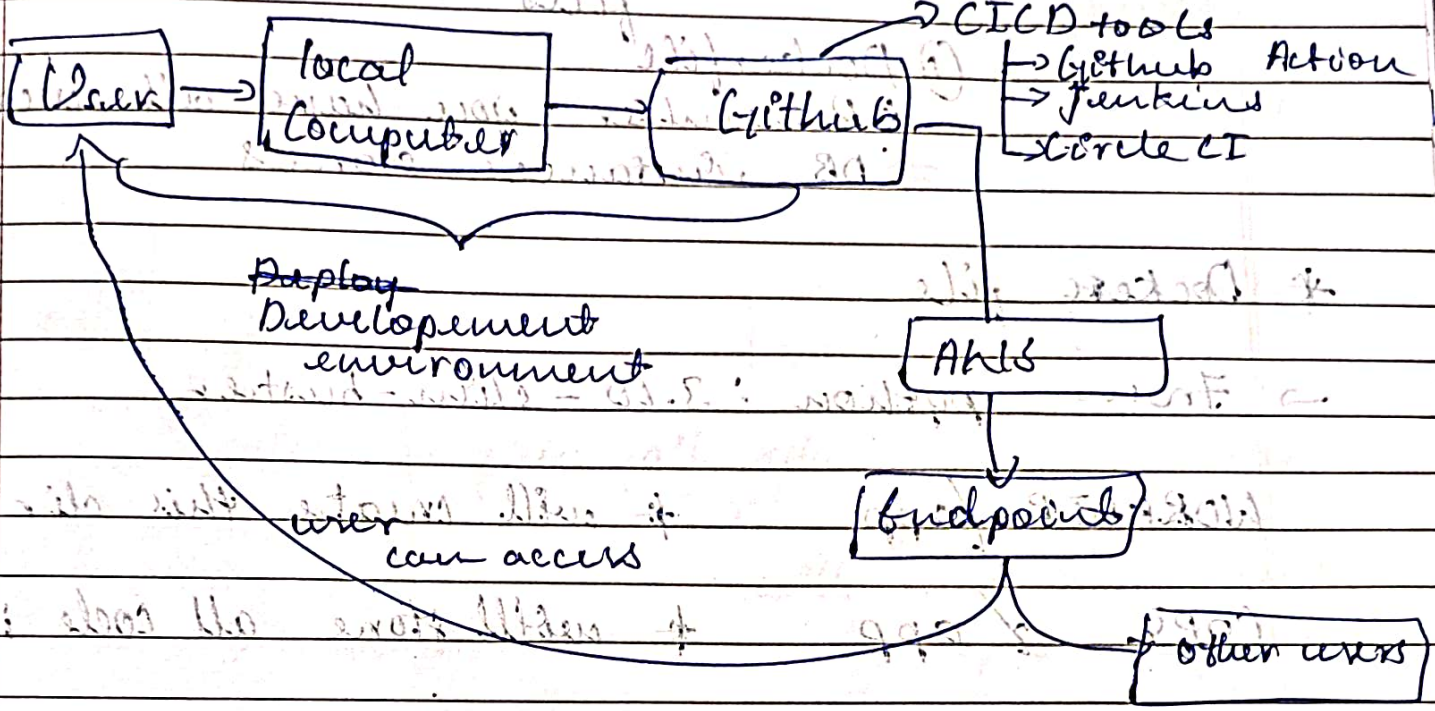
AWS CICD

Continuous integration
Continuous deployment



Services we will use

→ Docker

→ ECR → from AWS
Elastic Container Registery

→ EC2 → AWS

→ Github action – CICD

Lets begin

(1) but first in our code repo we need to create some files

(1) Dockerfile
* Remember you have all vector DB instances store's

* Docker file

→ From python : 3.10 - slim-buster

WORKDIR /app * will create this dir

COPY . /app * will store all code in th

RUN pip install -r requirements.txt
↳ installs all packa

CMD ["python3", "app.py"]
↳ will run our main endpoint app.

(2). Dockerignore

* all files to ignore

③ Create a folder called .github and inside that another folder workflows

and in that folder need to create a file cicd.yaml

this file

will contain all the CICD workflow

will also have route to api's keys, token and these keys are stored in github secrets

② Now AWS part.

① IAM user should be created

set up some policies

① Amazon EC2 ContainerRegistryFullAccess

② Amazon EC2 Full Access

(2) User is created in that go to Security credentials and create access key, select cmd interface and create the access key and keep it.

(3) Create a ECR repo to store/save docker image

(1) Search for ECR
Create repo
Keep name
keep everything default
Copy the URI

(4) Create a EC2 machine
search EC2
Launch instance
select ubuntu machine
keep name of machine
Instance type must be atleast 8gb ram
Create a key pair
In Network setting select all ticks
Configuration storage 30 Gib
Launch instance

We see our machine is running
open the instance
we have to connect it now
will open a new tab
a shows the prodiction server
in cmd interface
execute all these commands one by one

→ sudo apt-get update -y

→ sudo apt-get upgrade

now to setup docker in EC2 machine

→ curl -fsSL https://get.docker.com -o get-docker.sh

→ sudo sh get-docker.sh

→ sudo usermod -aG docker ubuntu

→ newgrp docker

5 Configure EC2 as Self-hosted runner

means we have to connect our Github

means whenever the developer will push the code to the Github ~~the~~ it will automatically update in AWS

* Go to your committed Git repo
  * Go to settings
  * click on option called Actions/Runner
  * Then create a new self-hosted runner
  * Select Linux
  * Execute all the commands from downloads given, in the AWS cmd.
    this will connect EC2 to Github
  * keep the name of Runner: self-hosted
    ↓
    have to keep this only
  * Now the code of Configure must also be pasted in AWS cmd

will show Connected to Github

6) Setup Github Secrets

- AWS_ACCESS_KEY_ID =
- AWS_SECRET_ACCESS_KEY =

} for both of these you have to enter the key you got from a downloaded file

- AWS_DEFAULT_REGION = for this you must set your region eg: if mumbai = ap-south-1
- ECR_REPO = remember we copied a URI
- PINECONE_API_KEY
- OPENAI_API_KEY

* in setting of repo click on secretes and Variables/Actio
  * Click on new repo secret
  * just copy the ID Names and keep the keys

* clone now just push the code in VSCode to git repo

will start running the action

we see everything is getting done

and after execution is done

in the EC2 instance you will get a public IP address copy that and open it wout run as we remember our site is running on localhost/8080

so we have to do port mapping

* Now go to Instance of EC2 go to security option and click on Security Groups → edit inbounding rules

Add rule → custom TCP / 8080 / 0.0.0.0/0

Save rules

→ Now save IP Address of EC2

e.g. 13.232.37.249:8080

↳ add our port no

we see our site is running

now any user can use this now now we can buy a domain and make as a url