PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY
Department of Information Technology
(NBA Accredited)

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

Group No. 17

Vedant Jitendra Patil – 19104065
Anvit Dinesh Mirjurkar – 19104059
Mudra Dinesh Limbasia – 19104058

Project Guide
Dr. Kiran Deshpande

Contents

- **Abstract**
- **Introduction**
- **Objectives**
- **Literature Survey**
- **Problem Definition**
- **Technological Stack**
- **Proposed System Architecture/Prototype**
- **Suggestions given during Review 1**

1. Abstract

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, manual installation with dependencies are time consuming and tedious. The aim of this project is to orchestrate dynamically scalable container based lab environment for educational institutes which incorporate various lab dependencies that contain all the necessary files needed for a perfectly working laboratory, and can be used whenever required. Containers are the upcoming technology, they provide rapid and flexible deployment, fine-grained resource sharing, and lightweight performance isolation.

2. Introduction

- In the current scenario, installing and updating software and technologies in multiple systems in IT labs is a tedious task, as the staff has to manually do the whole process of installation in each and every system.
- The fix of this problem is the implementation of an user-friendly platform which works conjointly with Kubernetes containers deployed on the cloud
- Dynamically scalable container based lab environment will eliminate the demand of manually installing, and updating software in individual systems, hence making it effortless for professors as well as the students.

3. 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.

3. Objectives

- 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.

4. Literature Survey

Sr. No.	Paper Name	Paper Summary
1.	J. Shah and D. Dubaria, ”Building Modern Clouds: Using Docker, Kubernetes and Google Cloud Platform,” 2019 IEEE 9th Annual Computing & Communication Workshop & Conference (CCWC), 2019, pp. 0184-0189, doi: 10.1109/CCWC.2019.8666479.	To develop and build a modern cloud infrastructure or DevOps implementation than both Docker and Kubernetes have revolutionized the era of software development and operations. Docker is used to build, ship and run any application anywhere. These containers can be used to make deployments much faster. Containers use less space, are reliable and are very fast. Docker Swarm helps to manage the docker container. Kubernetes is an automated container management, deployment and scaling platform.. Kubernetes provides key features like deployment, easy ways to scale, and monitoring.

4. Literature Survey

Sr. No.	Paper Name	Paper Summary
2.	P. Dewi, A. Noertjahyana, H. N. Palit and K. Yedutun, ”Server Scalability Using Kubernetes,” 2019 4th Technology Innovation Management and Engineering Science International Conference (TIMES-iCON), 2019, pp. 1-4, doi: 10.1109/TIMESiCON47539.2019.9024501.	Kubernetes is commonly used to automatically deploy and scale application containers. The scalability of these application containers can be applied to Kubernetes with several supporting parameters. This research focuses on applying the scalability in Kubernetes and evaluating its performance on overcoming the increasing number of concurrent users accessing academic data. This research employed 3 computers: one computer as the master node and two others as worker nodes. Two scenarios were designed to evaluate the CPU load on single and multiple servers. On multiple servers, the server scalability was enabled to serve the user requests.

4. Literature Survey

Sr. No.	Paper Name	Paper Summary
3.	H. V. Netto, A. F. Luiz, M. Correia, L. de Oliveira Rech and C. P. Oliveira, "Koordinator: A Service Approach for Replicating Docker Containers in Kubernetes," 2018 IEEE Symposium on Computers and Communications (ISCC), 2018, pp. 00058-00063, doi: 10.1109/ISCC.2018.8538452.	Container-based virtualization technologies such as Docker and Kubernetes are being adopted by cloud service providers due to their simpler deployment, better performance, and lower memory footprint in relation to hypervisor-based virtualization. Kubernetes supports basic replication for availability, but does not provide strong consistency and may corrupt application state in case there is a fault. This paper presents a state machine replication scheme for Kubernetes that provides high availability and integrity with strong consistency.

5. Problem Definition

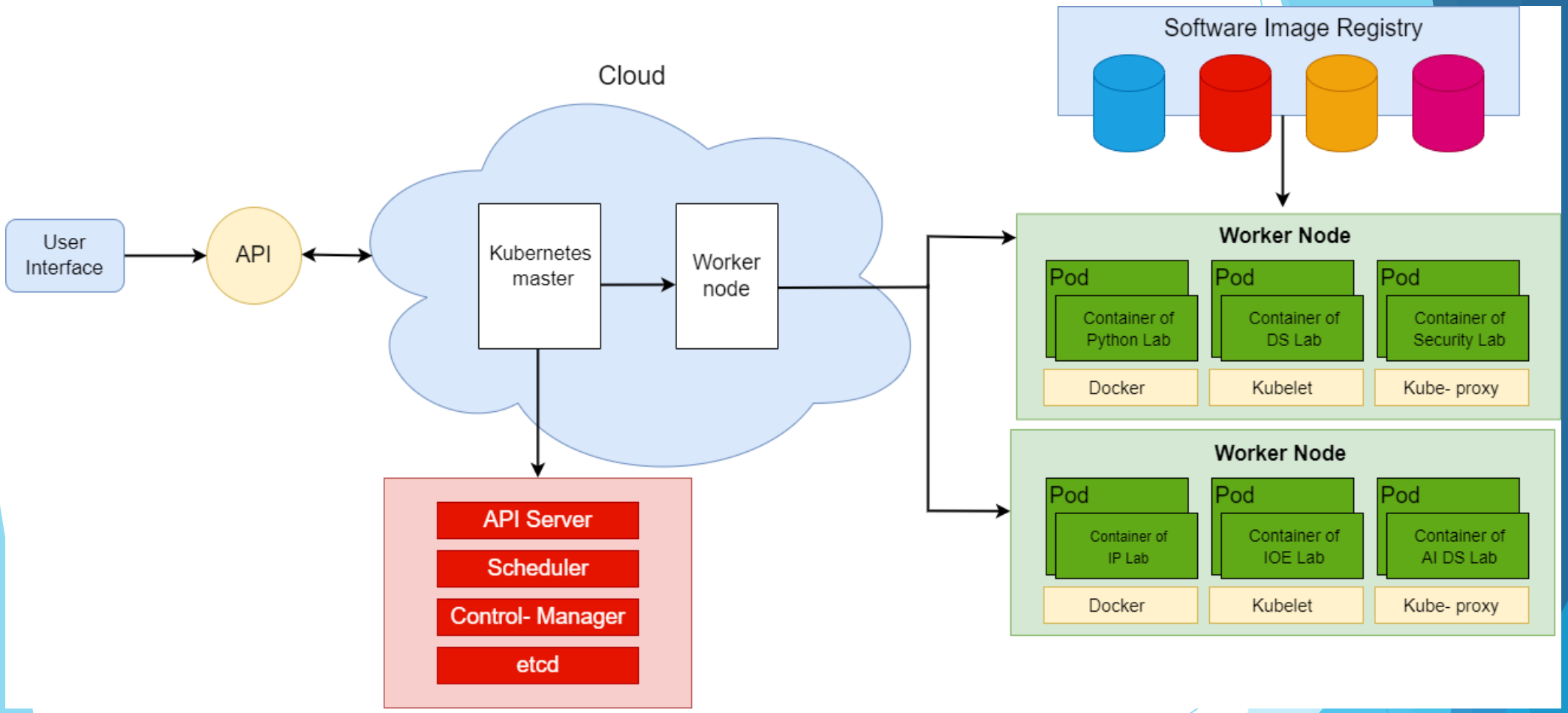
- Currently, upgrading software and technologies across numerous systems in IT laboratories is a time-consuming effort since staff members must manually complete the installation procedure across each and every system.

- Educational institutes in the field of Engineering and technology are liable to have various labs in order to educate students with the new and upcoming technologies which is necessary for their growth in the technical industry. It is a laborious task for any institute to manually set up labs for each and every technology.

6. Technology Stack

- Kubernetes
- OpenStack(or any other free Cloud Platform)
- Ansible

7. Proposed System Architecture/Prototype



7. Proposed System Architecture/Prototype

- **Master Node**
  - Key Kubernetes processes that run and manage the cluster. Key features of Master node:
    - API Server: User interface, API, and command line interface are the entry points to the Kubernetes cluster
    - Controller Manager: It controls and keeps track of the cluster's containers
    - Scheduler: According to the schedule of the application, it decides which worker nodes will be employed when.
    - Etcd: The state of the cluster is stored in a key-value store.

7. Proposed System Architecture/Prototype

- **Worker Node**
  - It controls the containers and pods.
    - Key Features:
    - Kubelet - It manages communication with the master and registers messages.
    - Pods - There are several pods per worker node.
    - Containers –It operates within the pods. Along with the OS and other resources required for the application to function, it is the location where the application operates.

Thank You...!!