

I have created separate pipelines for the dev and prod environments. Whenever a developer pushes code to the Git repository, the project is automatically built and deployed to the Docker repository. I have attached a screenshot for reference

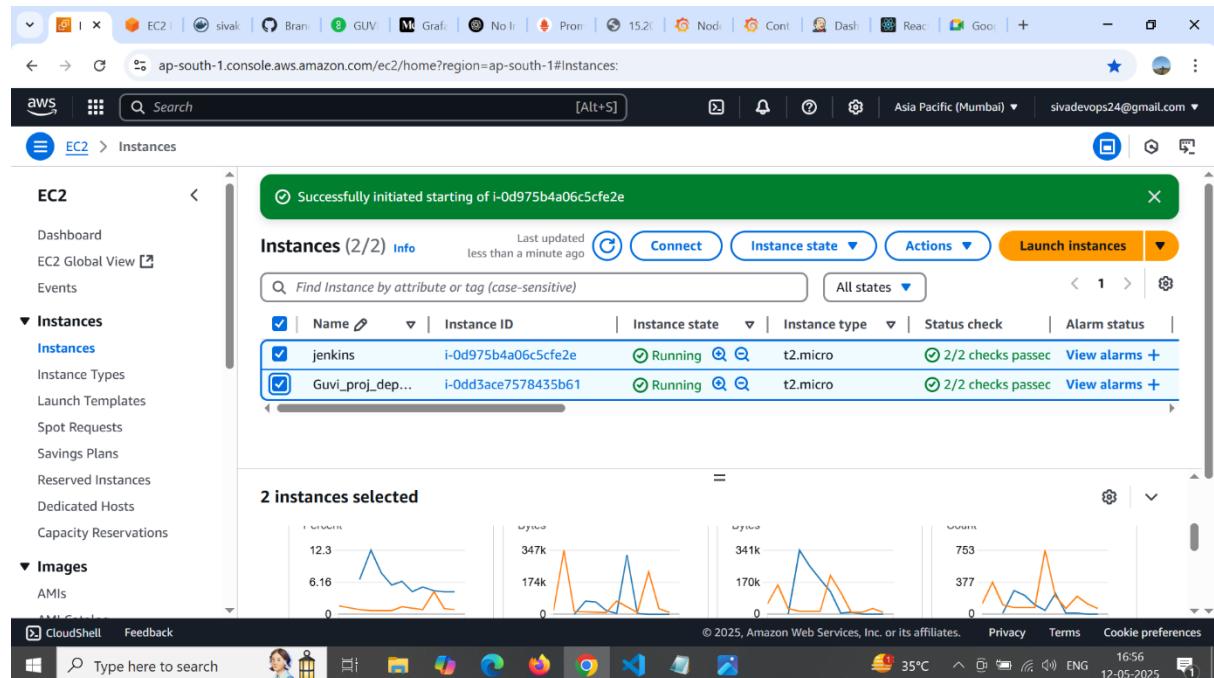
Git hub Url : <https://github.com/Sivarepo24/project1>

Docker dev Image : docker pull sivakumar135/guvi_project_dev

Docker prod Image : docker pull sivakumar135/guvi_project_prod

Deploy site url : <http://65.0.138.58>

Ec2 Instance

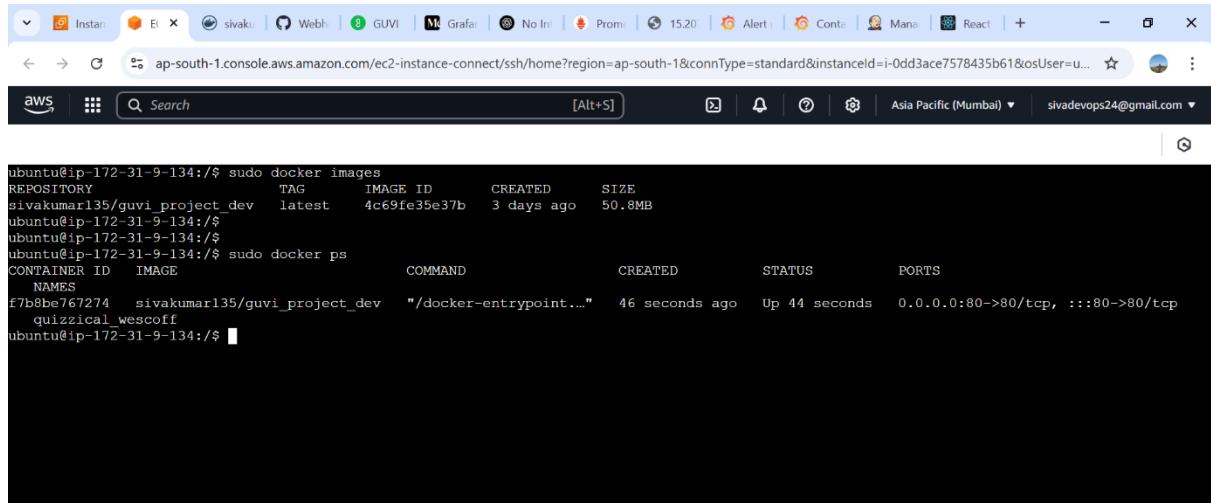


The screenshot shows the AWS EC2 Instances page. A success message at the top indicates "Successfully initiated starting of i-0d975b4a06c5cfe2e". The main table displays two instances:

Name	Instance ID	Instance state	Instance type	Status check
jenkins	i-0d975b4a06c5cfe2e	Running	t2.micro	2/2 checks passed
Guvi_proj_dep...	i-0dd3ace7578435b61	Running	t2.micro	2/2 checks passed

Below the table, it says "2 instances selected". There are four line charts showing monitoring data for each instance.

Pull image to docker repo



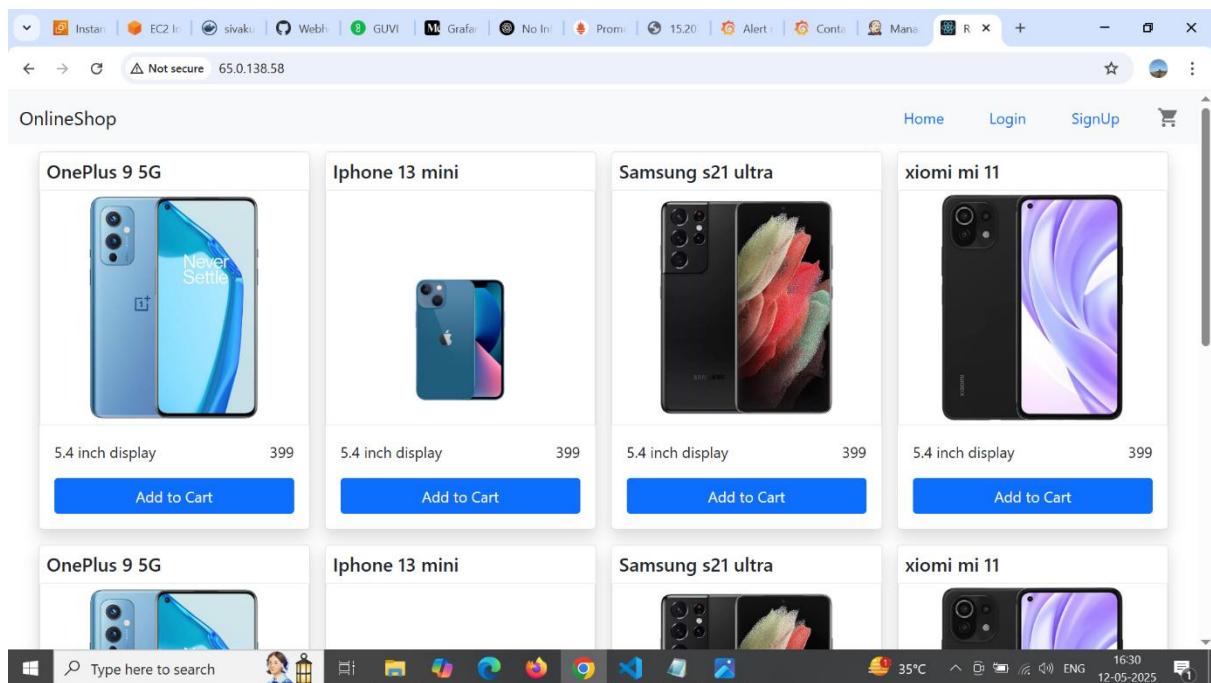
```
ubuntu@ip-172-31-9-134:/$ sudo docker images
REPOSITORY          TAG      IMAGE ID      CREATED        SIZE
sivakumar135/guvi_project_dev  latest   4c69fe35e37b  3 days ago   50.8MB
ubuntu@ip-172-31-9-134:/$
ubuntu@ip-172-31-9-134:/$ sudo docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED          STATUS          PORTS
 NAMES
f7b8be767274   sivakumar135/guvi_project_dev   "/docker-entrypoint..."   46 seconds ago   Up 44 seconds   0.0.0.0:80->80/tcp, :::80->80/tcp
ubuntu@ip-172-31-9-134:/$
```

i-0dd3ace7578435b61 (Guvi_proj_deploy)

Public IPs: 65.0.138.58 Private IPs: 172.31.9.134



Output



The screenshot shows a web browser displaying a mobile phone shopping page titled "OnlineShop". The page lists four phones in a grid:

Phone Model	Display Size	Price	Action
OnePlus 9 5G	5.4 inch display	399	Add to Cart
Iphone 13 mini	5.4 inch display	399	Add to Cart
Samsung s21 ultra	5.4 inch display	399	Add to Cart
xiom mi 11	5.4 inch display	399	Add to Cart

Below this grid, there is another row of three phones:

Phone Model	Display Size	Price	Action
OnePlus 9 5G			
Iphone 13 mini			
Samsung s21 ultra			
xiom mi 11			

The browser's address bar shows the URL as "65.0.138.58". The taskbar at the bottom of the screen shows various application icons and system status.

SG – Guvi_proj_deploy

The screenshot shows the AWS Management Console interface for modifying inbound security group rules. The URL in the address bar is `ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#ModifyInboundSecurityGroupRules:securityGroupId=sg-01b098ed904fff33a`. The page title is "Edit inbound rules". The "Inbound rules" section lists four rules:

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-0fbcb8fb0dc3fb855	HTTP	TCP	80	Cu... ▾	0.0.0.0/0
sgr-09c14cdacd0ca3698	All TCP	TCP	0 - 6553	Cu... ▾	0.0.0.0/0
sgr-0313b9abb207ec240	SSH	TCP	22	Cu... ▾	0.0.0.0/0
sgr-0f859f667287629a3	HTTPS	TCP	443	Cu... ▾	0.0.0.0/0

At the bottom left is a blue "Add rule" button. The bottom of the screen shows the Windows taskbar with various pinned icons and the system tray.

Jenkins instance security Group

The screenshot shows the AWS Management Console interface for modifying inbound security group rules. The URL in the address bar is `ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#ModifyInboundSecurityGroupRules:securityGroupId=sg-0289360fdfadc9164`. The page title is "Edit inbound rules". The "Inbound rules" section lists four rules:

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-0af49da07eb288641	Custom TCP	TCP	8080	Cu... ▾	0.0.0.0/0
sgr-0d8c6ed128da4e1eb	HTTPS	TCP	443	Cu... ▾	0.0.0.0/0
sgr-098843b35711a1a34	SSH	TCP	22	Cu... ▾	0.0.0.0/0
sgr-0eafb15c15bdd3f53	HTTP	TCP	80	Cu... ▾	0.0.0.0/0

At the bottom left is a blue "Add rule" button. The bottom of the screen shows the Windows taskbar with various pinned icons and the system tray.

Jenkins

The screenshot shows the Jenkins dashboard. At the top, there is a header bar with various links like 'Instances', 'EC2 In...', 'sivaku', 'Branch...', 'GUUI', 'Grafan...', 'No Int...', 'Prom...', '15.20...', 'Node...', 'Cont...', 'React...', and a '+' button. Below the header is a search bar and a toolbar with icons for 'New Item', 'Build History', 'All', and '+'. The main area has two sections: 'Build Queue' (empty) and 'Build Executor Status'. The 'Build Executor Status' section lists two executors: 'Guvi_dev_deploy' and 'Guvi_prod_deploy'. Both executors have a green checkmark icon, indicating they are healthy. The 'Guvi_dev_deploy' executor's last success was 19 min ago (#7) and its last failure was 3 days 4 hr ago (#4). The 'Guvi_prod_deploy' executor's last success was 22 min ago (#37) and its last failure was 3 days 4 hr ago (#35). At the bottom, there is a toolbar with icons for 'Icon', 'S', 'M', and 'L'.

Jenkins Configure (main-branch)

The screenshot shows the Jenkins configuration page for the 'Guvi_prod_deploy' pipeline. The URL in the browser is '13.233.150.139:8080/job/Guvi_prod_deploy/configure'. The left sidebar has navigation links for 'Jenkins', 'Guvi_prod_deploy', 'Configuration', 'General', 'Triggers', 'Pipeline', and 'Advanced'. The 'Triggers' link is currently selected. The main content area is titled 'Pipeline' and contains a 'Definition' section with a 'Pipeline script from SCM' dropdown set to 'SCM'. Below this, there is a 'Repositories' section with a 'Repository URL' input field containing 'https://github.com/Ssharepo24/project1.git'. There is also a 'Credentials' dropdown set to 'none'. Under 'Advanced', there is a 'Branch Specifier (blank for any)' input field with 'origin/main' entered. At the bottom of the page are 'Save' and 'Apply' buttons. The Windows taskbar at the bottom shows various application icons and the date/time as 12-05-2025 16:23.

Jenkins configure (stage-dev) branch

The screenshot shows the Jenkins Pipeline configuration page for a job named 'Guvi_dev_deploy'. The 'Pipeline' tab is selected in the left sidebar. In the main area, under 'Branches to build', a 'Branch Specifier' field contains 'stage_dev'. Below it, 'Repository browser' is set to '(Auto)'. Under 'Additional Behaviours', there is a 'Script Path' field with 'build/jenkinsfile' and a checked 'Lightweight checkout' option. At the bottom, there are 'Save' and 'Apply' buttons.

GitHub webhook

The screenshot shows the GitHub settings page for a repository. The left sidebar has 'Webhooks' selected. The main form is for creating a new webhook. It includes fields for 'Payload URL' (set to 'http://13.233.150.139:8080/github-webhook/'), 'Content type' (set to 'application/x-www-form-urlencoded'), and 'SSL verification' (set to 'Enable SSL verification'). Below these, under 'Which events would you like to trigger this webhook?', the 'Just the push event' option is selected. At the bottom right, there are 'Save' and 'Test' buttons.

Jenkins: GitHub configure

The screenshot shows the Jenkins job configuration page for the 'Guvi_dev_deploy' job. The 'General' tab is selected. The 'Description' field is empty. Under 'Build triggers', the 'GitHub hook trigger for GITScm polling' option is checked. Other options like 'Build periodically' and 'Poll SCM' are also present. The 'Pipeline' section is collapsed. At the bottom, there are 'Save' and 'Apply' buttons.

The screenshot shows the Jenkins 'Manage Jenkins / System' configuration page. Under 'GitHub', a 'GitHub Server' is defined with the name 'GitHub-Server', API URL 'https://api.github.com', and no credentials. The 'Manage hooks' checkbox is checked. There are 'Save' and 'Apply' buttons at the bottom.

GitHub Branches

The screenshot shows a browser window with multiple tabs open at the top. The active tab is 'github.com/Sivarepo24/project1/branches'. The page displays three sections: 'Default', 'Your branches', and 'Active branches'. Each section contains a table with columns for 'Branch', 'Updated', 'Check status', 'Behind', 'Ahead', and 'Pull request'. The 'Default' section has one row for 'main'. The 'Your branches' section has two rows: 'stage_dev' and 'master'. The 'Active branches' section also has two rows: 'stage_dev' and 'master'. The bottom of the screen shows the Windows taskbar with various pinned icons and system status.

Docker Repo

The screenshot shows a browser window with multiple tabs open at the top. The active tab is 'hub.docker.com/repositories/sivakumar135'. The page displays a sidebar on the left with options like 'Repositories', 'Settings', 'Notifications', 'Billing', 'Usage', 'Pulls', and 'Storage'. The main content area is titled 'Repositories' and shows a table of repositories under the namespace 'sivakumar135'. The table has columns for 'Name', 'Last Pushed', 'Contains', 'Visibility', and 'Scout'. Two repositories are listed: 'sivakumar135/guvi_project_dev' (Last Pushed: 2 minutes ago, Contains: IMAGE, Visibility: Public, Scout: Inactive) and 'sivakumar135/guvi_project_prod' (Last Pushed: 6 minutes ago, Contains: IMAGE, Visibility: Private, Scout: Inactive). The bottom of the screen shows the Windows taskbar with various pinned icons and system status.

Docker: project prod

The screenshot shows the Docker Hub interface for the repository `sivakumar135/guvi_project_prod`. The repository was last pushed 6 minutes ago and has a size of 23.6 MB. It contains one tag, `latest`, which is an image type pulled less than 1 day ago and pushed 6 minutes ago. A Docker Build Cloud card is present, advertising faster builds through shared caching and optimized infrastructure.

Docker commands
To push a new tag to this repository:
`docker push sivakumar135/guvi_project_prod:tagname`

Tag	OS	Type	Pulled	Pushed
latest	Ubuntu	Image	less than 1 day	6 minutes

[See all](#)

Docker : project dev

The screenshot shows the Docker Hub interface for the repository `sivakumar135/guvi_project_dev`. The repository was last pushed 3 minutes ago and has a size of 34.3 MB. It contains one tag, `latest`, which is an image type pulled less than 1 day ago and pushed 3 minutes ago. A Docker Build Cloud card is present, advertising faster builds through shared caching and optimized infrastructure.

Docker commands
To push a new tag to this repository:
`docker push sivakumar135/guvi_project_dev:tagname`

Tag	OS	Type	Pulled	Pushed
latest	Ubuntu	Image	less than 1 day	3 minutes

[See all](#)

Grafana: Alerting Rule

The screenshot shows the 'New alert rule' configuration page in Grafana. The left sidebar is open, showing the 'Alert rules' section. The main area is divided into two steps:

- 1. Enter alert rule name**: A text input field labeled 'Name' contains the value 'Instance_health'.
- 2. Define query and alert condition**: This section includes a Prometheus query editor. The query is set to 'node_cpu_seconds_total' with a label filter 'instance:localhost:9100'. The results table shows several metrics, with the first one being '{name="node_cpu_seconds_total",cpu="0",instance="localhost:9100",job="node_exporter_metrics",mode="idle"} 1839'.

At the top right, there are 'Save rule and exit' and 'Cancel' buttons.

This screenshot continues from the previous one, showing the second step of defining the alert rule.

- 1. Enter alert rule name**: The 'Name' field still contains 'Instance_health'.
- 2. Define query and alert condition**: The Prometheus query editor shows the same setup as before, with the results table displaying metric data. The first metric is again '{name="node_cpu_seconds_total",cpu="0",instance="localhost:9100",job="node_exporter_metrics",mode="idle"} 1839'.

A new 'Options' section is visible at the top of the configuration area, containing settings for 'Legend', 'Min step', 'Format', and 'Type'. The 'Type' dropdown is currently set to 'Time series'.

The screenshot shows the Grafana Alerting interface. On the left, a sidebar menu is open under the 'Alerting' section, with 'Alert rules' selected. The main content area is titled 'Home > Alerting > Alert rules > New alert rule'. A sub-section titled '4. Set evaluation behavior' is displayed, which asks 'Define how the alert rule is evaluated.' It includes a dropdown for 'Evaluation group and interval' set to 'evaluation_group_30s', and a 'Pending period' dropdown set to '1m'. Below these are buttons for 'None', '30s', '1m', '1m30s', '2m', and '2m30s'. At the top right of the content area are 'Save rule and exit' and 'Cancel' buttons.

The screenshot shows the Grafana Alerting interface continuing from the previous step. The 'Recipient' section is visible, with the 'Alertmanager' field set to 'grafana'. Under 'Contact point', the name 'siva' is entered into a dropdown menu, with a link 'View or create contact points' next to it. The 'Email' field contains the value 'sivaravichandran11@gmail.com'. The 'Muting, grouping and timings (optional)' section is also present. At the top right of the content area are 'Save rule and exit' and 'Cancel' buttons. A 'Configure no data and error handling' link is also visible in the previous step's section.

SMTP configure in instance

The screenshot shows a terminal window on an EC2 instance. The user is editing the file `grafana.ini` using `GNU nano 7.2`. The configuration section [smtp] is as follows:

```
[smtp]
enabled = true
host = smtp.gmail.com:465
user = sivaravichandran11@gmail.com
# If the password contains # or ; you have to wrap it with triple quotes. Ex """#password;"""
password =
;cert_file =
;key_file =
skip_verify = true
from_address = sivaravichandran11@gmail.com
from_name = Grafana
;Ehlo identity in SMTP dialog (defaults to instance_name)
;ehlo_identity = dashboard.example.com
; SMTP startTLS policy (defaults to 'OpportunisticStartTLS')
;startTLS_policy = NoStartTLS
; Enable trace propagation in e-mail headers, using the 'traceparent', 'tracestate' and (optionally) 'baggage' fields (defaults to false)
;enable_tracing = false
```

At the bottom of the terminal, the command `i-0dd3ace7578435b61 (Guvi_proj_deploy)` is shown, along with its Public IP (65.0.138.58) and Private IP (172.31.9.134).

The system tray at the bottom of the screen shows the date (12-05-2025), time (15:44), and weather (35°C Haze).

Grafana contact point

The screenshot shows the Grafana interface on a web browser. The user is navigating to the 'Contact points' section under 'Alerting'. They are creating a new contact point named 'siva' using the 'Email' integration.

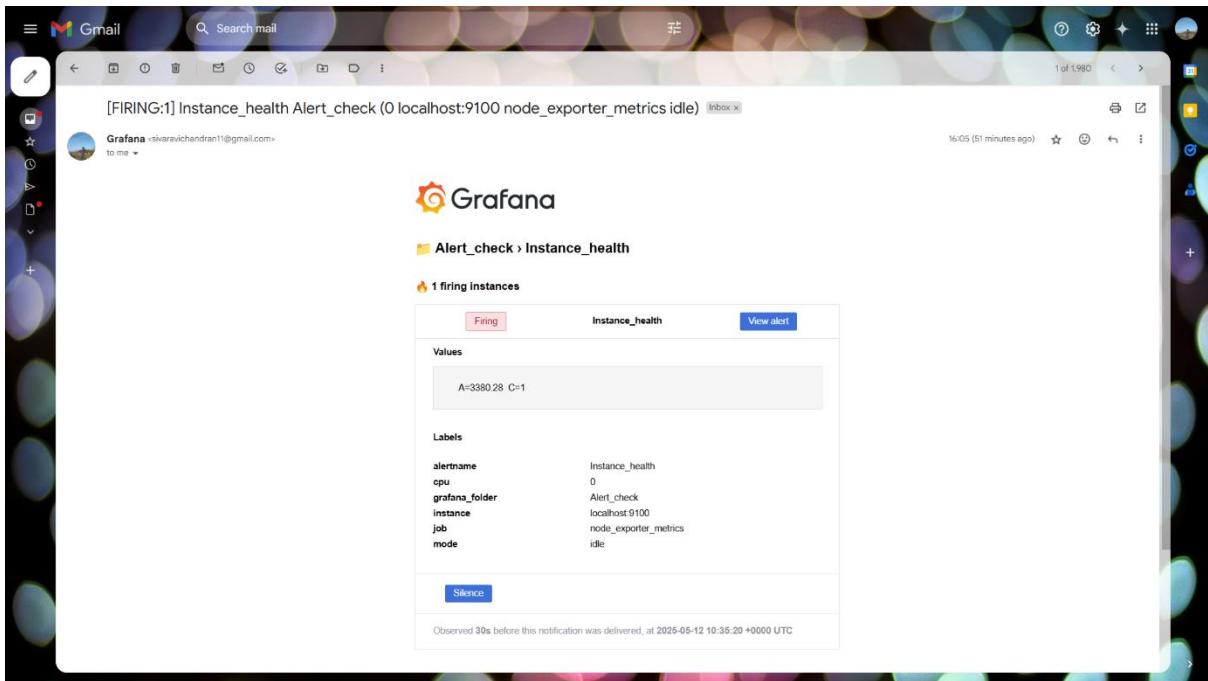
The 'Integration' dropdown is set to 'Email'. The 'Address' field contains the email address `sivaravichandran11@gmail.com`.

Below the integration settings, there are sections for 'Optional Email settings' and 'Notification settings'. A button at the bottom right of the form says '+ Add contact point integration'.

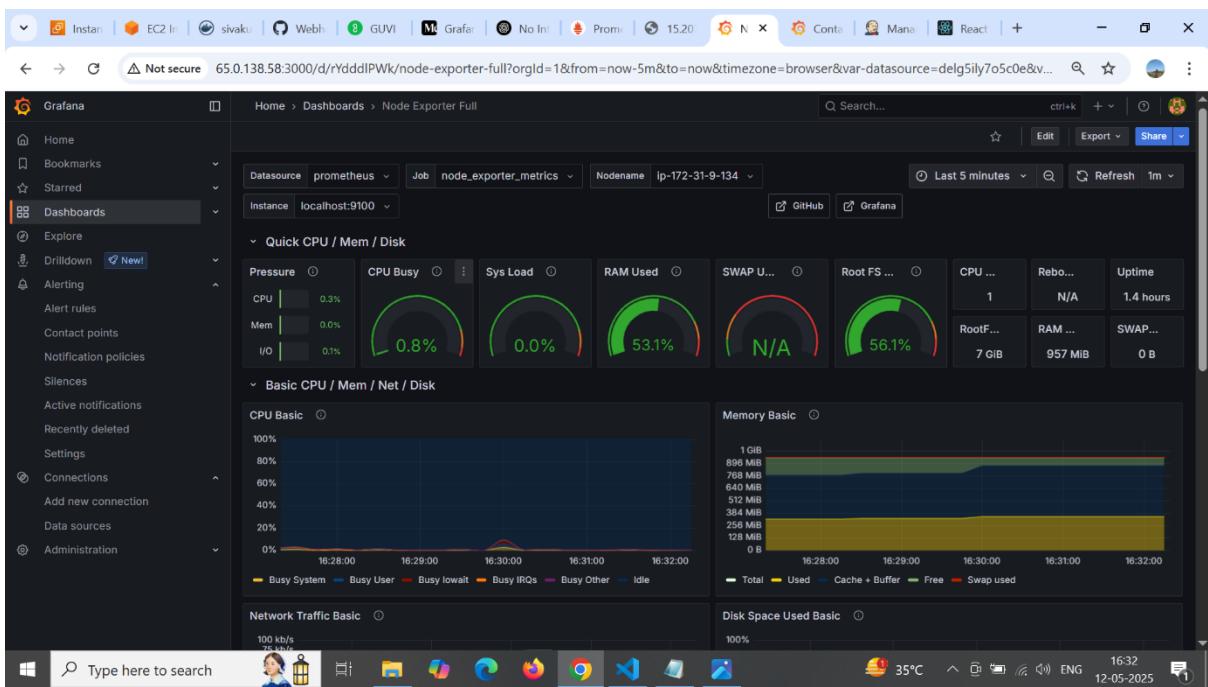
The left sidebar shows other navigation options like Home, Bookmarks, Starred, Dashboards, Explore, Alert rules, and Contact points (which is currently selected).

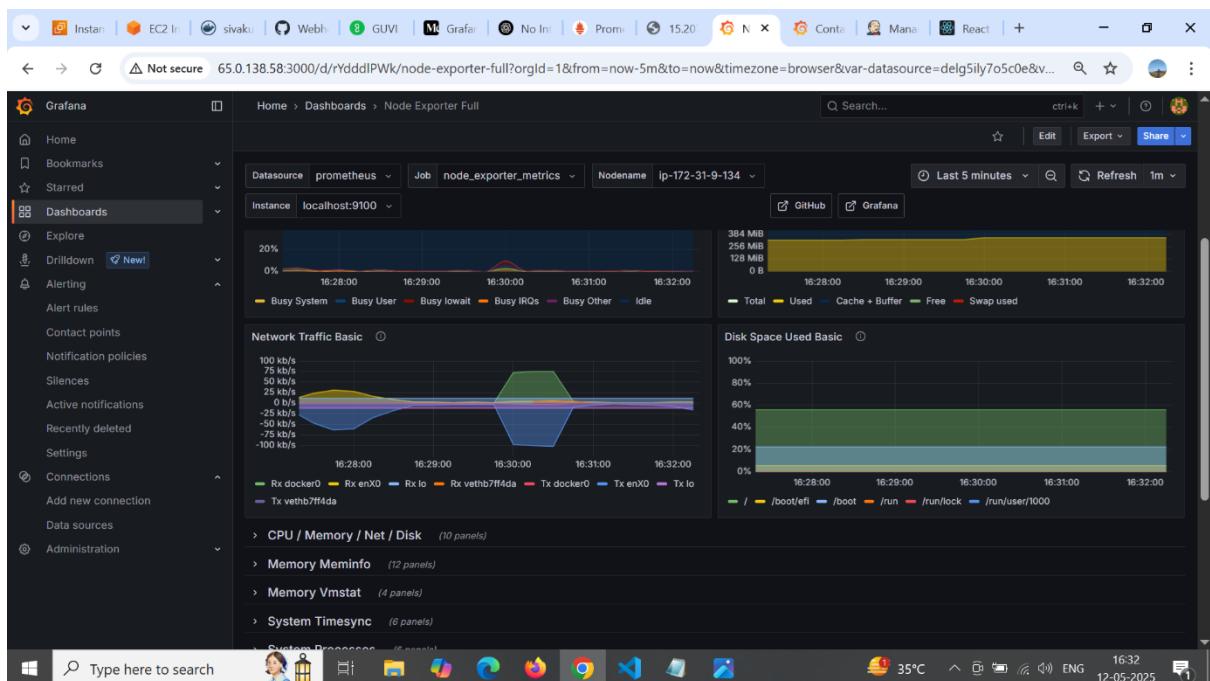
The system tray at the bottom of the screen shows the date (12-05-2025), time (16:00), and weather (35°C Haze).

Grafana Alert E-mail

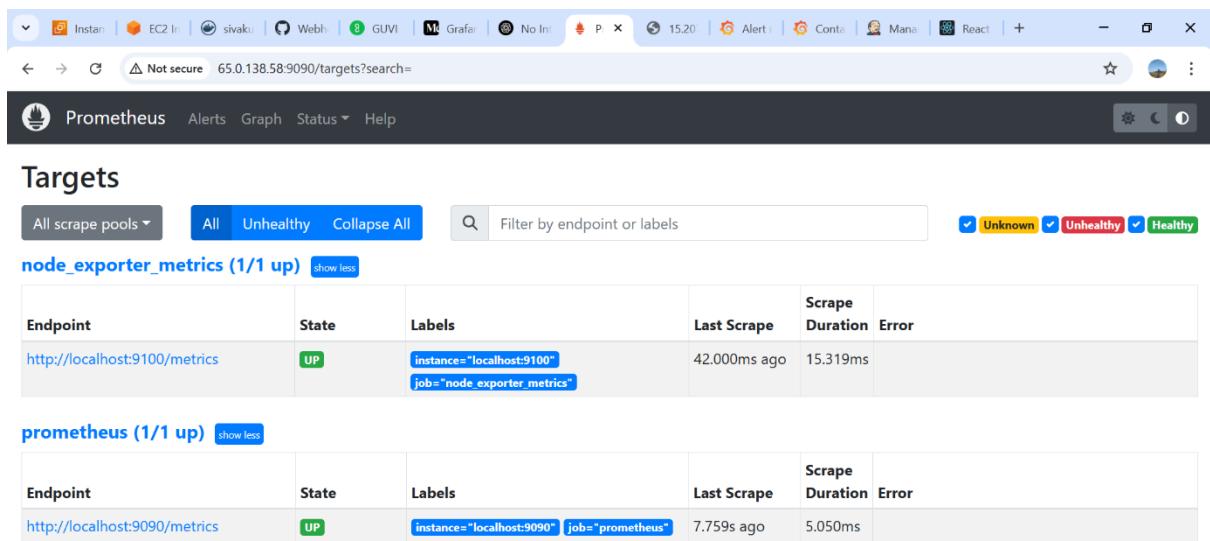


Grafana Dashboard





Prometheus



Jenkins : Docker Credentials

The screenshot shows the Jenkins 'Update credentials' page. The URL is [Manage Jenkins / Credentials / System / Global credentials \(unrestricted\) / sivakumar135/*****](#). The page title is 'Update credentials'. On the left, there are buttons for 'Update', 'Delete', and 'Move'. The 'Scope' dropdown is set to 'Global (Jenkins, nodes, items, all child items, etc)'. The 'Username' field contains 'sivakumar135'. There is an unchecked checkbox 'Treat username as secret'. The 'Password' field is concealed. A 'Change Password' button is next to it. The 'ID' field contains 'dockerhub-credentials'. At the bottom, a 'Save' button is visible. The taskbar at the bottom of the screen shows various application icons and system status.