

Documentation

Code: [GitHub Code](#)

API: The rest API is developed by using SpringBoot.

Testing: manually tested endpoints with test Cases in src/test folder

Authentication: Authenticated using OAuth. Tried to integrate API Keys and OAuth for MFA but only either one worked out.

Daemon: After deployment on AWS EC2 instance the api is made as a daemon process by using systemd-service file .

Deployment: To deploy the API on AWS EC2 instance, a docker image of the api is created and pushed onto the docker hub. Below is the code of the dockerfile(this is present in github as well)

Dockerfile

```
# Use the official OpenJDK image as a base
FROM openjdk
```

```
# Set the working directory in the container
WORKDIR /app
```

```
# Copy the JAR file into the container
COPY target/*.jar app.jar
```

```
# Expose the port on which your API runs
EXPOSE 8080
```

```
# Run the JAR file when the container starts
CMD ["java", "-jar", "app.jar"]
```

With the help of the docker file the api is dockerized and available at [DockerHub](#)

Steps for deploying on EC2 instance before daemonizing the API:

- 1) Sudo yum install docker
- 2) Sudo systemctl start docker
- 3) sudo docker pull kranthil/repo:homework1-image2
- 4) sudo docker run -d -p 8000:8000 kranthil/repo:homework1-image2
- 5) curl -L -H "KEY: secret" <http://localhost:8000/api/cpu>
The same command can be extended to other endpoints as well.
- 6) Sudo docker ps - check the container ID to stop
- 7) Sudo docker stop <cont_id>
- 8) Sudo docker rm <cont_id>

After testing the API on EC2, systemd is used for daemonizing. The steps below gives the flow.

- 9) Create a bash file
- 10) Vim rest_api.sh

Rest api.sh

```
#!/bin/bash  
sudo docker run -d -p 8000:8000
```

```
kranthil/repo:homework1-image2
```

```
Sudo chmod +x rest_api.sh
```

Create a service file in /etc/systemd/system and name it as rest_api.service

```
Sudo chmod +x rest_api.service
```

[Unit]

Description= System Monitoring Service

After= network.target

[Service]

ExecStart=/home/ec2-user/rest_api.sh

Restart=Always

StandardOutput=file:/var/log/rest_api.log

StandardError=file:/var/log/rest_api.err

[Install]

WantedBy=multi-user.target

Sudo systemctl enable rest_api.service

Sudo systemctl start rest_api

Sudo systemctl status

Sudo docker ps #to check whether the docker image is running on re-starting the EC2 instance.