



# CONESTOGA

Connect Life and Learning

<b>Student Names:</b>	Shivani Lahori (8950024) – Team Leader Kirtirajsinh Solanki (8946269) Akshay Tembulkar (8813420) Heli Bhavsar (8839005) Dhvanil Patel (8952685)
<b>Deliverable:</b>	Capstone Implementation
<b>Course Name:</b>	SYST8201 - Winter 2024 - Section 2
<b>Date Due:</b>	9 <sup>th</sup> April, 2024
<b>Group:</b>	Group 5

---

# Table of Contents

About This Document .....	4
Project Overview .....	4
Requirements and Design.....	4
<b>Server infrastructure</b> .....	4
IP structures and DNS .....	5
Configuration Steps: .....	6
<b>Setting Up OpenStack Compute, Controller, and Block Storage Nodes</b> .....	6
<b>Setting up Web Servers</b> .....	8
<b>Setting up Database Servers</b> .....	8
<b>Setting up Hosts</b> .....	9
Configuring Monitoring Server with Zabbix.....	9
Project implementation .....	9
Screenshots .....	10
<b>OpenStack infrastructure:</b> .....	10
Admin Panel of our application: .....	18
Front-end: .....	21
Dockerize application: .....	24
Automatic Fail over mechanism:.....	24
High Availability Using Docker Swarm:.....	25
Zabbix Monitoring: .....	27
Network Diagram: .....	30
Project Testing .....	31
Functional Testing:.....	31
Performance Testing: .....	31
Security Testing:.....	31
High Availability Testing:.....	31
Database Server Testing.....	31
Functional Testing:.....	31
Performance Testing: .....	31
Security Testing: .....	31
Data Integrity Testing: .....	31
Monitoring Server Testing .....	31
Functional Testing:.....	31
Performance Testing: .....	32

---

<b>Security Testing:</b> .....	<b>32</b>
<b>Alerting and Notification Testing:</b> .....	<b>32</b>
Schedule Detailed task list is captured as part of the Detailed Business Implementation Task List. A sample matrix is captured here to identify key milestones.....	32
<b>Roles &amp; Responsibilities</b> .....	<b>34</b>
<b>Reference:</b> .....	<b>35</b>

---

# About This Document

This document contains detailed information about all the configurations used during the development and construction of the project. These configurations cover various aspects, including software versions, environment settings, dependencies, and any other relevant parameters necessary for building and running the project successfully.

## Project Overview

Our project aims to develop a comprehensive web application for event management, providing users with a seamless platform to plan, organize, and execute various types of events. From corporate conferences to weddings and parties, our application will streamline the event planning process, making it easier and more efficient for both event organizers and attendees.

## Requirements and Design

User Registration and Authentication: Allow users to create accounts, log in securely, and manage their profiles.

Event Creation and Management: Enable users to create new events, set event details (date, time, location, etc.), and manage event settings.

Frontend: HTML, CSS, JavaScript, jQuery, Ajax, bootstrap (or similar frontend framework)

Backend: PHP, MySQL

Containerization: Docker for containerization of application components

Deployment: Kubernetes for container orchestration, deployed on OpenStack infrastructure

Monitoring and Logging: Zabbix for monitoring application performance

Agile Methodology: Adopt an Agile development approach with sprints and regular iteration cycles.

Version Control: Use Git for version control, with repositories hosted on platforms like GitHub or Bitbucket.

Collaboration Tools: Utilize project management and communication tools such as Project Planner, Microsoft Teams for team collaboration and coordination.

Testing: Implement end-to-end testing to ensure application reliability and quality.

Security: Follow best practices for web application security, including secure authentication, data encryption, and protection against common vulnerabilities such as SQL injection and cross-site scripting (XSS).

## Server infrastructure

We utilized the following specifications for the virtual machines (VMs) in our OpenStack infrastructure setup:

### Controller Node:

- **CPU:** 4 CPUs
- **RAM:** 16 GB
- **Storage:** 100GB
- **Network Interfaces:** 2
- **Operating System:** Ubuntu 22.04.1 (amd64)
- **ISO:** ubuntu-22.04.1-live-server-amd64.iso

### Compute Node:

- **CPU:** 4 CPUs
- **RAM:** 16GB
- **Storage:** 100GB
- **Network Interfaces:** 2
- **Operating System:** Ubuntu 22.04.1 (amd64)

- 
- ISO: ubuntu-22.04.1-live-server-amd64.iso

**Block Storage Node:**

- CPU: 4 CPUs
- RAM: 16GB
- Storage: 150GB
- Network Interfaces: 2
- Operating System: Ubuntu 22.04.1 (amd64)
- ISO: ubuntu-22.04.1-live-server-amd64.iso

These configurations were chosen to ensure optimal performance, resource allocation, and compatibility with the OpenStack services being deployed.

**Software Requirements:**

- Operating System: Ubuntu 22.04.1 (amd64)
- Monitoring Tool: Zabbix (latest version).

**Dependencies:**

- Ensure network connectivity between all servers.
- Configure DNS resolution for hostnames.

## IP structures and DNS

The IP structure and DNS for all the machines is as below:

Role	Machine Name	IP Address	Subnet Mask	Default Gateway
Controller Node	Controller	10.173.143.11	255.255.255.0	10.173.143.1
Compute Node	Compute	10.173.143.12	255.255.255.0	10.173.143.1
Block Storage	storage_block	10.173.143.22	255.255.255.0	10.173.143.1
Web server	webserver1	10.173.143.112	255.255.255.0	10.173.143.1
Database server	dbserver	10.173.143.113	255.255.255.0	10.173.143.1
Zabbix Moniroring server	monitorserver	10.173.77.16	255.255.255.0	10.173.77.1

---

# Configuration Steps:

## Setting Up OpenStack Compute, Controller, and Block Storage Nodes

### 1. Preparing the Environment:

- Ensure that you have three separate Ubuntu virtual machines provisioned for each role: Compute, Controller, and Block Storage nodes.
- Verify that each node has internet connectivity by pinging a reliable external IP address

### 2. Configuration of Controller Node:

#### Step 1: Initial Setup:

- Install Ubuntu on the virtual machine with the specified configurations.
- Set the management interface with IP address **10.173.143.11**, network mask **255.255.255.0**, and default gateway **10.173.143.1**.
- Set the hostname of the node to **controller**.
- Edit the **/etc/hosts** file accordingly.

#### Step 2: Installing and Configuring Chrony:

- `apt install chrony`
- Allow other nodes to connect to the Chrony daemon by editing `/etc/chrony/chrony.conf` and adding `Allow 10.173.143.0/24`.

### 3. Configuration of Compute Node:

#### Step 1: Initial Setup:

- Install Ubuntu on the virtual machine with the specified configurations.
- Set the management interface with IP address **10.173.143.12**, network mask **255.255.255.0**, and default gateway **10.173.143.1**.
- Set the hostname of the node to **compute**.
- Edit the **/etc/hosts** file accordingly.

#### Step 2: Installing and Configuring Chrony:

- Configure the **chrony.conf** file to reference the controller node.
- `apt install chrony`

### 4. Configuration of Block Storage Node:

#### Step 1: Initial Setup:

- Install Ubuntu on the virtual machine with the specified configurations.
- Set the management interface with IP address **10.173.143.22**, network mask **255.255.255.0**, and default gateway **10.173.143.1**.
- Set the hostname of the node to **block**.
- Edit the **/etc/hosts** file accordingly.

#### Step 2: Installing and Configuring Chrony:

- Configure the **chrony.conf** file to reference the controller node.
- `apt install chrony`

### 5. Verifying Connectivity:

- 
- Ping each node from the others to ensure connectivity between them.

#### **6. Install OpenStack Packages:**

- Follow the provided instructions to add the Bobcat repository and install necessary packages on the controller node.

#### **7. Additional Configuration (Controller Node):**

- Install and configure SQL database, message queue service, Memcached, and Etcd on the controller node as per the provided instructions.

#### **8. Setting Up Identity Service (Keystone):**

- Create a database for Keystone, install Keystone, and perform necessary configurations on the controller node.

### **9. Configuring Networking Service (Neutron) on the Controller Node:**

#### **Step 1: Install Neutron Packages:**

- `apt install neutron-server neutron-plugin-ml2 neutron-linuxbridge-agent neutron-l3-agent neutron-dhcp-agent neutron-metadata-agent`

#### **Step 2: Configure Neutron:**

Follow the provided instructions to edit configuration files such as `/etc/neutron/neutron.conf`, `/etc/neutron/plugins/ml2/ml2_conf.ini`, `/etc/neutron/plugins/ml2/linuxbridge_agent.ini`, `/etc/neutron/l3_agent.ini`, `/etc/neutron/dhcp_agent.ini`, and `/etc/neutron/metadata_agent.ini`.

#### **Step 3: Restart Neutron Services:**

- `service nova-api restart`
- `service neutron-server restart`
- `service neutron-linuxbridge-agent restart`
- `service neutron-dhcp-agent restart`
- `service neutron-metadata-agent restart`
- `service neutron-l3-agent restart`

#### **Step 4: Verify Neutron Operation:**

Check the status of Neutron agents to ensure they are running without errors:

- `neutron agent-list`

### **10. Setting Up Compute Service (Nova) on the Controller Node:**

#### **Step 1: Install Nova Packages:**

- `apt install nova-compute`

#### **Step 2: Configure Nova:**

Follow the provided instructions to edit configuration files such as `/etc/nova/nova.conf` to set up the Compute service

#### **Step 3: Restart Nova Services:**

- `service nova-api restart`
- `service nova-scheduler restart`
- `service nova-conductor restart`
- `service nova-compute restart`

#### **Step 4: Verify Nova Operation:**

Check the status of Nova services to ensure they are running without errors:

- `nova service-list`

### **11. Configuring Block Storage Service (Cinder) on the Controller Node:**

- Cinder service has been installed and configured on the controller node.
- Storage backends such as LVM or Ceph have been configured for Cinder volumes.

- 
- Cinder services have been restarted, and their operation has been verified.

**12. Configuring Image Service (Glance) on the Controller Node:**

- Glance service has been installed and configured on the controller node.
- Image storage locations and permissions have been configured.
- Base images have been imported into Glance.

**13. Configuring Dashboard (Horizon) on the Controller Node:**

- Horizon dashboard service has been installed and configured on the controller node.
- Authentication settings have been configured, and SSL certificates have been applied if necessary.

**14. Setting Up Object Storage Service (Swift) on Separate Nodes (Completed If Required):**

- Swift services have been installed and configured on dedicated storage nodes, if necessary.
- Replication and storage policies for Swift containers have been configured.

**15. Configuring Orchestration Service (Heat) on the Controller Node:**

- Heat orchestration service has been installed and configured on the controller node.
- Heat templates have been configured for deploying and managing resources.

**16. Configuring Telemetry Service (Ceilometer) on the Controller Node:**

- Ceilometer telemetry service has been installed and configured on the controller node.
- Metering and monitoring for OpenStack resources have been configured.

**17. Configuring Networking (Neutron) for Advanced Features :**

- Advanced networking features such as VLANs, VXLANS, or GRE tunnels have been configured in Neutron.
- Integration with external network services for advanced networking functionality has been completed.

## Setting up Web Servers

**1. Install Nginx:**

- Run **sudo apt update** to update package lists.
- Install Nginx using **sudo apt install nginx**.
- Start Nginx service with **sudo systemctl start nginx**.
- Enable Nginx to start on boot with **sudo systemctl enable nginx**.

**2. Configure Firewall:**

- Allow HTTP and HTTPS traffic using **sudo ufw allow 'Nginx Full'**.
- Verify the firewall status with **sudo ufw status**.

**3. Deploy Application:**

- Copy application files to **/var/www/html** directory.
- Set appropriate permissions with **sudo chown -R www-data:www-data /var/www/html**.

## Setting up Database Servers

**1. Install MySQL:**

- Run **sudo apt update** to update package lists.
- Install MySQL server with **sudo apt install mysql-server**.
- Secure MySQL installation with **sudo mysql\_secure\_installation**.

**2. Create Database and User:**

- Log in to MySQL shell with **sudo mysql**.
- Create a new database and user with appropriate privileges.
- Exit MySQL shell with **exit**.

---

## Setting up Hosts

1. **Install Required Packages:**
  - Update package lists with **sudo apt update**.
  - Install required packages such as Git, Docker, and Node.js.
2. **Clone Project Repository:**
  - Clone the project repository using Git.
3. **Configure Environment Variables:**
  - Create a **.env** file and set required environment variables.

## Configuring Monitoring Server with Zabbix

1. **Install Zabbix Server:**
  - Follow the official documentation to install Zabbix Server on Ubuntu.
2. **Configure Zabbix Agent:**
  - Install Zabbix Agent on all servers except the monitoring server.
  - Configure Zabbix Agent to communicate with the Zabbix Server.

## Project implementation

### Install Docker:

- Update the apt package index: **sudo apt update**
- Install required packages to allow apt to use a repository over HTTPS:
- **sudo apt install apt-transport-https ca-certificates curl software-properties-common**
- Add Docker's official GPG key: **curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -**
- Add the Docker repository to APT sources: **sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu \$(lsb\_release -cs) stable"**
- Update the apt package index again: **sudo apt update**
- Install Docker CE (Community Edition): **sudo apt install docker-ce**

### Manage Docker as a Non-root User (Optional):

- Add your user to the docker group: **sudo usermod -aG docker your\_username**
- Log out and log back in for the changes to take effect.

### Pull Docker Images:

- Pull the required Docker images from Docker Hub or your private registry:
- **sudo docker pull image\_name:tag**

### Run Docker Containers:

- Start a Docker container from the pulled image:

- sudo docker run -d --name container\_name -p host\_port:container\_port image\_name:tag

#### Manage Docker Containers:

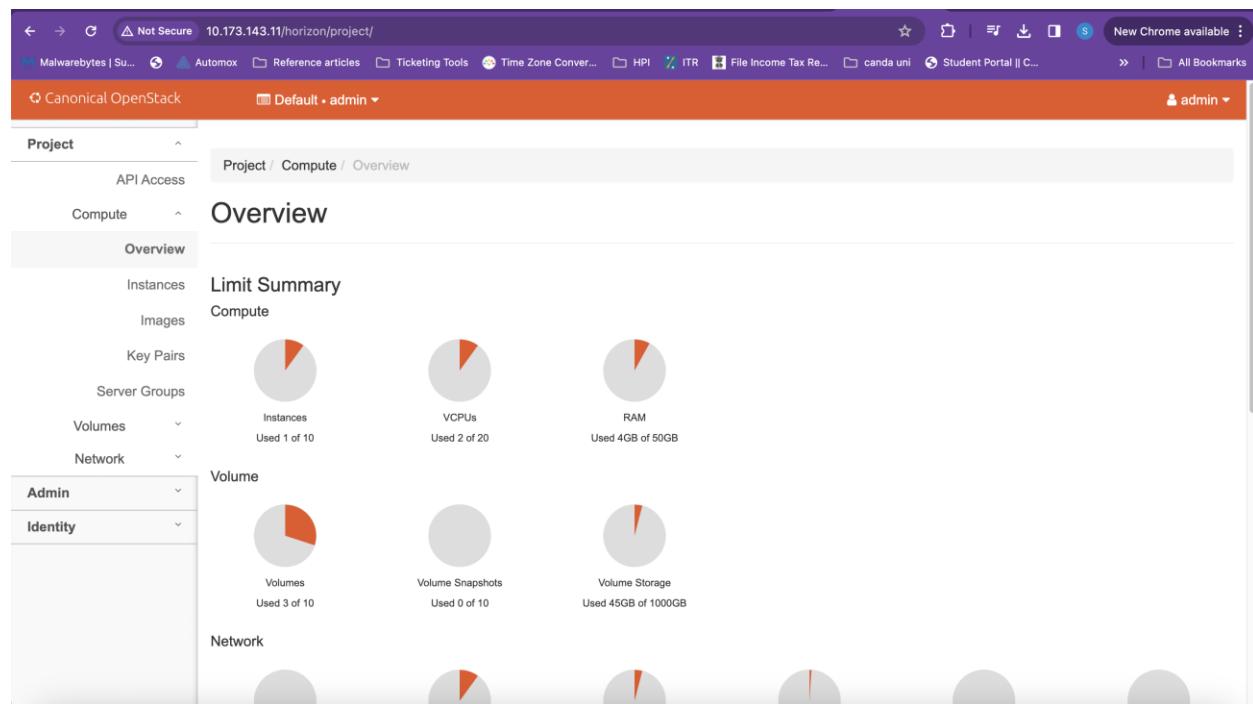
- List running containers: **sudo docker ps**
- Stop a running container: **sudo docker stop container\_id**
- Start a stopped container: **sudo docker start container\_id**
- Remove a container: **sudo docker rm container\_id**
- View container logs: **sudo docker logs container\_id**

#### Create Docker Compose File (Optional):

- Create a **docker-compose.yml** file to define multi-container Docker applications.
- Define services, networks, and volumes in the **docker-compose.yml** file.
- Use the **docker-compose** command to manage the application lifecycle.

## Screenshots

### OpenStack infrastructure:



Not Secure 10.173.143.11/horizon/project/instances/ New Chrome available : [Malwarebytes | Su...](#) [Automox](#) [Reference articles](#) [Ticketing Tools](#) [Time Zone Conver...](#) [HPI](#) [ITR](#) [File Income Tax Re...](#) [canda uni](#) [Student Portal || C...](#) > | All Bookmarks [admin](#)

Canonical OpenStack Default • admin ▾

**Project** API Access Compute Overview Instances Images Key Pairs Server Groups Volumes

**Instances**

Instance ID =  Filter [Launch Instance](#) [Delete Instances](#) More Actions ▾

Displaying 1 item

Instance Name	Image Name	IP Address	Flavor	Key Pair	Status	Availability Zone	Task	Power State	Age	Actions
ubuntu	-	10.173.143.112	ubuntu	-	Active	nova	None	Running	4 days, 16 hours	<a href="#">Create Snapshot</a> ▾

Not Secure 10.173.143.11/horizon/project/images/ New Chrome available : [Malwarebytes | Su...](#) [Automox](#) [Reference articles](#) [Ticketing Tools](#) [Time Zone Conver...](#) [HPI](#) [ITR](#) [File Income Tax Re...](#) [canda uni](#) [Student Portal || C...](#) > | All Bookmarks [admin](#)

Canonical OpenStack Default • admin ▾

**Project** API Access Compute Overview Instances Images Key Pairs Server Groups Volumes Network Admin Identity

**Images**

Click here for filters or full text search. [+ Create Image](#) [Delete Images](#)

Displaying 2 items

Owner	Name	Type	Status	Visibility	Protected	Disk Format	Size	
admin	cirros	Image	Active	Public	No	QCOW2	12.13 MB	<a href="#">Launch</a> ▾
admin	ubuntu	Image	Active	Public	No	ISO	1.96 GB	<a href="#">Launch</a> ▾

Displaying 2 items

Not Secure 10.173.143.11/horizon/project/volumes/ New Chrome available : [Malwarebytes | Su...](#) [Automox](#) [Reference articles](#) [Ticketing Tools](#) [Time Zone Conver...](#) [HPI](#) [ITR](#) [File Income Tax Re...](#) [canda uni](#) [Student Portal || C...](#) > | All Bookmarks [admin](#)

Canonical OpenStack Default • admin ▾

**Project** API Access Compute Volumes Volumes Snapshots Groups Group Snapshots Network Admin Identity

**Volumes**

Filter [+ Create Volume](#) [Accept Transfer](#) [Delete Volumes](#)

Displaying 3 items

Name	Description	Size	Status	Group	Type	Attached To	Availability Zone	Bootable	Encrypted	Actions
ubuntu 3	-	15GiB	Available	-	__DEFAULT__		nova	Yes	No	<a href="#">Edit Volume</a> ▾
ubuntu 2	-	15GiB	Available	-	__DEFAULT__		nova	Yes	No	<a href="#">Edit Volume</a> ▾
ubuntu	-	15GiB	In-use	-	__DEFAULT__	/dev/vda on ubuntu	nova	Yes	No	<a href="#">Edit Volume</a> ▾

Displaying 3 items

Not Secure 10.173.143.11/horizon/project/network\_topology/

Canonical OpenStack Default • admin

**Project**

- Compute
- Volumes
- Network
- Network Topology**
- Networks
- Routers
- Security Groups
- Floating IPs

Admin Identity

## Network Topology

Topology Graph

Small Normal

Launch Instance + Create Network + Create Router

Not Secure 10.173.143.11/horizon/project/networks/

Canonical OpenStack Default • admin

**Project**

- Compute
- Volumes
- Network
- Network Topology
- Networks**
- Routers
- Security Groups
- Floating IPs

Admin Identity

## Networks

Name = Filter + Create Network Delete Networks

Name	Subnets Associated	Shared	External	Status	Admin State	Availability Zones	Actions
ext-net	ext-subnet 10.173.143.0/24	No	Yes	Active	UP	nova	<a href="#">Edit Network</a>

Displaying 1 item

Not Secure 10.173.143.11/horizon/project/security\_groups/

Canonical OpenStack Default • admin

**Project**

- Compute
- Volumes
- Network
- Network Topology
- Networks
- Routers
- Security Groups**
- Floating IPs

Admin

## Security Groups

Filter + Create Security Group Delete Security Groups

Name	Security Group ID	Description	Shared	Actions
default	ea795f09-eb72-4c62-94a5-7f21f6e40d3c	Default security group	False	<a href="#">Manage Rules</a>

Displaying 1 item

Not Secure 10.173.143.11/horizon/admin/hypervisors/

Canonical OpenStack Default • admin admin

Project Admin Overview Compute Hypervisors Host Aggregates Instances Flavors Images Volume Network System Identity

All Hypervisors

Hypervisor Summary

VCPU Usage Used 2 of 8 Memory Usage Used 4.5GB of 19.5GB Local Disk Usage Used 1GB of 9GB

Hypervisor Compute Host

Displaying 1 item

Hostname	Type	VCPUs (used)	VCPUs (total)	RAM (used)	RAM (total)	Local Storage (used)	Local Storage (total)	Instances
compute	QEMU	2	8	4.5GB	19.5GB	1GB	9GB	1

Displaying 1 item

Not Secure 10.173.143.11/horizon/admin/aggregates/

Canonical OpenStack Default • admin admin

Project Admin Overview Compute Hypervisors Host Aggregates Instances Flavors Images Volume Network System Identity

Host Aggregates

Host Aggregates Filter + Create Host Aggregate

Name	Availability Zone	Hosts	Metadata	Actions
		No items to display.		

Availability Zones

Displaying 2 items

Availability Zone Name	Hosts	Available
internal	• controller (Services Up)	Yes
nova	• compute (Services Up)	Yes

Displaying 2 items

Not Secure 10.173.143.11/horizon/admin/instances/

Canonical OpenStack Default • admin admin

Project Admin Overview Compute Hypervisors Host Aggregates Instances Flavors Images

Instances

Displaying 1 item

Project	Host	Name	Image Name	IP Address	Flavor	Status	Task	Power State	Age	Actions
admin	compute	ubuntu	-	10.173.143.112	ubuntu	Active	None	Running	4 days, 16 hours	<button>Rescue Instance</button>

Not Secure 10.173.143.11/horizon/admin/flavors/

Canonical OpenStack Default • admin

Project Admin

Overview Compute Hypervisors Host Aggregates Instances Flavors Images Volume Network

## Flavors

Displaying 2 items

<input type="checkbox"/>	Flavor Name	VCPUs	RAM	Root Disk	Ephemeral Disk	Swap Disk	RX/TX factor	ID	Public	Metadata	Actions
<input type="checkbox"/>	Template Machine	2	4GB	1GB	1GB	100MB	1.0	79fb621c-d0d6-46ee-902d-0471aa714847	Yes	No	<button>Update Metadata</button>
<input type="checkbox"/>	ubuntu	2	4GB	10GB	1GB	100MB	1.0	2e1ce057-5ab7-41b0-891e-0e58f947e5d9	Yes	No	<button>Update Metadata</button>

Not Secure 10.173.143.11/horizon/admin/images/

Canonical OpenStack Default • admin

Project Admin

Overview Compute Hypervisors Host Aggregates Instances Flavors Images Volume Network System

## Images

Click here for filters or full text search.

Displaying 2 items

<input type="checkbox"/>	Owner	Name	Type	Status	Visibility	Protected	Disk Format	Size	
<input type="checkbox"/>	> admin	cirros	Image	Active	Public	No	QCOW2	12.13 MB	<button>Launch</button>
<input type="checkbox"/>	> admin	ubuntu	Image	Active	Public	No	ISO	1.96 GB	<button>Launch</button>

Canonical OpenStack Default • admin

Project Admin

Overview Compute Volume Snapshots Volume Types Groups Group Snapshots Group Types Network System

## Volumes

Volume Name =  Filter + Manage Volume Delete Volumes

Displaying 3 items

<input type="checkbox"/>	Project	Host	Name	Size	Status	Group	Type	Attached To	Bootable	Encrypted	Actions
<input type="checkbox"/>	admin	block@lvm#LVM	ubuntu 3	15GiB	Available	-	__DEFAULT__		Yes	No	<button>Delete Volume</button>
<input type="checkbox"/>	admin	block@lvm#LVM	ubuntu 2	15GiB	Available	-	__DEFAULT__		Yes	No	<button>Delete Volume</button>
<input type="checkbox"/>	admin	block@lvm#LVM	ubuntu	15GiB	In-use	-	__DEFAULT__	/dev/vda on ubuntu	Yes	No	<button>Update Volume Status</button>

Not Secure 10.173.143.11/horizon/admin/defaults/

Malwarebytes | Su... Automox Reference articles Ticketing Tools Time Zone Conver... HPI ITR File Income Tax Re... canda uni Student Portal || C...

Canonical OpenStack Default • admin admin ▾

Project Admin Overview Compute Quotas Volume Quotas Network Quotas Filter

Compute Volume Network System Defaults Metadata Definitions System Information Identity

Displaying 9 items

Quota Name	Limit
Networks	100
Subnets	100
Subnet Pool	-1
Ports	500
Routers	10
Floating IPs	50
RBAC Policies	10
Security Groups	10
Security Group Rules	100

Displaying 9 items

Not Secure 10.173.143.11/horizon/admin/defaults/

Malwarebytes | Su... Automox Reference articles Ticketing Tools Time Zone Conver... HPI ITR File Income Tax Re... canda uni Student Portal || C...

Canonical OpenStack Default • admin admin ▾

Project Admin Overview Compute Quotas Volume Quotas Network Quotas Filter Update Defaults

Compute Volume Network System Defaults Metadata Definitions System Information Identity

Displaying 10 items

Quota Name	Limit
VCPUs	20
Injected File Content (B)	10240
Length of Injected File Path	255
Injected Files	5
Instances	10
Key Pairs	100
Metadata Items	128
RAM (MB)	51200
Server Group Members	10
Server Groups	10

Displaying 10 items

Not Secure 10.173.143.11/horizon/admin/info/

Canonical OpenStack Default • admin

Project Admin Overview Compute Volume Network System Defaults Metadata Definitions System Information Identity

Admin / System / System Information

## System Information

Services Compute Services Block Storage Services Network Agents

Displaying 6 items

Name	Service	Region	Endpoints
glance	image	RegionOne	<b>Admin</b> http://controller:9292 <b>Internal</b> http://controller:9292 <b>Public</b> http://controller:9292
placement	placement	RegionOne	<b>Admin</b> http://controller:8778 <b>Internal</b> http://controller:8778 <b>Public</b> http://controller:8778
neutron	network	RegionOne	<b>Admin</b> http://controller:9696 <b>Internal</b> http://controller:9696 <b>Public</b> http://controller:9696
keystone	identity	RegionOne	<b>Admin</b> http://controller:5000/v3/ <b>Internal</b> http://controller:5000/v3/ <b>Public</b> http://controller:5000/v3/

Not Secure 10.173.143.11/horizon/admin/info/

Canonical OpenStack Default • admin

Project Admin Overview Compute Volume Network System Defaults Metadata Definitions System Information Identity

Admin / System / System Information

Name	Service	Region	Endpoints
glance	image	RegionOne	<b>Admin</b> http://controller:9292 <b>Internal</b> http://controller:9292 <b>Public</b> http://controller:9292
placement	placement	RegionOne	<b>Admin</b> http://controller:8778 <b>Internal</b> http://controller:8778 <b>Public</b> http://controller:8778
neutron	network	RegionOne	<b>Admin</b> http://controller:9696 <b>Internal</b> http://controller:9696 <b>Public</b> http://controller:9696
keystone	identity	RegionOne	<b>Admin</b> http://controller:5000/v3/ <b>Internal</b> http://controller:5000/v3/ <b>Public</b> http://controller:5000/v3/
nova	compute	RegionOne	<b>Admin</b> http://controller:8774/v2.1 <b>Internal</b> http://controller:8774/v2.1 <b>Public</b> http://controller:8774/v2.1
cinderv3	volumev3	RegionOne	<b>Admin</b> http://controller:8776/v3/ce2041eb03564d868ebed2d84cea21d5 <b>Internal</b> http://controller:8776/v3/ce2041eb03564d868ebed2d84cea21d5 <b>Public</b> http://controller:8776/v3/ce2041eb03564d868ebed2d84cea21d5

Displaying 6 items

Version: 23.3.0

Not Secure 10.173.143.11/horizon/identity/users/

Canonical OpenStack Default • admin

Identity / Users

## Users

Displaying 7 items

User Name	Description	Email	User ID	Enabled	Domain Name	Actions
admin	-		dac2fd4013084d06af483d2987f58018	Yes	Default	<button>Edit</button>
myuser	-		6393eb953e2a411cafbc9b51397b08c0	Yes	Default	<button>Edit</button>
glance	-		416ad0526f7b40848fafd23ff72e55da	Yes	Default	<button>Edit</button>
placement	-		1c657c7703734d50b39c58e89c4dff9f	Yes	Default	<button>Edit</button>
nova	-		70c21e1466af493eb50503ea6fd1c97	Yes	Default	<button>Edit</button>
neutron	-		d93de4158d5343358670270ac57e792f	Yes	Default	<button>Edit</button>
cinder	-		40dcc0e4c52b42d394020e16b1465dfe	Yes	Default	<button>Edit</button>

Displaying 7 items

Not Secure 10.173.143.11/horizon/identity/projects/

Canonical OpenStack Default • admin

Identity / Projects

## Projects

Displaying 3 items

Name	Description	Project ID	Domain Name	Enabled	Actions
myproject	Demo Project	691239c6f99c4882931000c0dfbdd8ba	Default	Yes	<button>Manage Members</button>
service	Service Project	6fe46963c1cc4ea0a079c6ca4cddee53c	Default	Yes	<button>Manage Members</button>
admin	Bootstrap project for initializing the cloud.	ce2041eb03564d868ebed2d84cea21d5	Default	Yes	<button>Manage Members</button>

Displaying 3 items

Not Secure 10.173.143.11/horizon/identity/roles

Malwarebytes | Su... Automox Reference articles Ticketing Tools Time Zone Conver... HPI ITR File Income Tax Re... canda uni Student Portal || C...

Canonical OpenStack Default - admin admin ▾

Project Admin Identity

Identity / Roles

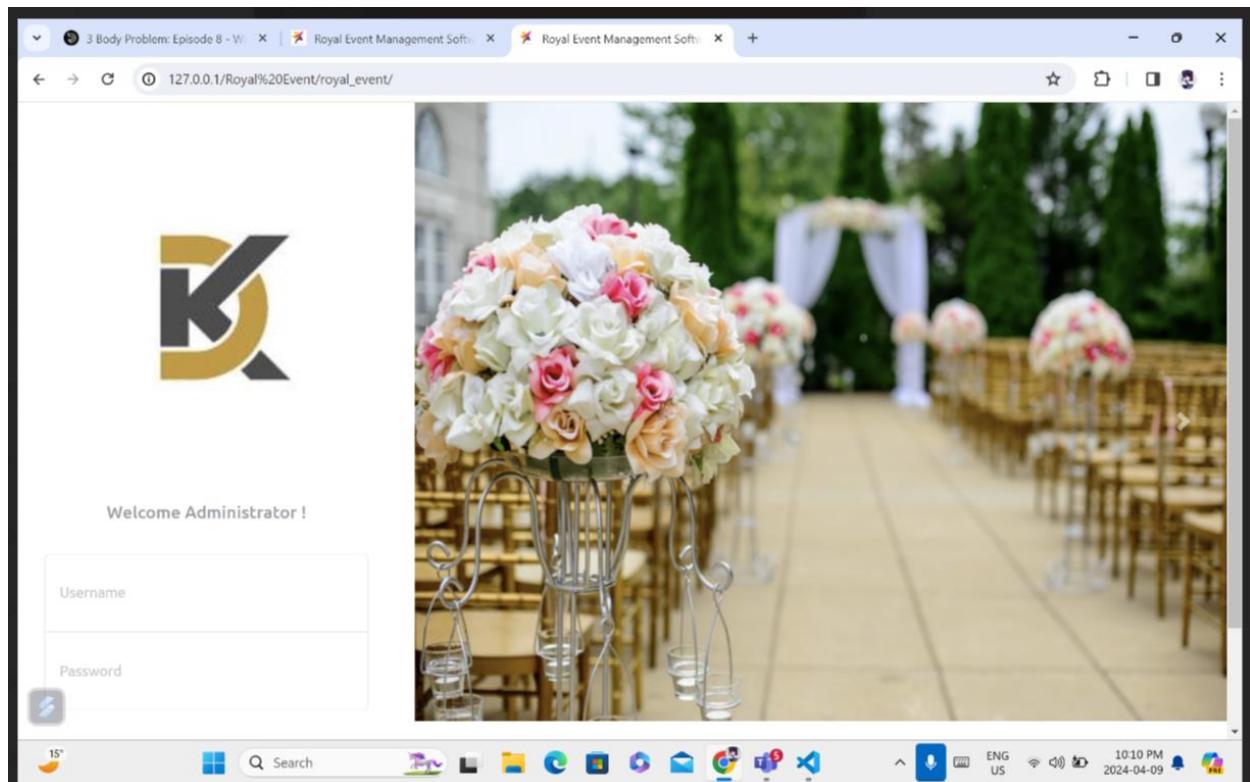
## Roles

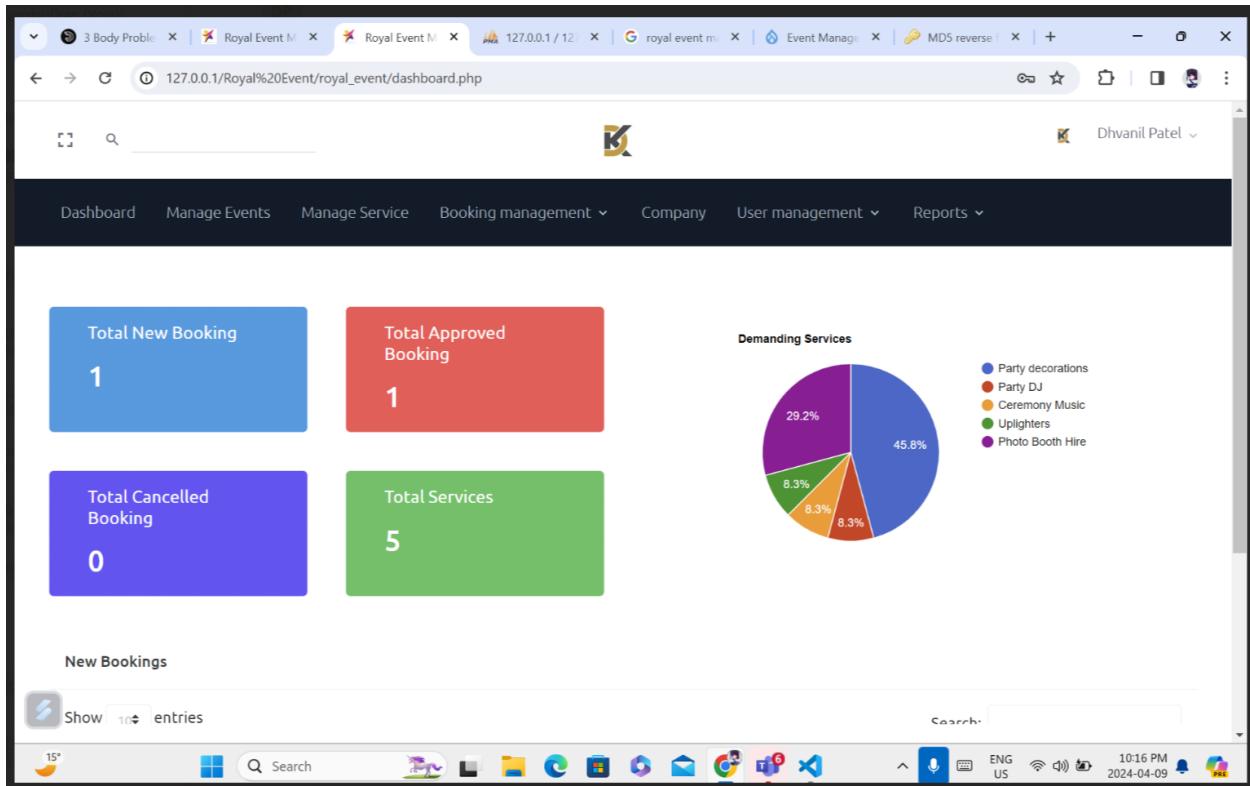
Click here for filters or full text search. + Create Role Delete Roles

Name	ID	Actions
admin	0759f5ea99dc4d8a83c332ec4211e513	Edit Role
manager	29e63bc6e92f47cbb421923dd40358e9	Edit Role
member	c7b72c6811394d45b4250b51c1f2b363	Edit Role
myrole	6133d087b67246dd84e8e8e5d5219356	Edit Role
reader	ebe2a0ebb05243059b47a5cb93d2249c	Edit Role
service	10512e6c82874b13aad7bdce9f7160e0	Edit Role

Displaying 6 items

## Admin Panel of our application:





The screenshot shows the "Service register" page. It includes a header with "Service register" and "Add Event" buttons, a search bar, and a table listing five events:

No.	Event Name	Creation Date	Action
1	Anniversary	2022-01-22 02:01:39	
2	Birthday Party	2022-01-22 02:02:34	
3	Charity	2022-01-22 02:02:43	
4	Cocktail	2022-01-22 02:03:00	
5	College	2022-01-22 02:03:11	

15° ENG US 10:17 PM 2024-04-09

The screenshot shows a table titled "New Bookings" with one entry. The columns are: Booking ID, Customer Name, Mobile Number, Email, Booking Date, Status, and Action. The entry details are: Booking ID 977361722, Customer Name Jayesh Panghwane, Mobile Number 07070707070, Email jayesh768@gmail.com, Booking Date 2022-03-22 05:29:18, Status Not Updated Yet, and Action (checkbox). Below the table, it says "Showing 1 to 1 of 1 entries".

	Booking ID	Customer Name	Mobile Number	Email	Booking Date	Status	Action
1	977361722	Jayesh Panghwane	07070707070	jayesh768@gmail.com	2022-03-22 05:29:18	Not Updated Yet	<input type="checkbox"/>

The screenshot shows a table titled "Register user" with two entries. The columns are: No., Name, Mobile number, Email, Date registered, and Action. The entries are: No. 1, Name Dhvanil Patel, Mobile number 0942397933, Email dhvanilpatel2542@gmail.com, Date registered 21-07-2022, and Action (checkbox, red circle). No. 2, Name Suraj kale, Mobile number 0942397933, Email Suraj@gmail.com, Date registered 25-07-2022, and Action (checkbox, red circle).

No.	Name	Mobile number	Email	Date registered	Action
1	Dhvanil Patel	0942397933	dhvanilpatel2542@gmail.com	21-07-2022	<input type="checkbox"/>
2	Suraj kale	0942397933	Suraj@gmail.com	25-07-2022	<input type="checkbox"/>

## Front-end:

A screenshot of a web browser window displaying an event gallery and footer links. The browser address bar shows "localhost:3000/eventdetail". The main content area is titled "Event Gallery" and contains a grid of six event-related images. Below the gallery, there are three columns of footer links: "Useful Links" (About, Events, Terms & Conditions, Privacy Policy), "Social Connections" (Facebook, Twitter, Instagram, LinkedIn), and "Contact" (Send us a message, Find us on Map). To the right of the links is a blue icon of a calendar with a star and some icons. At the bottom of the page, the copyright notice "Copyright © 2024 All Rights Reserved by Capstone Group." is visible.

localhost:3000/eventdetail

**Event Gallery**

**Useful Links**

- About
- Events
- Terms & Conditions
- Privacy Policy

**Social Connections**

- Facebook
- Twitter
- Instagram
- LinkedIn

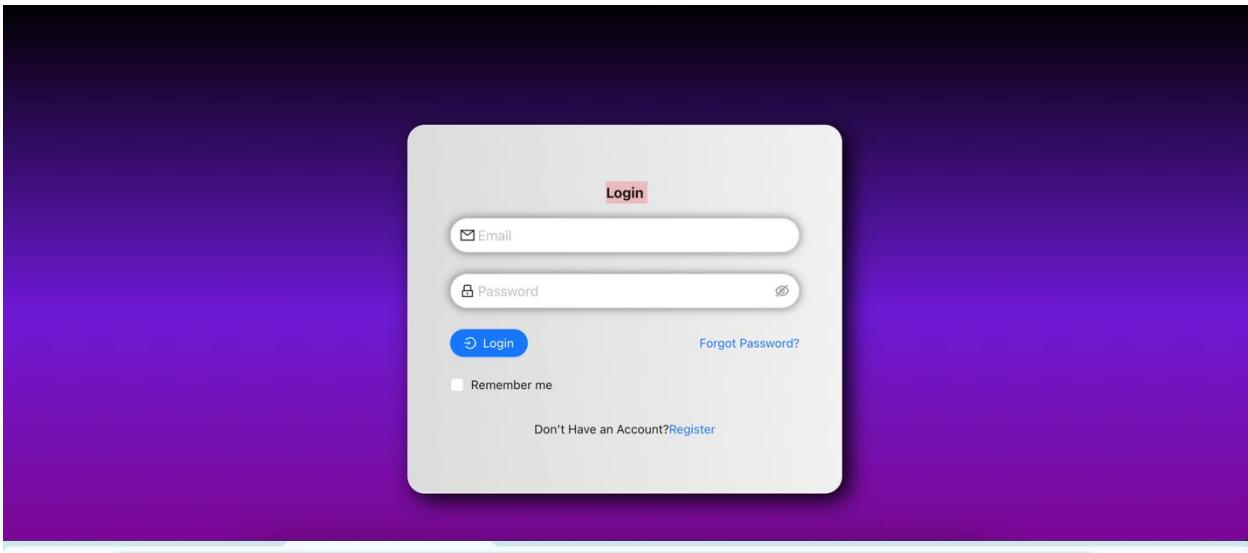
**Contact**

- Send us a message
- Find us on Map

Copyright © 2024 All Rights Reserved by Capstone Group.

The screenshot shows a web browser window with the URL [localhost:3000/eventdetail](http://localhost:3000/eventdetail). The page title is "Global Marketing Conference". On the left, there's a large image of a person. Below it, a section titled "Overview" contains a long, dense paragraph of placeholder text. To the right, there's a "Subscribe Now" button and a "Where" section with details for "Wheelbarrow Well" in Coomalbidgup, WA, Australia. The date and time listed are "01 Oct 2025 02:00 PM (UTC) - 31 Oct 2025 04:30 PM (UTC)".

The screenshot shows a mobile browser on an iPhone with the URL [localhost:3000](http://localhost:3000). The header bar includes the Apple logo, signal strength, battery level (59%), and the date/time (Fri Mar 29 3:06 PM). The main content features a large image of a concert stage with the text "Start hosting your event for free". Below this are four smaller images with dates: "01 MAY 2024" (concert), "01 JUN 2024" (people at a table), "01 JUL 2024" (party scene), and "01 Nov 2024" (people at a table). The bottom of the screen shows the iOS navigation bar with various app icons.



localhost:3000

A grid of eight event cards, each with a thumbnail image, date, title, category, and location. The cards are arranged in two rows of four.

01 MAY 2024 <b>Global Marketing Conference</b> Business & Seminars Location 1	01 JUN 2024 <b>Caribbean Karaoke Nights</b> Music & Concerts Location 2	01 JUL 2024 <b>VR assistant Workshop</b> Business & Seminars Location 3	01 Nov 2024 <b>Winter Wine Night</b> Food & Drink Location 4
02 Nov 2024 <b>Digital Marketing Seminar</b> Business & Seminars Location 5	03 Nov 2024 <b>Digital Marketing Seminar</b> Business & Seminars Location 5	03 Nov 2024 <b>Digital Marketing Seminar</b> Business & Seminars Location 5	04 Nov 2024 <b>Digital Marketing Seminar</b> Business & Seminars Location 5



## Dockerize application:

The screenshot shows a VS Code interface with the following details:

- Explorer:** Shows the project structure for "EVENT-FRONTEND-APP" containing files like Header.js, dockerfile, package-lock.json, package.json, and README.md.
- Editor:** Displays the contents of the "dockerfile" file:

```
FROM node:alpine
WORKDIR /app
COPY package*.json .
RUN npm install
COPY .
EXPOSE 3000
CMD ["npm", "start"]
```
- Terminal:** Shows the command "user@users-MacBook-Pro event-frontend-app %".
- Status Bar:** Shows Dockerfile, Prettier, and other status indicators.

## Automatic Fail over mechanism:

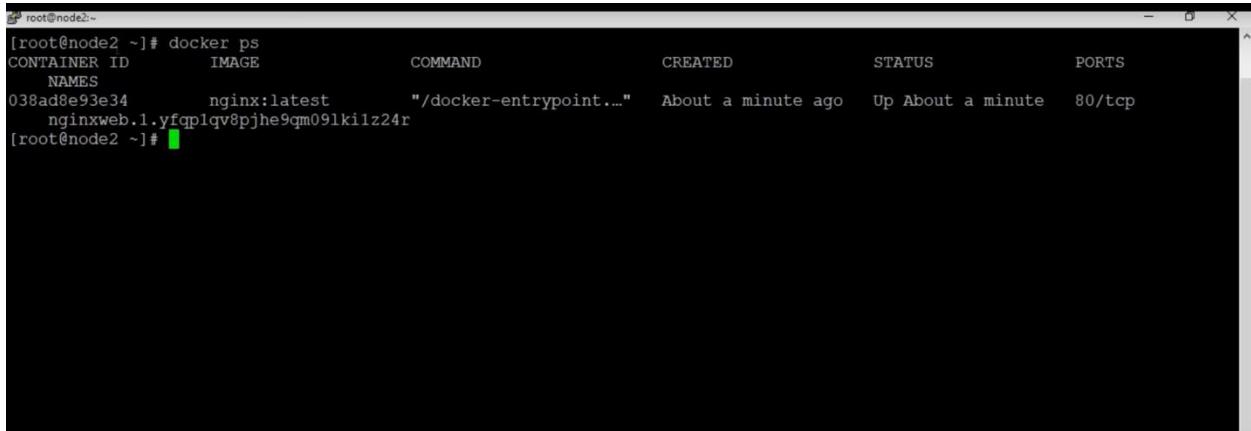
- If anything happens with container like crash due to higher resource usage, application failure. In these type of conditions docker will automatically re-create the container and restart it.

The terminal output shows the following sequence of commands and their results:

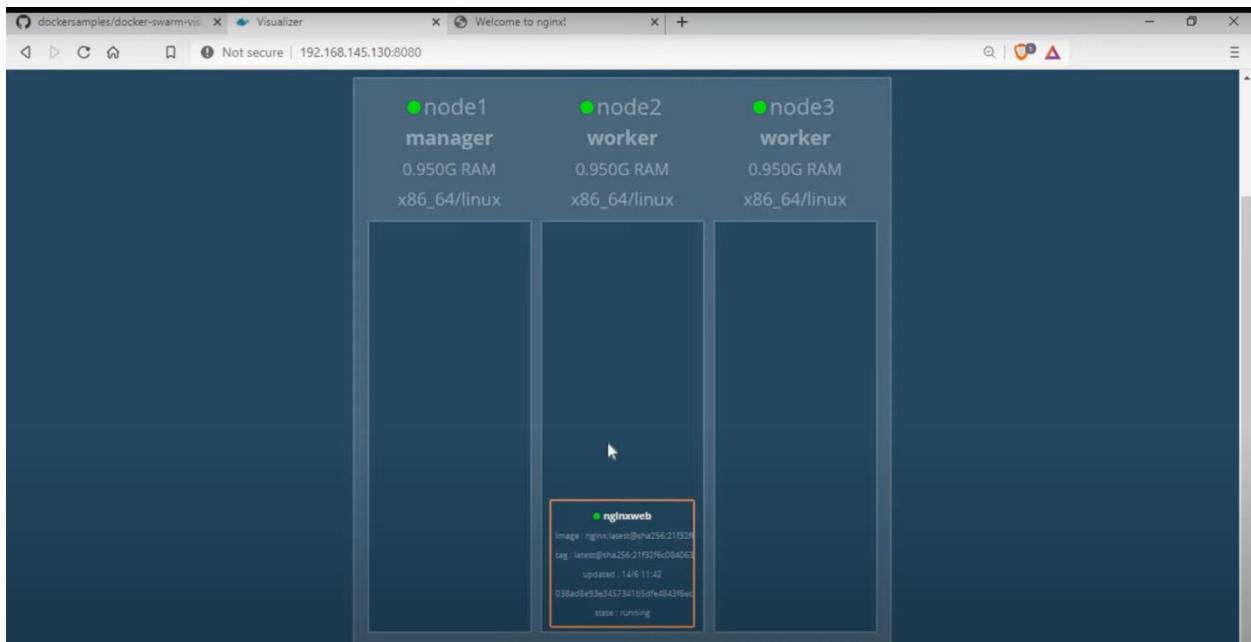
```
manuel@Manuels-MBP: ~
> docker start failing-demo
failing-demo
> docker ps
CONTAINER ID        IMAGE       COMMAND                  STATUS        NAMES
db745effa1fa        alpine      "sh -c 'sleep 10; ex..."   Up 6 seconds   failing-demo
> docker ps
CONTAINER ID        IMAGE       COMMAND                  STATUS        NAMES
db745effa1fa        alpine      "sh -c 'sleep 10; ex..."   Up 2 seconds   failing-demo
> docker ps
CONTAINER ID        IMAGE       COMMAND                  STATUS        NAMES
db745effa1fa        alpine      "sh -c 'sleep 10; ex..."   Up 7 seconds   failing-demo
> docker ps
CONTAINER ID        IMAGE       COMMAND                  STATUS        NAMES
db745effa1fa        alpine      "sh -c 'sleep 10; ex..."   Up 2 seconds   failing-demo
> docker ps
CONTAINER ID        IMAGE       COMMAND                  STATUS        NAMES
db745effa1fa        alpine      "sh -c 'sleep 10; ex..."   Up 6 seconds   failing-demo
> docker ps
CONTAINER ID        IMAGE       COMMAND                  STATUS        NAMES
db745effa1fa        alpine      "sh -c 'sleep 10; ex..."   Restarting (1) Less than a second ago   failing-demo
```

## High Availability Using Docker Swarm:

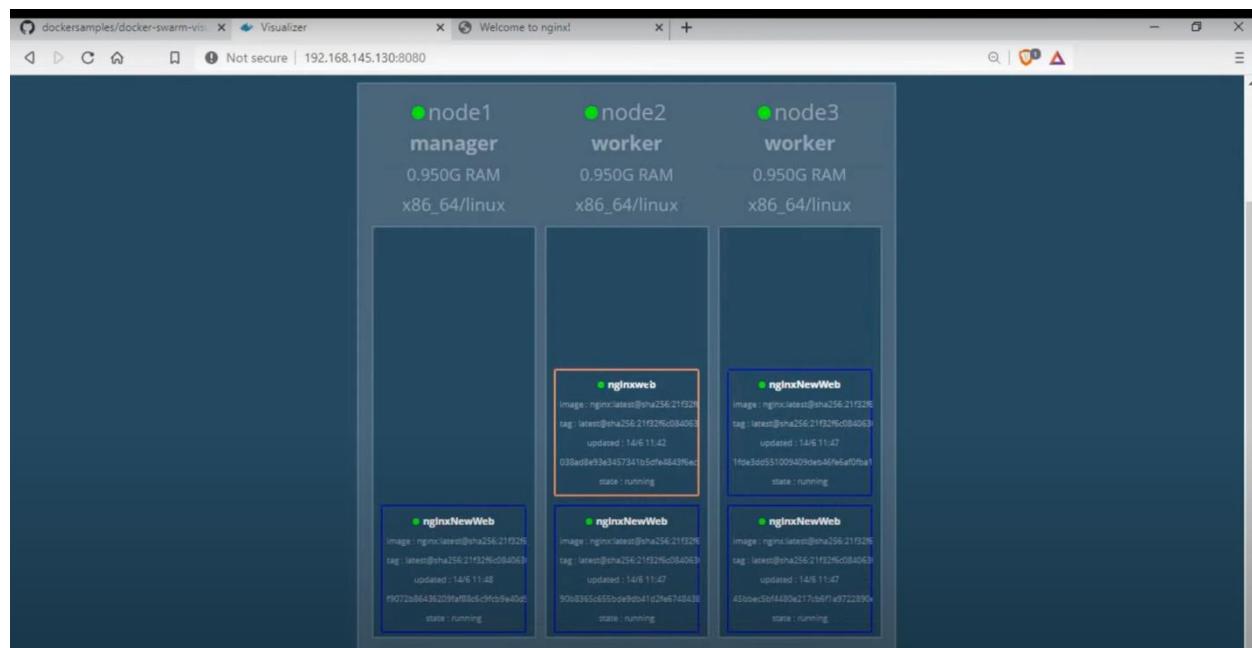
- We create replica of our containers so if one container fails the system will not shut down and docker automatically redirect the traffic to another container.



```
[root@node2 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND
 NAMES
038ad8e93e34      nginx:latest       "/docker-entrypoint..."   About a minute ago   Up About a minute   80/tcp
nginxweb.1.yfcqlqv8pjhe9qm091ki1z24r
[root@node2 ~]#
```

A terminal window showing the output of the docker ps command. It lists a single container named 'nginxweb.1.yfcqlqv8pjhe9qm091ki1z24r' running on port 80/tcp.

```
[root@node1 ~]# docker service create --name nginxxNewWeb -p 8081:80 --replicas 4 nginx
Error response from daemon: rpc error: code = InvalidArgument desc = port '8081' is already in use by service 'nginxxweb' (01q5meilmh0jv1lzfhl3v66o) as an ingress port
[root@node1 ~]# docker service create --name nginxxNewWeb -p 8082:80 --replicas 4 nginx
su7u79drqbi39slcuc39cyxnq
overall progress: 4 out of 4 tasks
1/4: running [=====>]
2/4: running [=====>]
3/4: running [=====>]
4/4: running [=====>]
verify: Service converged
[root@node1 ~]# docker service ls
ID           NAME      MODE      REPLICAS      IMAGE          PORTS
su7u79drqbi3  nginxxNewWeb replicated  4/4           nginx:latest *:8082->80/tcp
01q5meilmh0j  nginxxweb   replicated  1/1           nginx:latest *:8081->80/tcp
[root@node1 ~]#
```



```

root@node1:~# docker service create --name nginxxNewWeb -p 8081:80 --replicas 4 nginx
Error response from daemon: rpc error: code = InvalidArgument desc = port '8081' is already in use by service 'nginxweb' (01q5meilmh0jvilzffhl3v66o) as an ingress port
[root@node1 ~]# docker service create --name nginxxNewWeb -p 8082:80 --replicas 4 nginx
su7u79drqbi39slcuc39cyxng
overall progress: 4 out of 4 tasks
1/4: running [=====>]
2/4: running [=====>]
3/4: running [=====>]
4/4: running [=====>]
verify: service converged
[root@node1 ~]# docker service ls
ID           NAME      MODE      REPLICAS  IMAGE          PORTS
su7u79drqbi3  nginxxNewWeb replicated  4/4      nginx:latest *:8082->80/tcp
01q5meilmh0j  nginxxweb   replicated  1/1      nginx:latest *:8081->80/tcp
[root@node1 ~]# docker service ps nginxxNewWeb
ID           NAME      IMAGE      NODE      DESIRED STATE  CURRENT STATE
ERROR        PORTS
lospndvkstb5  nginxxNewWeb.1  nginx:latest  node1     Running       Running 55 seconds ago
6j066kfeu3ir  nginxxNewWeb.2  nginx:latest  node2     Running       Running 2 minutes ago
zem3cxe6f0p5  nginxxNewWeb.3  nginx:latest  node3     Running       Running about a minute
ago
z7jss7mdyuuh7  nginxxNewWeb.4  nginx:latest  node3     Running       Running about a minute
[root@node1 ~]#

```

## Zabbix Monitoring:

The screenshot shows the Zabbix Global view dashboard. On the left, there's a sidebar with navigation links for Dashboards, Monitoring, Services, Inventory, Reports, Data collection, Alerts, Users, Administration, Support, Integrations, Help, User settings, and Sign out. The main content area includes:

- Top hosts by CPU utilization:** Shows a chart for the Zabbix server with utilization at 1.49%.
- System information:** A table showing parameters like "Zabbix server is running" (Value: Yes), "Number of hosts (enabled/disabled)" (5 / 0), "Number of templates" (303), and "Number of items (enabled/disabled/not supported)" (435).
- Host availability:** A summary table showing 5 Available, 0 Not available, 0 Unknown, and 5 Total hosts.
- Problems by severity:** A summary table showing 0 Disaster, 0 High, 0 Average, 0 Warning, 0 Information, and 0 Not classified problems.
- Current problems:** A table with columns for Time, Info, Host, Problem + Severity, Duration, Update, Actions, and Tags. It displays "No data found."
- Geomap:** A map of Riga, Latvia, showing various monitoring points and routes.

**ZABBIX** < | Hosts

Hosts

Name: [ ] Status: Any Enabled Disabled

Host groups: type here to search Select Tags: And/Or Or

IP: [ ] tag: [ ] Contains value: [ ] Remove

DNS: Port: [ ] Add

Show hosts in maintenance: [ ] Show suppressed problems: [ ]

Severity: Not classified Warning High Information Average Disaster

Save as Apply Reset

Name	Interface	Availability	Tags	Status	Latest data	Problems	Graphs	Dashboards	Web
dbserver1	10.173.143.13:10050	ZBX	class: os target: linux	Enabled	Latest data 73	Problems 14	Graphs 14	Dashboards 2	Web
dbserver2	10.173.143.14:10050	ZBX	class: os target: linux	Enabled	Latest data 55	Problems 10	Graphs 10	Dashboards 2	Web
webserver1	10.173.143.11:10050	ZBX	class: os target: linux	Enabled	Latest data 82	Problems 15	Graphs 15	Dashboards 2	Web
webserver2	10.173.143.12:10050	ZBX	class: os target: linux	Enabled	Latest data 73	Problems 14	Graphs 14	Dashboards 2	Web
Zabbix server	127.0.0.1:10050	ZBX	class: os class: software target: linux ...	Enabled	Latest data 134	Problems 25	Graphs 25	Dashboards 4	Web

Displaying 5 of found

Zabbix 6.4.13. © 2001–2024, Zabbix SIA

**ZABBIX** < | Network interfaces

All hosts / dbserver1 / Network interfaces

From: now-1h To: now Apply

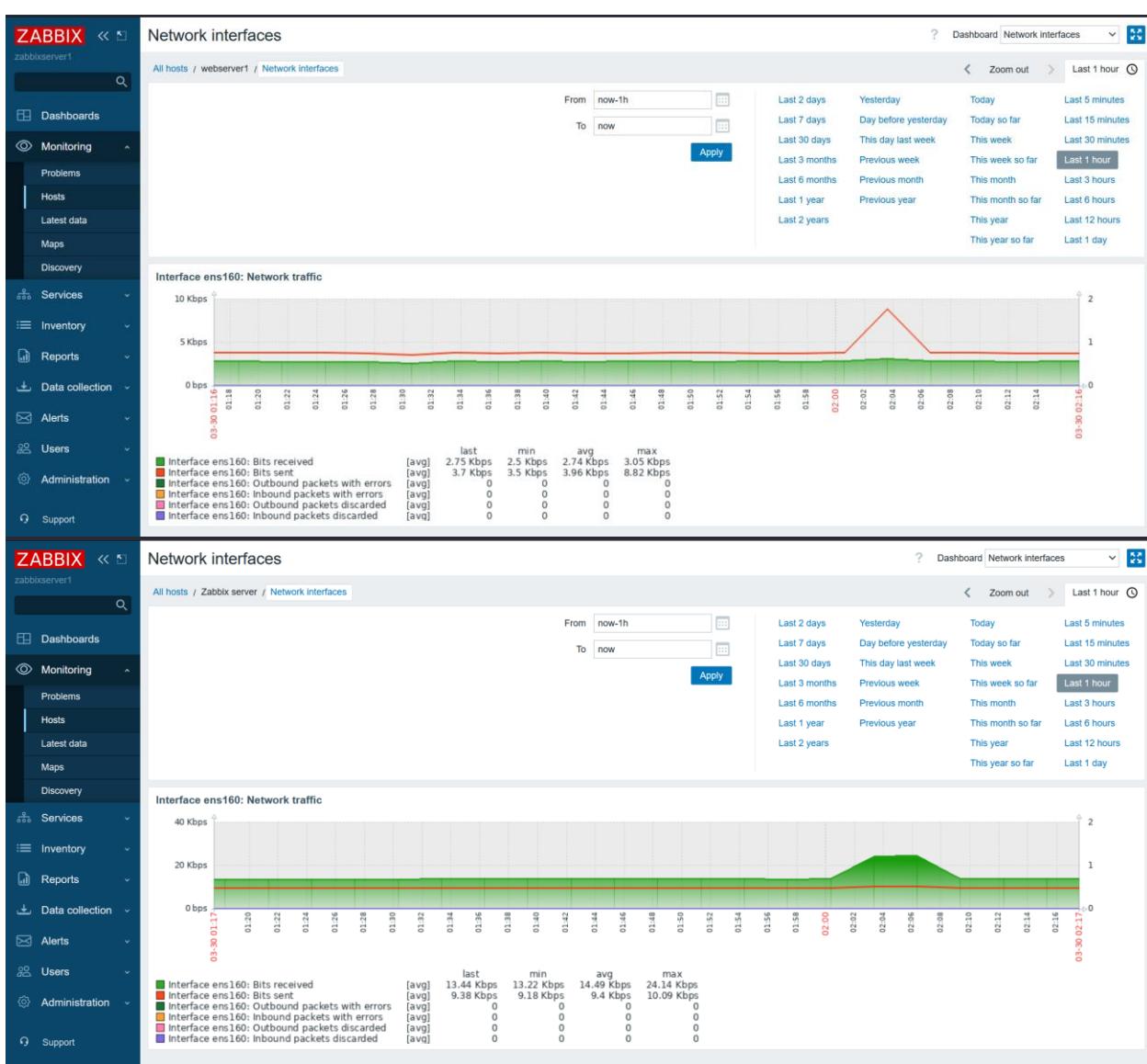
Last 2 days Yesterday Today Last 5 minutes  
Last 7 days Day before yesterday Today so far Last 15 minutes  
Last 30 days This day last week This week Last 30 minutes  
Last 3 months Previous week This week so far Last 1 hour  
Last 6 months Previous month This month Last 3 hours  
Last 1 year Previous year This month so far Last 6 hours  
Last 2 years This year Last 12 hours

Interface {#IFNAME}: Network traffic

Interface ens160: Network traffic

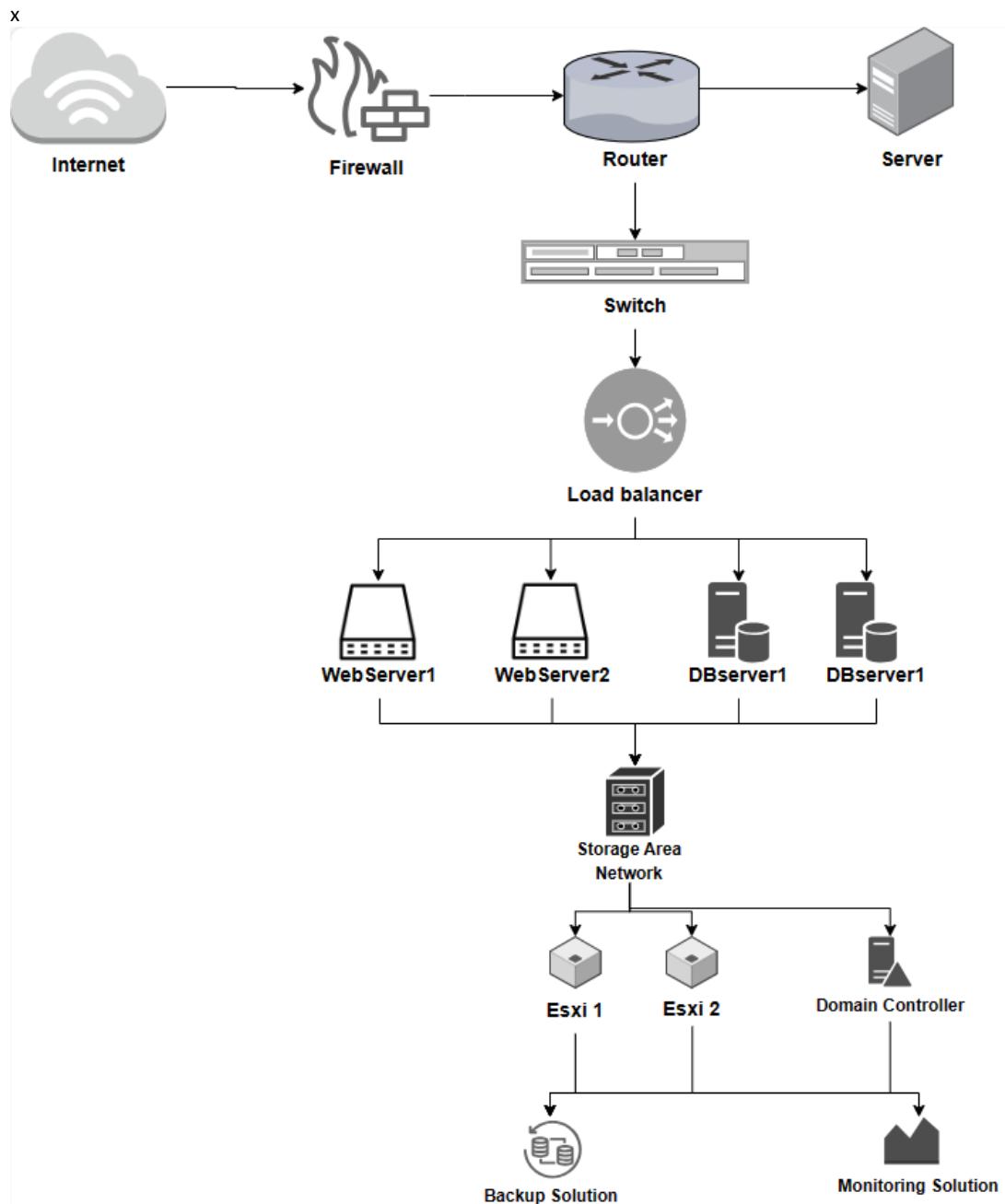
03:30 01:18 01:20 01:22 01:24 01:26 01:28 01:30 01:32 01:34 01:36 01:38 01:40 01:42 01:44 01:46 01:48 01:50 01:52 01:54 01:56 01:58 02:00 02:02 02:04 02:06 02:08 02:10 02:12 02:14 02:16 02:18 02:20 02:22 02:24 02:26 02:28 02:30 02:32 02:34 02:36 02:38 02:40 02:42 02:44 02:46 02:48 02:50 02:52 02:54 02:56 02:58 02:59 03:00

Interface ens160: Bits received	[avg]	last	min	avg	max
Interface ens160: Bits sent	[avg]	2.48 Kbps	2.42 Kbps	2.47 Kbps	8.8 Kbps
Interface ens160: Inbound packets with errors	[avg]	3.47 Kbps	3.45 Kbps	3.75 Kbps	0
Interface ens160: Outbound packets with errors	[avg]	0	0	0	0
Interface ens160: Outbound packets discarded	[avg]	0	0	0	0
Interface ens160: Inbound packets discarded	[avg]	0	0	0	0



---

## Network Diagram:



---

## Project Testing

### Functional Testing:

- Verify that web servers respond to HTTP and HTTPS requests.
- Test Nginx configuration to ensure proper routing and serving of web application content.
- Check PHP integration with Nginx to ensure dynamic content generation.

### Performance Testing:

- Conduct load testing using tools like Apache JMeter to evaluate server response times under various traffic conditions.
- Monitor server resource utilization (CPU, memory, disk I/O) during load testing to identify performance bottlenecks.

### Security Testing:

- Perform vulnerability scanning using tools like Nessus or OpenVAS to identify and mitigate security vulnerabilities.
- Review Nginx access logs and error logs for suspicious activity or unauthorized access attempts.

### High Availability Testing:

- Test failover scenarios by simulating server failures and ensuring seamless traffic redirection to healthy servers.

## Database Server Testing

### Functional Testing:

- Verify database connectivity from web servers using MySQL command-line tools or PHP scripts.
- Test database replication to ensure data consistency between master and slave servers.
- Check database backup and restore procedures to ensure data integrity and disaster recovery capability.

### Performance Testing:

- Execute database queries under various load conditions to evaluate query response times and database throughput.
- Monitor database server resource utilization (CPU, memory, disk I/O) during performance testing to identify optimization opportunities.

### Security Testing:

- Perform database security assessments to identify and remediate security vulnerabilities such as weak authentication methods or unencrypted connections.
- Review MySQL error logs for suspicious activity or unauthorized access attempts.

### Data Integrity Testing:

- Validate data integrity by comparing data between master and slave servers and ensuring replication consistency.

## Monitoring Server Testing

### Functional Testing:

- Verify Zabbix server and frontend accessibility through web browsers.
- Test Zabbix agent connectivity from monitored hosts to ensure data collection and monitoring capability.

- 
- Create test triggers and alerts to verify notification functionality.

## Performance Testing:

- Monitor Zabbix server resource utilization (CPU, memory, disk I/O) under various monitoring loads to ensure scalability.
- Evaluate Zabbix frontend responsiveness and performance when viewing large datasets or generating reports.

## Security Testing:

- Review Zabbix configuration settings to ensure secure communication between agents and the server.
- Test user authentication and access control mechanisms to prevent unauthorized access to monitoring data.

## Alerting and Notification Testing:

- Trigger test events on monitored hosts to verify alerting and notification workflows.

**Schedule Detailed task list is captured as part of the Detailed Business Implementation Task List. A sample matrix is captured here to identify key milestones.**

Planner

CC\_Capstone-Group5 CC\_Capstone-Group6

Grid Board Charts Schedule ...

HB AT DP SL +1 Members Filter (0)

Title	Assignment	Start Date	Due Date	Bucket	Progress	Priority
Network Topology Analysis	AT Akshay Janardan Te	2/3/2024	2/17/2024	To do	Completed	Medium
Network for Cloud platform Designing	AT Akshay Janardan Te	2/19/2024	3/1/2024	To do	Completed	Medium
Network Testing	AT Akshay Janardan Te	3/2/2024	3/9/2024	To do	Completed	Medium
Network Deployment over the cloud	SL AT	3/10/2024	3/14/2024	To do	Completed	Medium
Cloud Network Troubleshooting and Resolving issues	SL AT	3/15/2024	3/23/2024	To do	Completed	Medium
Analysis of Failover capabilities on cloud	HB Heli Alpeshbhai Bh	2/3/2024	2/10/2024	To do	Completed	Medium
Analysis of fault tolerance capabilities on cloud	HB Heli Alpeshbhai Bh	2/11/2024	2/18/2024	To do	Completed	Medium
Configuration of Failover on cloud	HB Heli Alpeshbhai Bh	2/19/2024	3/2/2024	To do	Completed	Medium
Configuration of Fault Tolerance on Cloud	HB Heli Alpeshbhai Bh	3/4/2024	3/10/2024	To do	Completed	Medium
Analysis of Monitoring tools	HB Heli Alpeshbhai Bh	3/11/2024	3/17/2024	To do	Completed	Medium
Configuration and Deployment of tools	SL HB	3/18/2024	3/20/2024	To do	Completed	Medium
Monitor the cloud component using monitoring tool	HB Heli Alpeshbhai Bh	2/24/2024	2/24/2024	To do	Completed	Medium
Cloud platform and application final phase and testing	SL HB AT +2	3/25/2024	3/25/2024	To do	Completed	Medium

+ Add new task

Planner

CC\_Capstone-Group5 CC\_Capstone-Group6

Grid Board Charts Schedule ...

HB AT DP SL +1 Members Filter (0)

Title	Assignment	Start Date	Due Date	Bucket	Progress	Priority
Front-end Application Containerization Deployment	HB DP	3/23/2024	3/24/2024	To do	Completed	Mark task as incomplete
Backend-end Application Analysis	Kirtirajsinh Rajendr.	2/3/2024	2/18/2024	To do	Completed	Medium
Backend-end Application Implementation	Kirtirajsinh Rajendr.	2/19/2024	2/25/2024	To do	Completed	Medium
Backend-end Application Testing	Kirtirajsinh Rajendr.	2/26/2024	2/29/2024	To do	Completed	Medium
Backend-end Application Feedback and Troubleshooting	Kirtirajsinh Rajendr.	3/1/2024	3/15/2024	To do	Completed	Medium
Backend-end Application Deployment	Kirtirajsinh Rajendr.	3/16/2024	3/19/2024	To do	Completed	Medium
Backend-end Application Containerization	Kirtirajsinh Rajendr.	3/20/2024	3/21/2024	To do	Completed	Medium
Backend-end Application Containerization Troubleshootin...	Kirtirajsinh Rajendr.	3/22/2024	3/23/2024	To do	Completed	Medium
Backend-end Application Containerization Deployment	HB	3/23/2024	3/24/2024	To do	Completed	Medium
Cloud-platform Analysis	SL Shivani Vijaykumar	2/3/2024	2/11/2024	To do	Completed	Medium
Cloud-platform Configuration and Implementation	SL Shivani Vijaykumar	2/12/2024	2/29/2024	To do	Completed	Medium
Cloud-Platform Deployment	SL Shivani Vijaykumar	3/1/2024	3/8/2024	To do	Completed	Medium
Cloud-Platform Testing and Troubleshooting	SL AT	3/9/2024	3/24/2024	To do	Completed	Medium

+ Add new task

Planner

CC\_Capstone-Group5

Title	Assignment	Start Date	Due Date	Bucket	Progress	Priority
work instruction document	SL, AT +2	3/19/2024	4/2/2024	To do	In progress	Medium
Final Presentation Preparation	SL, AT +2	3/19/2024	4/2/2024	To do	In progress	Medium
Build book Creation	SL, AT +2	3/19/2024	4/2/2024	To do	In progress	Medium
Requirement Gathering and Technology Analysis	SL, AT +2	1/5/2024	1/19/2024	To do	Completed	Medium
Project Planning and Feasibility Study	SL, AT +2	1/20/2024	2/2/2024	To do	Completed	Medium
Front-end Application Analysis	DP Dhvanikumar Kalp	2/3/2024	2/18/2024	To do	Completed	Medium
Front-end Application Implementation	DP Dhvanikumar Kalp	2/19/2024	2/25/2024	To do	Completed	Medium
Front-end Application Testing	DP Dhvanikumar Kalp	2/26/2024	2/29/2024	To do	Completed	Medium
Front-end Application Feedback and Troubleshooting	DP Dhvanikumar Kalp	3/1/2024	3/15/2024	To do	Completed	Medium
Front-end Application Deployment	DP Dhvanikumar Kalp	3/16/2024	3/19/2024	To do	Completed	Medium
Front-end Application Containerization	DP Dhvanikumar Kalp	3/20/2024	3/21/2024	To do	Completed	Medium
Front-end Application Containerization Troubleshooting a...	DP Dhvanikumar Kalp	2/22/2024	3/22/2024	To do	Completed	Medium
Front-end Application Containerization Deployment	HB, DP	3/23/2024	3/24/2024	To do	Completed	Medium

+ Add new task

## Roles & Responsibilities

Name /Role	Responsibilities
Project Leader	<ul style="list-style-type: none"> <li>Lead the team.</li> <li>Assigned work.</li> <li>Keep track on task complication.</li> <li>Solving any doubts of all the team members</li> <li>Final inspection of the project.</li> <li>Created documentation with the help of all the team members taking inputs from each equally.</li> <li>Checked the infrastructure working as expected.</li> <li>Checked all VM's and functionalities.</li> </ul>
Shivani/Akshay	<ul style="list-style-type: none"> <li>Created Detailed implementation task list for implementation created open-stack and servers in infrastructure</li> <li>Added features of file server.</li> <li>Created GPOs</li> <li>Created monitoring infrastructure via installing monitoring client and agents.</li> <li>Design diagram</li> </ul>
Heli	<ul style="list-style-type: none"> <li>Designed application's user interface</li> <li>Analysis and research of UI/UX interface of our system</li> <li>Help in configuring network infrastructure</li> </ul>

	<ul style="list-style-type: none"> <li>· design all the pages for whole website</li> </ul>
Kirtiraj	<ul style="list-style-type: none"> <li>· Created Detailed implementation task list for implementation of back-end source code.</li> </ul>
	<ul style="list-style-type: none"> <li>· Added features if required and make changes to source code</li> </ul>
	<ul style="list-style-type: none"> <li>· Created Docker files and configure application.</li> </ul>
	<ul style="list-style-type: none"> <li>· Run and test the backend code</li> </ul>
Dhvanil	<ul style="list-style-type: none"> <li>· Created Detailed implementation task list for implementation of front end.</li> </ul>
	<ul style="list-style-type: none"> <li>· Added features if required and make changes to source code for front end code</li> </ul>
	<ul style="list-style-type: none"> <li>· Tested and verified all the issues and bottlenecks</li> </ul>
	<ul style="list-style-type: none"> <li>· Created and implemented web application</li> </ul>

## Reference:

*OpenStack Installation Guide — Installation Guide documentation. (n.d.).*

<https://docs.openstack.org/install-guide/>