

Parte 1: Passo a Passo para Criar a Fila SQS (Feito por apenas UMA pessoa do grupo)

O objetivo aqui é criar a "caixa de correio" central que todos os 6 servidores do jogo usarão para se comunicar.

1. Acesse o Serviço SQS:
 - No Console da AWS, na barra de busca no topo, digite SQS e clique no serviço "Simple Queue Service" que aparecer.
2. Inicie a Criação da Fila:
 - Na tela do SQS, clique no botão laranja "Create queue" (Criar fila).
3. Configure a Fila:
 - Type (Tipo): Deixe a opção padrão, "Standard", selecionada. Ela é mais flexível para o nosso caso.
 - Name (Nome): Dê um nome único para a fila do seu grupo. Por exemplo: banco-imobiliario-grupo-alpha.
 - Outras Configurações: Para este projeto, todas as outras configurações podem ser deixadas como padrão. Role a página até o final.
4. Finalize a Criação:
 - Clique no botão laranja "Create queue".

O único item que é compartilhado entre todos é a URL da Fila SQS. É por isso que todos usam o mesmo comando **export**
SQS_QUEUE_URL='...'.

USE ESSE LINK DA FILA, SO UAM PESSOA CRIA A FILA E MANDA O LINK PARA OS DEMAIS

LINK DO REPOSITORIO: <https://github.com/KayqueLopes99/jogo.git>

O nome da chave.pem muda ok?

Lançar uma Nova Instância (Todos)

1. **Inicie o Processo:**
 - Na mesma tela do EC2, clique no grande botão laranja **"Launch instances"** (Lançar instâncias).
2. **Configurações Básicas:**
 - **Name:** Dê um nome novo, como `servidor-jogo-versao-final`.
 - **Application and OS Images (AMI):** Selecione `ubuntu`.
 - **Instance type:** Selecione `t2.micro` (deve estar marcado como "Free tier eligible").
3. **Par de Chaves (O PASSO MAIS IMPORTANTE):**
 - Clique em **"Create new key pair"** (Criar novo par de chaves).
 - **Key pair name:** Dê um nome completamente novo e fácil de identificar, como `chave-final-do-jogo-kayque`.

- **Key pair type:** Deixe em `RSA`.
 - **Private key file format:** Deixe em `.pem`.
 - Clique no botão azul **"Create key pair"**.
 - O seu navegador irá baixar automaticamente o novo arquivo: `chave-final-do-jogo-kayque.pem`. **Salve-o na sua pasta de Downloads. É ESTE arquivo que usaremos.**
4. **Configurações de Rede (Firewall):**
- No painel "Network settings", clique em **"Edit"**.
 - Em "Security group name", dê um nome novo, como `regras-do-jogo-final`.
 - Adicione as **três** regras de entrada ("Inbound security groups rules") que precisamos:
 - **Regra 1 (já existe):**
 - Type: `SSH`
 - Source type: `Anywhere`
 - **Regra 2 (adicionar):** Clique em "Add security group rule"
 - Type: `HTTP`
 - Source type: `Anywhere`
 - **Regra 3 (adicionar):** Clique em "Add security group rule"
 - Type: `Custom TCP`
 - Port range: `5000`
 - Source type: `Anywhere`
5. **Configurações Avançadas (IAM Role):**
- Role a página para baixo e expanda a seção **"Advanced details"**.
 - Procure pelo campo **"IAM instance profile"**.
 - Clique no menu suspenso e selecione o perfil `LabInstanceProfile`. (Este passo é obrigatório no AWS Academy para dar permissão à instância).
6. **Lançar a Instância:**
- Revise o resumo no painel à direita e clique no botão laranja **"Launch instance"**.

Passo 3: Conectar à Nova Instância (A Hora da Verdade)

1. **Aguarde a Instância Ficar Pronta:**
 - Volte para a lista de "Instances". Você verá sua nova instância com o estado "Pending".
 - Aguarde alguns minutos até que o "Instance state" mude para **"Running"** e o "Status check" fique **"3/3 ou 2/2 checks passed"**.
2. **Copie o Novo Endereço:**
 - Selecione a **NOVA** instância na lista.
 - Na aba "Details" abaixo, encontre e copie o **"Public IPv4 DNS"**. Será um endereço novo e diferente do anterior.
3. **Conecte via PowerShell:**
 - Abra o PowerShell no seu computador.

- Navegue até a pasta onde salvou a **nova** chave:
- PowerShell

`cd Downloads`

Substitua o nome da chave e o endereço DNS pelos seus novos valores

```
ssh -i "chave-final-do-jogo-kayque.pem" ubuntu@SEU-NOVO-ENDERECO-DNS.com
```

E, conseqüentemente, o prompt que aparecerá após a conexão será `[ubuntu@ip-...]`

-
- Como esta é a primeira vez que você se conecta a este novo endereço, ele perguntará: `Are you sure you want to continue connecting (yes/no/[fingerprint])?`. Digite `yes` e pressione Enter.
- Se tudo correu bem, seu prompt do PowerShell mudará para `[ec2-user@ip-...]`, confirmando que você está conectado!

Guia de Comandos para o PowerShell

Parte 1: Conectando à sua Instância EC2 (Executado no seu PC)

1. **Abra o PowerShell** no seu computador. **Navegue até a pasta** onde você salvou o seu arquivo de chave `.pem`. Geralmente, é a pasta `Downloads`.
cd Downloads
2. **Use o comando `ssh` para conectar**. Substitua o nome do seu arquivo `.pem` e o seu DNS Público.
PowerShell
Exemplo: `ssh -i "chave-final-do-jogo-kayque.pem" ubuntu@ec2-54-167-67-85.compute-1.amazonaws.com`

```
ssh -i "NOME-DA-SUA-CHAVE.pem" ec2-18-207-151-89.compute-1.amazonaws.com
```

3. Na primeira vez, ele perguntará se você quer continuar. Digite `yes` e pressione Enter. Quando a conexão for bem-sucedida, seu prompt mudará de `PS C:\...>` para `ubuntu@ip-...:~$`.

Parte 2: Configurando e Rodando o Jogo (Executado *dentro* da conexão SSH)

Agora que seu PowerShell está controlando o servidor Ubuntu, execute os seguintes comandos em sequência.

Atualize o servidor e instale as ferramentas necessárias:

```
sudo apt update && sudo apt install git python3-pip python3.12-venv -y
git clone https://github.com/KayqueLopes99/jogo.git
cd jogo/
# Cria a pasta 'venv'
python3 -m venv venv
# Ativa o ambiente (seu prompt vai ganhar um '(venv)' no início)
source venv/bin/activate

pip3 install Flask boto3
export
SQS_QUEUE_URL='https://sqs.us-east-1.amazonaws.com/131407549918/banco-imobiliario-grupo-alpha'
python3 app.py
```

Resultado

Após o último comando, seu terminal ficará "preso", mostrando mensagens do servidor Flask (como ** Running on http://...*). Isso significa que seu jogo está no ar! Mantenha esta janela do PowerShell aberta enquanto você e seu grupo jogam.