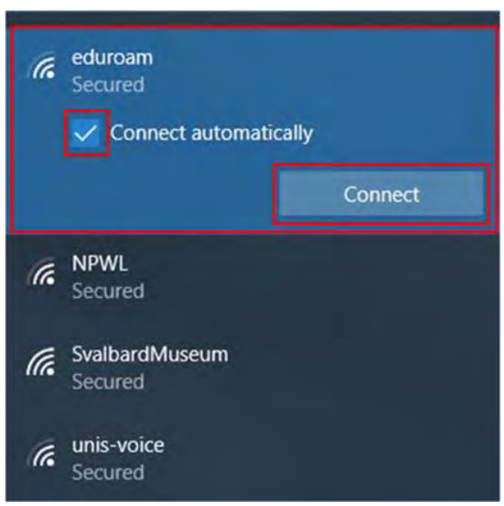
1. Connect the PC to nearby accessible network outlet. This most possibly would be an ethernet cable connection.



2. After a while, verify that you are connected to EDUROAM by hovering cursor on network toolbar.

2.1.2 Procedure for Wifi Connection:

1. Search and locate EDUROAM in available WiFi networks.



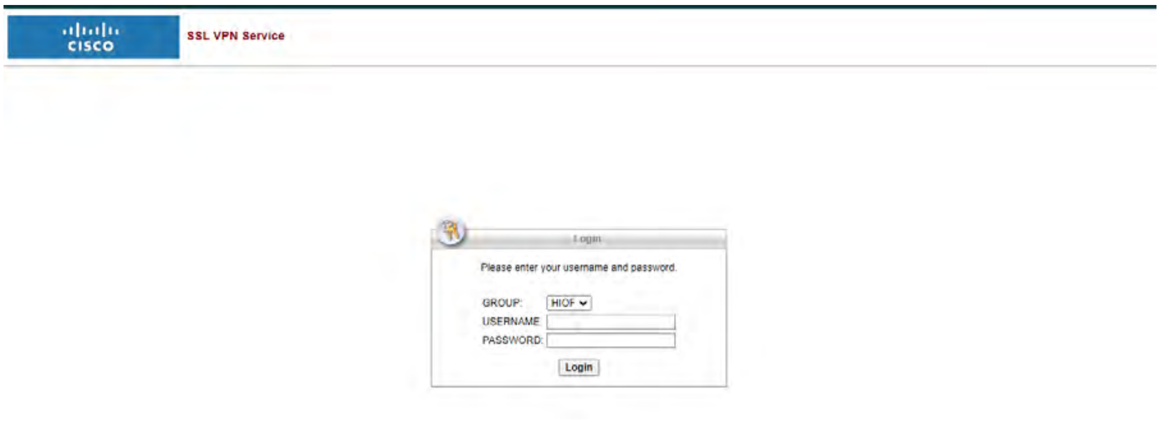
2. Connect to it using HIOF's student credentials.

2.2 Through VPN:

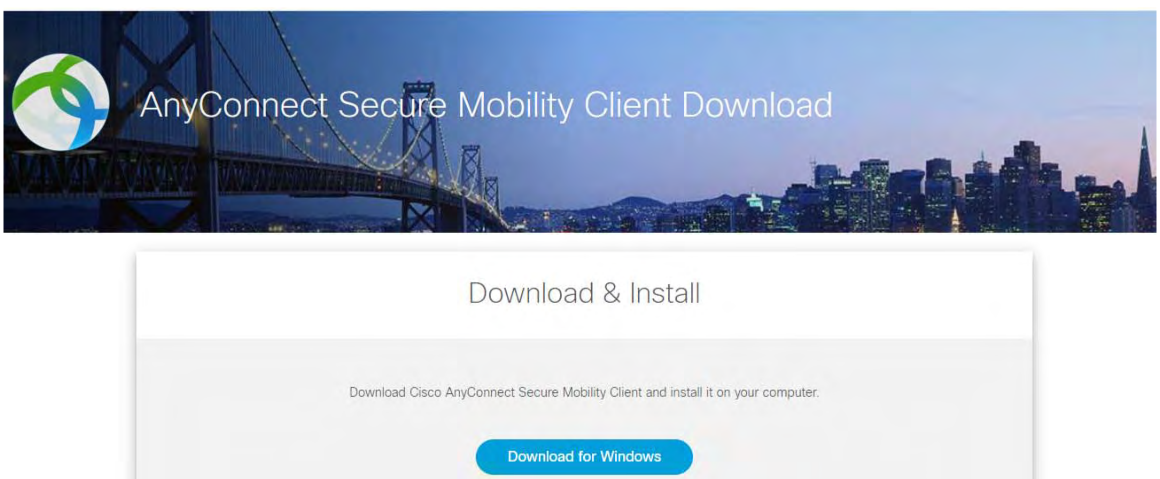
In case it is not possible to connect with EDUROAM, user is ought to follow below mentioned procedure to have a connection via VPN.

2.2.1 Procedures for VPN connection.

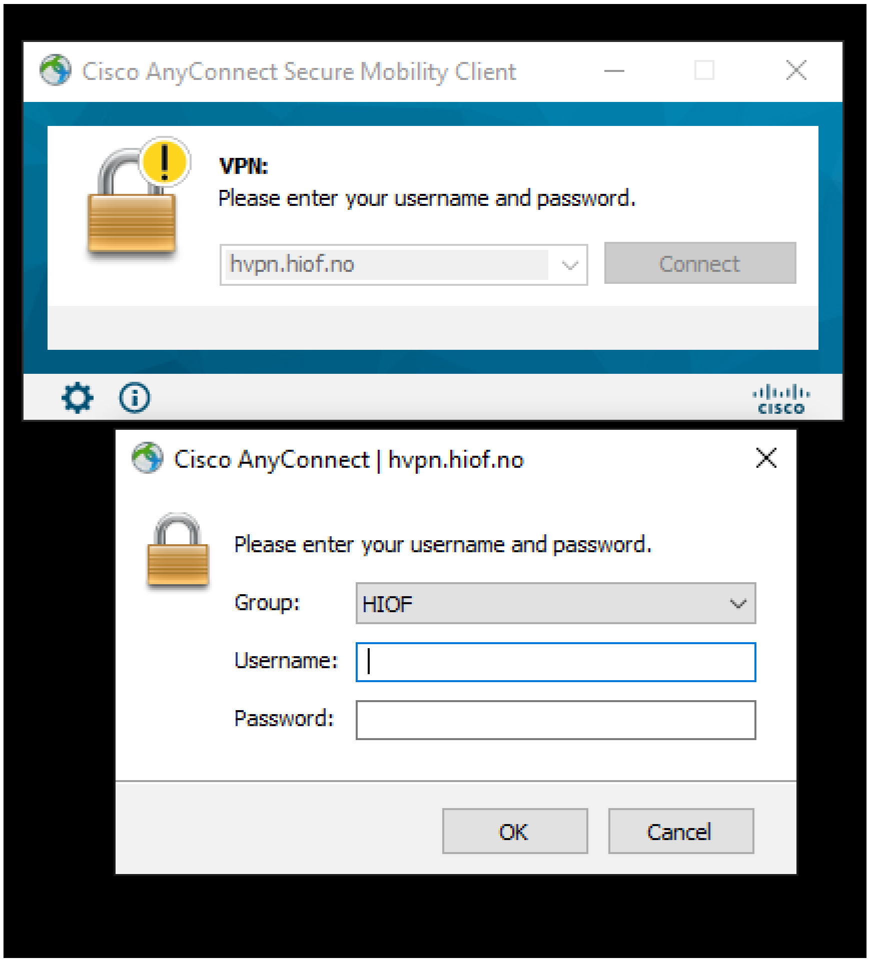
1. Click the LINK to download AnyConnect. The webpage will appear up as below first, the field GROUP shall be HIOF. While the username and student as provided by the university.



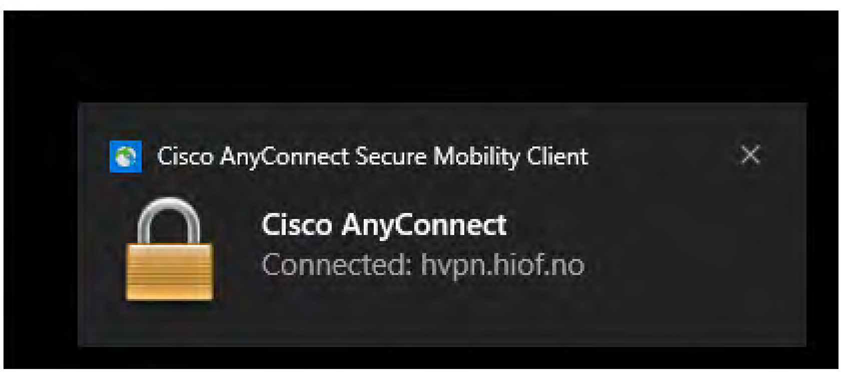
2. Follow the instructions along to download and install AnyConnect.



3. Once the installation is done, run the AnyConnect and use connect to the hvpn.hiof.no. The software will ask for the credentials to login, the username and password is same as provided by the university.



4. On successful connection, following notification would appear up.

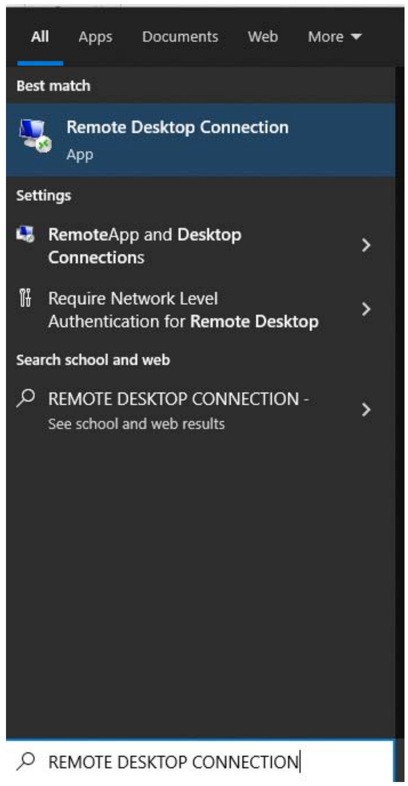


3 Accessing Arrowhead Virtual Lab:

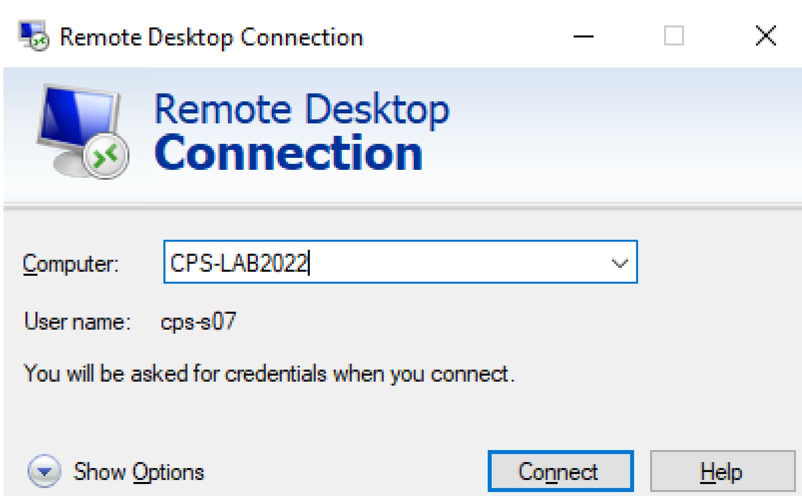
After successful connection to HIOF's network. User can remotely connect to the lab machine using the Remote Desktop Connection (RDP). Remote Desktop Connection is pre-installed on Windows OS.

3.1 RDP Session:

1. From the Windows search bar, type "**Remote Desktop Connection**". And open the application.



2. Once opened, it will be requiring you to enter the hostname of in the **Computer** section. Type in "**CPS-LAB2022**". And click on **Connect** afterwards.



3. Upon clicking Connect, dialog box will be prompted for username and password. Enter the following then:

   Username: cps-s07

   Password: atlanta8190

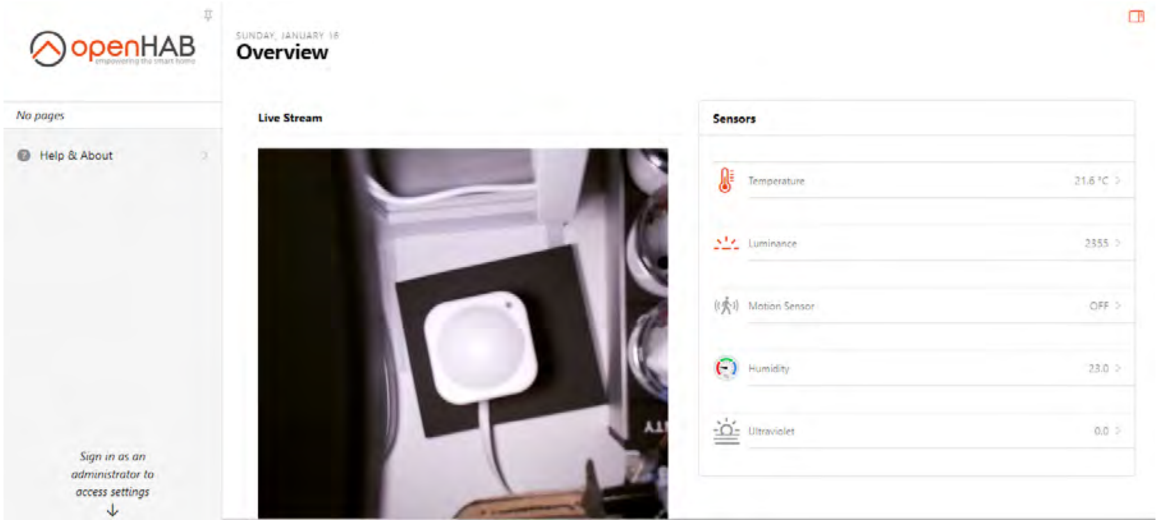
4. From here on, after a while, the window for OpenHabian web dashboard open.

4 OpenHabian Web Dashboard:

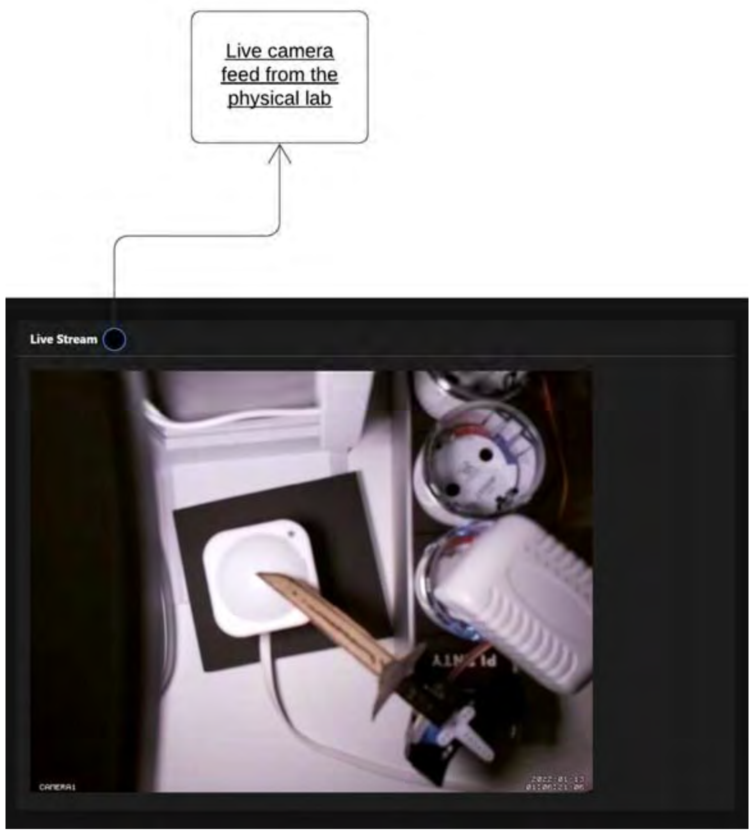
Upon logging in to CPS-LAB2022, the web dashboard would open automatically.

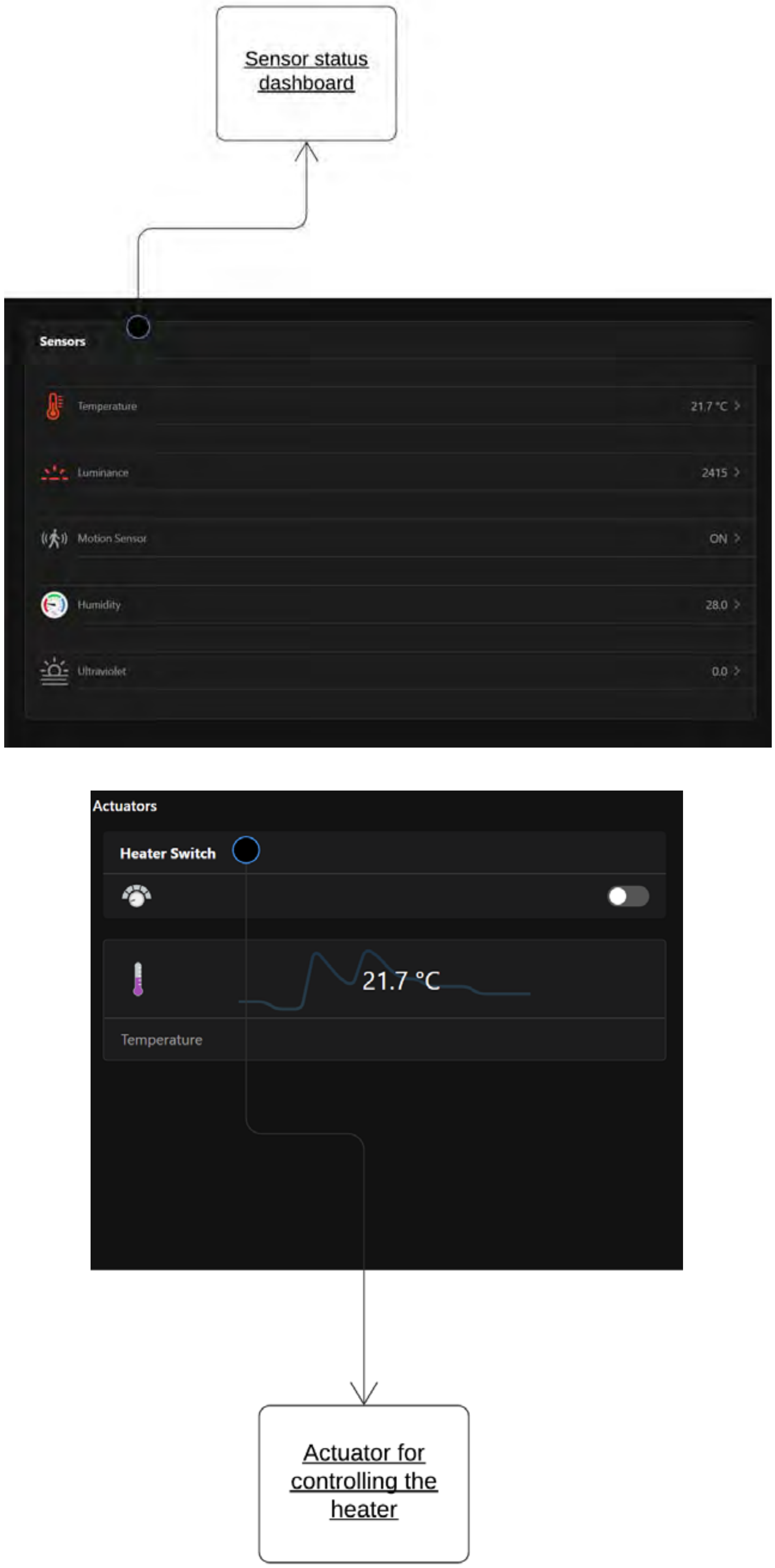
4.1 Web Dashboard:

The view of dashboard looks like following, where each component is briefly explain further below.

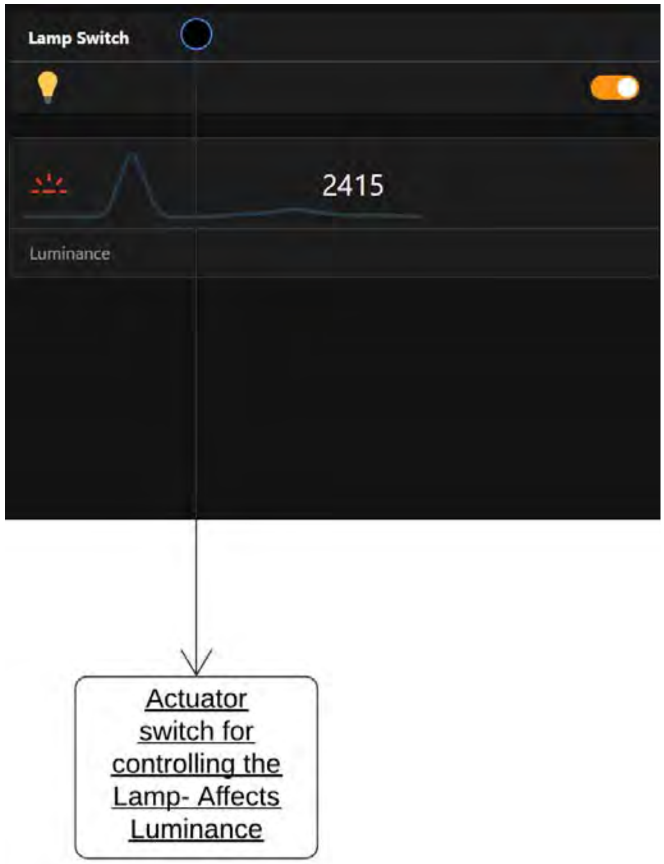


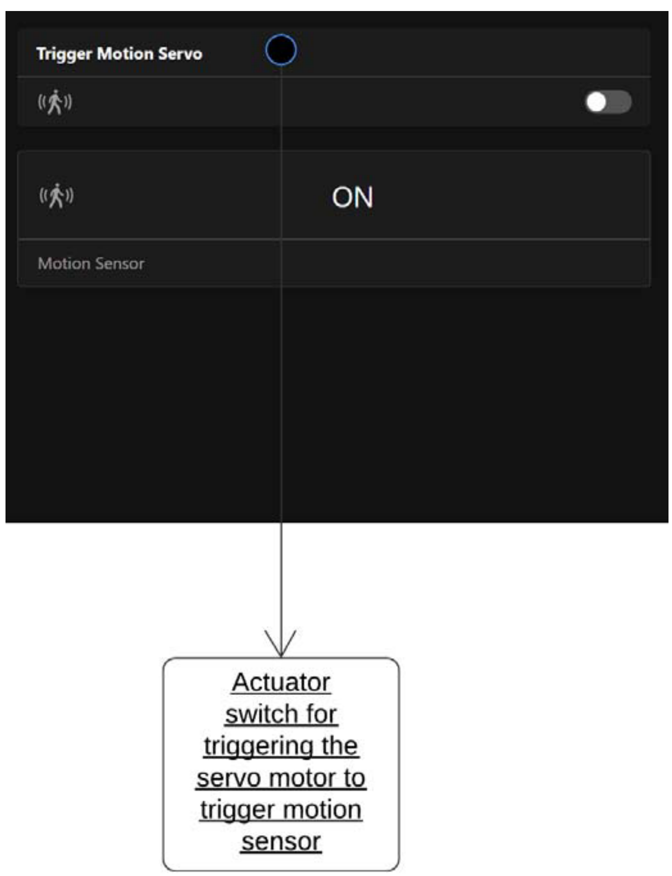
4.1.1 Components:





Sensor status dashboard
Sensors
Temperature
21.7 °C
Luminance
2415
Motion Sensor
ON
Humidity
28.0
Ultraviolet
0.0
Actuators
Heater Switch
21.7 °C
Temperature
Actuator for controlling the heater


Lamp Switch
2415
Luminance
Actuator switch for controlling the Lamp- Affects Luminance


Trigger Motion Servo
ON
Motion Sensor
Actuator switch for triggering the servo motor to trigger motion sensor