Department of Information Technology
NBA Accredited

A.P. Shah Institute of Technology

G.B.Road,Kasarvadavli, Thane(W), Mumbai-400615

UNIVERSITY OF MUMBAI

Academic Year 2022-2023

A Project Report on

Orchestrating dynamically scalable container based lab environment for an Educational Institute.

Submitted in partial fulfillment of the degree of
Bachelor of Engineering(Sem-8)
in
INFORMATION TECHNOLOGY

By
Vedant Patil(19104065)
Mudra Limbasia(19104058)
Anvit Mirjurkar(19104059)

Under the Guidance of
Dr. Kiran Deshpande

1.Project Conception and Initiation

1.1 Abstract

- Containers are the upcoming technology in the field of cloud computing, they provide rapid and flexible deployment, fine-grained resource sharing, and lightweight performance isolation.
- Educational institutes tend to have various laboratories pertaining to diverse technical aspects by which students learn the upcoming technologies which make them industry apt. Educational institutes are obliged to have the software in laboratories ready for student operation. The task of preparing the labs is a colossal exercise for large institutes, as they would have to manually install all the dependencies required for all the laboratories.

1.2 Objectives

In this project implementation we intend to fulfill the following objectives:

- To model and orchestrate container based environment for IT laboratories to provide hassle free lab environments to students.
- To provide dynamic scalability to container based lab environment through Kubernetes cluster deployed over cloud.
- To model and build user friendly interface for using container based lab environment created which will provide ease to professors and students during lab hours.
- To extend use of dynamically scalable container based lab environment created for collaborative project management and assessment.

1.3 Literature Review

Sr. No.	Title	Key Findings	Year
1.	Building Modern Clouds: Using Docker, Kubernetes and Google Cloud Platform	• This paper discusses the use of Docker and Kubernetes in building modern cloud infrastructure. • Docker is used for building, shipping, and running applications, while Kubernetes is an automated platform for container management, deployment, and scaling. • The Google Cloud Platform is used to deploy containers on Kubernetes Engine for rapid application development and management.	2019

1.3 Literature Review

Sr. No.	Title	Key Findings	Year
2.	Server Scalability Using Kubernetes	• The study focuses on applying scalability to Kubernetes and evaluating its performance in handling increasing numbers of concurrent users accessing academic data. • The results show that scalability enabled on multiple servers can reduce container load due to the distribution of loads to containers scattered across many workers, but the server response time may take longer due to the overhead delay of scaling containers.	2019

1.3 Literature Review

Sr. No.	Title	Key Findings	Year
3.	Deploying Microservice Based Applications with Kubernetes: Experiments and Lessons Learned	• The paper explores the use of Kubernetes for building microservice-based applications and evaluates its effectiveness in achieving high availability. • The study focuses on the default configuration of Kubernetes and identifies potential issues with service outage, highlighting the need for further investigation and optimization. • The paper also discusses lessons learned from deploying microservices with Kubernetes, such as the importance of proper resource allocation, network configuration, and security measures.	2018

1.5 Scope

The scope of our project is:

- The implementation of an orchestrating system using Kubernetes for dynamically scaling container-based lab environment in an educational institute.
- This includes the creation of a container image registry, automated deployment and scaling of lab environments, integration with an identity management system for user authentication and access control, and monitoring and logging for system maintenance and troubleshooting.
- The system will be designed to support multiple courses and allow for customization of lab environments based on specific course requirements.

1.6 Technology stack

- Kubernetes
- Google Cloud Platform
- GitHub
- JupyterHub

1.7 Benefits for environment & Society

The project can bring several benefits to the environment and society, such as:

- **Reduced energy consumption:** By consolidating multiple servers into fewer host servers and using containerization, the project can lead to reduced power requirements and energy consumption, which can contribute to environmental sustainability.
- **Improved accessibility to education:** By providing a scalable and flexible lab environment, the project can improve access to education for students, especially those who are unable to attend physical labs. This can lead to increased educational opportunities and ultimately contribute to social development.

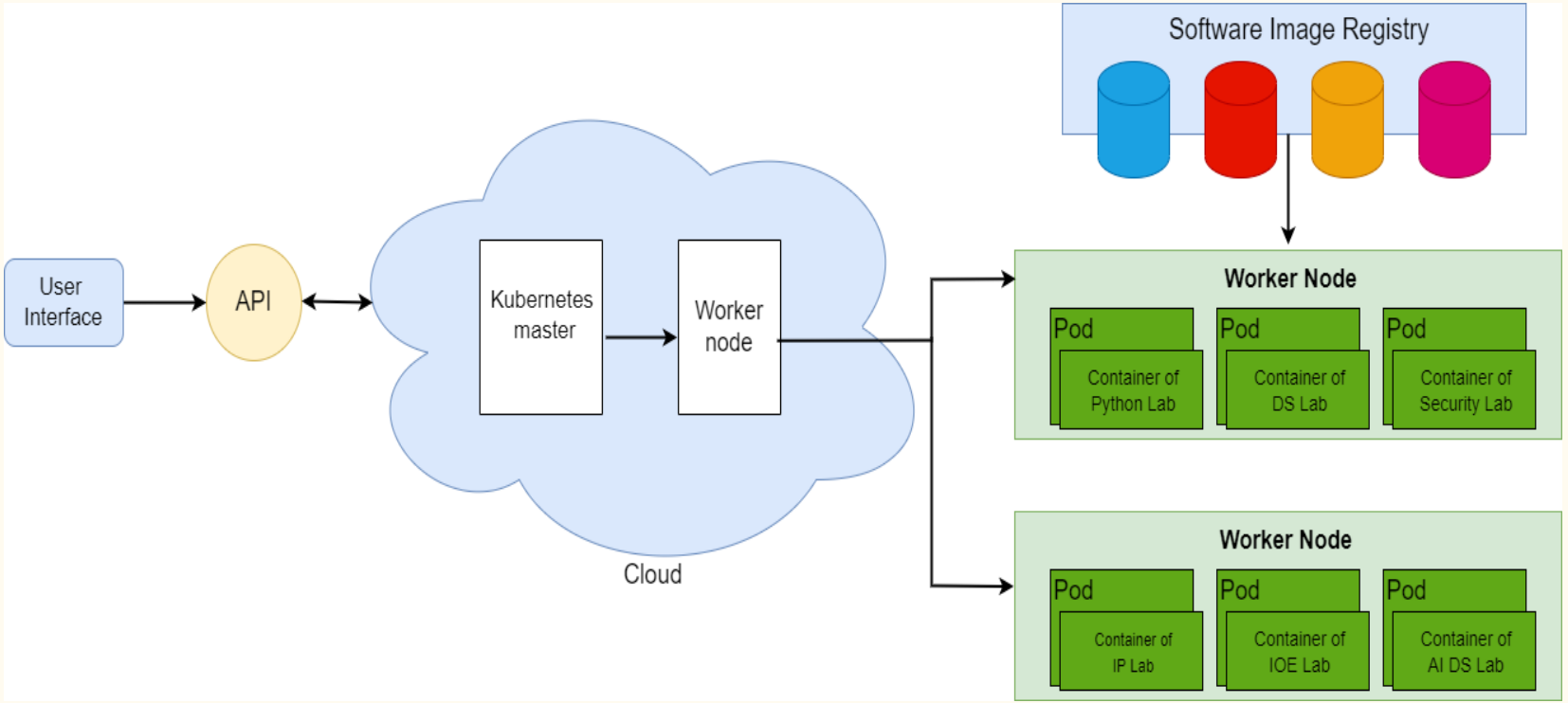
1.7 Benefits for environment & Society

The project can bring several benefits to the environment and society, such as:

- **Reduced waste:** Containerization enables more efficient use of resources and reduces the waste associated with traditional virtualization approaches. This can contribute to a reduction in electronic waste and environmental impact.

2. Project Design

2.1 Proposed System

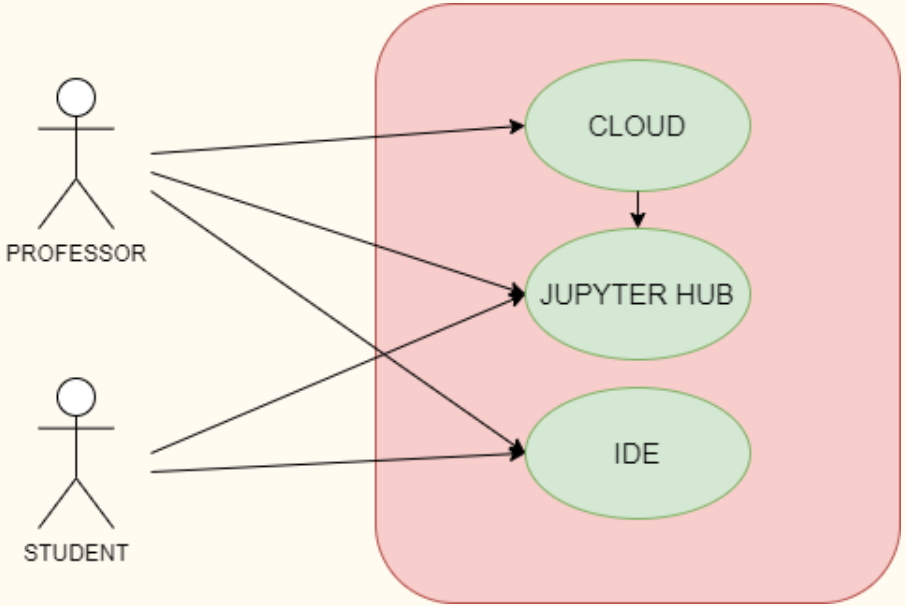


2.2 Design(Flow Of Modules)

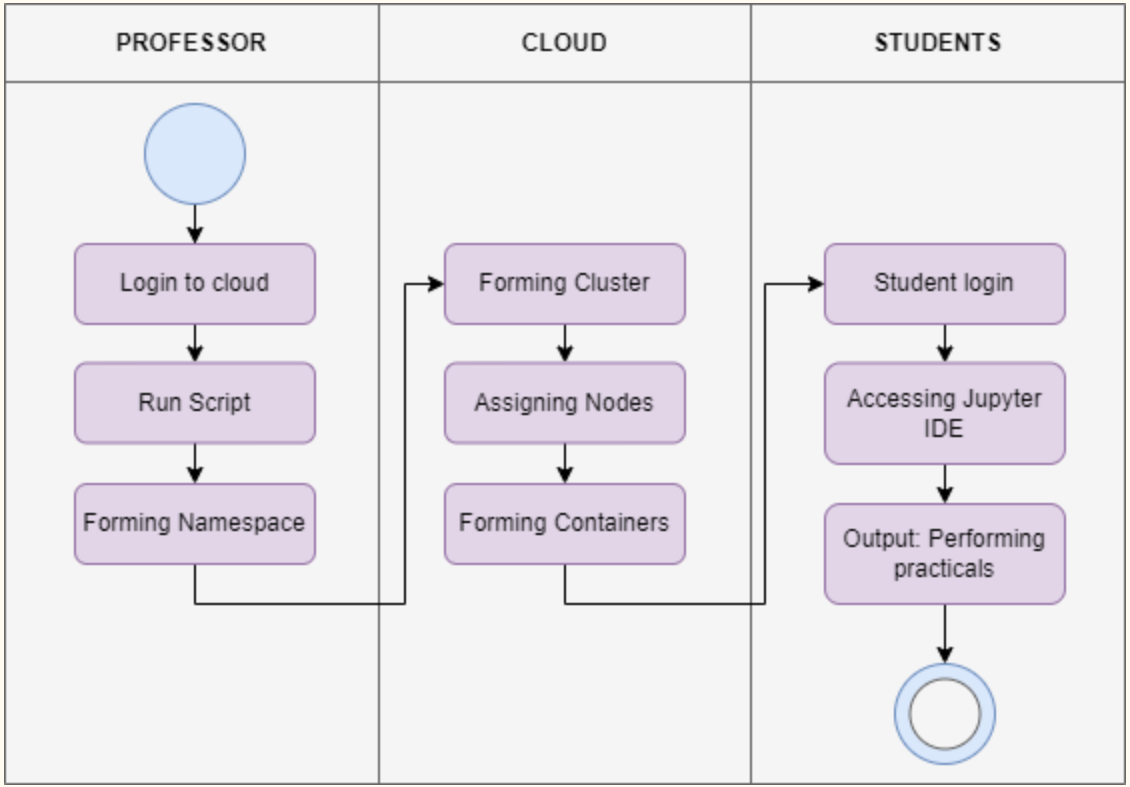
- Professor: The professor logs into the Google Cloud Platform and creates the Kubernetes cluster. Professor then clones the GitHub repository where the source files of the designed system is stored. Once the designed system is cloned on the cloud platform, the professor will run the script for starting the JupyterHub lab environment.
- Students: Once the script is successfully compiled, the command line gives an output of the IP address of the server where the JupyterHub worker node is deployed. The students have to enter the IP address in the browser window to access the designed system.

2.3 Description Of Use Case

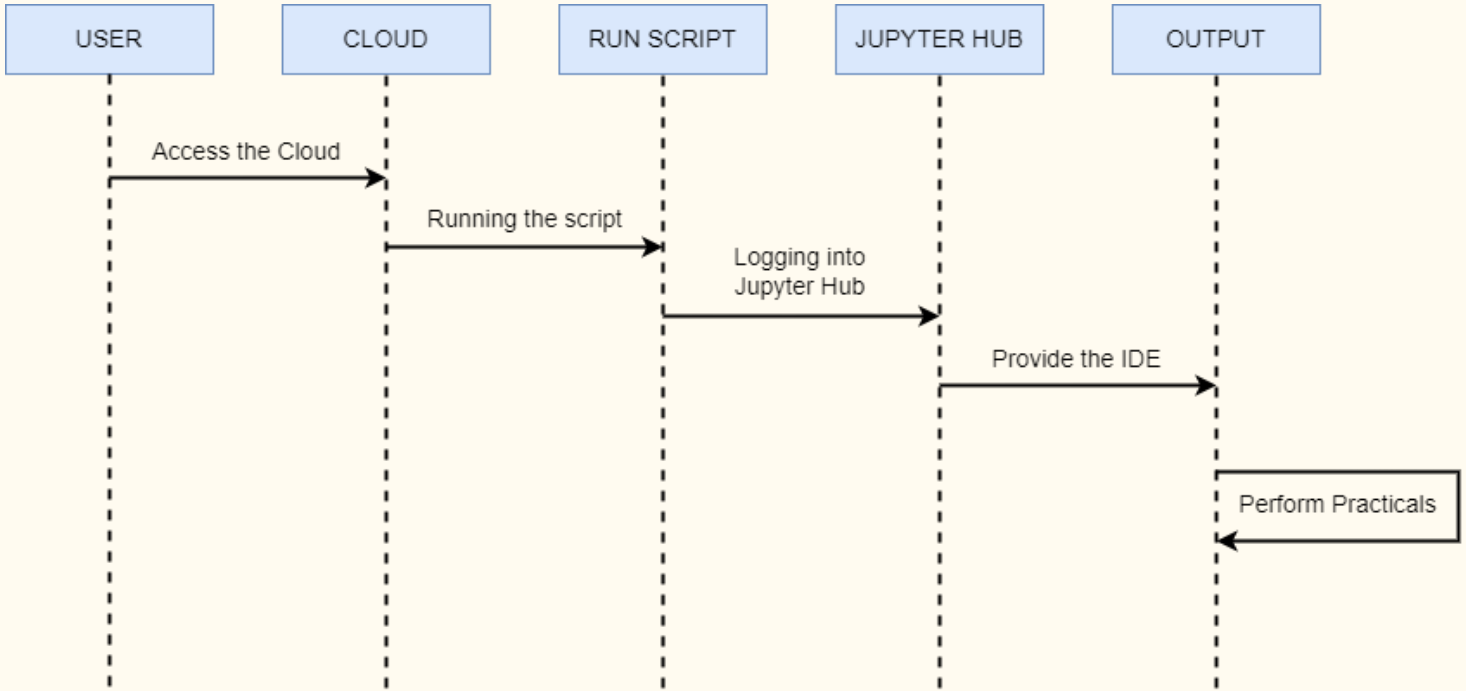
- The use case Diagram illustrates the different roles of users such as Professors and Students, their relationship with the system, and the actions they can perform with it.
- The system will serve as a platform for conducting laboratory experiments with various programming languages.



2.4 Activity diagram

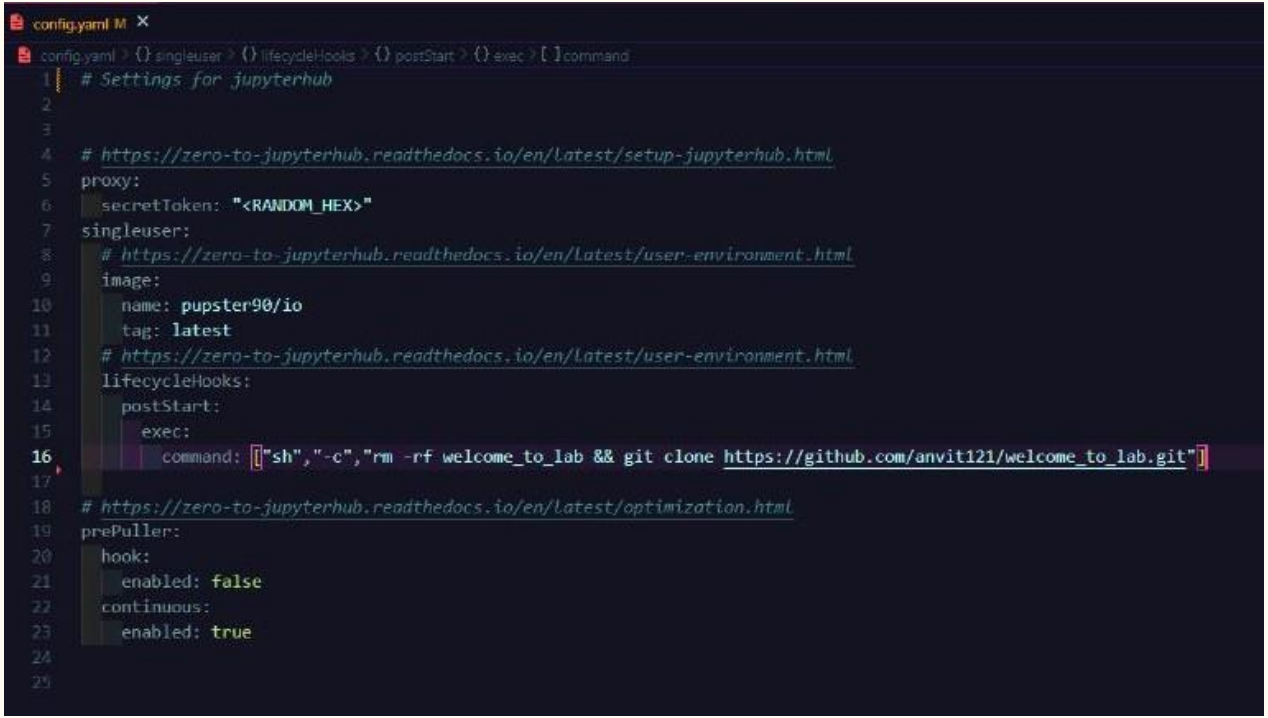


2.5 Sequence Diagram

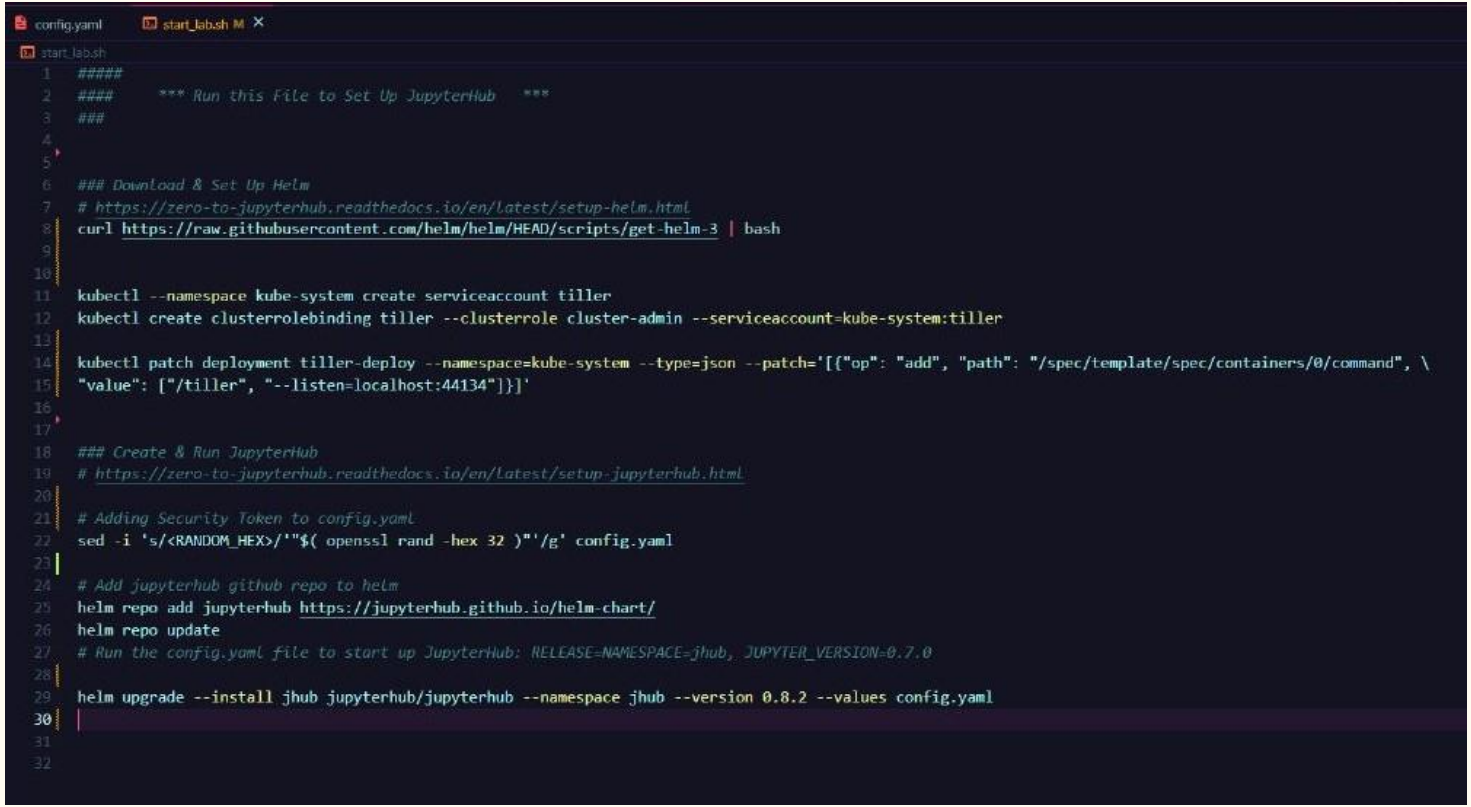


3. Implementation

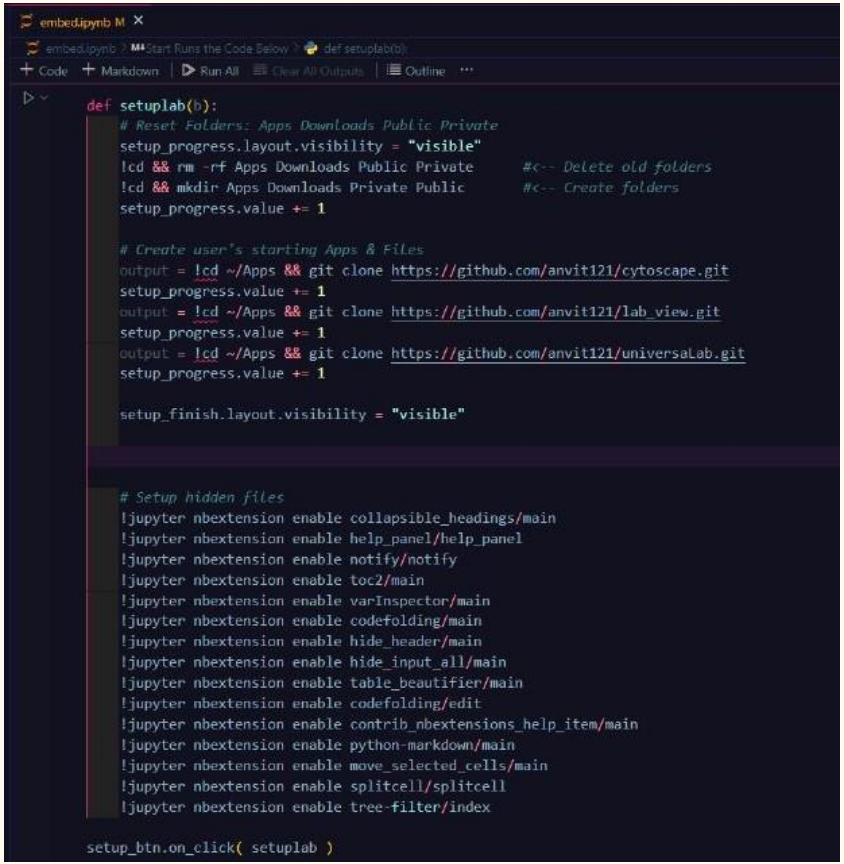
3.1 Implementation – Code Snippets



Configuration file of the system.



This code initializes the project and provides the user with the server's IP address to access the JupyterHub platform.



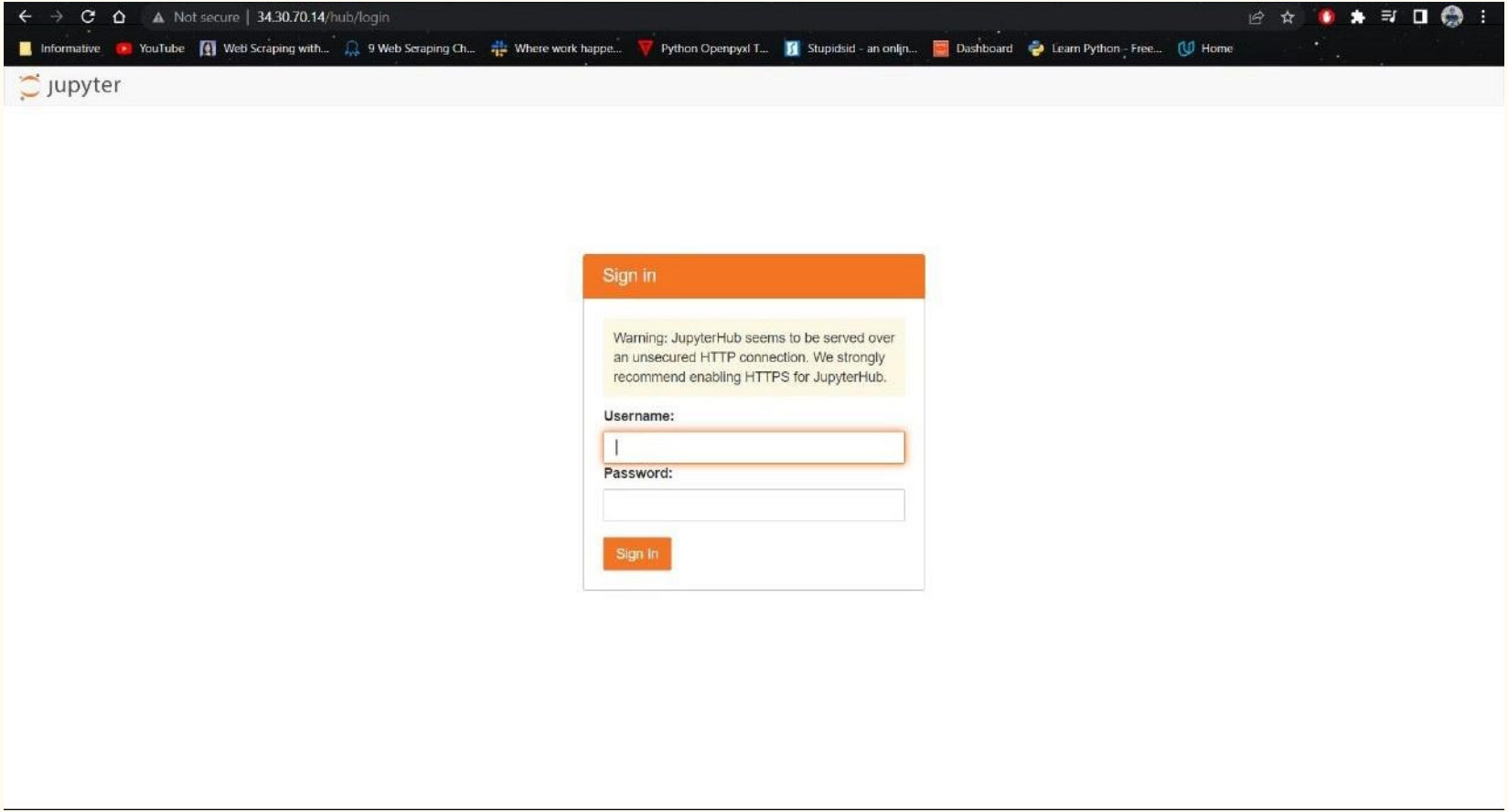
This code displays the folder structure in JupyterHub, which is a web-based interactive computing environment for Jupyter notebooks.

4. Testing

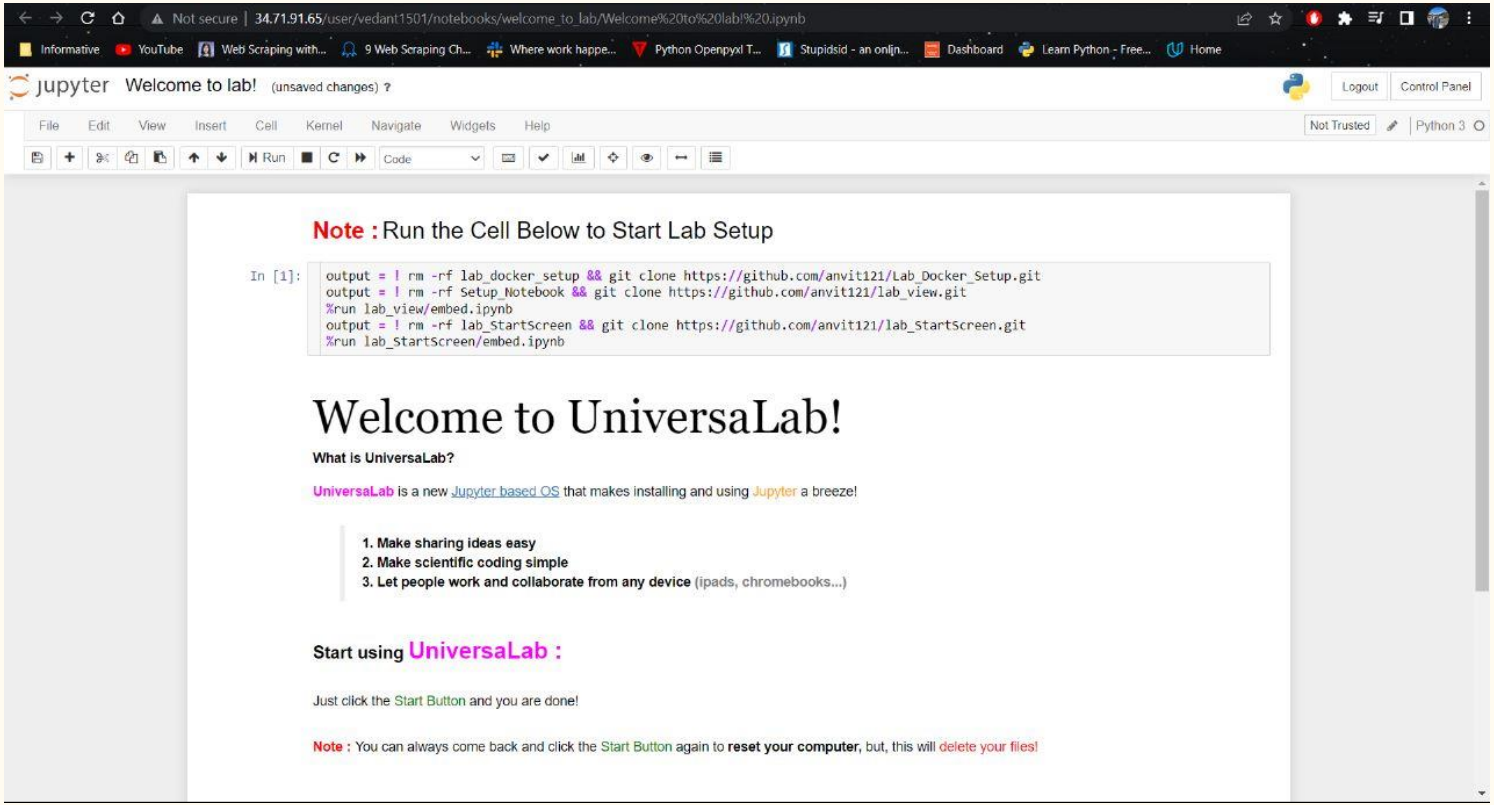
4.1 Integration Testing

Module	Integration Testing objective	Expected Result	Actual Result	Pass/Fail
Kubernetes Cluster Setup	Verify that the cluster is integrated with other components	Cluster is integrated with other components and can communicate with them	Cluster communicates successfully	Pass
JupyterHub Deployment	Verify that JupyterHub is integrated with other components	JupyterHub is integrated with other components and can access and manage them	JupyterHub is successfully integrated	Pass

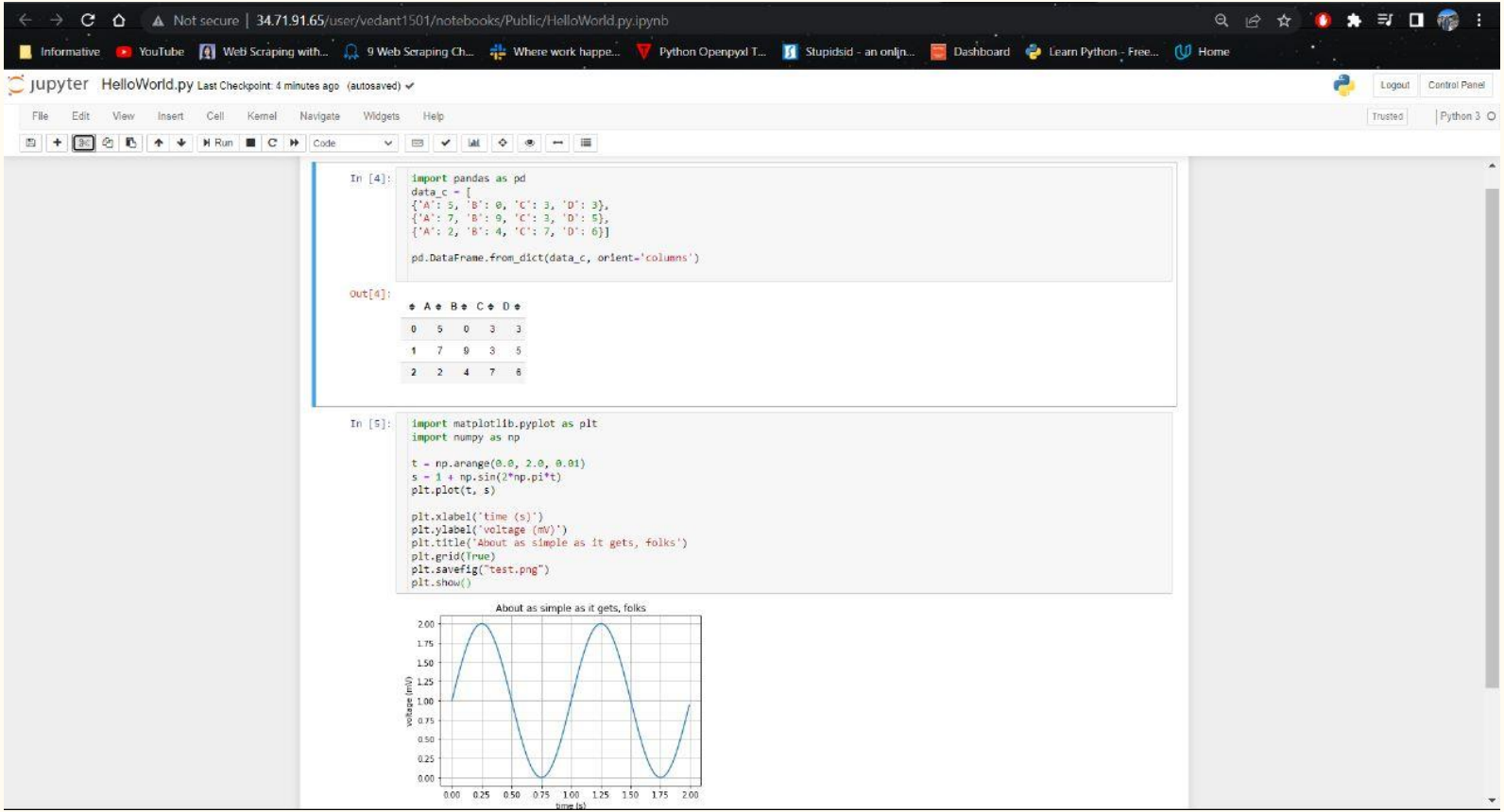
5. Result



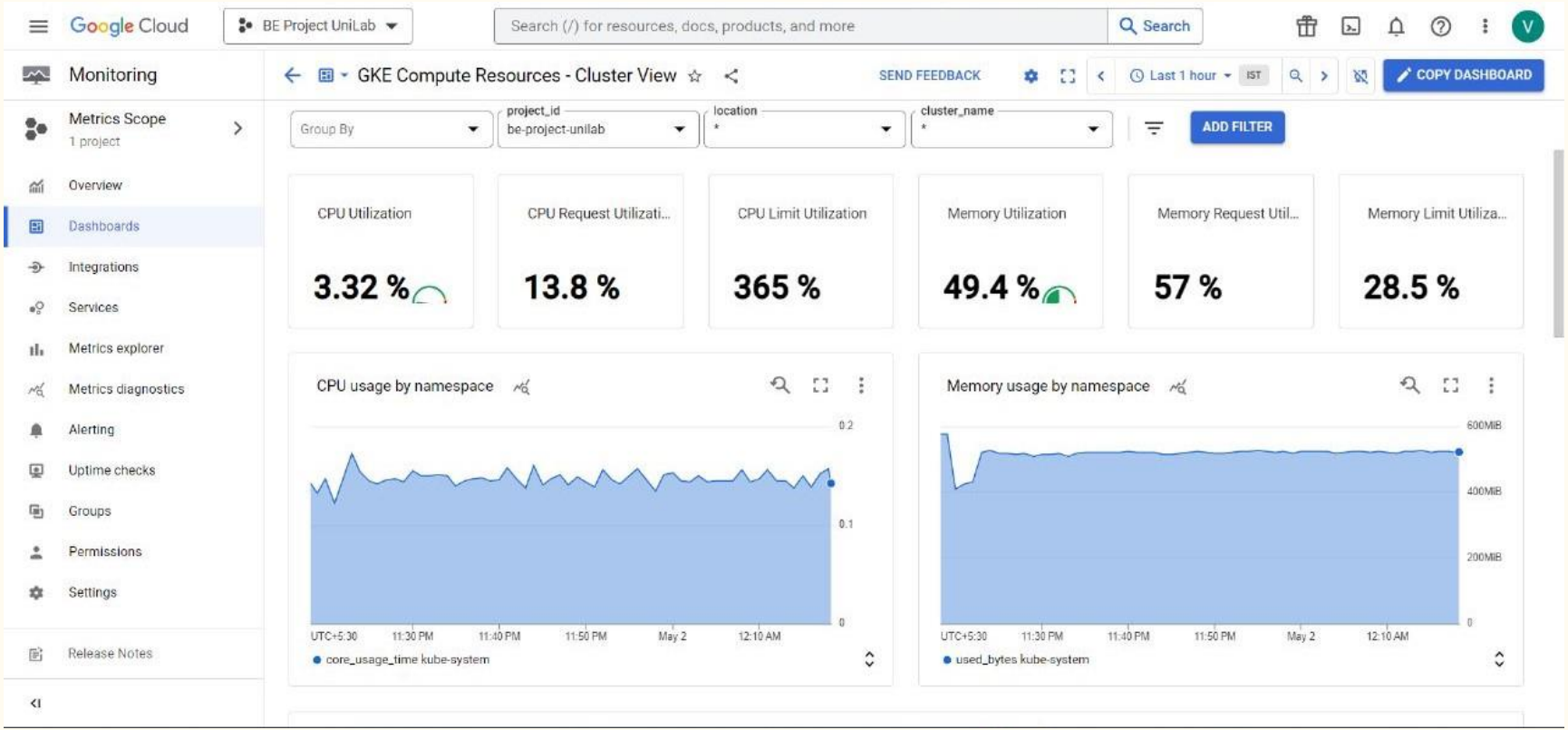
Accessing system designed through authentication



Snippet reflecting usage of software environment deployed over kubernetes pod



Snippet reflecting usage of Python coding environment



Snippet representing GKE Cluster view

6. Conclusion and Future Scope

6.1 Conclusion

- Container orchestration provides a powerful tool for Educational Institutes to build and manage dynamically scalable lab environments. By automating the deployment, management, and scaling of containerized workloads, container orchestration can help Educational Institutes save time and reduce costs while providing a secure and reliable infrastructure for students and faculties. Flexibility and Customization of container orchestration allow Educational Institutes to support a wide range of workloads, from traditional teaching to collaborative project management. Thus, the system designed and implemented is an excellent solution for Educational Institutes that are looking to build and manage scalable and efficient lab environments.

6.1 Future Scope

- The solution proposed can be extended to use this platform as a project collaboration environment, where all the necessary source files needed for the project will be stored on the platform. These files can be pulled onto any machine thereby creating a suitable environment that will enable the project to be run on any desired machine, without having to install all the dependencies again on a new machine.
- In addition, experiments performed successfully by students can be saved and used by later batches as a reference to debug any difficulties faced during execution through the effective use of GitHub. Moreover, the system designed can be modified so that the students will only be able to access the lab environments that are mapped to their respective academic year and semester.

References

- J. Shah and D. Dubaria, “Building Modern Clouds: Using Docker, Kubernetes and Google Cloud Platform,” 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC), 2019, pp. 0184-0189, doi: 10.1109/CCWC.2019.8666479.”
- Lily Puspa Dewi, Agustinus Noertjahyana, Henry Novianus Palit and Kezia Yedutun, “Server Scalability Using Kubernetes”, IEEE Xplore Digital Library 2019. [Online]. Available: https://ieeexplore.ieee.org/document/9024501.
- L. Abdollahi Vayghan, M. A. Saied, M. Toeroe and F. Khendek, “Deploying Microservice Based Applications with Kubernetes: Experiments and Lessons Learned,” 2018 IEEE 11th International Conference on Cloud Computing (CLOUD), 2018, pp. 970- 973, doi: 10.1109/CLOUD.2018.00148.

Paper Publication

Paper entitled “**Orchestrating dynamically scalable container-based lab environment for an Educational Institute**” is submitted at “**International Conference on Advanced Computing Technologies and Applications-2023**” by “**Vedant Patil, Mudra Limbasia, Anvit Mirjurkar,** and **Dr. Kiran Deshpande**”.

Thank You