Final Year B.Tech. Project Report

On

Private Cloud for WCE

For the Degree of

Bachelor of Technology

In

Computer Science and Engineering

Submitted by

2019BTECS00041 Suyash Chavan

2019BTECS00078 Om Khairnar

2019BTECS00085 Yash Pandhare

Under the Guidance of

Prof. M. K. Chavan

Department of Computer Science and Engineering,

Walchand College of Engineering, Sangli.

(*An Autonomous Institute*)

2022-23

Walchand College of Engineering, Sangli.

(*An Autonomous Institute*)

Department of Computer Science and Engineering

CERTIFICATE

This is to certify that the Project Report entitled, **"Private Cloud for WCE"** submitted by

2019BTECS00041 Suyash Chavan,

2019BTECS00078 Om Khairnar,

2019BTECS00085 Yash Pandhare

to Walchand College of Engineering, Sangli, India, is a record of bonafide Project work of course **"PROJECT"** carried out by him under my/our supervision and guidance and is worthy of consideration for the award of the degree of Bachelor of Technology in Computer Science & Engineering during the academic year ***2022-23***.

Prof. M. K. Chavan	**Prof. S. S. Rokade**	**Prof. A. M. Chimanna**
Guide	Panel Member	Panel Member

Mrs. Dr. M. A. Shah

Head

Department of Computer Science and Engineering

Acknowledgement

We would like to take this opportunity to express our sincere gratitude to the Director, HOD of CSE Department for giving us a chance to showcase our talent in the form of a project. We would like to convey our gratitude to our project guide **Prof. M.K. Chavan** for his valuable suggestions. We have put effort into this project. However, it would not have been possible without the kind support and help of our respected **H.O.D. M.A. Shah**. Your valuable guidance and suggestions helped us in various phases of this project. We will always be thankful to you in this regard.

Declaration

I hereby declare that work presented in this project report titled **"Private Cloud for WCE"** submitted by us in the partial fulfillment of the requirement of the award of the degree of Bachelor of Technology (B.Tech) Submitted in the Department of Computer Science & Engineering, Walchand College of Engineering, Sangli, is an authentic record of my project work carried out under the guidance of **Prof. M. K. Chavan.**

2019BTECS00041

Suyash Chavan

2019BTECS00078

Om Khairnar

2019BTECS00085

Yash Pandhare

Date :

Place : Sangli

Abstract

The project aims to develop and design a private cloud infrastructure tailored specifically for college students, offering them cost-effective deployment of cloud services while leveraging the underutilized compute and storage resources available on college servers. The implementation of this private cloud will provide students with a platform to host their applications, websites, and other cloud-based services at a fraction of the cost compared to commercial cloud providers.

By harnessing the existing compute and storage capabilities of college servers that are typically underutilized, the private cloud will maximize resource efficiency, reducing wastage and optimizing the utilization of available resources. This approach not only contributes to cost savings but also promotes sustainable resource allocation within the college ecosystem.

The project will encompass various stages, starting with the design of a robust and scalable private cloud architecture that ensures high availability, fault tolerance, and security. Virtualization technologies will be employed to enable the efficient partitioning and management of server resources, allowing multiple students to concurrently utilize the cloud infrastructure without compromising performance.

To facilitate seamless deployment and management of cloud services, a user-friendly interface and a set of management tools will be developed. Students will have the ability to provision and configure virtual machines through a web-based portal.

The successful implementation of the private cloud for college students will empower them to explore and experiment with cloud-based technologies without the financial burden associated with commercial cloud providers. Moreover, the utilization of underutilized resources on college servers will foster a more sustainable and resource-efficient environment. This project can serve as a model for educational institutions looking to provide cost-effective cloud services to their students while optimizing their own infrastructure utilization.

Table Of Contents

1. Problem Statement

The absence of a cost-effective private cloud infrastructure tailored to college students' needs limits their ability to deploy cloud services efficiently, while underutilized compute and storage resources on college servers go untapped. There is a pressing need to develop a solution that enables students to leverage these resources effectively, facilitating affordable cloud service deployment and optimizing resource utilization within the college ecosystem.

2. Introduction

Educational institutions are constantly seeking ways to optimize resource management and improve operational efficiency. Cloud computing, as an emerging paradigm, offers promising opportunities to deliver a wide range of computing services in an unprecedented manner. However, many colleges face challenges in effectively utilizing and managing their underutilized resources.

This project proposes the implementation of a private cloud platform within the college to address these issues. By establishing a private cloud infrastructure, faculties and students can maximize resource utilization and eliminate the need for external dependencies. This report will explore the benefits of a private cloud platform in streamlining resource management and enhancing the overall productivity and innovation within the educational environment.

By deploying a private cloud, WCE will address key challenges faced by educational institutions, including the need for scalability, centralized resource management, security, and collaboration. This transformative project will revolutionize the way students, faculty, and staff interact with technology, enabling them to access computing resources on-demand, collaborate seamlessly, and leverage advanced services for academic and administrative purposes.

3. Motivation

The motivation behind developing and designing a private cloud infrastructure for college students stems from the need to provide them with a robust and accessible platform for deploying their cloud services while utilizing the underutilized compute and storage resources available on college servers. This section highlights the key motivations driving this project and the potential benefits it brings to college students and the college ecosystem.

1. Accessible Cloud Services: College students often require cloud services to host their projects, applications, and websites. However, relying solely on commercial cloud providers can be costly and may impose financial barriers for students with limited budgets. By developing a private cloud infrastructure, students gain access to a cost-effective platform where they can deploy their cloud services without incurring significant expenses. This accessibility enables a wider range of students to leverage the benefits of cloud technology for their academic and personal endeavors.

2. Efficient Resource Utilization: College servers often operate with underutilized compute and storage resources, resulting in wastage and inefficiency. By harnessing these unused resources, a private cloud solution can optimize resource utilization within the college ecosystem. This approach not only reduces waste but also promotes sustainable resource management by effectively utilizing existing infrastructure. By making efficient use of college server resources, the private cloud infrastructure minimizes the need for additional hardware investments, benefiting both students and the college.

3. Learning and Skill Development: The private cloud infrastructure provides college students with an opportunity to gain practical experience and develop skills in cloud computing and related technologies. By deploying their cloud services on the private cloud, students can learn about virtualization, system administration, and cloud management. This hands-on experience enables students to deepen their understanding of cloud computing concepts and enhances their ability to design, deploy, and manage scalable cloud applications. These skills are highly valuable in today's job market, where cloud computing plays a crucial role.

4. Significance

1. By utilizing this application, students can access and utilize the resources provided by our college, eliminating the need to spend money on external resources. This allows colleges to optimize their existing resources and reduces the necessity for additional infrastructure investments. Consequently, it leads to cost savings by eliminating reliance on external cloud service providers.

2. The private cloud infrastructure streamlines the allocation of resources, centralizes management, and enhances operational efficiency, resulting in cost savings and increased productivity for WCE.

3. The private cloud platform provides WCE with a scalable solution to meet the growing demands for computing resources, storage, and applications as the institution expands.

4. The successful implementation of a private cloud at WCE serves as a model for other educational institutions, inspiring them to explore similar solutions in addressing their IT challenges and improving their capabilities.

5. By utilizing this solution, colleges gain complete control over their computing environment. They can tailor the infrastructure to their specific requirements, ensuring alignment with their academic and research needs.

6. The implementation of a private cloud project offers students and faculty the opportunity to acquire practical experience with cloud technologies, equipping them with valuable skills in managing and utilizing cloud resources.

7. The private cloud implementation at WCE positions the institution to adapt to future technological advancements and emerging trends, ensuring it remains competitive and prepared for the evolving educational landscape.

5. Literature Survey

An example of a university implementing a private cloud is the Indian Institute of Technology (IIT) Kharagpur, which has utilized OpenStack. The institute issued a tender for the procurement of a Cloud Computing System specifically for IIT Kharagpur.

INDIAN INSTITUTE OF TECHNOLOGY KHARAGPUR - 721302 Sub: Procurement of Cloud Computing System for IIT Kharagpur

Various platforms available on the internet, such as OpenStack, offer the capability to create custom private clouds. OpenStack, for instance, provides a range of services including compute, storage, network, and frontend. These services can be utilized to construct a private cloud tailored to the specific needs of the university.

6. Objectives

The proposed project work aims to achieve several objectives:

1. Creating On-Demand Virtual Machine Instances: The project intends to establish a system that allows the creation of virtual machine instances as needed for the college's laboratories and club activities. This enables flexibility and scalability in resource allocation.

2. Monitoring Underutilized Resources: One of the goals is to monitor and identify underutilized resources within the college's infrastructure. This helps optimize resource allocation, ensuring that resources are efficiently utilized and unnecessary costs are minimized.

3. Cost Efficiency for Students: The project aims to offer a more cost-efficient solution for students when it comes to deploying applications and utilizing computing resources. By utilizing the private cloud infrastructure, students can avoid additional expenses associated with external cloud service providers.

4. No Time Bound Restrictions: Unlike many free public cloud services that impose time limitations or restrictions, the proposed private cloud solution does not have such time-bound constraints. This ensures that students and users have continuous and uninterrupted access to the cloud resources they require.

5. Improved Workload Management: The private cloud implementation intends to provide better workload management capabilities. This includes efficient resource allocation, load balancing, and prioritization of tasks, ensuring optimal performance and responsiveness.

6. Faster Data Transfer within the Organization: By accessing the private cloud within the organization's firewall, users can benefit from faster data transfer rates. The internal network structure facilitates quicker and more efficient data communication and processing, enhancing overall system performance and user experience.

7. Methodology

We have designed and developed a user friendly UI with functionality of spinning up compute instances with all the necessary needs. It under the hood utilizes Openstack backend to configure the instance.

OpenStack is a cloud operating system which will enable and control large pools of compute, storage, and networking resources throughout a datacenter, all managed and provisioned through APIs with common authentication mechanisms.

A dashboard is also available, giving administrators control while empowering their users to provision resources through a web interface.

Following are the components of Openstack:

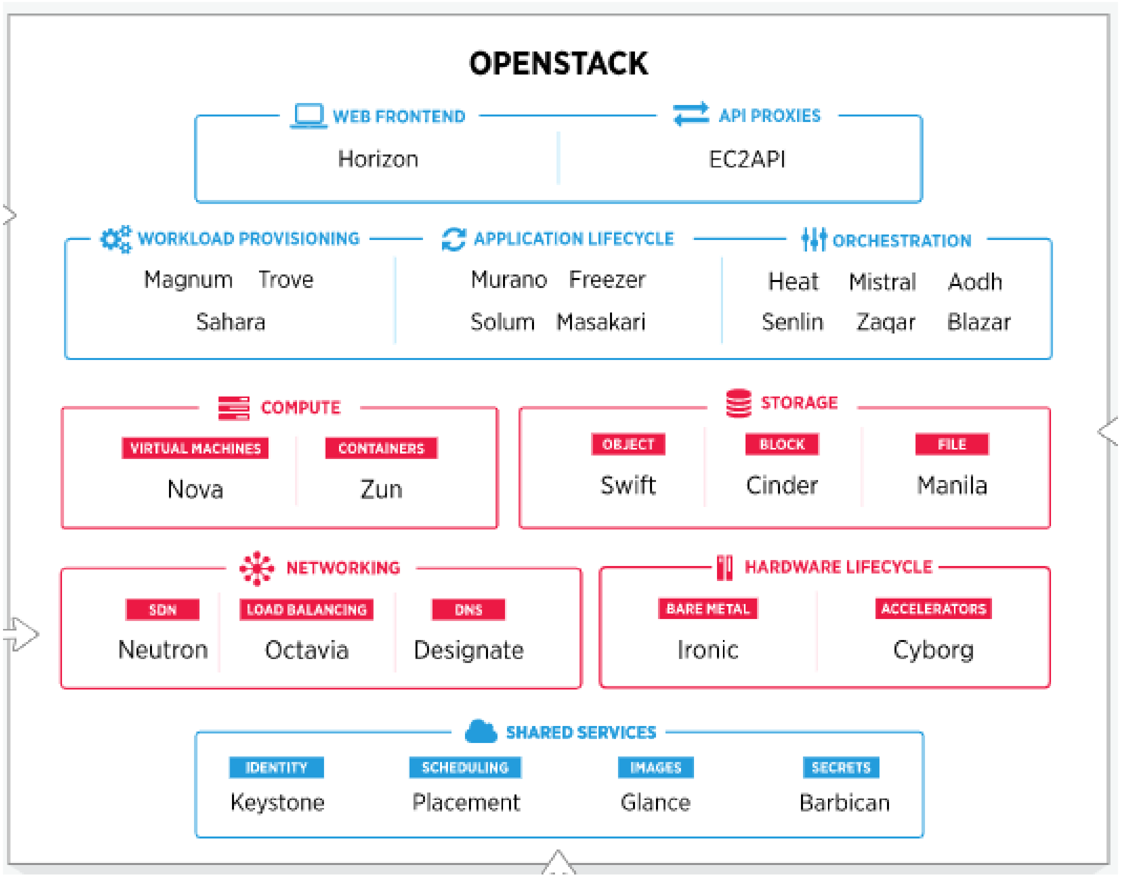


Figure 1: Components of Openstack

Phase 1: Software and API

Open source software for setting up a private cloud is widely available in the market such as OpenStack, Eucalyptus, OpenNebula, etc. They are the main competitors in the private cloud area among which OpenStack has a more user-friendly and widely used.

It is a cloud operating system, a new management layer that adds automation and control for a pool of resources such as computation, storage and network, empower administrators & users via self-service portals and empower developers to make apps cloud-aware via APIs.

In this phase, we focus on setting up the software components and APIs required for private cloud.

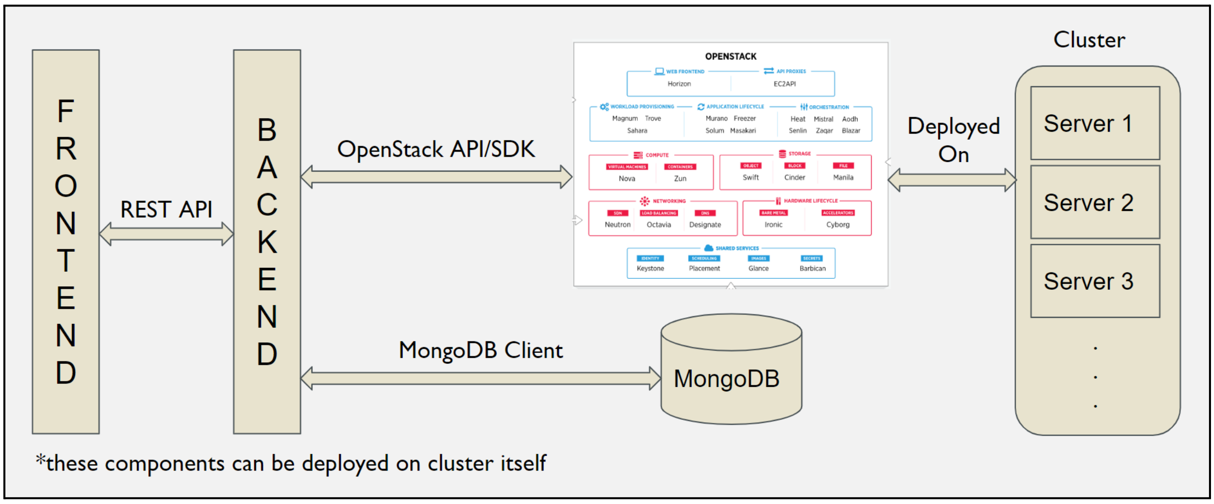


Figure 2: Overview of system interaction in Phase 1

Above figure shows how systems interact with each other through protocols. Frontend and Backennd are implemented in Streamlit and Python where they interact with REST APIs. Backend is connected to MongoDB to store user and instance information that is created on Openstack which in this phase is hosted on Openmetal infra.

Phase 2: Proxy for External Access

We will be deploying Private Cloud in campus intranet so VMs will be accessible from Intranet with their Private IP. Talking about internet access, we can have internet access for each VM if we allocate public IP for each VM(similar to AWS Elastic IP).

Getting a pool of IPs is costly and time consuming. Instead we can use a single public IP and deploy Proxy software so that we can interact with VMs through this proxy server.

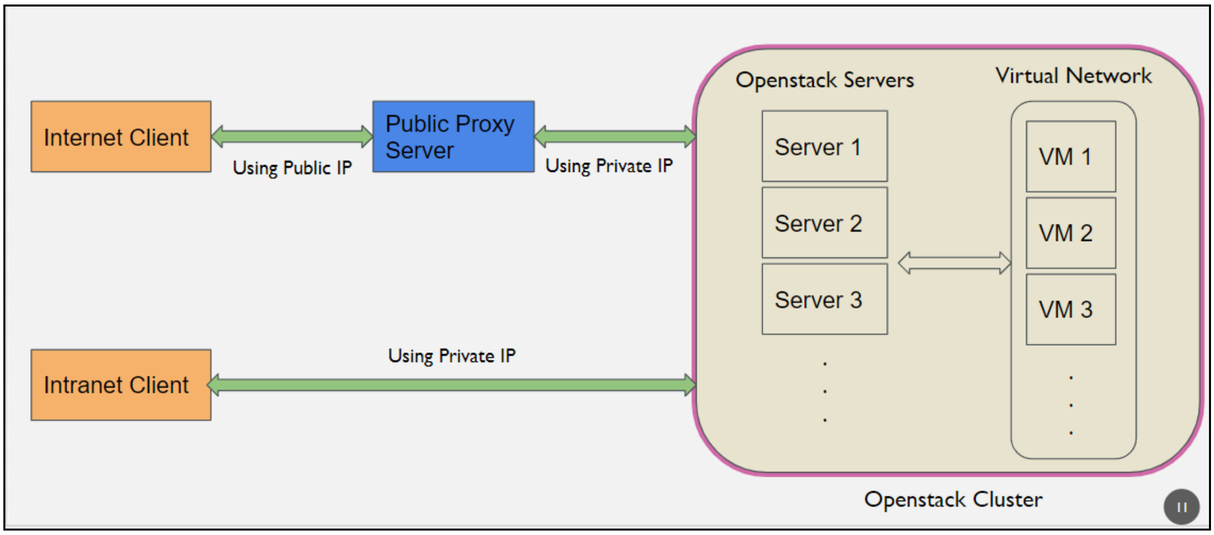


Figure 3: Access of VM through network

As illustrated in Figure 3, a user can access the VM through the intranet using the Private IP address that has been assigned and access all the ports that he needs.

When we access a VM from the internet we can't use a Private IP address as it is only valid for Intranet. As discussed earlier we will be using one Public IP address to access the VM by using a Proxy in between which will redirect all the traffic from ports that are mapped to ports of VM. There needs to be a policy that governs the maximum number of ports that a user can associate to proxy.

Phase 3: Deployment on Hardware

In this phase the private cloud will be deployed on infrastructure provided by the college. In previous phases we used Openmetal who provided Openstack as a service.

There are various ways in which Openstack can be deployed. Most simple approach is to use Microstack on Ubuntu Server 22.04. Microstack is a snap app which can be installed through snap as snap being by default on Ubuntu.

There will be a cluster of 2 nodes one of which will be a controller and another compute node. Following is the hardware config provided by college for both controller and compute node:

```
suyash@openstack:~$ lscpu
Architecture:            x86_64
  CPU op-mode(s):        32-bit, 64-bit
  Address sizes:         39 bits physical, 48 bits virtual
  Byte Order:            Little Endian
CPU(s):                  4
  On-line CPU(s) list:   0-3
Vendor ID:               GenuineIntel
  Model name:            Intel(R) Xeon(R) CPU E3-1225 v3 @ 3.20GHz
    CPU family:          6
    Model:               60
    Thread(s) per core:  1
    Core(s) per socket:  4
    Socket(s):           1
    Stepping:            3
    CPU max MHz:         3600.0000
    CPU min MHz:         800.0000
    BogoMIPS:            6384.95
    Flags:               fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts acpi mmx fxsr sse sse2 s
                         p_good nopl xtopology nonstop_tsc cpuid aperfmperf pni pclmulqdq dtes64 monitor ds_cpl vmx smx est tm2 ssse3
                         imer aes xsave avx f16c rdrand lahf_lm abm cpuid_fault epb invpcid_single pti ssbd ibrs ibpb stibp tpr_shado
                         ms invpcid xsaveopt dtherm ida arat pln pts md_clear flush_l1d
Virtualization features:
  Virtualization:        VT-x
Caches (sum of all):
  L1d:                   128 KiB (4 instances)
  L1i:                   128 KiB (4 instances)
  L2:                    1 MiB (4 instances)
  L3:                    8 MiB (1 instance)
NUMA:
  NUMA node(s):          1
  NUMA node0 CPU(s):     0-3
Vulnerabilities:
  Itlb multihit:         KVM: Mitigation: VMX disabled
  L1tf:                  Mitigation; PTE Inversion; VMX conditional cache flushes, SMT disabled
  Mds:                   Mitigation; Clear CPU buffers; SMT disabled
  Meltdown:              Mitigation; PTI
  Mmio stale data:       Unknown: No mitigations
  Retbleed:              Not affected
  Spec store bypass:     Mitigation; Speculative Store Bypass disabled via prctl and seccomp
  Spectre v1:            Mitigation; usercopy/swapgs barriers and __user pointer sanitization
  Spectre v2:            Mitigation; Retpolines, IBPB conditional, IBRS_FW, STIBP disabled, RSB filling, PBRSB-eIBRS Not affected
  Srbds:                 Mitigation; Microcode
  Tsx async abort:       Not affected
suyash@openstack:~$
```

Figure 4: Specification of Controller and Compute Hardware

8. Implementation and Results

For creating a cluster we need 2 computers in the same network, let's say one a Controller Node and other a Compute Node. Configure them with a static IP as restarts can initiate a DHCP and IP address might change. Also make sure these machines have VT support if they are virtual machines on some hypervisor like VMware ESXi.

Following steps will provide a path to configure these nodes from scratch and spin up an instance from Private Cloud Console and associate Proxy ports for external access.

<u>Setting up Controller Node:</u>

1. Install Microstack from snap

```
suyash@INTERN-SUYASHC:~$ ssh suyash@10.7.3.162
suyash@10.7.3.162's password:
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 5.15.0-72-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Sat May 27 11:42:57 AM UTC 2023

  System load:  0.00244140625      Users logged in:                 1
  Usage of /:   19.5% of 97.87GB   IPv4 address for br-2da36b8a03d1: 172.19.0.1
  Memory usage: 14%                IPv4 address for br-f76e99203afb: 172.18.0.1
  Swap usage:   0%                 IPv4 address for docker0:         172.17.0.1
  Temperature:  50.0 C             IPv4 address for eno1:            10.7.3.162
  Processes:    171


 * Introducing Expanded Security Maintenance for Applications.
   Receive updates to over 25,000 software packages with your
   Ubuntu Pro subscription. Free for personal use.

     https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

56 updates can be applied immediately.
7 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings


Last login: Sat May 27 11:40:22 2023 from 10.7.5.131
suyash@openstack:~$ sudo snap install microstack --devmode --beta
[sudo] password for suyash:
microstack (beta) ussuri from Canonical✓ installed
suyash@openstack:~$
```

2. Initiate Microstack as control node and configure cinder to use LVM backend. Also add ssl certificate to glance config file and restart the *microstack.cinder.uwsgi*, *microstack.cinder.scheduler* and *microstack.cinder.volume* services.

```
suyash@openstack:~$ sudo microstack init --auto --control --setup-loop-based-cinder-lvm-backend --loop-device-file-size 50
2023-05-27 11:46:02,206 - microstack_init - INFO - Configuring clustering ...
2023-05-27 11:46:02,581 - microstack_init - INFO - Setting up as a control node.
2023-05-27 11:46:05,677 - microstack_init - INFO - Generating TLS Certificate and Key
2023-05-27 11:46:07,230 - microstack_init - INFO - Configuring networking ...
2023-05-29 07:58:29,142 - microstack_init - INFO - Opening horizon dashboard up to *
2023-05-29 07:58:30,138 - microstack_init - INFO - Waiting for RabbitMQ to start ...
Waiting for 10.7.3.162:5672
2023-05-29 07:58:38,374 - microstack_init - INFO - RabbitMQ started!
2023-05-29 07:58:38,375 - microstack_init - INFO - Configuring RabbitMQ ...
2023-05-29 07:58:39,291 - microstack_init - INFO - RabbitMQ Configured!
2023-05-29 07:58:39,365 - microstack_init - INFO - Waiting for MySQL server to start ...
Waiting for 10.7.3.162:3306
2023-05-29 08:00:04,512 - microstack_init - INFO - Mysql server started! Creating databases ...
2023-05-29 08:00:09,821 - microstack_init - INFO - Configuring Keystone Fernet Keys ...
2023-05-29 08:03:38,099 - microstack_init - INFO - Bootstrapping Keystone ...
2023-05-29 08:03:50,290 - microstack_init - INFO - Creating service project ...
2023-05-29 08:03:56,039 - microstack_init - INFO - Keystone configured!
2023-05-29 08:03:56,110 - microstack_init - INFO - Configuring the Placement service...
2023-05-29 08:04:17,120 - microstack_init - INFO - Running Placement DB migrations...
2023-05-29 08:04:39,791 - microstack_init - INFO - Configuring nova control plane services ...
2023-05-29 08:04:52,038 - microstack_init - INFO - Running Nova API DB migrations (this may take a lot of time)...
2023-05-29 08:06:41,849 - microstack_init - INFO - Running Nova DB migrations (this may take a lot of time)...
Waiting for 10.7.3.162:8774
2023-05-29 08:18:55,688 - microstack_init - INFO - Creating default flavors...
2023-05-29 08:19:22,151 - microstack_init - INFO - Configuring nova compute hypervisor ...
2023-05-29 08:19:22,152 - microstack_init - INFO - Checking virtualization extensions presence on the host
2023-05-29 08:19:22,180 - microstack_init - INFO - Hardware virtualization is supported - KVM will be used for Nova instances
2023-05-29 08:19:28,637 - microstack_init - INFO - Configuring the Spice HTML5 console service...
2023-05-29 08:19:30,466 - microstack_init - INFO - Configuring Neutron
Waiting for 10.7.3.162:9696
2023-05-29 08:27:15,144 - microstack_init - INFO - Configuring Glance ...
Waiting for 10.7.3.162:9292
2023-05-29 08:28:53,252 - microstack_init - INFO - Adding cirros image ...
2023-05-29 08:28:56,865 - microstack_init - INFO - Creating security group rules ...
2023-05-29 08:29:09,041 - microstack_init - INFO - Configuring the Cinder services...
2023-05-29 08:30:07,366 - microstack_init - INFO - Running Cinder DB migrations...
2023-05-29 08:31:24,446 - microstack_init - INFO - Setting up cinder-volume service with the LVM backend...
2023-05-29 08:31:32,600 - microstack_init - INFO - restarting libvirt and virtlogd ...
2023-05-29 08:32:06,289 - microstack_init - INFO - Complete. Marked microstack as initialized!
suyash@openstack:~$ sudo tee /var/snap/microstack/common/etc/cinder/cinder.conf.d/glance.conf <<EOF
glance_ca_certificates_file = /var/snap/microstack/common/etc/ssl/certs/cacert.pem
EOF
[sudo] password for suyash:
glance_ca_certificates_file = /var/snap/microstack/common/etc/ssl/certs/cacert.pem
suyash@openstack:~$ sudo snap restart microstack.cinder-{uwsgi,scheduler,volume}
2023-05-29T08:33:56Z INFO Waiting for "snap.microstack.cinder-scheduler.service" to stop.
2023-05-29T08:34:04Z INFO Waiting for "snap.microstack.cinder-uwsgi.service" to stop.
Restarted.
```

3. Command microstack to add another compute node to the cluster. It gives us the connection string that we will use while setting up microstack on the compute node.

```
suyash@openstack:~$ sudo microstack add-compute
[sudo] password for suyash:
Use the following connection string to add a new compute node to the cluster (valid for 20 minutes from this moment):
hKhob3N0bmFtZaoxMC43LjMuMTYyq2ZpbmdlcnByaW50xCB47zWa/W6Qh6jPo0HdXoIaYwyyVNM90MOkyqPL9Zow36JpZNkgNmQwNzhiOTJmZTBkNDJjNWIzMGEwNzIxZT
suyash@openstack:~$
```

Setting up Compute Node:

1. Install Microstack from snap

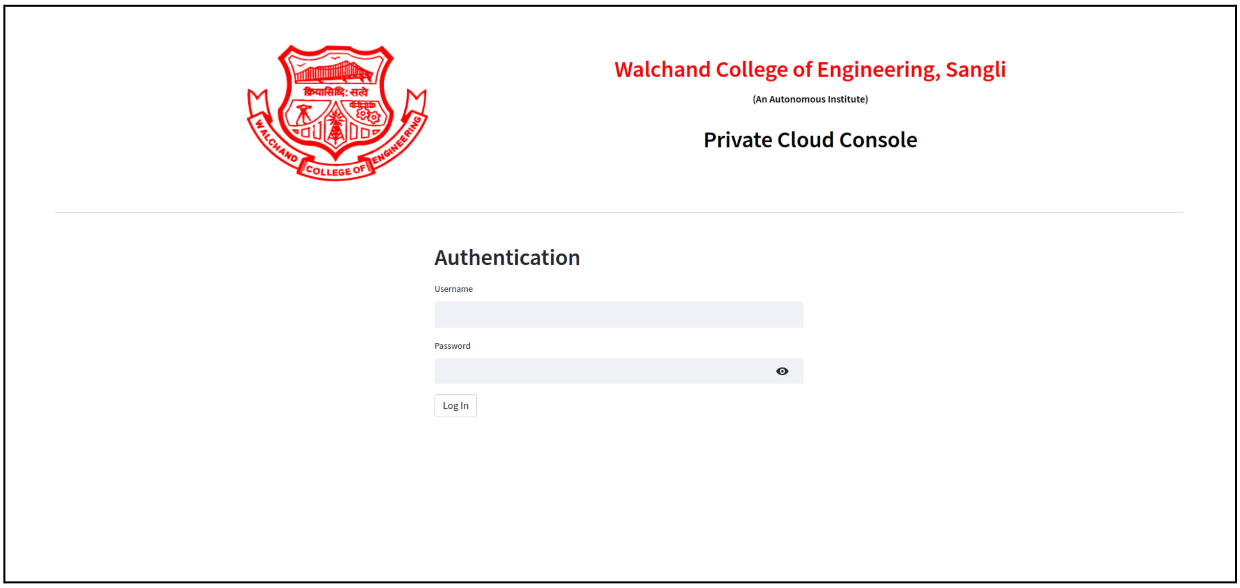
```
suyash@INTERN-SUYASHC:~$ ssh suyash@10.7.3.163
suyash@10.7.3.163's password:
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 5.15.0-60-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Mon May 29 09:56:33 AM UTC 2023

  System load:  0.26806640625     Temperature:           55.0 C
  Usage of /:   7.4% of 97.87GB   Processes:             170
  Memory usage: 2%                Users logged in:       1
  Swap usage:   0%                IPv4 address for eno1: 10.7.3.163


 * Introducing Expanded Security Maintenance for Applications.
   Receive updates to over 25,000 software packages with your
   Ubuntu Pro subscription. Free for personal use.

     https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings


Last login: Mon May 29 09:54:50 2023 from 10.7.5.131
suyash@openstack-1:~$ sudo snap install microstack --beta
[sudo] password for suyash:
microstack (beta) ussuri from Canonical✓ installed
suyash@openstack-1:~$
```

2. Initiate Microstack as a compute node following with the connection key generated by the control node.

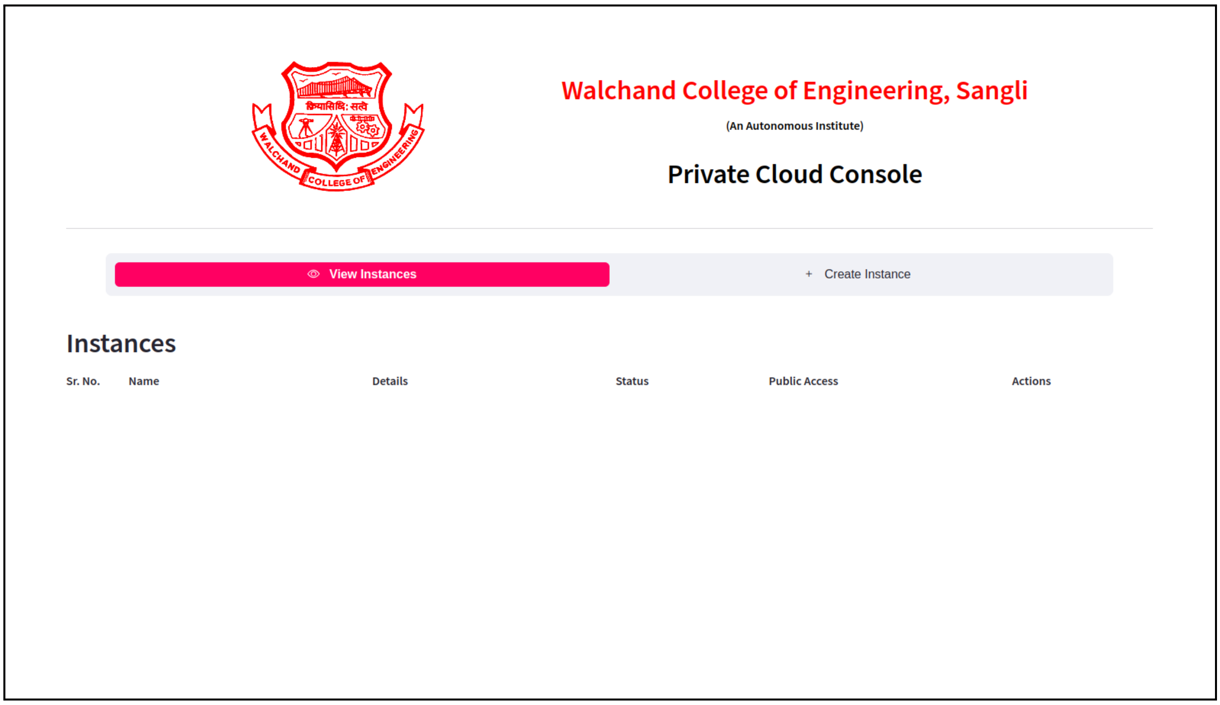
```
suyash@openstack-1:~$ sudo microstack init --auto --compute --join hKhob3N0bmFtZaoxMC43LjMuMTYyq2ZpbmdlcnByaW50xCB47zWa/W6Qh6jPo0HdXoIaY
cmV02SB1bnlkTWhVVDVXVlR5SVFxbDQ4elIyWWR5SXExNVZqeA==
2023-05-29 10:02:32,890 - microstack_init - INFO - Configuring clustering ...
2023-05-29 10:02:33,255 - microstack_init - INFO - Setting up as a compute node.
2023-05-29 10:02:38,047 - microstack_init - INFO - TLS certificates must be provided: config.tls.cacert-path, config.tls.cert-path, and
2023-05-29 10:02:39,317 - microstack_init - INFO - Configuring networking ...
2023-05-29 10:02:45,746 - microstack_init - INFO - Opening horizon dashboard up to *
2023-05-29 10:02:46,680 - microstack_init - INFO - Disabling local rabbit ...
2023-05-29 10:02:47,895 - microstack_init - INFO - Disabling local MySQL ...
2023-05-29 10:02:49,064 - microstack_init - INFO - Disabling the Placement service...
2023-05-29 10:02:50,203 - microstack_init - INFO - Disabling nova control plane services ...
2023-05-29 10:02:54,638 - microstack_init - INFO - Configuring nova compute hypervisor ...
2023-05-29 10:02:54,638 - microstack_init - INFO - Checking virtualization extensions presence on the host
2023-05-29 10:02:54,646 - microstack_init - INFO - Hardware virtualization is supported - KVM will be used for Nova instances
2023-05-29 10:02:58,960 - microstack_init - INFO - Configuring the Spice HTML5 console service...
2023-05-29 10:03:16,838 - microstack_init - INFO - Creating security group rules ...
2023-05-29 10:03:22,607 - microstack_init - INFO - Disabling Cinder services...
2023-05-29 10:03:31,853 - microstack_init - INFO - restarting libvirt and virtlogd ...
2023-05-29 10:03:53,465 - microstack_init - INFO - Complete. Marked microstack as initialized!
suyash@openstack-1:~$
```

Private Cloud Console:

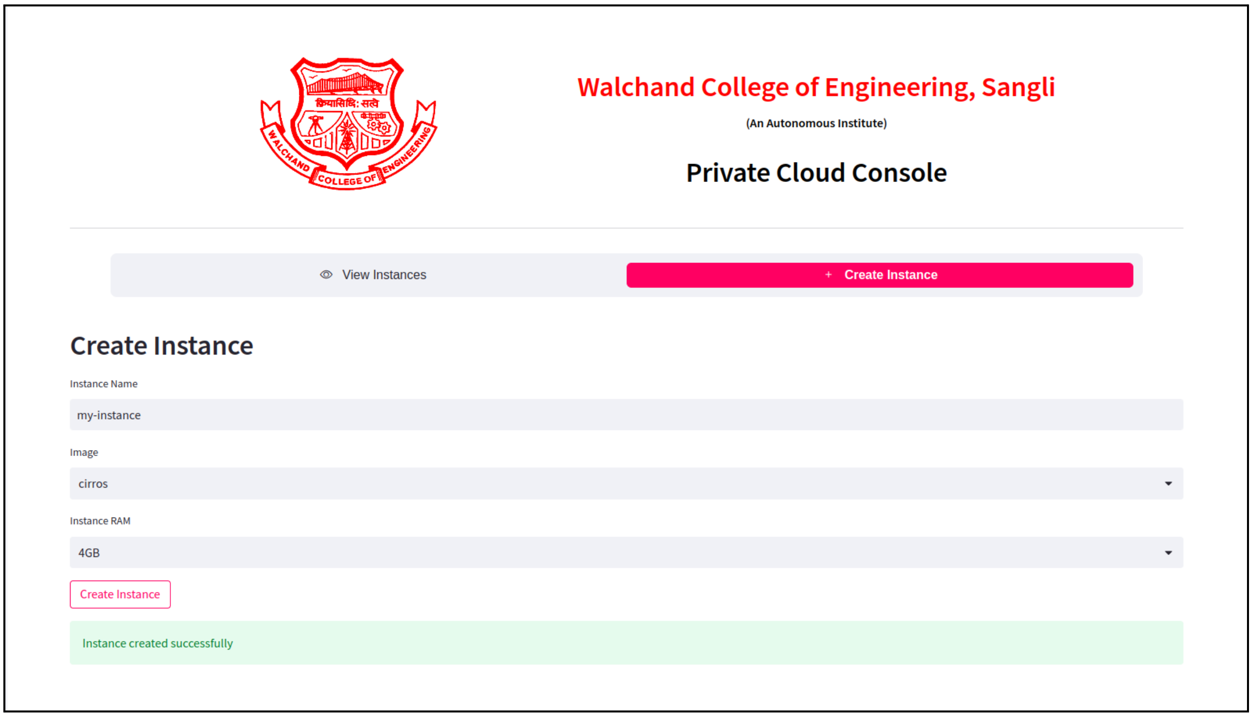
1. Login using Moodle Credentials



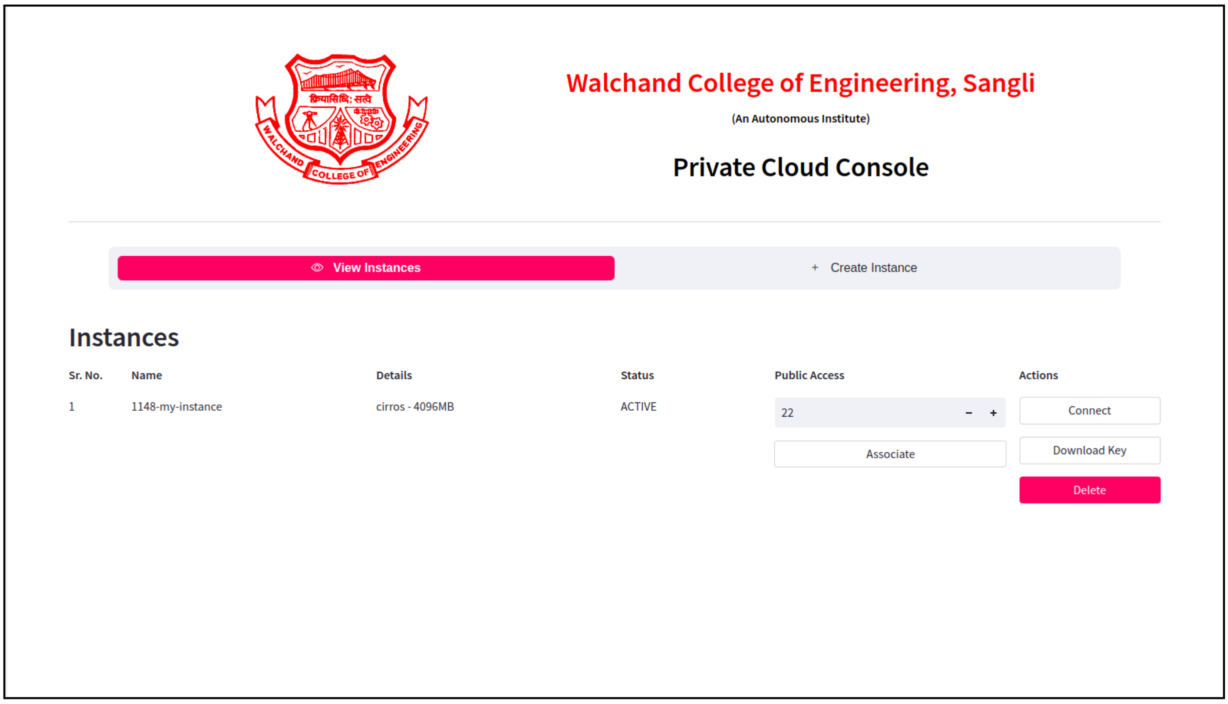
2. Following is the interface for viewing, configuring and creating instances.



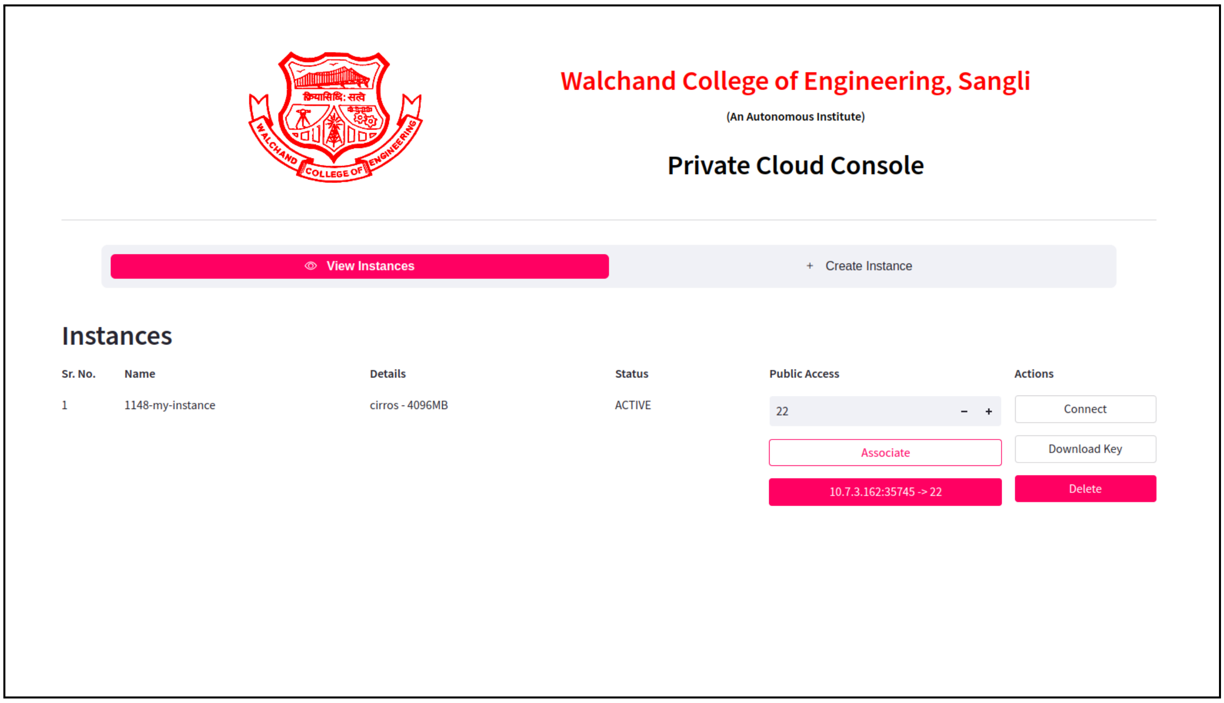
3. Create Instance by naming it and choosing image and RAM size.

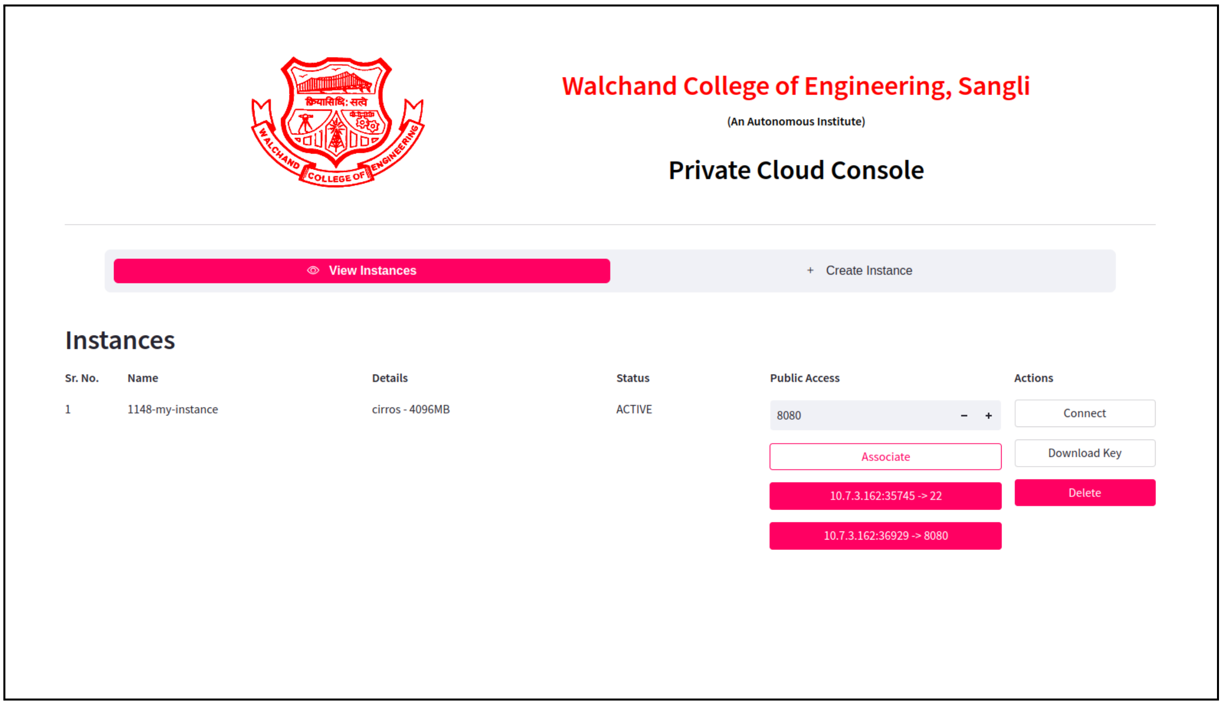


4. View instances that is created



5. Associate one or more public ports. Here maximum 3 public associations are allowed.





9. Conclusion and future scope

While the development and design of the private cloud infrastructure for college students serve as a significant step forward, there are several potential areas for future improvement and expansion

1. Advanced Resource Management: Future iterations of the private cloud infrastructure can incorporate advanced resource management techniques such as dynamic resource allocation, intelligent workload balancing, and automated scaling to further optimize resource utilization and enhance performance.

2. Integration of Emerging Technologies: Integration with emerging technologies like containers, serverless computing, and edge computing can expand the capabilities of the private cloud infrastructure, providing students with access to a wider range of cutting-edge technologies and deployment options.

3. Integration with Public Cloud Services: Developing mechanisms for integrating the private cloud infrastructure with public cloud services can provide students with a hybrid cloud environment, enabling them to seamlessly migrate or extend their services to public cloud platforms when required.

4. Collaboration and Sharing Features: Expanding the collaborative features of the private cloud infrastructure, such as facilitating resource sharing, collaborative development environments, and real-time communication tools, can further enhance teamwork and innovation among students.

5. Energy Efficiency: Future improvements can focus on optimizing the energy efficiency of the private cloud infrastructure by implementing power management techniques, workload consolidation, and energy-aware resource allocation algorithms, reducing the environmental impact and operational costs.

References

[1] Sarvesh Kumar, Omkara Murthy: Cloud Computing for Universities: A Prototype Suggestion and use of Cloud Computing in Academic Institutions, International Journal of Computer Applications (0975 – 8887) Volume 70– No, May 2013.

[2] Guoxia Zou: The Design Of Private Cloud Platform For Colleges And Universities Education Resources Based On Openstack, Dec 2015.

[3] Nikhil Wagh, Vikul Pawar and Kailash Kharat, Implementation of Stable Private Cloud using OpenStack with Virtual Machine Results, International Journal of Computer Engineering and Technology, 10(2), 2019, pp. 258-269.

[4] SudhaDevi, D. & Yamuna Devi, Loganathan & Thilagavathy, K. & P, Aruna & Priya, Navya & Vasantha, s. (2013). Private Cloud in Educational Institutions: An Implementation using UEC. International Journal of Computer Applications. 78. 8-12. 10.5120/13451-1059.