Devops

3 tier app project

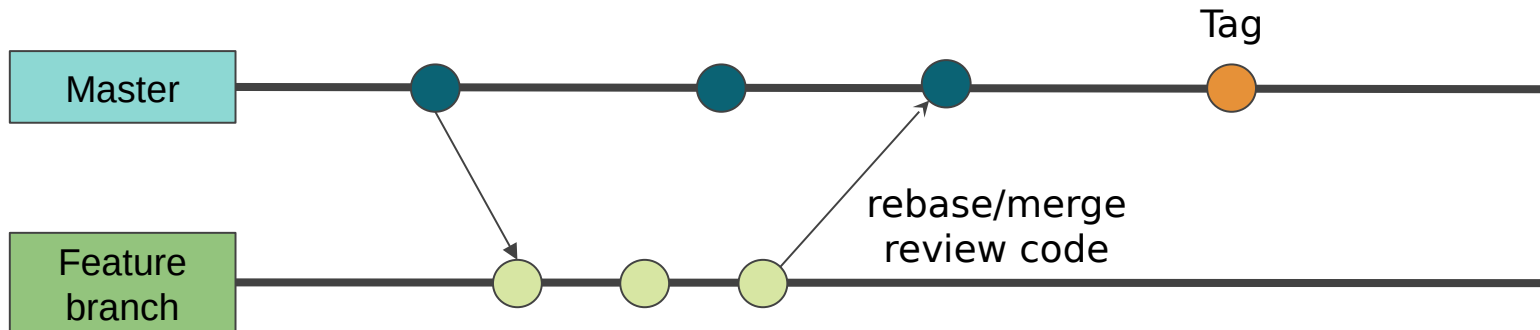# Architecture

:80

VPC

/api/**   default   /grafana/**

AZ-1 Private subnet

API   Web   Grafana

AZ-2 Private subnet

Internal ALB

API   Web   Grafana

AZ-2 Private subnet

API   Web   Grafana

AZ-1 Database subnet   AZ-2 Database subnet   AZ-3 Database subnet

PostgreSQL   PostgreSQL

master   standy

ECR
Registry

API

Cloudwatch

Grafana

WEB

AWS Cloud

content   CDN

GIT Workflow

Tag

Master

Feature
branch

rebase/merge
review code
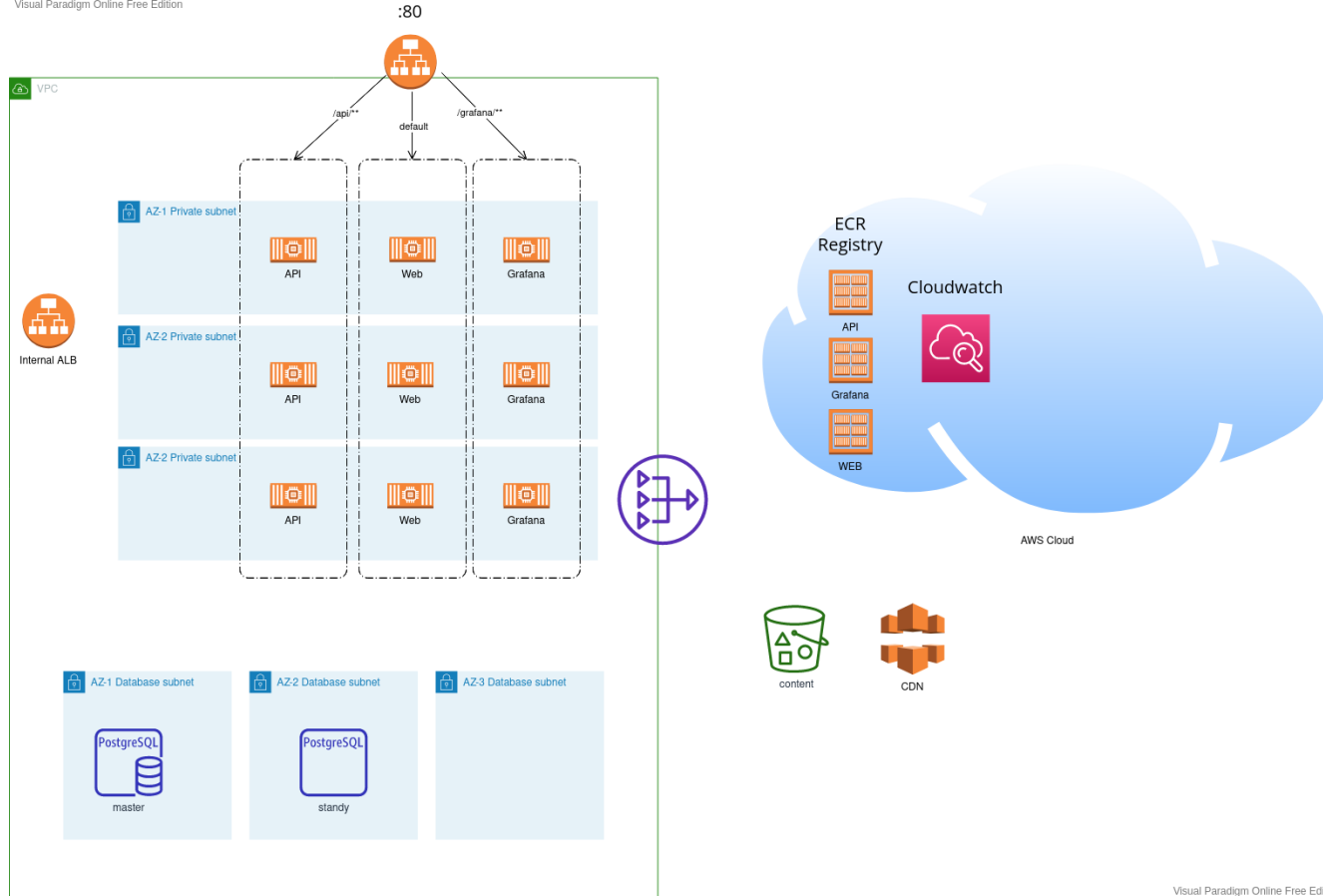
Commits in master are deployed in **DEVELOPMENT**

Tags are deployed in **STAGING**

QA validated tags in staging are deployed to **PRODUCTION**

Next steps: Domain/HTTPS

- Use Custom DNS domain
  - each env own domain
    - prod: *.mydomain.com/mydomain.com
    - stg: *.stg.mydomain.com/stg.mydomain.com
    - dev: *.dev.mydomain.com/dev.mydomain.com
  - specific domain for api component
    - api.dev.mydomain
- Create certificates from a well-known authorities, maybe use Let's encrypt

Next steps: CI/CD

- Split pipeline in different github actions jobs
- Create staging/production environments
- Deploy feature branches of WEB component in custom DNS domain for testing
  - feature-add-dog-gif.dev.mydomain.com
- Add VPC private endpoint to ECR

Next steps: CDN

- Implement a cache expiration strategy. Different options to discuss depending on the use case:
  - Force distribution invalidation
  - Use different filename on each deployment
- Use custom cdn DNS domain, something like cdn.myapp.com using Route 53 alias.

Next steps: Database

- Evaluate and implement Point in time restore.
- RDS AWS private link

Next steps: Monitoring

- Move Grafana to its management VPC
- Specific domain for api component
  - grafana.mydomain
- Only allow certain IPs access to monitoring VPC
- Add alarms to detect problems in real time