

# Etapa do meu projeto

1- Adicionar as imagens para o meu site com o código :

```
scp -i /home/marjorie/Downloads/chaveprojeto.pem /home/marjorie/Downloads/capaside.png.png  
ubuntu@3.92.15.199:/home/ubuntu/imgsite/capaside.png.png
```

## Explicação de cada parte:

1. scp:

scp significa Secure Copy Protocol (Protocolo de Cópia Segura).

Ele é usado para transferir arquivos entre sistemas de maneira segura, utilizando SSH (Secure Shell).

-i /home/marjorie/Downloads/chaveprojeto.pem:

O -i especifica o caminho para a chave privada que você usará para autenticação **SSH**.

Nesse caso, a chave privada é o arquivo chaveprojeto.pem, localizado no diretório /home/marjorie/Downloads/ no seu computador local

/home/marjorie/Downloads/img1.png:

Este é o caminho local do arquivo que você quer transferir.

Aqui, o arquivo img1.png está no diretório /home/marjorie/Downloads/ do seu computador local.

ubuntu@3.92.15.199:/home/ubuntu/imgsite/:

ubuntu: Este é o nome de usuário na sua instância EC2 (a conta de usuário do EC2 que você está utilizando para acessar o servidor remoto).

3.92.15.199: Este é o IP público da sua instância EC2 na AWS, ou seja, o endereço da sua máquina remota.

:/home/ubuntu/imgsite/: O caminho remoto onde o arquivo será copiado. No caso, você está transferindo o arquivo para a pasta /home/ubuntu/imgsite/ dentro da sua instância EC2.

O que o comando faz?

- scp -i /home/marjorie/Downloads/chaveprojeto.pem: Usa a chave privada chaveprojeto.pem para autenticar sua sessão SSH de forma segura.
- /home/marjorie/Downloads/img1.png: Seleciona o arquivo img1.png no seu computador local.

- ubuntu@3.92.15.199:/home/ubuntu/imgsite/: Especifica que você deseja transferir o arquivo para o diretório /home/ubuntu/imgsite/ na instância EC2, onde o usuário é ubuntu e a instância tem o IP público 3.92.15.199.

```
marjorie@nginx01:~$ ls -l /home/marjorie/Downloads/imgsite
total 0
marjorie@nginx01:~$ scp -i /home/marjorie/Downloads/chaveprojeto.pem /home/marjorie/Downloads/capaside.png.png ubuntu@3.92.15.199:/home/ubuntu/imgsite/
capaside.png.png          100% 1386KB 140.5KB/s   00:09
marjorie@nginx01:~$ scp -i /home/marjorie/Downloads/chaveprojeto.pem /home/marjorie/Downloads/logouol.png ubuntu@3.92.15.199:/home/ubuntu/imgsite/
scp: stat local "/home/marjorie/Downloads/logouol.png": No such file or directory
marjorie@nginx01:~$ scp -i /home/marjorie/Downloads/chaveprojeto.pem /home/marjorie/Downloads/logo.png ubuntu@3.92.15.199:/home/ubuntu/imgsite/
logo.png                  100% 29KB 73.4KB/s   00:00
marjorie@nginx01:~$ scp -i /home/marjorie/Downloads/chaveprojeto.pem /home/marjorie/Downloads/img1.jpg ubuntu@3.92.15.199:/home/ubuntu/imgsite/
img1.jpg                  100% 5369KB 105.1KB/s   00:51
marjorie@nginx01:~$ scp -i /home/marjorie/Downloads/chaveprojeto.pem /home/marjorie/Downloads/img2.jpg ubuntu@3.92.15.199:/home/ubuntu/imgsite/
img2.jpg                  100% 563KB 99.0KB/s   00:05
marjorie@nginx01:~$ scp -i /home/marjorie/Downloads/chaveprojeto.pem /home/marjorie/Downloads/img3.png ubuntu@3.92.15.199:/home/ubuntu/imgsite/
img3.png                  100% 325KB 101.5KB/s   00:03
marjorie@nginx01:~$ sudo mv /home/ubuntu/imgsite/logouol.png /var/www/html/projetosite/imagens/
sudo mv /home/ubuntu/imgsite/logo.png /var/www/html/projetosite/imagens/
sudo mv /home/ubuntu/imgsite/img2.jpg /var/www/html/projetosite/imagens/
sudo mv /home/ubuntu/imgsite/img3.png /var/www/html/projetosite/imagens/
sudo mv /home/ubuntu/imgsite/capaside.png.png /var/www/html/projetosite/imagens/
sudo mv /home/ubuntu/imgsite/img1.jpg /var/www/html/projetosite/imagens/
[sudo] senha para marjorie:
```

## Observação:

Tenho que estar no meu usuario para funcionar , EX:

**marjorie@**

### Conectar à EC2:

Primeiro, conecte-se à instância EC2 com o comando SSH:

```
sh -i /home/marjorie/Downloads/chaveprojeto.pem ubuntu@3.92.15.199s
```

### Verificar o conteúdo do diretório imgsite na EC2:

Acesse a máquina EC2 e verifique se as imagens realmente estão no diretório /home/ubuntu/imgsite/:

```
ls -l /home/ubuntu/imgsite/
```

Isso deve listar as imagens que você copiou. Se não houver nada, significa que a cópia não foi realizada corretamente.

### Mover as imagens para o diretório do projeto:

Se as imagens estiverem na pasta correta (/home/ubuntu/imgsite/), mova-as para o diretório do seu projeto:

```
bash
```

```
sudo mv /home/ubuntu/imgsite/*.png /var/www/html/projetosite/imagens/
```

## Verificar as permissões:

Se você estiver com problemas de permissão, verifique se o diretório imagens tem as permissões adequadas para que o Nginx possa acessar as imagens:

```
sudo chown -R www-data:www-data /var/www/html/projetosite/imagens/
```

## Programação

```
marjorie@nginx01:~$ scp -i /home/marjorie/Downloads/chaveprojeto.pem
```

```
/home/marjorie/Downloads/capaside.png.png ubuntu@3.92.15.199:/home/ubuntu/imgsite/capaside.png.png
```

---

```
ssh -i /home/marjorie/Downloads/chaveprojeto.pem ubuntu@3.92.15.199
```

```
ubuntu@ip-10-0-1-77:~$ ls -l /home/ubuntu/imgsite/
```

```
-rw-rw-r-- 1 ubuntu ubuntu 1418887 Mar 25 19:51 capaside.png.png
```

```
-rw-rw-r-- 1 ubuntu ubuntu 5498321 Mar 25 19:57 img1.jpg
```

```
-rw-rw-r-- 1 ubuntu ubuntu 576482 Mar 25 19:57 img2.jpg
```

```
-rw-rw-r-- 1 ubuntu ubuntu 333211 Mar 25 19:57 img3.png
```

```
-rw-rw-r-- 1 ubuntu ubuntu 29278 Mar 25 19:54 logo.png
```

```
sudo mv /home/ubuntu/imgsite/logouol.png /var/www/html/projetosite/imagens/
```

---

# Etapa SITE

## Verificar os arquivos na pasta projetosite:

Para listar os arquivos dentro do diretório **/var/www/html/projetosite**, use o comando:

```
ls -l /var/www/html/projetosite/
```

## Verificar o arquivo site.html:

Caso o site não esteja aparecendo corretamente, você pode revisar o arquivo **site.html** dentro do diretório **/var/www/html/projetosite**. Use o comando:

```
sudo nano /var/www/html/projetosite/site.html
```

## Verificar permissões de arquivos e pastas:

Se o Nginx não conseguir acessar os arquivos, pode ser um problema de permissões. Verifique as permissões da pasta **projetosite** com:

```
buntu@ip-10-0-1-77:~$ ls -l /var/www/html/
total 8
-rw-r--r-- 1 root root 615 Mar 25 18:06 index.nginx-debian.html
-rwxr-xr-x 3 www-data www-data 4096 Mar 25 20:46 projetosite
buntu@ip-10-0-1-77:~$
```

Se necessário, ajuste as permissões para garantir que o usuário **www-data** (que é o usuário padrão do Nginx) tenha acesso adequado:

```
sudo chown -R www-data:www-data /var/www/html/projetosite/
```

```
sudo chmod -R 755 /var/www/html/projetosite/
```

Resumo:

1. Verifique se o Nginx está em execução (`sudo systemctl status nginx`).
2. Liste os arquivos dentro de `/var/www/html/projetosite/` (`ls -l /var/www/html/projetosite/`).
3. Acesse o site no navegador: `http://<IP público da EC2>`.
4. Verifique as permissões dos arquivos e pastas para garantir que o Nginx tenha acesso adequado.

# Etapa Nginx:

## Passo 1: Acessar a Instância EC2 via SSH

1. Abra o terminal do seu computador local e acesse sua instância EC2 usando o SSH com a chave PEM:

```
ssh -i /caminho/para/sua/chave.pem ubuntu@<IP-público-da-instancia-EC2>
```

## Passo 2: Instalar o Nginx (Se ainda não estiver instalado)

Após acessar a instância, instale o Nginx com o seguinte comando:

```
sudo apt update
```

```
sudo apt install nginx -y
```

**Pós a instalação, verifique se o Nginx está rodando:**

```
sudo systemctl status nginx
```

**Se o Nginx não estiver rodando, inicie-o com :**

```
sudo systemctl start nginx
```

## Passo 5: Configurar o Nginx

1. Verifique o arquivo de configuração do Nginx para garantir que ele está apontando para o diretório onde você colocou os arquivos do seu site. O arquivo de configuração padrão do Nginx geralmente está em:

```
sudo nano /etc/nginx/sites-available/default
```

**No arquivo de configuração, altere a linha root para apontar para o diretório correto onde você colocou os arquivos do site, como projetosite**

**Teste a configuração do Nginx** para garantir que não há erros:

```
sudo nginx -t
```

Se não houver erros, você verá algo como:

```
nginx: configuration file /etc/nginx/nginx.conf test is successful
```

**Recarregue o Nginx** para aplicar as mudanças:

```
sudo systemctl reload nginx
```

## Passo 6: Verificar se o Site Está Funcionando

1. Agora, no navegador do seu computador, acesse o IP público da sua instância EC2.  
Exemplo:
2. `http://<IP-público-da-instancia-EC2>`

## Passo 7: Diagnóstico e Resolução de Problemas

- Se o site não aparecer:
  - Verifique se o Nginx está em execução com:

```
sudo systemctl status nginx
```

- Certifique-se de que o diretório `/var/www/html/projetosite/` contém o arquivo `site.html` e outros arquivos necessários.

Verifique se o Nginx está ouvindo na porta 80 (HTTP). Você pode verificar com:

```
sudo netstat -tuln | grep :80
```

Se não aparecer, verifique o firewall da sua instância EC2 para garantir que a porta 80 esteja aberta.

### Resumo do Processo:

1. **Acessar a EC2** via SSH.
2. **Instalar o Nginx**, caso não esteja instalado.
3. **Transferir os arquivos do site** para a instância EC2.
4. **Mover os arquivos do site** para o diretório correto no Nginx.
5. **Configurar o Nginx** para apontar para os arquivos do site.
6. **Recarregar o Nginx** e verificar se o site está sendo servido corretamente.
7. **Testar o site no navegador** acessando o IP público da EC2.

## Como criar o cript de monitoramento que verifica a disponibilidade do site e envia notificações via Discord.

### Criar o Arquivo do Script

No terminal, crie um diretório para armazenar o script (se ainda não tiver):

```
mkdir -p ~/scripts
```

Agora, crie o arquivo do script:

```
nano ~/scripts/monitorar_site
```

Para entrar na pasta onde o script **monitorar\_site.sh** está localizado, basta usar o comando **cd** para navegar até o diretório correto. Com base no seu caminho fornecido, siga este comando:

```
cd /home/ubuntu/monitoramento/
```

Dentro da pasta **monitoramento**, crie ou edite o arquivo **monitorar\_site.sh**:

```
nano monitorar_site.sh
```

## Código no Script

Cole o código do script **monitorar\_site.sh** no editor.

## Tornar o Script Executável

Agora, torne o script executável com o comando:

```
chmod +x monitorar_site.sh
```

## Executar o Script

Para executar o script, use:

```
./monitorar_site.sh
```

Se você quiser rodá-lo em segundo plano para que ele continue monitorando o site mesmo que você saia da sessão SSH, use:

```
nohup ./monitorar_site.sh &
```