

🚀 Guia de Deploy AWS (EC2 + Docker) - Projeto Integrador

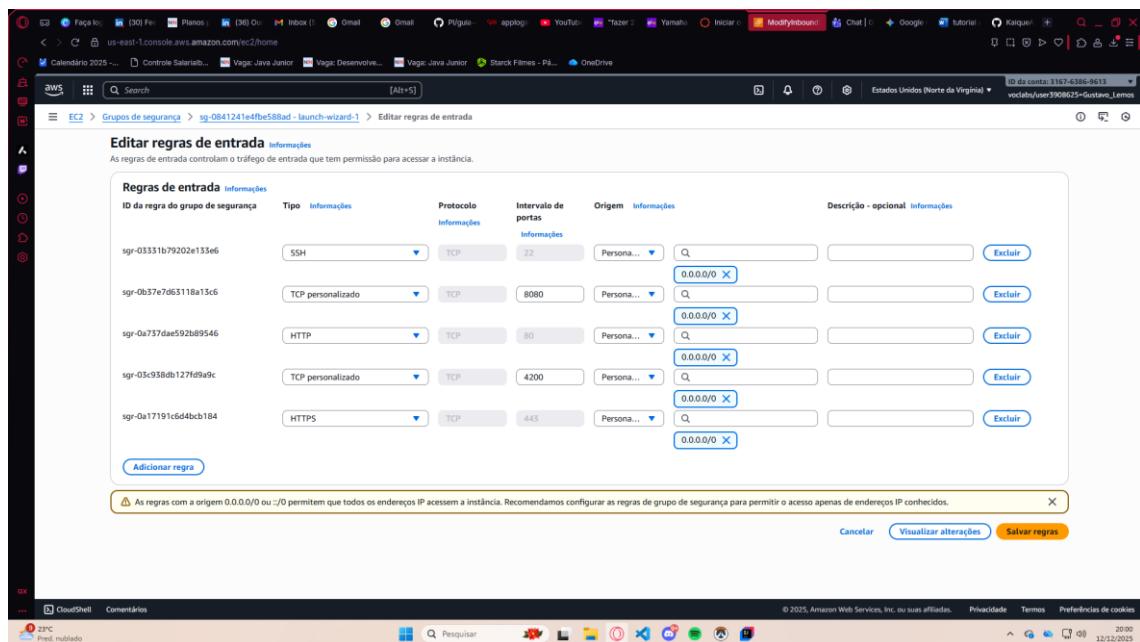
1. Criação e Configuração da Instância (AWS Console)

1. Lançar Instância:

- Acesse o EC2 Dashboard e clique em **Launch Instance**.
- Nome:** Digite um nome (ex: Estoque_SEUNOME).
- OS:** Amazon Linux 2023.
- Tipo:** t2.micro (Free tier).
- Key Pair:** Selecione a chave **vockey**.

2. Configurar Firewall (Security Group):

- Em "Network settings", crie um novo Security Group.
- Adicione as seguintes **Inbound Rules** (Regras de Entrada):
 - SSH (22):** 0.0.0.0/0 (Para acessar o terminal)
 - Custom TCP (8080):** 0.0.0.0/0 (Backend Spring Boot)
 - Custom TCP (4200):** 0.0.0.0/0 (Porta Dev Angular - Opcional)
 - HTTP (80):** 0.0.0.0/0 (Frontend Produção Nginx - **Importante** adicionar esta também)



3. Configurar IP Fixo (Elastic IP):

- No menu lateral esquerdo, vá em **Network & Security -> Elastic IPs**.
- Clique em **Allocate Elastic IP address** -> Clique em **Allocate**.
- Selecione o IP criado, clique em **Actions -> Associate Elastic IP address**.
- Escolha a sua instância criada e clique em **Associate**.
- ⚠ ANOTE ESSE IP!** Você precisará dele no Passo 3.

2. Configuração do Servidor (Terminal)

Conecte-se à instância (via "Connect" > "EC2 Instance Connect").

Passo A: Instalação das Ferramentas

Copie e cole o bloco abaixo inteiro no terminal (Ctrl + Shift + V):

```
Bash
# 1. Atualizar sistema e instalar Git e Docker
sudo dnf update -y
sudo dnf install -y git docker
sudo systemctl start docker
sudo systemctl enable docker

# 2. Adicionar usuário ao grupo docker
sudo usermod -aG docker ec2-user

# 3. Instalar Docker Compose (Versão v2.24.0)
sudo curl -L
"https://github.com/docker/compose/releases/download/v2.24.0/docker-compose-\$\(uname -s\)-\$\(uname -m\)" -o /usr/local/bin/docker-compose
sudo chmod +x /usr/local/bin/docker-compose

# 4. Verificar versões
docker --version
docker-compose --version
```

Passo B: Configurar Memória Swap (CRÍTICO para t2.micro)

Como a máquina gratuita tem pouca memória (1GB), esses comandos evitam que o Java/Angular travem o servidor durante o deploy.

```
Bash
sudo dd if=/dev/zero of=/swapfile bs=128M count=16
sudo chmod 600 /swapfile
sudo mkswap /swapfile
sudo swapon /swapfile
echo '/swapfile swap swap defaults 0 0' | sudo tee -a /etc/fstab
```

Agora digite exit e conecte novamente para aplicar as permissões do Docker.

3. Deploy do Projeto (Ajuste de IP)

1. Clonar a Branch Develop:

- Substitua o link abaixo pelo link do seu repositório.

```
git clone -b develop https://github.com/KaiqueAMesquita/Estoque_LSVP.git
```

```
# Entrar na pasta do projeto  
cd Estoque_LSVP
```

2. Configurar o IP no Frontend (OBRIGATÓRIO):

Você precisa colocar o seu Elastic IP no código do Angular, senão o site não acha o backend.

- Edite o arquivo de produção:

```
nano frontend/Estoque_WebApp/src/environments/environment.prod.ts
```

- Procure a linha API_URL e mude para o seu IP (mantenha o :8080/api):

```
TypeScript  
export const environment = {  
  production: true,  
  // Mude o numero abaixo para o SEU Elastic IP  
  API_URL: "http://SEU_ELASTIC_IP_AQUI:8080/api",  
  TOKEN_KEY: "Centro_Paula_Souza_Token"  
};
```

- Para Salvar: Aperte Ctrl + O e depois Enter.
- Para Sair: Aperte Ctrl + X.

IMPORTANTE: FAÇA O MESMO PRO enviroment.development.ts, ou seja

```
nano frontend/Estoque_WebApp/src/environments/environment.development.ts
```

E mude o ip lá também.

3. Subir a Aplicação:

Agora que o IP está certo, mande o Docker construir tudo:

```
sudo docker-compose up -d --build
```

4. Manutenção e Atualizações

Sempre que você atualizar o código no GitHub e quiser passar para a AWS, use este procedimento de "Limpeza Nuclear" para garantir que o Docker não use versões velhas:

```
Bash  
# 1. Baixar as novidades do Git  
git pull  
  
# 2. Derrubar e limpar containers antigos
```

```
sudo docker-compose down

# 3. Apagar imagens antigas (Força o rebuild total)
sudo docker rmi estoque_lsvp-frontend estoque_lsvp-backend

# 4. Construir e subir sem cache
sudo docker-compose build --no-cache
sudo docker-compose up -d
```

5. Links de Acesso

- **Frontend:** <http://SEU ELASTIC IP>
- **Backend (Swagger):** <http://SEU ELASTIC IP:8080/swagger-ui/index.html>