Docker Monitoring Tool
BTP Evaluation 1

What is Docker?

Docker is all about making it easier to **create, deploy, and run applications** by using containers. Containers allow a developer to package up an application with all of the parts it needs, such as libraries and other dependencies, and ship it all out as one package. He can rest assured that the application will run on any other **Linux machine** regardless of any customized settings that machine might have that could differ from the machine used for writing and testing the code.

How was it developed?

Docker Monitor Tool was developed using Python curses library and docker-py.

What is our Tool about ?

- Docker Monitoring Tool analyzes Resource usage (CPU%) and Start/Stop Action for containers.
- It automatically discovers and monitors new containers when they are created.
- The user is given option to stop the specific container as well.
- All information about container's is shown in a TUI.

Docker v/s VM

Docker is a bit like a virtual machine. But unlike a virtual machine, rather than creating a **whole virtual operating system**, Docker allows applications to use the same Linux kernel as the system that they're running on and only requires applications be shipped with things not already running on the host computer. This gives a significant **performance boost** and reduces the **size** of the application.

And importantly, Docker is **open source**. This means that anyone can contribute to Docker and extend it to meet their own needs if they need additional features that aren't available out of the box.

Who is it for?

Docker is a tool that is designed to benefit both **developers and system administrators**, making it a part of many DevOps (developers + operations) tool chains.

Pros

- With a perfectly tuned container system, you can have as many as **four-to-six** times the number of server application instances as you can using Xen or KVM VMs on the same hardware.
- Containers use shared operating systems. That means they are much **more efficient** than hypervisors in system resource terms. Instead of virtualizing hardware, containers rest on top of a single Linux instance.

Docker Monitoring Tool
BTP Evaluation 1

What is Docker?

Docker is all about making it easier to **create, deploy, and run applications** by using containers. Containers allow a developer to package up an application with all of the parts it needs, such as libraries and other dependencies, and ship it all out as one package. He can rest assured that the application will run on any other **Linux machine** regardless of any customized settings that machine might have that could differ from the machine used for writing and testing the code.

How was it developed?

Docker Monitor Tool was developed using Python curses library and docker-py.

What is our Tool about ?

- Docker Monitoring Tool analyzes Resource usage (CPU%) and Start/Stop Action for containers.
- It automatically discovers and monitors new containers when they are created.
- The user is given option to stop the specific container as well.
- All information about container's is shown in a TUI.

Docker v/s VM

Docker is a bit like a virtual machine. But unlike a virtual machine, rather than creating a **whole virtual operating system**, Docker allows applications to use the same Linux kernel as the system that they're running on and only requires applications be shipped with things not already running on the host computer. This gives a significant **performance boost** and reduces the **size** of the application.

And importantly, Docker is **open source**. This means that anyone can contribute to Docker and extend it to meet their own needs if they need additional features that aren't available out of the box.

Who is it for?

Docker is a tool that is designed to benefit both **developers and system administrators**, making it a part of many DevOps (developers + operations) tool chains.

Pros

- With a perfectly tuned container system, you can have as many as **four-to-six** times the number of server application instances as you can using Xen or KVM VMs on the same hardware.
- Containers use shared operating systems. That means they are much **more efficient** than hypervisors in system resource terms. Instead of virtualizing hardware, containers rest on top of a single Linux instance.

What is Docker?

Docker is all about making it easier to **create, deploy, and run applications** by using containers. Containers allow a developer to package up an application with all of the parts it needs, such as libraries and other dependencies, and ship it all out as one package. He can rest assured that the application will run on any other **Linux machine** regardless of any customized settings that machine might have that could differ from the machine used for writing and testing the code.

Docker v/s VM

Docker is a bit like a virtual machine. But unlike a virtual machine, rather than creating a **whole virtual operating system,** Docker allows applications to use the same Linux kernel as the system that they're running on and only requires applications be shipped with things not already running on the host computer. This gives a significant **performance boost** and **reduces the size** of the application.

And importantly, Docker is **open source**. This means that anyone can contribute to Docker and extend it to meet their own needs if they need additional features that aren't available out of the box.

Who is it for?

Docker is a tool that is designed to benefit both **developers and system administrators**, making it a part of many Dev Ops (developers + operations) tool chains.

Pros

- With a perfectly tuned container system, you can have as many as **four-to-six** times the number of server application instances as you can using Xen or KVM VMs on the same hardware.
- Containers use shared operating systems. That means they are much **more efficient** than hypervisors in system resource terms. Instead of virtualizing hardware, containers rest on top of a single Linux instance.

What is our Tool about ?

- Docker Monitoring Tool analyzes Resource usage (CPU%) and Start/Stop Action for containers.
- It automatically discovers and monitors new containers when they are created.
- The user is given option to stop the specific container as well.
- All information about containers is shown in a TUI.

How was it developed?

Docker Monitor Tool was developed using Python curses library and docker-py.

How to Deploy

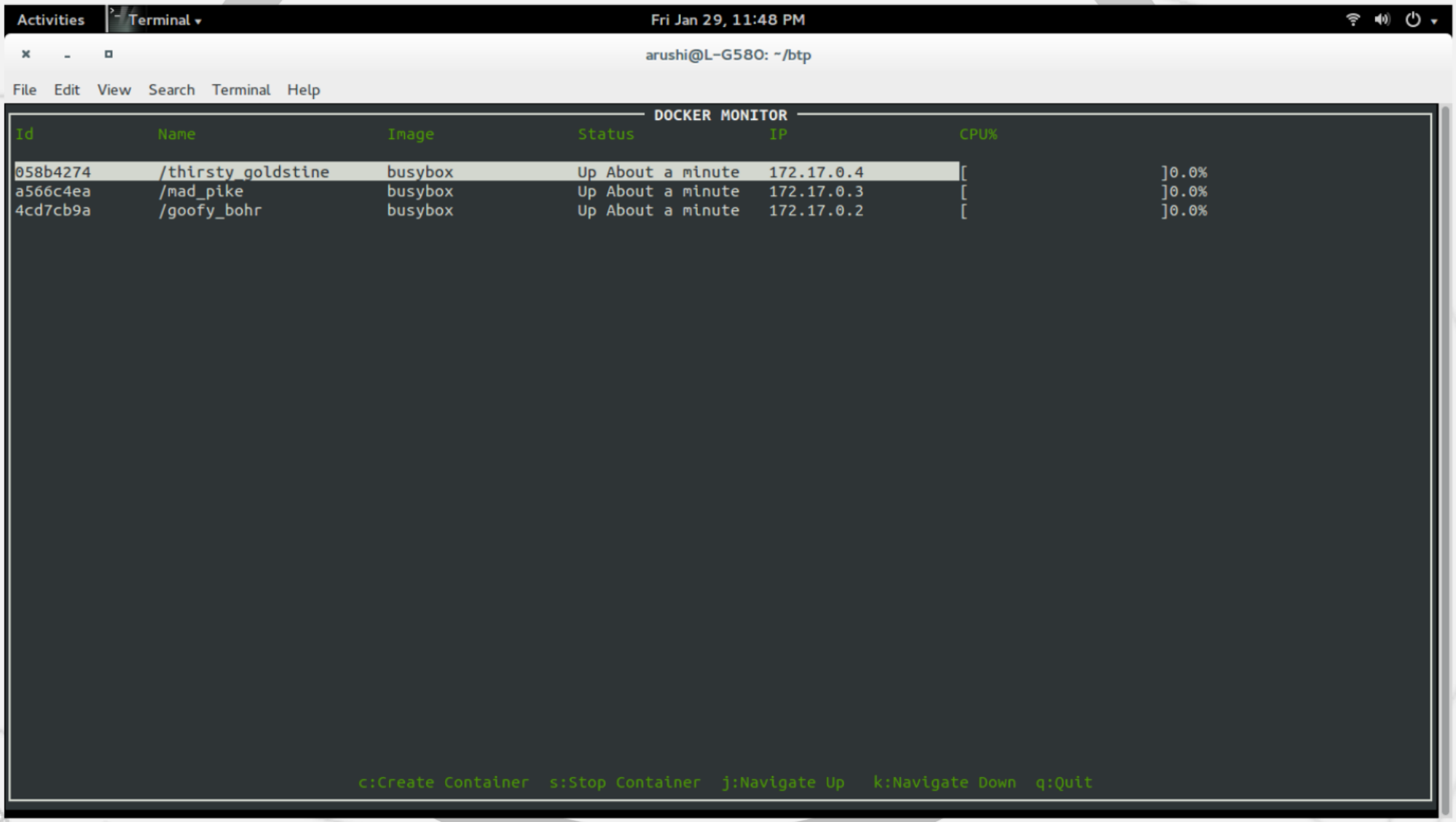
pip install -r requirements.txt

python docker-monitor.py

Future Scope

- Start/Stop arguments for containers
- Graphs for depicting CPU and memory usage
- Detailed Information on each container

Screenshot of Docker Monitor



Docker Monitoring Tool
BTP Evaluation 1

What is Docker?

Docker is all about making it easier to **create, deploy, and run applications** by using containers. Containers allow a developer to package up an application with all of the parts it needs, such as libraries and other dependencies, and ship it all out as one package. He can rest assured that the application will run on any other **Linux machine** regardless of any customized settings that machine might have that could differ from the machine used for writing and testing the code.

How was it developed?

Docker Monitor Tool was developed using Python curses library and docker-py.

What is our Tool about ?

- Docker Monitoring Tool analyzes Resource usage (CPU%) and Start/Stop Action for containers.
- It automatically discovers and monitors new containers when they are created.
- The user is given option to stop the specific container as well.
- All information about container's is shown in a TUI.

Docker v/s VM

Docker is a bit like a virtual machine. But unlike a virtual machine, rather than creating a **whole virtual operating system**, Docker allows applications to use the same Linux kernel as the system that they're running on and only requires applications be shipped with things not already running on the host computer. This gives a significant **performance boost** and reduces the **size** of the application.

And importantly, Docker is **open source**. This means that anyone can contribute to Docker and extend it to meet their own needs if they need additional features that aren't available out of the box.

Who is it for?

Docker is a tool that is designed to benefit both **developers and system administrators**, making it a part of many DevOps (developers + operations) tool chains.

Pros

- With a perfectly tuned container system, you can have as many as **four-to-six** times the number of server application instances as you can using Xen or KVM VMs on the same hardware.
- Containers use shared operating systems. That means they are much **more efficient** than hypervisors in system resource terms. Instead of virtualizing hardware, containers rest on top of a single Linux instance.