

## 00 - Setup e Configuração de ambiente

Você irá utilizar 3 ferramentas para desenvolver os laboratórios:

- . Uma conta da AWS criada por você.
- . [AWS Cloud 9](#). Uma IDE cloud dentro da AWS para todos terem um ambiente igual e com a autorização para executar os comandos dos exercícios.
- . [Postman](#). Uma ferramenta para testar APIs.

### Conta AWS

[!IMPORTANT] O [Free Tier](#) da AWS é uma oferta promocional que permite aos usuários experimentarem gratuitamente certos serviços da Amazon Web Services (AWS) até um limite especificado durante o primeiro ano após a criação da conta. Essa oferta é projetada para ajudar novos clientes a começar a explorar os serviços da AWS sem custo inicial, oferecendo quantidades generosas de recursos em serviços populares como Amazon EC2 (Elastic Compute Cloud), Amazon S3 (Simple Storage Service) e Amazon DynamoDB, entre outros. A necessidade de cadastrar um cartão de crédito ao se inscrever para o Free Tier serve a dois propósitos principais: primeiro, ajuda a AWS a verificar a identidade do usuário e a autenticidade da conta, assegurando que os serviços estão sendo solicitados por um indivíduo real ou uma entidade empresarial legítima; segundo, permite à AWS cobrar automaticamente quaisquer taxas associadas ao uso de serviços que excedam os limites do Free Tier ou ao uso de serviços não incluídos na oferta gratuita. Isso garante que os clientes possam continuar usando os serviços da AWS sem interrupção após ultrapassarem os limites de uso gratuito, ao mesmo tempo em que protege a AWS contra abusos e fraude, mantendo sua infraestrutura segura e disponível para todos os usuários.

## Guia para Criar uma Conta na AWS e Gerenciar Usuários IAM

Este guia fornece instruções passo a passo sobre como criar uma conta na Amazon Web Services (AWS), por que você deve evitar usar a conta root para operações diárias e como criar e usar um usuário IAM com permissões de administrador.

### Criar uma Conta na AWS

- Visitar o Site da AWS  
Acesse o [site oficial da AWS](#) e clique em Criar uma conta da AWS.
- Preencher os Dados de Cadastro  
Insira seu endereço de e-mail, senha, nome da conta da AWS e clique em Continuar.
- Escolher o Tipo de Conta  
Escolha entre uma conta Pessoal ou Empresarial, preencha as informações necessárias e clique em Continuar.

- . Inserir Informações de Contato  
Complete o formulário de contato e prossiga.
- . Adicionar Informações de Pagamento  
Insira os detalhes do seu cartão de crédito. A AWS utiliza essas informações para verificar sua identidade e pode aplicar uma pequena cobrança de validação.
- . Verificar por Telefone  
Forneça um número de telefone e aguarde a ligação ou SMS para verificação.
- . Acessar o Console de Gerenciamento da AWS  
Após concluir a configuração, acesse o console de gerenciamento da AWS.

## Por Que Evitar o Uso do Usuário Root

O usuário root da sua conta AWS possui acesso irrestrito a todos os recursos e serviços. Usá-lo regularmente pode representar um risco significativo de segurança, incluindo:

Risco de Alterações Acidentais: Com acesso total, é fácil fazer alterações não intencionais em recursos críticos.

Exposição a Ataques: Se as credenciais root forem comprometidas, um invasor teria controle total sobre a conta.

## Criar e Usar um Usuário IAM Administrador

- . Acessar o IAM no Console da AWS  
No console da AWS, procure e clique em IAM para abrir o painel do IAM.
- . Criar um Novo Usuário IAM  
Clique em Usuários e depois em Adicionar usuário.  
Escolha um nome de usuário e selecione os tipos de acesso Acesso programático e Acesso ao console de gerenciamento.  
Clique em Próximo: Permissões.
- . Atribuir Permissões de Administrador  
Selecione Anexar políticas existentes de forma direta.  
Procure e selecione AdministratorAccess para conceder permissões de administrador.  
Clique em Próximo: Tags (opcional) e depois em Próximo: Revisar.
- . Revisar e Criar o Usuário  
Revise as configurações e clique em Criar usuário.  
Anote o ID de acesso e a chave de acesso secreta fornecidos.

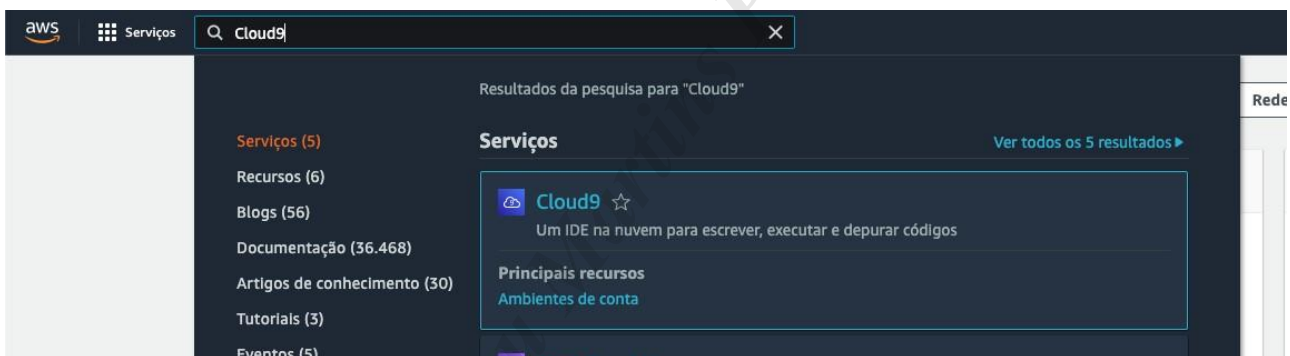
Parabéns! Agora você tem uma conta AWS e entende a importância de evitar o uso do usuário root para tarefas diárias. Com o usuário IAM administrador criado, você pode gerenciar sua conta AWS com maior

segurança e eficiência.

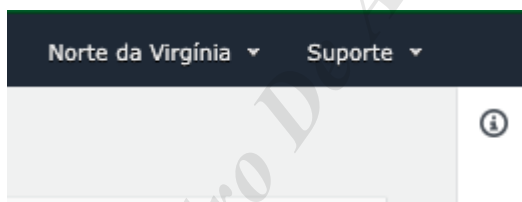
## Cloud 9

O [AWS Cloud 9](#) é um ambiente de desenvolvimento integrado (IDE) baseado na nuvem que fornece aos desenvolvedores uma plataforma rica e flexível para escrever, executar e depurar código diretamente de qualquer navegador, sem a necessidade de instalar ou manter um ambiente de desenvolvimento local. Este serviço suporta várias linguagens de programação e está integrado a outros serviços da AWS, facilitando a construção, teste e depuração de aplicações na nuvem. Além disso, o Cloud9 promove a colaboração em tempo real entre desenvolvedores, permitindo que equipes distribuídas geograficamente trabalhem juntas em projetos com facilidade. Com recursos como um editor de código, depurador e terminal embutidos, o Cloud9 oferece uma solução tudo-em-um para o desenvolvimento de aplicações serverless, web e de software, proporcionando uma experiência de desenvolvimento eficiente e escalável diretamente na nuvem.

. Abra o console da AWS e vá para o serviço **Cloud 9** através da barra de pesquisas.



. Garanta que a região que está utilizando é **us-east-1/ Norte da Virgínia**. Você consegue ver isso no canto superior direito da tela.



. Clique em **Criar ambiente** no lado direito da tela.

. Coloque o nome **lab-usp**.

Criar ambiente [Info](#)

## Detalhes

## Nome

lab-usp

Limite de 60 caracteres, alfanumérico e exclusivo por usuário.

## Descrição - opcional

Limite de 200 caracteres.

Tipo de ambiente [Info](#)

Determina em que o Cloud9 IDE será executado.

☒ Nova instância do EC2

O Cloud9 cria uma instância do EC2 em sua conta. A configuração da sua instância do EC2 não pode ser alterada pelo Cloud9 após a criação.

☐ Computação existente

Você tem uma instância ou servidor existente que gostaria de usar.

- . Deixe as configurações como na imagem a seguir. Se atente ao tipo da máquina que deve ser **t3.medium** e o sistema operacional **Ubuntu Server 22.04 LTS**:

## Nova instância do EC2

Tipo de instância [Info](#)

A memória e a CPU da instância do EC2 que será criada para que o Cloud9 seja executado.

☐ t2.micro (1 GiB RAM + 1 vCPU)

Qualificado para nível gratuito. Ideal para usuários educacionais e exploração.

☐ t3.small (2 GiB RAM + 2 vCPU)

Recomendado para pequenos projetos da web.

☐ m5.large (8 GiB RAM + 2 vCPU)

Recomendado para produção e desenvolvimento de uso geral.

☒ Tipos de instância adicionais

Explore instâncias adicionais para atender às suas necessidades.

## Tipos de instância adicionais

t3.medium

Plataforma [Info](#)

Isso será instalado na sua instância do EC2. Recomendamos o Amazon Linux 2023.

Ubuntu Server 22.04 LTS

## Tempo limite

Por quanto tempo o Cloud9 pode ficar inativo (sem entrada do usuário) antes da hibernação automática. Isso ajuda a evitar cobranças desnecessárias.

30 minutos

- . Em **Configurações de rede** deixe como na imagem a seguir:

Configurações de rede [Info](#)

## Conexão

Como seu ambiente é acessado.

☐ AWS Systems Manager (SSM)

Acessa o ambiente via SSM sem abrir portas de entrada (sem entrada).

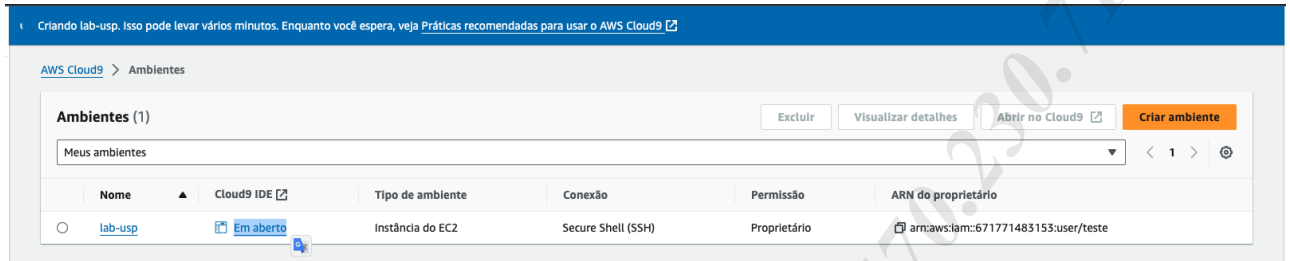
☒ Secure Shell (SSH)

Acessa o ambiente diretamente via SSH, abre portas de entrada.

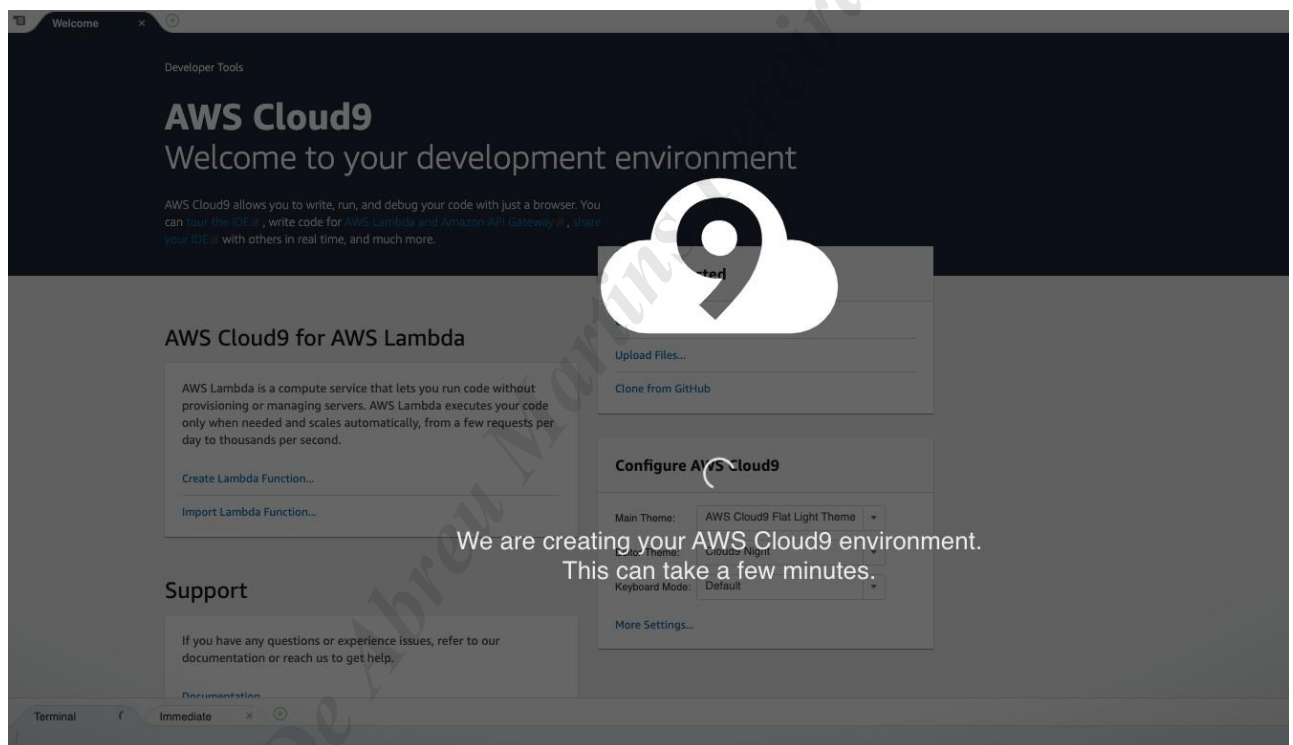
▶ Configurações de VPC [Info](#)

- . Caso todos os passos anteriores estejam corretos, clique em **Criar**.

- . Após a criação clique em **Em aberto**, caso o IDE não tenha aberto automaticamente.

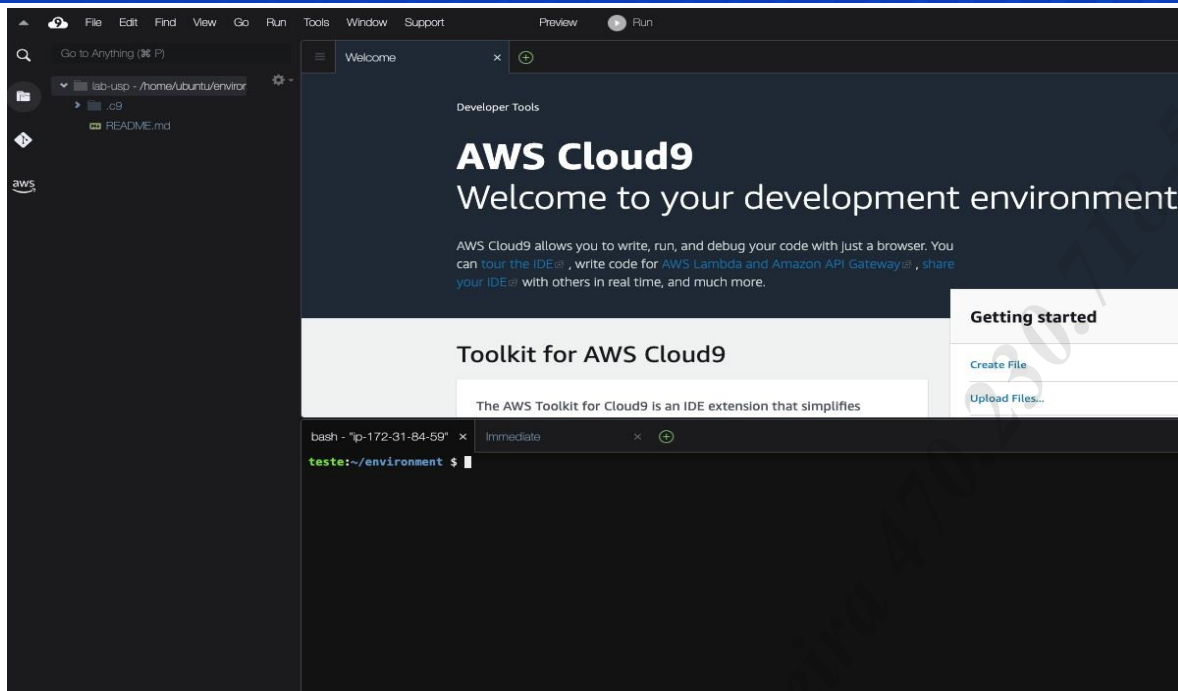


- . A criação do ambiente pode levar alguns minutos.



- . Para os próximos comandos utilize o console bash que fica no canto inferior do seu IDE.





[!IMPORTANT] O [Serverless Framework](#) é uma ferramenta de código aberto projetada para construir e implementar aplicações serverless de maneira eficiente e simplificada, abstraindo a complexidade de gerenciar a infraestrutura subjacente. Ele permite aos desenvolvedores focar no código da aplicação, automatizando o processo de configuração e implementação de serviços serverless, como funções do AWS Lambda, Amazon API Gateway, triggers do AWS Lambda, entre outros, em várias plataformas de nuvem. Utilizando uma simples configuração em arquivos YAML, o Serverless Framework facilita a definição de recursos, gerenciamento de permissões e integrações entre serviços, promovendo um desenvolvimento ágil e uma experiência de implantação contínua. Ao optar pelo Serverless Framework para criar e gerenciar componentes como o Amazon API Gateway e AWS Lambda, você pode acelerar significativamente o ciclo de desenvolvimento de suas aplicações serverless, garantindo escalabilidade, eficiência e redução de custos operacionais, aproveitando ao máximo a infraestrutura serverless da AWS.

. Execute o comando `npm install -g serverless` para instalar o serverless framework.

```
bash - "p-172-31-21-183" x
teste:~/environment $ npm install -g serverless
npm WARN deprecated querystring@2.2.1: The querystring API is considered Legacy. new code should use the URLSearchParams API instead.
npm WARN deprecated querystring@2.2.0: The querystring API is considered Legacy. new code should use the URLSearchParams API instead.
npm WARN deprecated querystring@2.2.0: The querystring API is considered Legacy. new code should use the URLSearchParams API instead.
npm WARN deprecated superagent@7.1.6: Please downgrade to v7.1.5 if you need IE/ActiveXObject support OR upgrade to v8.0.0 as we no longer support IE and published an incorrect patch version (see https://github.com/vision
a/superagent/issues/1731)
added 514 packages in 1m
73 packages are looking for funding
  run 'npm fund' for details
teste:~/environment $
```

- . Execute o comando `sudo apt update -y && sudo apt install jq -y` para instalar o software que irá nos ajudar a ler e manipular Jsons no terminal
- . Execute o comando `npm install -g c9` para baixar a extenãp que ajudará o Cloud9 a lidar melhor com o como abrir arquivos no IDE. Semelhando ao comando code no vscode.
- . Execute o comando `git clone https://github.com/vamperst/usp-serverless-computing.git` para clonar o repositório com os exercícios.
- . Execute o comando `cd usp-serverless-computing/` para entrar na pasta criada pelo git

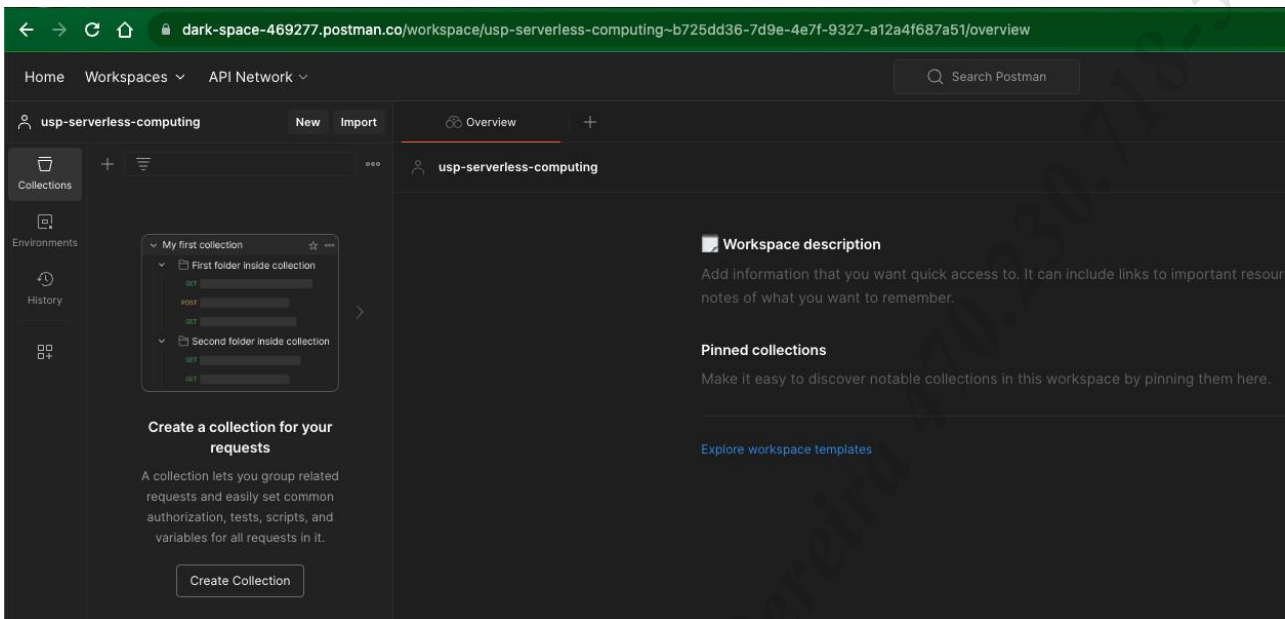
- . Execute o comando `cd 01-criando-o-ambiente-de-desenvolvimento` para entrar na pasta com os scripts de Configuração.
- . Precisamos aumentar o tamanho do volume(HD) do cloud9. Para isso execute o comando `sh resize.sh`

```
teste:~/environment $ sh resize.sh
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
jq is already the newest version (1.6-2.1ubuntu3).
0 upgraded, 0 newly installed, 0 to remove and 10 not upgraded.
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100    19  100    19    0     0  2710      0 --:--:-- --:--:-- --:--:--  3166
{
  "VolumeModification": {
    "VolumeId": "vol-0145f0b2cd0e6a382",
    "ModificationState": "modifying",
    "TargetSize": 50,
    "TargetIops": 3000,
    "TargetVolumeType": "gp3",
    "TargetThroughput": 125,
    "TargetMultiAttachEnabled": false,
    "OriginalSize": 10,
    "OriginalIops": 3000,
    "OriginalVolumeType": "gp3",
    "OriginalThroughput": 125,
    "OriginalMultiAttachEnabled": false,
    "Progress": 0,
    "StartTime": "2024-02-12T14:30:19+00:00"
  }
}
waiting volume...
waiting volume...
NAME            MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
loop0             7:0    0  55.7M  1 loop /snap/core18/2812
loop1             7:1    0  24.9M  1 loop /snap/amazon-ssm-agent/7628
loop2             7:2    0  63.9M  1 loop /snap/core20/2105
loop3             7:3    0  74.1M  1 loop /snap/core22/1033
loop4             7:4    0   60M  1 loop /snap/go/10489
loop5             7:5    0 111.9M  1 loop /snap/lxd/24322
loop6             7:6    0   87M  1 loop /snap/lxd/27037
loop7             7:7    0  40.4M  1 loop /snap/snapd/20671
nvme0n1          259:0    0   50G  0 disk
├─nvme0n1p1      259:1    0   9.9G  0 part /
├─nvme0n1p14     259:2    0    4M  0 part
└─nvme0n1p15     259:3    0  106M  0 part /boot/efi
CHANGED: partition=1 start=227328 old: size=20744159 end=20971487 new: size=104630239 end=104857567
resize2fs 1.46.5 (30-Dec-2021)
Filesystem at /dev/nvme0n1p1 is mounted on /; on-line resizing required
old_desc_blocks = 2, new_desc_blocks = 7
The filesystem on /dev/nvme0n1p1 is now 13078779 (4k) blocks long.

teste:~/environment $ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        49G   6.7G   42G  14% /
tmpfs           463M    0   463M   0% /dev/shm
```

Postman online

- . Acesse o [postman online](#)
- . Faça login com sua conta Postman ou crie uma nova conta se necessário.



. No canto superior esquerdo clique em **Workspaces** e depois em **Create a workspace**

. Crie uma workspace pessoal (personal) com o nome **usp-serverless-computing**.

Workspaces do Postman são ambientes colaborativos que permitem aos usuários organizar, desenvolver, testar e compartilhar APIs dentro de um contexto específico ou projeto. Eles funcionam como espaços de trabalho dedicados onde equipes ou indivíduos podem agrupar coleções de requisições, ambientes, mocks, documentações e monitores de API relacionados a um determinado objetivo ou projeto. Os workspaces facilitam a colaboração em tempo real, permitindo que membros da equipe trabalhem juntos em APIs, vejam atualizações instantâneas e gerenciem o acesso através de configurações de visibilidade e permissões. Disponíveis nas versões pessoal e de equipe, os workspaces do Postman oferecem uma maneira eficiente de segmentar o trabalho de API, promovendo organização, eficiência e sinergia entre os desenvolvedores, independentemente de sua localização geográfica.