BHARATI VIDYAPEETH'S
INSTITUTE OF COMPUTER APPLICATIONS & MANAGEMENT
(Affiliated to Guru Gobind Singh Indraprastha University, Approved by AICTE, New Delhi)

Project Synopsis

Submitted By:
Pushpender Rana (02911604418)
MCA 6th Sem, Sec 1

INDEX

S. No.	Title	Page No.
1.	Project Title	1
2.	Problem Statement	1
3.	Present state of the art	1
4	Objectives	1
5	Hardware Requirements	2
6	Software Requirements	2
7	Technology Stack	2
8	Testing Technologies	2
9	Value addition by the project	3
10	Limitations	3
11	Future Scope	3
12	References and Bibliography	4

1. Project Title

KVM Virtual Machines and Docker control panel

2. Problem Statement

To develop a hypervisor system for Grappus Technologies Private Limited capable of creating, managing and using KVM based virtual machines using a web application. The system should also have the capabilities to manage docker subsystem running on the host operating system along with the capabilities to perform basic sysadmin and maintenance tasks on the host os. The system will be used for managing their testing and production Virtual Machine instances and applications running inside containers.

3. Present state of the art

The following points sum up the current problems that are faced.

- The current web based management system for virtual machines and docker containers require the user to install a custom operating system.
- They require one or more gigabyte of RAM to maintain their own functioning.
- They do not have components that can be easily replaced by custom ones according to requirement without modifying the source code.
- Users have to modify their host file system in order to install them.
- These management applications once installed are not easy to remove or replace.
- They usually don't play well with existing installations of virtualization platforms.
- They usually contain advanced levels of functionality that a common user may not require at all on leads to increase in complexity.
- The user either has to run the entire software suite otherwise they have to modify the source code.
- These virtualization platforms don't support managing docker containers.
- The existing platforms are not made such that they can easily be ported for other virtualization platforms if the need arises.

4. Objective and scope of the project

The proposed system aims at achieving the following targets in its course of action.

- To allow the user to create virtual machines on Linux based systems with KVM support.
- To allow the users to manage virtual machines running on a Linux host..
- To allow the users to create docker containers on their systems.
- To allow the users to manage running and existing docker containers on their system.
- To allow the users to access the logs and terminal of running docker containers.

- To allow the users to access the terminal of the host system via a web interface
- To allow the users to access the display of virtual machines via a web interface.
- To allow the users to easily replace components of the with their own custom ones.
- To allow the users to manage their virtual machine environment without modifying their own file system or environment.
- To develop a system that can be easily ported to other virtualization platforms.

6. Hardware Requirements

- Processor: 2GHz or more with minimum 4 cores.
- Ethernet connection (LAN) with 1 Gigabits / second speeds.
- Hard Drive: Minimum 100 GB; Recommended 1TB or more.
- Memory (RAM): Minimum 4 GB; Recommended 8 GB or above.

7. Software Requirements

- Docker v19.3 or higher
- Docker Compose v1.2 or higher.
- Web Browser - Google Chrome 75 or Firefox 40
- KVM module and tools
- Linux operating system with kernel 2.6 or higher
- QEMU and libvirt
- Virsh tools.
- Keycloak or any other openid provider.
- Nginx reverse proxy server v1.16 or higher.

8. Technology Stack

- **Frontend:** ReactJS
- **Backend:** Django
- **Database:** SQLite

8. Testing Technologies

Manual testing to be carried out.

9. What contribution / value addition would the project make?

- The proposed system will make it much easier to turn any linux based host with KVM tools and docker into a virtualization hypervisor.
- The system will allow the user to run only the parts they require due to its modularity thus leading into lesser resource consumption.
- The system architecture will support porting to other virtualization platforms such as Xen, Vbox with the least amount of effort as only a single layer would need to be rewritten thus supporting code reuse.
- The ability to perform basic sysadmin tasks using a web based terminal will make it much easier to remotely maintain a host machine.
- Freedom of choosing an openid provider for authentication and using docker as a container platform will allow the system to be easily deployed to any Linux, Windows or Mac systems (Provided a platform specific daemon is written for the target platform).
- The resource consumption of the system will be minimal and would not require any additional dependencies other than docker to be installed in order to function thus preventing dependency pollution in the host os. The removal of the system is also trivial as it will refrain from making breaking changes to the host.

10. Limitations / constraints of the project

- The system will not support podman or podman-compose.
- No support for docker-compose will be present.
- Authentication will be dependent on third party open id connect providers.
- No access control system will be present
- Only qcow2 disk format will be supported.
- Limited networking support.
- Host health checks and health monitors will not be supported.

Future Scope

- Add full networking support for VMs and containers.
- Support for platform specific auth protocols for allowing multiple user roles.
- Implement an access control list.
- Support for USB and PCI passthrough for devices.
- Support for VM disk formats.
- Support for podman, podman compose and docker compose.
- Disk and system health monitoring and email notifications for the same.

References and Bibliography

- https://libvirt.org/drvqemu.html
- https://docs.docker.com/
- https://docs.djangoproject.com/en/3.2/intro/tutorial01/
- https://www.freecodecamp.org/news/tag/react/
- https://www.w3schools.com/html/html_css.asp
- https://www.linux-kvm.org/page/Main_Page
- https://libvirt.org/manpages/virsh.html
- https://qemu-project.gitlab.io/qemu/
- https://github.com/novnc/websockify