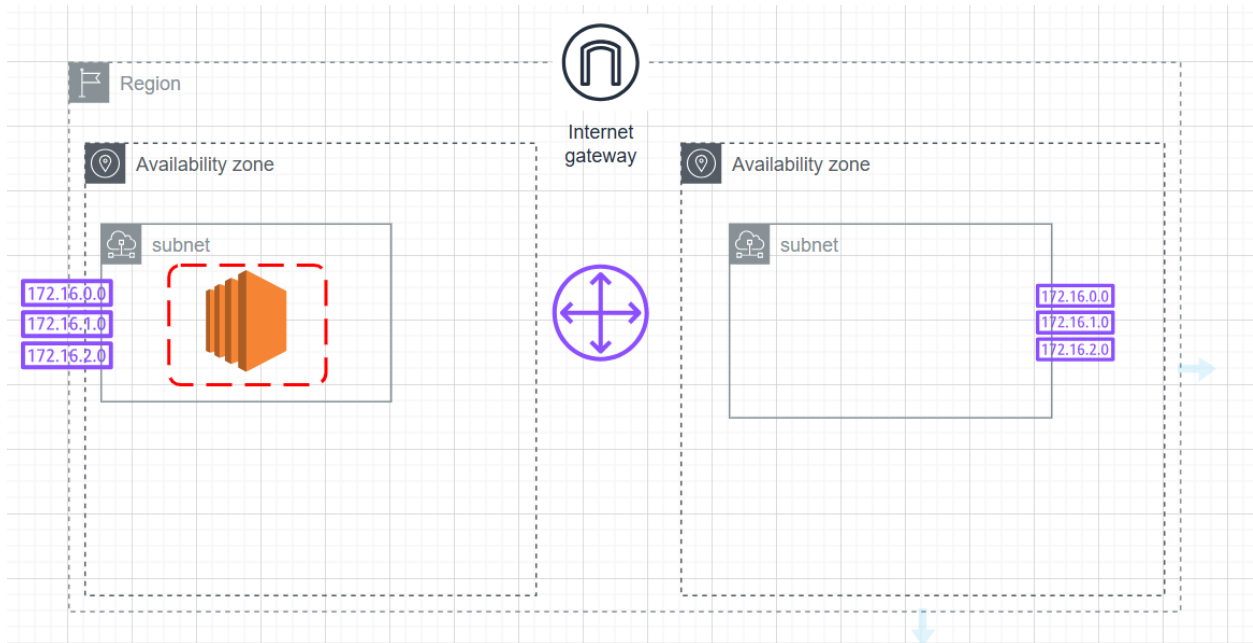


Application images:

Simple architecture but for the best architecture we should use public and private subnet with vpn server and load balancer but this architecture for the task only



Running instance after run terraform code:

The screenshot shows the AWS Management Console 'Instances' page. The table below lists the instances:

<input checked="" type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check
<input checked="" type="checkbox"/>	sparkrock-staging-instance	i-0107ddbc26411ee72	Running	t3.micro	3/3 checks passed

Cloudwatch alarm

aws [Search] [Alt+S] Europe (Frankfurt) Account ID: 1200-1917-7759 Devops_Amr

CloudWatch > Alarms

CloudWatch <

Favorites and recents ▶

Dashboards

▶ AI Operations [New](#)

▼ Alarms 0 1 0

In alarm

[All alarms](#)

Alarms (1)

☐ Hide Auto Scaling alarms

Alarm state: OK Alarm type: Any Actions status: Any < 1 > ⚙️

<input type="checkbox"/>	Name	State	Last state update (UTC)	Conditions
<input type="checkbox"/>	sparkrock-staging-cpu-high	OK	2025-08-24 00:08:51	CPUUtilization > 70 for 2 datapoints within 7 minutes

ECR Registry:

aws [Search] [Alt+S] Europe (Frankfurt) Account ID: 1200-1917-7759 Devops_Amr

Amazon ECR > Private registry > Repositories

Amazon Elastic Container Registry <

▼ Private registry

[Repositories](#)

Features & Settings

▼ Public registry

Repositories

Settings

Private repositories (2)

<input type="radio"/>	Repository name	URI	Created at	Tag immutability	Encryption type
<input type="radio"/>	sparkrock-api	120019177759.dkr.ecr.eu-central-1.amazonaws.com/sparkrock-api	August 24, 2025, 03:06:28 (UTC+03)	Mutable	AES-256
<input type="radio"/>	sparkrock-web	120019177759.dkr.ecr.eu-central-1.amazonaws.com/sparkrock-web	August 24, 2025, 03:06:28 (UTC+03)	Mutable	AES-256

VPC:

aws [Search] [Alt+S] Europe (Frankfurt) Account ID: 1200-1917-7759 Devops_Amr

VPC > Your VPCs

VPC dashboard <

EC2 Global View

▼ Virtual private cloud

[Your VPCs](#)

Subnets

Route tables

Your VPCs (2) [Info](#) Last updated less than a minute ago

<input type="checkbox"/>	Name	VPC ID	State	Block Public...	IPv4 CIDR
<input type="checkbox"/>	sparkrock-staging-vpc	vpc-097cb520d6691ef0c	Available	Off	10.0.0.0/16
<input type="checkbox"/>	-	vpc-0ecbdbef8677ba078	Available	Off	172.31.0.0/16

Terraform code after running:

```

aws_instance.app_instance: Creating...
aws_route_table_association.public_b: Creating...
aws_route_table_association.public_a: Creation complete after 0s [id=rtbassoc-06a499fa4ce2c15ef]
aws_route_table_association.public_b: Creation complete after 0s [id=rtbassoc-0148c830c92e790f1]
aws_instance.app_instance: Still creating... [10s elapsed]
aws_instance.app_instance: Creation complete after 13s [id=i-043ee90cf68dd77d9]
aws_cloudwatch_metric_alarm.cpu_high: Creating...
aws_cloudwatch_metric_alarm.cpu_high: Creation complete after 0s [id=sparkrock-staging-cpu-high]

Apply complete! Resources: 21 added, 0 changed, 0 destroyed.

Outputs:

application_direct_url = "http://3.127.221.235"
application_http_url = "http://3.127.221.235:8080"
application_https_url = "https://3.127.221.235"
ecr_api_repo_url = "120019177759.dkr.ecr.eu-central-1.amazonaws.com/sparkrock-api"
ecr_web_repo_url = "120019177759.dkr.ecr.eu-central-1.amazonaws.com/sparkrock-web"
instance_public_ip = "3.127.221.235"
amribrahim@LAPTOP-3KQROCM8:~/sparkrock-assignment/infrastructure$ |

```

Github action pipeline:

The screenshot shows the GitHub Actions interface for the repository 'amribrahim / Sparkrock-assignment'. The left sidebar contains navigation links: Code, Issues, Pull requests, Actions (selected), Projects, Wiki, Security, Insights, and Settings. Below these are links for 'All workflows', 'Build and Deploy (staging)', and a 'Management' section with links to Caches, Attestations, Runners, Usage metrics, and Performance metrics. The main content area is titled 'All workflows' and shows 'Showing runs from all workflows'. A 'Filter workflow runs' search bar is present. Below this is a 'Help us improve GitHub Actions' banner. The '18 workflow runs' section shows two recent runs for the 'Update deploy.yml' workflow, both successful (green checkmark). The first run is for commit c0825f7, pushed by amribrahim, and the second is for commit b050b2e, also pushed by amribrahim. Both runs are on the 'main' branch and took 54s and 49s respectively.

APP:



Sparkrock Dashboard (Demo Frontend)

This is a tiny static front-end served via Nginx.

Call API

Click the button to call `/api/hello`