



Manual do usuário

# AWS CodeStar



# AWS CodeStar: Manual do usuário

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

As marcas comerciais e imagens comerciais da Amazon não podem ser usadas no contexto de nenhum produto ou serviço que não seja da Amazon, nem de qualquer maneira que possa gerar confusão entre os clientes ou que deprecie ou desprestige a Amazon. Todas as outras marcas comerciais que não são propriedade da Amazon pertencem aos respectivos proprietários, os quais podem ou não ser afiliados, estar conectados ou ser patrocinados pela Amazon.

---

# Table of Contents

.....	viii
O que é o AWS CodeStar? .....	1
O que posso fazer com o AWS CodeStar? .....	1
Como começo a usar o AWS CodeStar? .....	2
Configurar .....	3
Etapa 1: Criar uma conta da .....	3
Inscreva-se para um Conta da AWS .....	3
Criar um usuário com acesso administrativo .....	4
Etapa 2: Criar a função AWS CodeStar de serviço .....	5
Etapa 3: configurar as permissões do IAM do usuário .....	5
Etapa 4: criar um par de chaves do Amazon EC2 para projetos do AWS CodeStar .....	6
Etapa 5: abrir o AWS CodeStar console .....	6
Próximos Passos .....	7
Conceitos básicos do AWS CodeStar .....	8
Etapa 1: Criar um projeto do AWS CodeStar .....	9
Etapa 2: Adicionar informações de exibição a seu perfil de usuário do AWS CodeStar .....	14
Etapa 3: Visualizar o projeto .....	15
Etapa 4: Confirmar uma alteração .....	16
Etapa 5: Adicionar mais membros à equipe .....	21
Etapa 6: Limpeza .....	24
Etapa 7: preparar o projeto para um ambiente de produção .....	24
Próximas etapas .....	25
Tutorial do projeto com tecnologia sem servidor .....	25
Visão geral .....	26
Etapa 1: Criar o projeto .....	27
Etapa 2: Explorar os recursos do projeto .....	28
Etapa 3: Testar o serviço da web .....	31
Etapa 4: Configurar sua estação de trabalho local para editar o código do projeto .....	32
Etapa 5: Adicionar lógica ao serviço da web .....	33
Etapa 6: Testar o serviço da web aprimorado .....	35
Etapa 7: Adicionar um teste de unidade ao serviço da web .....	36
Etapa 8: Visualizar os resultados do teste de unidade .....	39
Etapa 9: Limpeza .....	39
Próximas etapas .....	40

Tutorial do projeto da AWS CLI .....	41
Etapa 1: Fazer download e analisar o código-fonte de exemplo .....	42
Etapa 2: Fazer download do modelo da cadeia de ferramentas de exemplo .....	42
Etapa 3: Testar o modelo da cadeia de ferramentas no AWS CloudFormation .....	44
Etapa 4: Fazer upload do código-fonte e do modelo da cadeia de ferramentas .....	44
Etapa 5: Criar um projeto no AWS CodeStar .....	45
Projeto de habilidade do Alexa .....	48
Pré-requisitos .....	48
Etapa 1: Criar o projeto e conectar sua conta de desenvolvedor da Amazon .....	49
Etapa 2: Testar sua habilidade no Simulador do Alexa .....	51
Etapa 3: Explorar os recursos do projeto .....	51
Etapa 4: Fazer uma alteração na resposta da habilidade .....	51
Etapa 5: Configurar sua estação de trabalho local para se conectar ao repositório do projeto .....	52
Próximas etapas .....	53
Tutorial: Criar um projeto com um repositório de origem do GitHub .....	53
Etapa 1: Criar o projeto e criar seu repositório do GitHub .....	53
Etapa 2: exibir seu código-fonte .....	57
Etapa 3: Criar uma pull request do GitHub .....	58
Modelos de projeto .....	59
Arquivos de projeto e recursos do AWS CodeStar .....	59
Primeiros passos: escolher um modelo de projeto .....	61
Escolha uma plataforma de computação de modelo .....	61
Escolha um tipo de aplicativo de modelo .....	62
Escolha um modelo de linguagem de programação .....	63
Como fazer alterações no projeto do AWS CodeStar .....	63
Alterar o código-fonte do aplicativo e enviar alterações por push .....	64
Alterar recursos do aplicativo com o arquivo template.yml .....	64
.....	65
Melhores práticas do AWS CodeStar .....	66
Melhores práticas de segurança para recursos do AWS CodeStar .....	66
Melhores práticas a fim de configurar versões para dependências .....	66
Monitoramento e registro em log das melhores práticas para recursos do AWS CodeStar .....	67
Como trabalhar com projetos do .....	68
Criar um projeto .....	70
Criar um projeto no AWS CodeStar (console) .....	70

Criar um projeto no AWS CodeStar (AWS CLI) .....	76
Usar um IDE com o AWS CodeStar .....	83
Usar o AWS Cloud9 com a AWS CodeStar .....	84
Usar o Eclipse com o AWS CodeStar .....	92
Use o Visual Studio com o AWS CodeStar .....	97
Alterar recursos do projeto .....	99
Alterações de recursos compatíveis .....	99
Adicionar um estágio ao AWS CodePipeline .....	101
Alterar as configurações de ambiente do AWS Elastic Beanstalk .....	102
Alterar uma função AWS Lambda no código-fonte .....	102
Habilitar rastreamento para um projeto .....	102
Adicionar um recurso a um projeto .....	105
Adicionar uma função do IAM a um projeto .....	111
Adicionar um estágio e uma endpoint de produção a um projeto .....	112
Use com segurança os parâmetros SSM em um projeto AWS CodeStar .....	121
Mudar tráfego para um projeto do AWS Lambda .....	123
Transição do projeto do AWS CodeStar para Produção .....	131
Criar um repositório do GitHub .....	132
Trabalhar com tags de projetos .....	133
Adicionar uma tag a um projeto .....	133
Remover uma tag de um projeto .....	133
Obter uma lista de tags de um projeto .....	134
Excluir um projeto .....	134
Excluir um projeto no AWS CodeStar (console) .....	135
Excluir um projeto no AWS CodeStar (AWS CLI) .....	136
Como trabalhar com equipes do .....	138
Adicionar membros da equipe a um projeto .....	140
Adicionar um membro da equipe (Console) .....	142
Adicionar e exibir membros da equipe (AWS CLI) .....	144
Gerenciar permissões da equipe .....	145
Gerenciar permissões da equipe (console) .....	146
Gerenciar permissões da equipe (AWS CLI) .....	147
Remover membros da equipe de um projeto .....	147
Remover membros da equipe (console) .....	148
Remover membros da equipe (AWS CLI) .....	149
Como trabalhar com seu perfil de usuário do AWS CodeStar .....	150

Gerenciar informações de exibição .....	150
Gerenciar seu perfil de usuário (console) .....	151
Gerenciar perfis de usuário (AWS CLI) .....	152
Adicionar uma chave pública a seu perfil de usuário .....	155
Gerenciar a chave pública (console) .....	155
Gerenciar sua chave pública (AWS CLI) .....	156
Conectar-se à instância do Amazon EC2 com sua chave privada .....	157
Segurança .....	159
Proteção de dados .....	160
Criptografia de dados no AWS CodeStar .....	161
Identity and Access Management .....	161
Público .....	162
Autenticação com identidades .....	162
Gerenciamento do acesso usando políticas .....	166
Como o AWS CodeStar funciona com o IAM .....	168
Políticas e permissões em nível de projeto do AWS CodeStar .....	179
Exemplos de políticas baseadas em identidade .....	186
Solução de problemas .....	218
Log de chamadas de API do AWS CodeStar com o AWS CloudTrail .....	220
Informações do AWS CodeStar no CloudTrail .....	220
Noções básicas das entradas dos arquivos de log do AWS CodeStar .....	221
Validação de compatibilidade .....	223
Resiliência .....	223
Segurança da infraestrutura .....	223
Limites .....	225
Solução de problemas AWS CodeStar .....	227
Falha ao criar o projeto: um projeto não foi criado .....	227
Criação do projeto: vejo um erro quando tento editar a configuração do Amazon EC2 ao criar um projeto .....	228
Exclusão do projeto: um AWS CodeStar projeto foi excluído, mas os recursos ainda existem ..	229
Falha no gerenciamento da equipe: um usuário do IAM não pôde ser adicionado a uma equipe em um AWS CodeStar projeto .....	230
Falha de acesso: um usuário federado não pode acessar um projeto AWS CodeStar .....	231
Falha de acesso: um usuário federado não pode acessar ou criar um ambiente AWS Cloud9 ..	231
Falha de acesso: um usuário federado pode criar um AWS CodeStar projeto, mas não pode visualizar os recursos do projeto .....	231

---

Problema de função de serviço: a função de serviço não pôde ser criada .....	232
Problema de função de serviço: a função de serviço não é válida ou está ausente .....	232
Problema de função do projeto: as verificações do status de AWS Elastic Beanstalk saúde falham em instâncias em um AWS CodeStar projeto .....	232
Problema de função de projeto: uma função de projeto não é válida ou está ausente .....	233
Extensões de projetos: não é possível se conectar ao JIRA .....	234
GitHub: Não consigo acessar o histórico de commits, problemas ou código de um repositório .	234
AWS CloudFormation: criação da pilha revertida para permissões ausentes .....	235
AWS CloudFormation não está autorizado a realizar iam: PassRole na função de execução do Lambda .....	235
Não é possível criar a conexão para um GitHub repositório .....	236
Notas de versão .....	237
Glossário do AWS .....	243

Em 31 de julho de 2024, a Amazon Web Services (AWS) interromperá o suporte para criação e visualização AWS CodeStar de projetos. Depois de 31 de julho de 2024, você não poderá mais acessar o AWS CodeStar console nem criar novos projetos. No entanto, os AWS recursos criados por AWS CodeStar, incluindo seus repositórios de origem, pipelines e compilações, não serão afetados por essa alteração e continuarão funcionando. AWS CodeStar As conexões e AWS CodeStar notificações não serão afetadas por essa descontinuação.

Se você deseja monitorar o trabalho, desenvolver código e criar, testar e implantar seus aplicativos, a Amazon CodeCatalyst fornece um processo de introdução simplificado e funcionalidades adicionais para gerenciar seus projetos de software. Saiba mais sobre a [funcionalidade](#) e [os preços](#) da Amazon CodeCatalyst.

As traduções são geradas por tradução automática. Em caso de conflito entre o conteúdo da tradução e da versão original em inglês, a versão em inglês prevalecerá.



# O que é o AWS CodeStar?

O AWS CodeStar é um serviço baseado em nuvem para criar, gerenciar e trabalhar com projetos de desenvolvimento de software na AWS. Você pode desenvolver, projetar e implantar aplicativos na AWS rapidamente com um projeto do AWS CodeStar. Um projeto do AWS CodeStar cria e integra serviços da AWS para a cadeia de ferramentas de desenvolvimento do seu projeto. Dependendo da escolha de modelo de projeto do AWS CodeStar, essa cadeia de ferramentas pode incluir controle de código-fonte, compilação, implantação, servidores virtuais ou recursos sem servidor e muito mais. O AWS CodeStar também gerencia as permissões necessárias para os usuários do projeto (chamados de membros da equipe do projeto). Adicionando usuários como membros da equipe a um projeto do AWS CodeStar, os proprietários de projetos podem conceder a cada membro da equipe o acesso adequado à função para um projeto e seus recursos de maneira rápida e simples.

## Tópicos

- [O que posso fazer com o AWS CodeStar?](#)
- [Como começo a usar o AWS CodeStar?](#)

# O que posso fazer com o AWS CodeStar?

Você pode usar o AWS CodeStar para ajudar com o desenvolvimento de seu aplicativo na nuvem e gerenciar o desenvolvimento em um único painel de controle centralizado. Especificamente, é possível:

- Iniciar novos projetos de software na AWS em minutos usando modelos para aplicativos web, serviços web e muito mais: o AWS CodeStar inclui modelos de projeto para vários tipos de projeto e linguagens de programação. Como o AWS CodeStar cuida da configuração, todos os recursos do projeto são configurados para trabalhar em conjunto.
- Gerenciar o acesso ao projeto para a equipe: o AWS CodeStar fornece um console central onde você pode atribuir membros da equipe do projeto às funções necessárias para acessar ferramentas e recursos. Essas permissões são aplicadas automaticamente em todos os serviços do AWS usados no projeto, de modo que você não precisa criar ou gerenciar políticas complexas do IAM.
- Visualizar, operar e colaborar em seus projetos em um só lugar: o AWS CodeStar inclui um painel de projeto que fornece uma visão geral do projeto, sua cadeia de ferramentas e eventos importantes. Você pode monitorar as atividades mais recentes do projeto, como

confirmações recentes de código e acompanhar o status das alterações no código, criar resultados e implantações, tudo a partir da mesma página da web. Você pode monitorar o que está acontecendo no projeto a partir de um único painel e analisar os problemas a serem investigados.

- Iterar rapidamente com todas as ferramentas necessárias: o AWS CodeStar inclui uma cadeia de desenvolvimento integrada para o seu projeto. Os membros da equipe enviam código por push, e as alterações são implantadas automaticamente. A integração com o acompanhamento de problemas permite que os membros da equipe controlem o que precisa ser feito em seguida. Você e sua equipe podem trabalhar em conjunto de forma mais rápida e eficiente em todas as fases de entrega de código.

## Como começo a usar o AWS CodeStar?

Para começar a usar o AWS CodeStar:

1. Prepare-se para usar o AWS CodeStar seguindo as etapas em [Configuração AWS CodeStar](#).
2. Teste com AWS CodeStar seguindo as etapas no tutorial do [Conceitos básicos do AWS CodeStar](#).
3. Compartilhe seu projeto com outros desenvolvedores seguindo as etapas em [Adicionar membros da equipe a um projeto do AWS CodeStar](#).
4. Integre seu IDE favorito seguindo as etapas em [Usar um IDE com o AWS CodeStar](#).

# Configuração AWS CodeStar

Antes de começar a usar AWS CodeStar, você deve concluir as etapas a seguir.

## Tópicos

- [Etapa 1: Criar uma conta da](#)
- [Etapa 2: Criar a função AWS CodeStar de serviço](#)
- [Etapa 3: configurar as permissões do IAM do usuário](#)
- [Etapa 4: criar um par de chaves do Amazon EC2 para projetos do AWS CodeStar](#)
- [Etapa 5: abrir o AWS CodeStar console](#)
- [Próximos Passos](#)

## Etapa 1: Criar uma conta da

### Inscreva-se para um Conta da AWS

Se você não tiver um Conta da AWS, conclua as etapas a seguir para criar um.

Para se inscrever em um Conta da AWS

1. Abra <https://portal.aws.amazon.com/billing/signup>.
2. Siga as instruções on-line.

Parte do procedimento de inscrição envolve receber uma chamada telefônica e digitar um código de verificação no teclado do telefone.

Quando você se inscreve em um Conta da AWS, um Usuário raiz da conta da AWS é criado. O usuário-raiz tem acesso a todos os Serviços da AWS e recursos na conta. Como prática recomendada de segurança, atribua o acesso administrativo a um usuário e use somente o usuário-raiz para executar [tarefas que exigem acesso de usuário-raiz](#).

AWS envia um e-mail de confirmação após a conclusão do processo de inscrição. A qualquer momento, é possível visualizar as atividades da conta atual e gerenciar sua conta acessando <https://aws.amazon.com/> e selecionando Minha conta.

## Criar um usuário com acesso administrativo

Depois de se inscrever em um Conta da AWS, proteja seu Usuário raiz da conta da AWS AWS IAM Identity Center, habilite e crie um usuário administrativo para que você não use o usuário root nas tarefas diárias.

### Proteja seu Usuário raiz da conta da AWS

1. Faça login [AWS Management Console](#) como proprietário da conta escolhendo Usuário raiz e inserindo seu endereço de Conta da AWS e-mail. Na próxima página, digite sua senha.

Para obter ajuda ao fazer login usando o usuário raiz, consulte [Signing in as the root user](#) (Fazer login como usuário raiz) no Guia do usuário Início de Sessão da AWS .

2. Ative a autenticação multifator (MFA) para seu usuário raiz.

Para obter instruções, consulte [Habilitar um dispositivo de MFA virtual para seu usuário Conta da AWS raiz \(console\) no Guia](#) do usuário do IAM.

### Criar um usuário com acesso administrativo

1. Habilitar o IAM Identity Center.

Para obter instruções, consulte [Habilitar AWS IAM Identity Center](#) no Guia do usuário do AWS IAM Identity Center .

2. No Centro de Identidade do IAM, conceda o acesso administrativo para um usuário.

Para ver um tutorial sobre como usar o Diretório do Centro de Identidade do IAM como fonte de identidade, consulte [Configurar o acesso do usuário com o padrão Diretório do Centro de Identidade do IAM](#) no Guia AWS IAM Identity Center do usuário.

### Iniciar sessão como o usuário com acesso administrativo

- Para fazer login com seu usuário IAM Identity Center, use a URL de login enviada ao seu endereço de e-mail quando você criou o usuário IAM Identity Center user.

Para obter ajuda para fazer login usando um usuário do IAM Identity Center, consulte [Como fazer login no portal de AWS acesso](#) no Guia Início de Sessão da AWS do usuário.

## Atribuir acesso a usuários adicionais

1. No Centro de Identidade do IAM, crie um conjunto de permissões que siga as práticas recomendadas de aplicação de permissões com privilégio mínimo.

Para obter instruções, consulte [Create a permission set](#) no Guia do usuário do AWS IAM Identity Center .

2. Atribua usuários a um grupo e, em seguida, atribua o acesso de autenticação única ao grupo.

Para obter instruções, consulte [Add groups](#) no Guia do usuário do AWS IAM Identity Center .

## Etapa 2: Criar a função AWS CodeStar de serviço

Crie uma [função de serviço](#) que seja usada para dar AWS CodeStar permissão para administrar AWS recursos e permissões do IAM em seu nome. Você precisa criar a função de serviço apenas uma vez.

### Important

Você deve estar conectado como um usuário administrativo do (ou conta raiz) para criar uma função de serviço. Para obter informações, consulte [Criando seu primeiro grupo e usuário do IAM](#).

1. Abra o AWS CodeStar console em <https://console.aws.amazon.com/codestar/>.
2. Escolha Iniciar projeto.

Se você não vir Start project (Iniciar projeto) e for direcionado para a página da lista de projetos, a função de serviço já terá sido criada.

3. Em Create service role (Criar função de serviço), escolha Yes, create role (Sim, criar função).
4. Saia do assistente. Você vai retornar a ela mais tarde.

## Etapa 3: configurar as permissões do IAM do usuário

Além do usuário administrativo, você pode usar AWS CodeStar como usuário do IAM, usuário federado, usuário raiz ou função assumida. Para obter informações sobre o que AWS CodeStar pode fazer para usuários do IAM versus usuários federados, consulte [Perfis do IAM do AWS CodeStar](#).

Se você não tiver configurado usuários do IAM, consulte [Usuário do IAM](#).

Para conceder acesso, adicione as permissões aos seus usuários, grupos ou perfis:

- Usuários e grupos em AWS IAM Identity Center:

Crie um conjunto de permissões. Siga as instruções em [Criação de um conjunto de permissões](#) no Guia do usuário do AWS IAM Identity Center .

- Usuários gerenciados no IAM com provedor de identidades:

Crie um perfil para a federação de identidades. Siga as instruções em [Criar um perfil para um provedor de identidades de terceiros \(federação\)](#) no Guia do usuário do IAM.

- Usuários do IAM:

- Crie um perfil que seu usuário possa assumir. Siga as instruções em [Criação de um perfil para um usuário do IAM](#) no Guia do usuário do IAM.
- (Não recomendado) Vincule uma política diretamente a um usuário ou adicione um usuário a um grupo de usuários. Siga as instruções em [Adição de permissões a um usuário \(console\)](#) no Guia do usuário do IAM.

## Etapa 4: criar um par de chaves do Amazon EC2 para projetos do AWS CodeStar

Muitos AWS CodeStar projetos usam AWS CodeDeploy ou AWS Elastic Beanstalk implantam código em instâncias do Amazon EC2. Para acessar as instâncias do Amazon EC2 associadas ao projeto, crie um par de chaves do Amazon EC2 para o usuário do IAM. O usuário do IAM deve ter permissões para criar e gerenciar chaves do Amazon EC2 (por exemplo, a permissão para executar as ações `ec2:CreateKeyPair` e `ec2:ImportKeyPair`). Para obter mais informações, consulte [Pares de chaves do Amazon EC2](#).

## Etapa 5: abrir o AWS CodeStar console

Faça login no AWS Management Console e abra o AWS CodeStar console em <https://console.aws.amazon.com/codestar/>.

## Próximos Passos

Parabéns, você concluiu a configuração! Para começar a trabalhar com AWS CodeStar, consulte [Conceitos básicos do AWS CodeStar](#).

# Conceitos básicos do AWS CodeStar

Neste tutorial, você usa o AWS CodeStar para criar um aplicativo web. Este projeto inclui código de exemplo em um repositório de origem, uma cadeia de ferramentas de implantação e um painel do projeto onde você pode visualizar e monitorar o projeto.

Seguindo as etapas, você:

- Cria um projeto no AWS CodeStar.
- Explorar o projeto.
- Confirmar uma alteração no código.
- Ver a alteração no código implantada automaticamente.
- Adicionar outras pessoas para trabalhar no projeto.
- Limpar recursos do projeto quando não forem mais necessários.

## Note

Se você ainda não tiver feito isso, primeiro conclua as etapas em [Configuração AWS CodeStar](#), inclusive [Etapa 2: Criar a função AWS CodeStar de serviço](#). Você deve entrar usando uma conta que seja de um usuário administrativo no IAM. Para criar um projeto, entre no AWS Management Console usando um usuário do IAM que tenha a política de **AWSCodeStarFullAccess**.

## Tópicos

- [Etapa 1: Criar um projeto do AWS CodeStar](#)
- [Etapa 2: Adicionar informações de exibição a seu perfil de usuário do AWS CodeStar](#)
- [Etapa 3: Visualizar o projeto](#)
- [Etapa 4: Confirmar uma alteração](#)
- [Etapa 5: Adicionar mais membros à equipe](#)
- [Etapa 6: Limpeza](#)
- [Etapa 7: preparar o projeto para um ambiente de produção](#)
- [Próximas etapas](#)
- [Tutorial: Criação e gerenciamento de um projeto com tecnologia sem servidor no AWS CodeStar](#)



- [Tutorial: Criar um projeto no AWS CodeStar com a AWS CLI](#)
- [Tutorial: Criar um projeto de habilidade do Alexa no AWS CodeStar](#)
- [Tutorial: Criar um projeto com um repositório de origem do GitHub](#)

## Etapa 1: Criar um projeto do AWS CodeStar

Nesta etapa, você criará um projeto de desenvolvimento de software JavaScript (Node.js) para um aplicativo web. Você usará um modelo de projeto do AWS CodeStar para criar o projeto.

### Note

O modelo de projeto do AWS CodeStar usado neste tutorial utiliza as seguintes opções:

- Categoria do aplicativo: aplicativo Web
- Linguagem de programação: Node.js
- Serviço do AWS: Amazon EC2

Se você escolher outras opções, sua experiência poderá não corresponder ao que é documentado neste tutorial.

Para criar um projeto no AWS CodeStar

1. Faça login no AWS Management Console e então abra o console do AWS CodeStar em <https://console.aws.amazon.com/codestar/>.

Verifique se você está conectado à região da AWS onde você deseja criar o projeto e seus recursos. Por exemplo, para criar um projeto em no Leste dos EUA (Ohio), verifique se selecionou essa região do AWS. Para obter informações sobre as regiões da AWS onde o AWS CodeStar está disponível, consulte [Regiões e endpoints](#) na Referência geral da AWS.


2. Na página AWS CodeStar, escolha Criar um projeto.
3. Na página Escolher um modelo de projeto, escolha o tipo de projeto na lista de modelos de projeto do AWS CodeStar. Você pode usar a barra de filtro estreitar suas opções. Por exemplo, para um projeto de aplicativo web escrito em Node.js que será implantado em instâncias do Amazon EC2, selecione as caixas Aplicativo de web, Node.js, e Amazon EC2. Em seguida, escolha entre os modelos disponíveis para esse conjunto de opções.

Para obter mais informações, consulte [Modelos de projetos do AWS CodeStar](#).

- Escolha Next (Próximo).
- No campo de entrada de texto Nome do projeto, insira um nome para o projeto, como *Meu primeiro projeto*. Em ID do projeto, a ID para o projeto é derivada desse nome do projeto, mas é limitada a 15 caracteres.

Por exemplo, a ID padrão de um projeto chamado *Meu primeiro projeto* é *my-first-projec*. O ID do projeto é a base dos nomes de todos os recursos associados ao projeto. O AWS CodeStar usa esse ID do projeto como parte do URL para o repositório de códigos, bem como para os nomes de funções de acesso e políticas de segurança relacionados no IAM. Depois que o projeto for criado, o ID do projeto não pode ser alterado. Para editar a ID do projeto antes de criar o projeto, em ID do projeto, insira a ID que você deseja usar.


Para obter informações sobre os limites em nomes e IDs de projetos, consulte [Limites no AWS CodeStar](#).

 Note

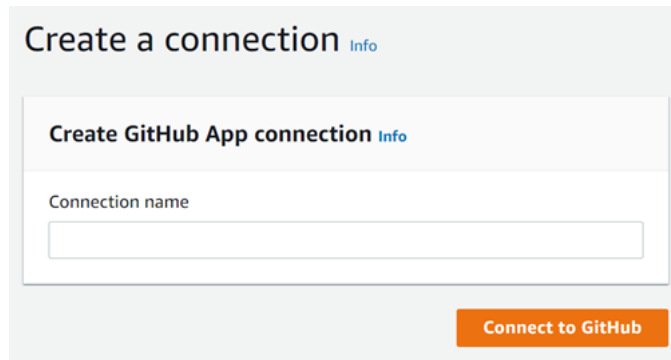
Os IDs dos projetos devem ser exclusivos para a conta da AWS em uma região da AWS.

- Escolha o provedor do repositório, AWS CodeCommit ou GitHub.
- Se você tiver escolhido AWS CodeCommit, para Repository name (Nome do repositório), aceite o nome do repositório do AWS CodeCommit padrão ou digite um outro. Em seguida, avance para a etapa 9.
- Se você escolheu o GitHub, precisará escolher ou criar um recurso de conexão. Se você tiver uma conexão existente, escolha-a no campo de pesquisa. Caso contrário, crie uma nova conexão agora. Escolha Conectar-se ao GitHub.

A página Criar uma conexão é exibida.

 Note

Para criar uma conexão, você deve ter uma conta do GitHub. Se você estiver criando uma conexão para uma organização, você deve ser o proprietário da organização.



- a. Em Criar conexão do App GitHub, no campo de entrada de texto Nome da conexão, insira um nome para sua conexão. Escolha Conectar-se ao GitHub.

A página de Conectar ao GitHub exibe e mostra o campo Apps do GitHub

- b. Em GitHub apps (Aplicações do GitHub), escolha uma instalação de aplicação ou Install a new app (Instalar uma nova aplicação) para criar uma.

**Note**

Você instala uma aplicação para todas as suas conexões com um provedor específico. Se você já instalou a aplicação AWS Connector para GitHub, selecione-a e ignore esta etapa.

- c. Na página de instalação do Connect para GitHub, selecione a conta na qual deseja instalar a aplicação.

**Note**

Se você instalou a aplicação anteriormente, poderá escolher Configure (Configurar) para prosseguir para uma página de modificação para a instalação da aplicação ou usar o botão Back (Voltar) para retornar ao console.

- d. Se a página Confirme a senha para continuar for exibida, digite sua senha do GitHub e escolha Entrar.
- e. Na página Instalar o AWS Connector para GitHub, mantenha os valores padrão e selecione Instalar.

- f. Na página Conexão ao GitHub o ID de conexão da sua nova instalação é exibido em Aplicações do GitHub.

Depois que a conexão for criada, na página de criação do projeto do CodeStar, a mensagem Pronto para conectar é exibida.

**Note**

É possível visualizar sua conexão em Configurações no console do Developer Tools. Para obter mais informações, consulte o tópico de [Conceitos básicos sobre conexões](#).

Select a repository provider

CodeCommit  
Use a new AWS CodeCommit repository for your project.

GitHub  
Use a new GitHub source repository for your project (requires an existing GitHub account).

**The GitHub repository provider now uses CodeStar Connections**  
To use a GitHub repository in CodeStar, create a connection. The connection will use GitHub Apps to access your repository. Use the following options to choose an existing connection or create a new one. [Learn more](#)

Connection  
Choose an existing connection or create a new one and then return to this task.

am:aws:codestar-connections:us-east-1 or [Connect to GitHub](#)

**Ready to connect**  
Your Github connection is ready for use.


Repository owner  
The owner of the new repository. This can be a personal GitHub account or a GitHub organization.

Repository name  
The name of the new repository.

Repository description  
An optional description of the new repository.


Public

- g. Em Proprietário do repositório, escolha a organização do GitHub ou a conta do GitHub pessoal.
- h. Para Repository name (Nome do repositório), aceite o nome do repositório do GitHub padrão ou digite um outro.
- i. Escolha Público ou Privado.

 Note


Para usar o AWS Cloud9 como ambiente de desenvolvimento, será necessário selecionar um repositório público.

- j. (Opcional) Em Repository description (Descrição do repositório), digite uma descrição para o repositório do GitHub.

 Note

Se você escolher um modelo de projeto Alexa Skill, precisará conectar uma conta de desenvolvedor da Amazon. Para obter mais informações sobre como trabalhar com projetos do Alexa Skill, consulte o [Tutorial: Criar um projeto de habilidade do Alexa no AWS CodeStar](#).

9. Se seu projeto estiver implantado em instâncias do Amazon EC2 e você quiser fazer alterações, configure suas instâncias do Amazon EC2 em Configuração do Amazon EC2. Por exemplo, você pode escolher entre os tipos de instância disponíveis para o projeto.

 Note

Os diferentes tipos de instância do Amazon EC2 fornecem diferentes níveis de potência de computação e podem ter custos associados diferentes. Para obter mais informações, consulte [Tipos de Instância do Amazon EC2](#) e [Preços do Amazon EC2](#).

Se tiver mais de uma nuvem privada virtual (VPC) ou várias sub-redes criadas no Amazon Virtual Private Cloud, você também poderá escolher a VPC e a sub-rede a serem utilizadas. No entanto, se escolher um tipo de instância do Amazon EC2 não compatível com instâncias dedicadas, você não poderá escolher uma VPC cuja localização de instância esteja definida como Dedicada.

Para obter mais informações, consulte [O que é o Amazon VPC?](#) e [Noções básicas de instâncias dedicadas](#).

Em Par de chaves, escolha o par de chaves do Amazon EC2 que você criou em [Etapa 4: criar um par de chaves do Amazon EC2 para projetos do AWS CodeStar](#). Selecione Eu reconheço que tenho acesso ao arquivo de chave privada.

10. Escolha Next (Próximo).
11. Analise os recursos e os detalhes da configuração.
12. Escolha Avançar ou Criar projeto. (A opção exibida depende do modelo do projeto.)

Pode levar alguns minutos para criar o projeto, incluindo o repositório.

13. Depois que seu projeto tiver um repositório, você poderá usar a página Repositório para configurar o acesso a ele. Use os links nas Próximas etapas para configurar um IDE, configurar o controle de problemas ou adicionar membros da equipe ao seu projeto.

## Etapa 2: Adicionar informações de exibição a seu perfil de usuário do AWS CodeStar

Quando cria um projeto, você é adicionado à equipe do projeto como proprietário. Se esta for a primeira vez em que usa o AWS CodeStar, você será solicitado a fornecer:

- O nome de exibição a ser mostrado aos outros usuários.
- O endereço de e-mail a ser mostrado aos outros usuários.

Essas informações são usadas em seu perfil de usuário do AWS CodeStar. Os perfis de usuário não são específicos ao projeto, mas são limitados a uma região da AWS. Você deve criar um perfil de usuário em cada região do AWS na qual pertence a projetos. Cada perfil pode conter informações diferentes, se você preferir.

Digite um nome de usuário e um endereço de e-mail e, em seguida, escolha Next (Avançar).

### Note

Esse nome de usuário e endereço de e-mail são usados em seu perfil de usuário do AWS CodeStar. Se o seu projeto usa recursos fora da AWS (por exemplo, um repositório GitHub

ou chamados no Atlassian JIRA), esses provedores de recursos podem ter os próprios perfis de usuário, com nomes de usuário e endereços de e-mail diferentes. Para obter mais informações, consulte a documentação do provedor do recurso.

## Etapa 3: Visualizar o projeto

A página do projeto do AWS CodeStar é onde você e sua equipe visualizam o status dos recursos do projeto, incluindo as confirmações mais recentes do projeto, o estado do pipeline de entrega contínua e a performance das instâncias. Para ver mais informações sobre qualquer um desses recursos, escolha a página correspondente na barra de navegação.

Em seu novo projeto, a barra de navegação contém as seguintes páginas:

- A página Visão geral contém informações sobre a atividade do seu projeto, os recursos do projeto e o conteúdo do README do seu projeto.
- A página IDE é onde você conecta seu projeto a um ambiente de desenvolvimento integrado (IDE) para modificar, testar e enviar alterações de código-fonte por push. Ele contém instruções para configurar IDEs para o GitHub e repositórios do AWS CodeCommit e informações sobre seus ambientes do AWS Cloud9.
- A página Repositório exibe os detalhes do seu repositório, incluindo nome, provedor, quando foi modificado pela última vez e URLs de clonagem. Você também pode ver informações sobre o commit mais recente e visualizar e criar pull requests.
- A página Pipeline exibe informações de CI/CD sobre seu pipeline. Você pode visualizar detalhes do pipeline, como nome, ação mais recente e status. Você pode ver o histórico do pipeline e lançar uma alteração. Você também pode ver o status das etapas individuais do seu pipeline.
- A página de monitoramento exibe o Amazon EC2 ou métricas do AWS Lambda, dependendo da configuração do seu projeto. Por exemplo, ela exibe a utilização de CPU de todas as instâncias do Amazon EC2 implantadas pelo AWS Elastic Beanstalk ou pelos recursos do CodeDeploy em seu pipeline. Em projetos que usam o AWS Lambda, ele exibe métricas de invocação e de erro da função do Lambda. Essas informações são exibidas por hora. Se você tiver usado o modelo de projeto AWS CodeStar sugerido para este tutorial, deverá ver um pico de atividades notável à medida que o aplicativo é implantado pela primeira vez nessas instâncias. Você pode atualizar o monitoramento para ver as alterações na integridade da instância, o que pode ajudar a identificar problemas ou a necessidade de mais recursos.

- A página Problemas é para integrar o projeto do AWS CodeStar com um projeto do Atlassian JIRA existente. A configuração desse bloco possibilitará que você e a equipe do projeto acompanhem problemas do JIRA no painel do projeto.

O painel de navegação no lado esquerdo do console é onde você pode navegar entre as páginas Projeto, Equipe e Configurações.

## Etapa 4: Confirmar uma alteração

Primeiro, examine o aplicativo de exemplo que foi incluído em seu projeto. Veja a aparência do aplicativo escolhendo Exibir aplicativo de qualquer lugar na navegação do seu projeto. O aplicativo web de exemplo será exibido em uma nova janela ou guia do navegador. Este é o código de exemplo que o AWS CodeStar compilou e implantou.

Para examinar o próprio código, na barra de navegação, escolha Repositório. Escolha o link em Nome do repositório e o repositório do seu projeto será aberto em uma nova guia ou janela. Leia o conteúdo do arquivo Leiam do repositório (README .md) e procure o conteúdo desses arquivos.

Nesta etapa, você altera o código e, em seguida, enviar essa alteração para o seu repositório. É possível fazer isso de algumas formas:

- Se o código do projeto estiver armazenado em um repositório do CodeCommit ou do GitHub, use o AWS Cloud9 para trabalhar com o código diretamente no seu navegador da web, sem instalar ferramentas. Para obter mais informações, consulte [Criar um ambiente do AWS Cloud9 para um projeto](#).
- Se o código do projeto estiver armazenado em um repositório do CodeCommit, e você tiver o Visual Studio ou o Eclipse instalado, poderá usar o AWS Toolkit for Visual Studio ou o AWS Toolkit for Eclipse para se conectar ao código com mais facilidade. Para obter mais informações, consulte [Usar um IDE com o AWS CodeStar](#). Se você não tiver o Visual Studio nem o Eclipse, instale um cliente do Git e siga as instruções mais adiante nesta etapa.
- Se o código do projeto estiver armazenado em um repositório do GitHub, você poderá usar suas ferramentas de IDE para se conectar ao GitHub.
  - Para o Visual Studio, você pode usar ferramentas como o GitHub Extension para Visual Studio. Para obter mais informações, consulte a página [Visão geral](#) no site do GitHub Extension para Visual Studio e [Conceitos básicos do GitHub para Visual Studio](#) no site do GitHub.
  - Para o Eclipse, você pode usar uma ferramenta como o EGit para Eclipse. Para obter mais informações, consulte a [Documentação do EGit](#) no site do EGit.



- Para outros IDEs, consulte a respectiva documentação.
- Para outros tipos de repositórios de código, consulte a documentação do provedor do repositório.

As instruções a seguir mostram como fazer uma alteração secundária na amostra.

Para configurar o computador para confirmar alterações (usuário do IAM)

#### Note

Neste procedimento, presumimos que o código de seu projeto esteja armazenado em um repositório do CodeCommit. Para outros tipos de repositório de código, consulte a documentação do provedor do repositório e, em seguida, passe para o próximo procedimento, "[Para clonar o repositório do projeto e fazer uma alteração](#)".

Se o código estiver armazenado no CodeCommit e já estiver usando o CodeCommit ou tiver usado o console do AWS CodeStar para criar um ambiente de desenvolvimento do AWS Cloud9 para o projeto, você não precisará de mais configuração. Passe para o próximo procedimento, [Para clonar o repositório do projeto e fazer uma alteração](#).

1. [Instale o Git](#) no computador local.
2. Faça login no AWS Management Console e abra o console do IAM em <https://console.aws.amazon.com/iam/>.

Faça login como o usuário do IAM que usará as credenciais do Git para conexões com o repositório do projeto do AWS CodeStar no CodeCommit.

3. No painel de navegação do console do IAM, selecione Usuários e, na lista de usuários, escolha o seu usuário do IAM.
4. Na página de detalhes do usuário, selecione a guia Credenciais de segurança e, em Credenciais do Git em HTTPS para o CodeCommit, escolha Gerar.

#### Note

Você não pode escolher suas próprias credenciais de login para as credenciais do Git. Para obter mais informações, consulte [Usar credenciais do Git e HTTPS com o CodeCommit](#).

5. Copie as credenciais de login que o IAM gerou para você. Você pode escolher Mostrar e, em seguida, copiar e colar essas informações em um arquivo seguro no computador local ou escolher Fazer download das credenciais para fazer download dessas informações como um arquivo .CSV. Você precisa dessas informações para se conectar ao CodeCommit.


Após salvar suas credenciais, selecione Close.

 Important

Essa é sua única chance de salvar as credenciais de login. Se você não salvá-los, poderá copiar o nome de usuário do console do IAM, mas não poderá pesquisar a senha. Você deverá reiniciar a senha e salvá-la.

Para configurar o computador para confirmar alterações (usuário federado)

Você pode usar o console para fazer upload dos arquivos para o repositório ou usar o Git para se conectar pelo computador local. Se você estiver usando acesso federado, siga estas etapas para usar o Git para se conectar e clonar o repositório do computador local.

 Note

Neste procedimento, presumimos que o código de seu projeto esteja armazenado em um repositório do CodeCommit. Para outros tipos de repositório de código, consulte a documentação do provedor do repositório e, em seguida, passe para o próximo procedimento, "[Para clonar o repositório do projeto e fazer uma alteração](#)".

1. [Instale o Git](#) no computador local.
2. [Instale o AWS CLI](#).
3. Configure as credenciais de segurança temporárias para um usuário federado. Para obter informações, consulte [Acesso temporário a repositórios](#). As credenciais temporárias consistem em:
  - Chave de acesso da AWS
  - Chave secreta do AWS.
  - Session token (Token da sessão)

Para obter mais informações sobre credenciais temporárias, consulte [Permissões para GetFederationToken](#).

4. Conecte-se ao repositório usando o auxiliar de credenciais da AWS CLI. Para obter mais informações, consulte [Etapas de configuração para conexões HTTPS com repositórios do CodeCommit no Linux, no macOS ou no Unix com o assistente de credenciais da CLI do AWS](#) ou [Etapas de configuração para conexões HTTPS com repositórios do AWS no Windows com o assistente de credenciais da CLI do](#)
5. O exemplo a seguir mostra como se conectar a um repositório do CodeCommit e enviar uma confirmação para ele.

Exemplo: para clonar o repositório do projeto e fazer uma alteração

#### Note

Este procedimento mostra como clonar o repositório de código do projeto em seu computador, fazer uma alteração no arquivo `index.html` do projeto e, em seguida, enviar as suas alterações para o repositório remoto. Neste procedimento, presumimos que o código do projeto esteja armazenado em um repositório do CodeCommit e que você esteja usando um cliente do Git a partir da linha de comando. Para outros tipos de repositórios de código ou ferramentas, consulte a documentação do provedor sobre como clonar o repositório, alterar o arquivo e, em seguida, enviar o código.

1. Se você tiver usado o console do AWS CodeStar para criar um ambiente de desenvolvimento do AWS Cloud9 para o projeto, abra o ambiente de desenvolvimento e vá para a etapa 3 deste procedimento. Para abrir o ambiente de desenvolvimento, consulte [Abrir um ambiente do AWS Cloud9 para um projeto](#).

Com seu projeto aberto no console do AWS CodeStar, na barra de navegação, selecione Repositório. Em Clonar URL, escolha o protocolo para o tipo de conexão configurado para o CodeCommit e copie o link. Por exemplo, se você seguiu as etapas no procedimento anterior para configurar as credenciais do Git para o CodeCommit, escolha HTTPS.

2. No computador local, abra um terminal ou janela de linha de comando e altere os diretórios para um diretório temporário. Execute o comando `git clone` para clonar o repositório no computador. Cole o link que você copiou. Por exemplo, para o CodeCommit usando HTTPS:

```
git clone https://git-codecommit.us-east-2.amazonaws.com/v1/repos/my-first-projec
```

Na primeira vez em que se conectar, será solicitado que você forneça as credenciais do repositório. Para o CodeCommit, insira as credenciais de login do Git que você baixou no procedimento anterior.

3. Navegue até o diretório clonado no computador e examine o conteúdo.
4. Abra o arquivo `index.html` (na pasta pública) e faça uma alteração no arquivo. Por exemplo, adicione um parágrafo após a tag `<H2>`, como:

```
<P>Hello, world!</P>
```

Salve o arquivo.

5. No terminal ou no prompt de comando, adicione o arquivo alterado, confirme e envie a alteração por push:

```
git add index.html
git commit -m "Making my first change to the web app"
git push
```

6. Na página Repositório, veja as alterações em andamento. Você verá que o histórico de confirmação do repositório é atualizado com sua confirmação incluindo a mensagem de confirmação. Na página pipeline, você pode ver o pipeline coletar sua alteração no repositório e começar a compilar e implantar. Depois que seu aplicativo de web for implementado, você pode escolher Exibir aplicativo para visualizar sua alteração.

#### Note

Se Failed (Com falha) for exibido para qualquer um dos estágios de pipeline, consulte o seguinte para obter ajuda na solução do problema:

- No estágio Origem, consulte [Solucionar problemas do AWS CodeCommit](#) no Guia do usuário do AWS CodeCommit.

- No estágio Criar, consulte [Solucionar problemas do AWS CodeBuild](#) no Guia do usuário do AWS CodeBuild.
- No estágio Implantar, consulte [Solucionar problemas do AWS CloudFormation](#) no Guia do usuário do AWS CloudFormation.
- Para outros problemas, consulte [Solução de problemas AWS CodeStar](#).

## Etapa 5: Adicionar mais membros à equipe

Cada projeto do AWS CodeStar já está configurado com três funções do AWS CodeStar. Cada função oferece seu próprio nível de acesso ao projeto e a seus recursos:

- Proprietário: pode adicionar e remover membros da equipe, alterar o painel do projeto e excluir o projeto.
- Colaborador: pode alterar o painel do projeto e contribuir com código, se o código estiver armazenado no CodeCommit, mas não pode adicionar nem remover membros da equipe nem excluir o projeto. Essa é a função que você deve escolher para a maioria dos membros da equipe em um projeto do AWS CodeStar.
- Visualizador: pode visualizar o painel do projeto, o código do projeto se o código estiver armazenado no CodeCommit e o estado do projeto, mas não pode mover, adicionar nem remover blocos do painel do projeto.

### Important

Se o seu projeto usa recursos fora da AWS (por exemplo, um repositório do GitHub ou chamados no Atlassian JIRA), o acesso a esses recursos é controlado pelo provedor de recursos, não pelo AWS CodeStar. Para obter mais informações, consulte a documentação do provedor do recurso.


Qualquer pessoa que tenha acesso a um projeto do AWS CodeStar pode usar o console do AWS CodeStar para acessar recursos que estão fora da AWS mas estão relacionados a esse projeto.

O AWS CodeStar não permite que os membros da equipe do projeto participem de ambientes de desenvolvimento relacionados ao AWS Cloud9 para um projeto. Para permitir que um membro da equipe participe de um ambiente compartilhado, consulte [Compartilhar um ambiente do AWS Cloud9 com um membro da equipe do projeto](#).

Para obter mais informações sobre equipes e funções de projeto, consulte [Como trabalhar com equipes AWS CodeStar](#).

Para adicionar um membro da equipe a um projeto do AWS CodeStar (console)

1. Abra o console do AWS CodeStar em <https://console.aws.amazon.com/codestar/>.
2. Selecione Projetos no painel de navegação e escolha seu projeto.
3. No painel lateral de navegação do projeto, escolha Equipe.
4. Na página Membros da equipe, escolha Adicionar membro da equipe.
5. Em Escolher usuário, proceda de uma das seguintes maneiras:
  - Se um usuário do IAM já existir para a pessoa que você deseja adicionar, escolha o usuário IAM na lista.

 Note

Os usuários que já foram adicionados a outro projeto do AWS CodeStar aparecerão na lista de Usuários existentes do AWS CodeStar.

Em Função do projeto, escolha a função do AWS CodeStar (proprietário, colaborador ou visualizador) para esse usuário. Essa é uma função em nível de projeto do AWS CodeStar que só pode ser alterada por um proprietário do projeto. Quando aplicada a um usuário do IAM, a função fornece todas as permissões necessárias para acessar os recursos do projeto do AWS CodeStar. Ela aplica as políticas necessárias para criar e gerenciar as credenciais do Git para o código armazenado no CodeCommit no IAM ou para fazer upload de chaves SSH do Amazon EC2 para o usuário no IAM.

**⚠ Important**

Você não pode fornecer ou alterar as informações de nome de exibição ou de e-mail de um usuário do IAM, a menos que você tenha feito login no console com esse usuário. Para obter mais informações, consulte [Gerenciar informações de exibição de seu perfil de usuário do AWS CodeStar](#).

Escolha Adicionar membro da equipe.

- Se um usuário do IAM não existir para a pessoa que você deseja adicionar ao projeto, escolha Criar novo usuário do IAM;. Você será redirecionado para o console do IAM, onde poderá criar um novo usuário do IAM. Consulte [Como criar usuários do IAM](#) no guia do usuário do IAM para obter mais informações. Depois de criar seu usuário do IAM, retorne ao console do AWS CodeStar, atualize a lista de usuários e escolha o usuário do IAM que você criou na lista suspensa. Digite o nome de exibição do AWS CodeStar, o endereço de e-mail e a função do projeto que você deseja aplicar a esse novo usuário e escolha Adicionar membro da equipe.

**ℹ Note**

Para facilitar o gerenciamento, pelo menos um usuário deve receber a função Proprietário do projeto.

6. Envie ao novo membro da equipe as seguintes informações:

- Informações sobre a conexão de seu projeto do AWS CodeStar.
- Se o código-fonte estiver armazenado no CodeCommit, [instruções para configurar o acesso com as credenciais do Git](#) ao repositório do CodeCommit de seus computadores locais.
- Informações sobre como o usuário pode gerenciar seu nome de exibição, endereço de e-mail e chave SSH pública do Amazon EC2, conforme descrito em [Como trabalhar com seu perfil de usuário do AWS CodeStar](#).
- Uma senha avulsa e informações de conexão, se o usuário for novo no AWS e você tiver criado um usuário do IAM para essa pessoa. A senha expira na primeira vez em que o usuário entra. O usuário deve escolher uma nova senha.

## Etapa 6: Limpeza

Parabéns! Você concluiu o tutorial. Se você não desejar continuar a usar esse projeto e seus recursos, você deverá excluí-lo para evitar possíveis cobranças contínuas em sua conta da AWS.

Para excluir um projeto no AWS CodeStar

1. Abra o console do AWS CodeStar em <https://console.aws.amazon.com/codestar/>.
2. Selecione Projetos no painel de navegação.
3. Selecione o projeto que deseja excluir e selecione Excluir.

Ou abra o projeto e escolha Settings do painel de navegação na esquerda do console. Na página de detalhes do projeto, escolha Excluir projeto.

4. Na Página de confirmação de exclusão, insira excluir. Mantenha a opção Excluir recursos selecionada se quiser excluir recursos do projeto. Escolha Delete (Excluir).

A exclusão de um projeto pode demorar vários minutos. Depois de ser excluído, o projeto não aparece mais na lista de projetos no console do AWS CodeStar.

### Important

Se o seu projeto usa recursos fora da AWS (por exemplo, um repositório do GitHub ou chamados no Atlassian JIRA), esses recursos não serão excluídos, mesmo se você marcar a caixa de seleção.

O projeto não poderá ser excluído se alguma política gerenciada do AWS CodeStar tiver sido anexada manualmente a funções que não sejam usuários do IAM. Se tiver anexado as políticas gerenciadas do projeto a uma função do usuário federado, você deverá desanexar a política para excluir o projeto. Para obter mais informações, consulte [???](#).

## Etapa 7: preparar o projeto para um ambiente de produção

Depois que tiver criado o projeto, você estará pronto para criar, testar e implantar código. Analise as seguintes considerações para manter o projeto em um ambiente de produção:



- Aplicar patches regularmente e revise as melhores práticas de segurança para as dependências usadas pelo aplicativo. Para obter mais informações, consulte [Melhores práticas de segurança para recursos do AWS CodeStar](#).
- Monitore regularmente as configurações de ambiente sugeridas pela linguagem de programação para o projeto.

## Próximas etapas

Aqui estão alguns outros recursos para ajudar a saber mais sobre o AWS CodeStar:

- O [Tutorial: Criação e gerenciamento de um projeto com tecnologia sem servidor no AWS CodeStar](#) usa um projeto que cria e implanta um serviço da Web usando uma lógica no AWS Lambda e pode ser chamado por uma API no Amazon API Gateway.
- [Modelos de projetos do AWS CodeStar](#) descreve outros tipos de projeto que você pode criar.
- [Como trabalhar com equipes AWS CodeStar](#) fornece informações sobre como habilitar outras pessoas para ajudá-lo a trabalhar em seus projetos.

## Tutorial: Criação e gerenciamento de um projeto com tecnologia sem servidor no AWS CodeStar

Nesse tutorial, você usa o AWS CodeStar para criar um projeto que use o AWS Serverless Application Model (AWS SAM) para criar e gerenciar recursos da AWS para um serviço da Web hospedado no AWS Lambda.

O AWS CodeStar usa o AWS SAM, que conta com o AWS CloudFormation, para oferecer uma maneira simplificada de criar e gerenciar os recursos suportados da AWS, incluindo as APIs do Amazon API Gateway, as funções do AWS Lambda e as tabelas do Amazon DynamoDB. (Esse projeto não usa nenhuma tabela do Amazon DynamoDB.)

Para obter mais informações, consulte [Modelo de aplicação de tecnologia sem servidor do AWS \(AWS SAM\)](#) no GitHub.

Pré-requisito: concluir as etapas em [Configuração AWS CodeStar](#).

**Note**

A conta da AWS pode ser cobrada pelos custos relacionados a este tutorial, inclusive os custos dos serviços da AWS usados pelo AWS CodeStar. Para obter mais informações, consulte [Preços do AWS CodeStar](#).

## Tópicos

- [Visão geral](#)
- [Etapa 1: Criar o projeto](#)
- [Etapa 2: Explorar os recursos do projeto](#)
- [Etapa 3: Testar o serviço da web](#)
- [Etapa 4: Configurar sua estação de trabalho local para editar o código do projeto](#)
- [Etapa 5: Adicionar lógica ao serviço da web](#)
- [Etapa 6: Testar o serviço da web aprimorado](#)
- [Etapa 7: Adicionar um teste de unidade ao serviço da web](#)
- [Etapa 8: Visualizar os resultados do teste de unidade](#)
- [Etapa 9: Limpeza](#)
- [Próximas etapas](#)

## Visão geral

Neste tutorial, você:

1. Usa o AWS CodeStar para criar um projeto que use o AWS SAM para criar e implantar um serviço Web baseado em Python. Esse serviço da Web está hospedado no AWS Lambda e pode ser acessado por meio do Amazon API Gateway.
2. Explore os recursos principais do projeto, que incluem:
  - O repositório do AWS CodeCommit no qual o código-fonte do projeto está armazenado. Esse código-fonte inclui a lógica do serviço Web e define os recursos relacionados da AWS.
  - O pipeline do AWS CodePipeline que automatiza a criação do código-fonte. Esse pipeline usa o AWS SAM para criar e implantar uma função no AWS Lambda, criar uma API relacionada no Amazon API Gateway e conectar a API à função.
  - A função que é implantada no AWS Lambda.

- A API criada no Amazon API Gateway.
3. Teste o serviço Web para confirmar que o AWS CodeStar criou e implantou o serviço Web conforme o esperado.
  4. Configure sua estação de trabalho local para trabalhar com o código-fonte do projeto.
  5. Altere o código-fonte do projeto usando sua estação de trabalho local. Quando você adiciona uma função ao projeto e, em seguida, envia as suas alterações para o código-fonte, o AWS CodeStar recompila e reimplanta o serviço da web.
  6. Teste o serviço da web novamente para confirmar que o AWS CodeStar o recriou e o reimplantou conforme o esperado.
  7. Escreva um teste de unidade usando sua estação de trabalho local para substituir alguns de seus testes manuais por um teste automatizado. Quando você envia o teste de unidade, o AWS CodeStar recompila e reimplanta o serviço da web e executa o teste de unidade.
  8. Visualize os resultados dos testes de unidade.
  9. Limpe o projeto. Essa etapa ajuda a evitar cobranças da conta da AWS os custos relacionados a esse tutorial.

## Etapa 1: Criar o projeto

Nesta etapa, use o console do AWS CodeStar para criar um projeto.

1. Faça login no AWS Management Console e então abra o console do AWS CodeStar em <https://console.aws.amazon.com/codestar/>.

### Note

Você deve fazer login no AWS Management Console usando as credenciais associadas ao usuário do IAM criado ou identificado em [Configuração AWS CodeStar](#). Esse usuário deve ter a política gerenciada **AWSCodeStarFullAccess** anexada.

2. Escolha a região da AWS onde você deseja criar o projeto e seus recursos.

Para obter informações sobre as regiões da AWS onde o AWS CodeStar está disponível, consulte [Regiões e endpoints](#) na Referência geral da AWS.

3. Escolha Create project (Criar projeto).
4. Na página Choose a project template (Escolher um modelo de projeto):

- Em Tipo de aplicação, selecione Serviço de web.
  - Em Linguagem de programação, selecione Python.
  - Em Serviço do AWS, selecione AWS Lambda.
5. Escolha a caixa que contém suas seleções. Escolha Next (Próximo).
  6. Em Project name (Nome do projeto), digite o nome do projeto (por exemplo, **My SAM Project**). Se você usar um nome diferente do exemplo, certifique-se de usá-lo ao longo de todo este tutorial.

Em Project ID (ID do projeto), o AWS CodeStar escolhe um identificador relacionado para esse projeto (por exemplo, my-sam-project). Se vir outro ID de projeto, certifique-se de usá-lo durante todo este tutorial.

Deixe AWS CodeCommit selecionado e não altere o valor Repository name (Nome do repositório).

7. Escolha Next (Próximo).
8. Revise suas configurações e escolha Criar projeto.

Se essa for a primeira vez que você usa o AWS CodeStar nessa região do AWS, em Nome de exibição e E-mail, digite o nome de exibição e o endereço de e-mail que o AWS CodeStar deverá usar para o usuário do IAM. Escolha Next (Próximo).

9. Aguarde enquanto o AWS CodeStar cria o projeto. Isso pode demorar vários minutos. Não continue até ver o banner provisionado do projeto ao atualizar.

## Etapa 2: Explorar os recursos do projeto

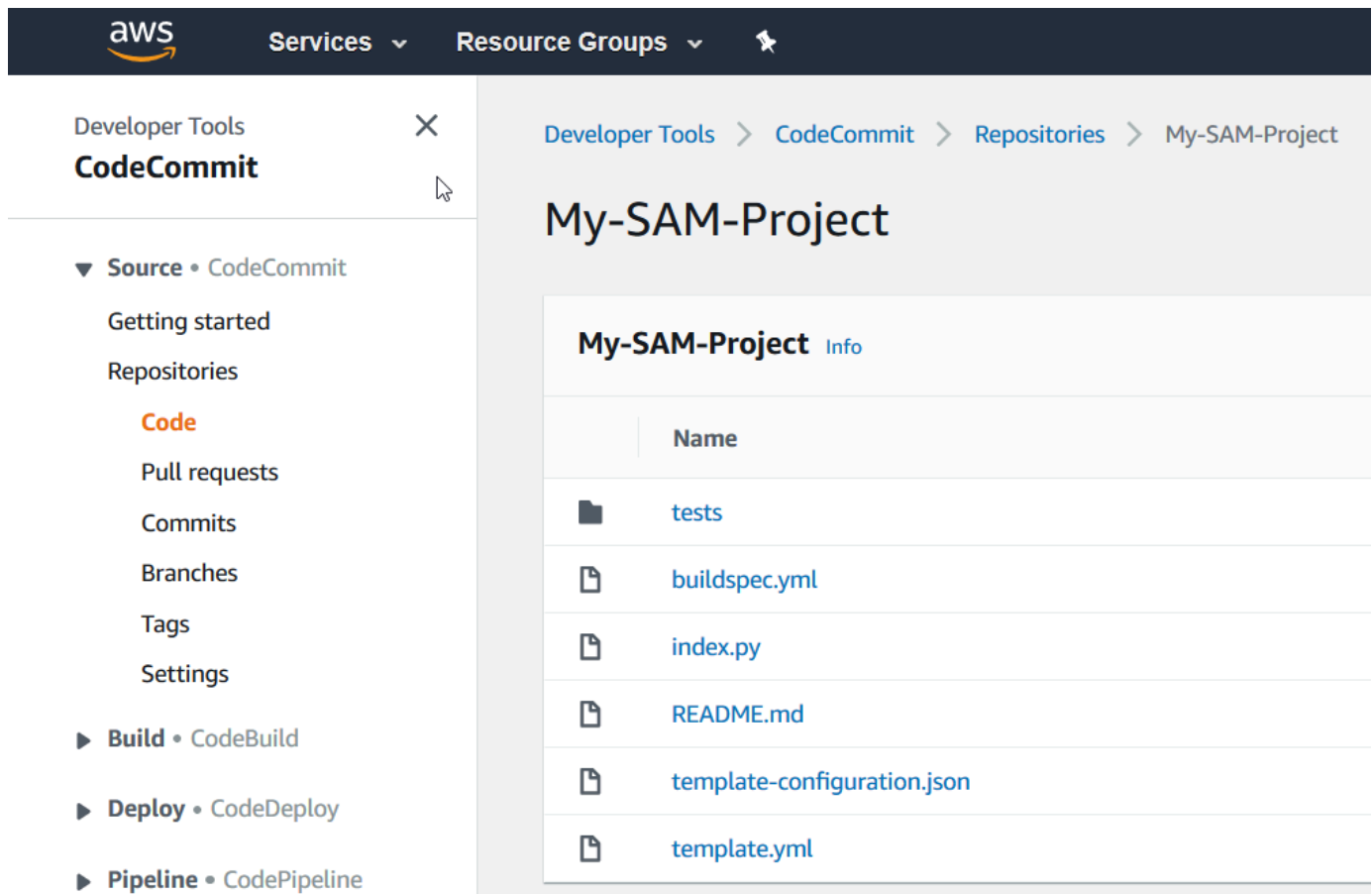
Nesta etapa, você explora quatro dos recursos da AWS para entender como o projeto funciona:

- O repositório do AWS CodeCommit no qual o código-fonte do projeto está armazenado. O AWS CodeStar dá ao repositório o nome my-sam-project, em que my-sam-project é o nome do projeto.
- O pipeline do AWS CodePipeline que usa o CodeBuild e o AWS SAM para automatizar a criação e a implantação da função do Lambda do serviço da web e a API no API Gateway. O AWS CodeStar dá ao pipeline o nome my-sam-project--Pipeline, onde my-sam-project é o ID do projeto.
- A função do Lambda que contém a lógica do serviço da web. O AWS CodeStar dá à função o nome awscodestar-my-sam-project-lambda-HelloWorld-**RANDOM\_ID**, onde:
  - my-sam-project é o ID do projeto.

- HelloWorld é o ID da função conforme especificado no arquivo `template.yaml` no repositório do AWS CodeCommit. Você explorará esse arquivo mais tarde.
- ***RANDOM\_ID*** é um ID aleatório que o SAM da AWS atribui à função para ajudar a garantir a exclusividade.
- A API no API Gateway que facilita chamar a função do Lambda. O AWS CodeStar dá à API o nome `awscodestar-my-sam-project--lambda`, onde `my-sam-project` é o ID do projeto.

Para explorar o repositório de código-fonte no CodeCommit

1. Com seu projeto aberto no console do AWS CodeStar, na barra de navegação, selecione Repositório.
2. Escolha o link para seu repositório do CodeCommit (**My-SAM-Project**) em Detalhes do repositório.
3. No console do CodeCommit, na página Código, os arquivos de código-fonte para o projeto são exibidos:
  - `buildspec.yaml`, que o CodePipeline instrui o CodeBuild a ser usado durante a fase de construção, para empacotar o serviço da web usando o AWS SAM.
  - `index.py`, que contém a lógica para a função do Lambda Essa função simplesmente gera a string `Hello World` e um carimbo de data e hora no formato ISO.
  - `README.md`, que contém informações gerais sobre o repositório.
  - `template-configuration.json`, que contém o ARN do projeto com espaços reservados usados para marcar recursos com o ID do projeto.
  - `template.yaml`, que o AWS SAM usa para empacotar o serviço da web e criar a API no API Gateway.



Para visualizar o conteúdo de um arquivo, escolha-o na lista.

Para obter mais informações sobre o uso do console CodeCommit, consulte o [Guia do usuário do AWS CodeCommit](#).

Para explorar o pipeline no CodePipeline

1. Para exibir informações sobre o pipeline, com seu projeto aberto no console do AWS CodeStar na barra de navegação lateral, selecione Pipeline e verá que o pipeline contém:
  - Uma fase Origem para obter o código-fonte do CodeCommit.
  - O estágio Criar para compilar o código-fonte com o CodeBuild.
  - Uma fase Deploy (Implantar) para implantar o código-fonte e os recursos da AWS com o AWS SAM.

2. Para ver mais informações sobre o pipeline, em Detalhes do pipeline, escolha seu pipeline para abri-lo no console do CodePipeline.

Para obter informações sobre o uso do console CodePipeline, consulte o [Guia do usuário do AWS CodePipeline](#).

Para explorar a atividade do projeto e os recursos de serviço do AWS na página Visão geral

1. Abra o projeto no console do AWS CodeStar e, na barra de navegação, escolha Visão geral.
2. Analise as listas Atividade do projeto e Recursos do projeto.

Para explorar a função no Lambda

1. Com seu projeto aberto no console do AWS CodeStar, na barra de navegação lateral, selecione Visão geral.
2. Em Recursos do projeto, na coluna ARN, escolha o link da função do Lambda.

O código da função é exibido no console do Lambda.

Para obter informações sobre o uso do console Lambda, consulte o [Manual do desenvolvedor do AWS Lambda](#).

Para explorar o API no API Gateway

1. Com seu projeto aberto no console do AWS CodeStar, na barra de navegação lateral, selecione Visão geral.
2. Em Recursos do projeto, na coluna ARN, escolha o link da API do Amazon API Gateway.

Os recursos da API são exibidos no console do API Gateway.

Para obter informações sobre o uso do console API Gateway, consulte o [Guia do desenvolvedor](#).

## Etapa 3: Testar o serviço da web

Nesta etapa, você testará o serviço da web que o AWS CodeStar acabou de criar e implantar.

1. Com o seu projeto ainda aberto da etapa anterior, na barra de navegação, selecione Pipeline.

2. No bloco Implantação contínua, verifique se Bem-sucedido é exibido para as fases Origem, Criar e Implantar antes de continuar. Isso pode demorar vários minutos.

 Note

Se Failed (Com falha) for exibido para qualquer uma das fases, consulte o seguinte para obter ajuda na solução do problema:

- No estágio Origem, consulte [Solucionar problemas do AWS CodeCommit](#) no Guia do usuário do AWS CodeCommit.
- No estágio Criar, consulte [Solucionar problemas do AWS CodeBuild](#) no Guia do usuário do AWS CodeBuild.
- No estágio Implantar, consulte [Solucionar problemas do AWS CloudFormation](#) no Guia do usuário do AWS CloudFormation.
- Para outros problemas, consulte [Solução de problemas AWS CodeStar](#).

3. Escolha Novo aplicativo

Na nova guia que é aberta no navegador da web, o serviço da web exibe a seguinte saída de resposta:

```
{"output": "Hello World", "timestamp": "2017-08-30T15:53:42.682839"}
```

## Etapa 4: Configurar sua estação de trabalho local para editar o código do projeto

Nesta etapa, você configura a estação de trabalho local para editar o código-fonte no projeto do AWS CodeStar. Sua estação de trabalho local pode ser um computador físico ou virtual que execute macOS, Windows ou Linux.

1. Com o projeto ainda aberto da etapa anterior:
  - Na barra de navegação, escolha IDE e, em seguida, expanda Acesse o código do seu projeto.



- Escolha Exibir instruções abaixo da interface da linha de comando.

Se você tiver o Visual Studio ou o Eclipse instalado, escolha Exibir as instruções sob Visual Studio, ou Eclipse, siga as instruções e, em seguida, vá para [Etapa 5: Adicionar lógica ao serviço da web](#).

2. Siga as instruções para concluir as tarefas a seguir:
  - a. Configure o Git na sua estação de trabalho local.
  - b. Use o console do IAM para gerar credenciais do Git para o seu usuário do IAM.
  - c. Clone o repositório do CodeCommit do projeto em sua estação de trabalho local.
3. No painel de navegação esquerdo, escolha Projeto para retornar à visão geral do projeto.

## Etapa 5: Adicionar lógica ao serviço da web

Nesta etapa, você usa a estação de trabalho local para adicionar lógica ao serviço da web. Especificamente, você adiciona uma função do Lambda e, em seguida, a conecta à API no API Gateway.

1. Na estação de trabalho local, vá até o diretório que contém o repositório de código-fonte clonado.
2. Nesse diretório, crie um arquivo chamado `hello.py`. Adicione o seguinte código e salve-o em seguida:

```
import json


def handler(event, context):
    data = {
        'output': 'Hello ' + event["pathParameters"]["name"]
    }
    return {
        'statusCode': 200,
        'body': json.dumps(data),
        'headers': {'Content-Type': 'application/json'}
    }
```

O código anterior gera a string Hello e a string que o chamador enviar para a função

3. No mesmo diretório, abra o arquivo `template.yml`. Adicione o seguinte código ao final do arquivo e salve-o em seguida:


```
Hello:
  Type: AWS::Serverless::Function
  Properties:
    FunctionName: !Sub 'awscodestar-${ProjectId}-lambda-Hello'
    Handler: hello.handler
    Runtime: python3.7
    Role:
      Fn::GetAtt:
        - LambdaExecutionRole
        - Arn
    Events:
      GetEvent:
        Type: Api
        Properties:
          Path: /hello/{name}
          Method: get
```

O AWS SAM usa esse código anterior para criar uma função no Lambda, adicionar um novo método e o caminho à API no API Gateway e, em seguida, conectar esse método e o caminho à nova função.

 Note

O recuo do código anterior é importante. Se você não adicionar o código exatamente como for mostrado, o projeto poderá não ser criado corretamente.

4. Execute `git add .` para adicionar as alterações do arquivo à área de preparação do repositório clonado. Não se esqueça do ponto (`.`), que adiciona todos os arquivos alterados.

 Note

Se você estiver usando o Visual Studio ou o Eclipse em vez da linha de comando, as instruções para o uso do Git podem ser diferentes. Consulte a documentação do Eclipse ou do Visual Studio.

5. Executar `git commit -m "Added hello.py and updated template.yaml."` para confirmar os arquivos preparados no repositório clonado
6. Execute `git push` para enviar a confirmação para o repositório remoto por push.

**Note**

Você pode ser solicitado a fornecer as credenciais de login geradas anteriormente. Para que não sejam solicitados toda vez que você interagir com o repositório remoto, considere instalar e configurar um gerenciador de credenciais do Git. Por exemplo, no macOS ou no Linux, você pode executar `git config credential.helper 'cache --timeout 900'` no terminal para serem solicitados a cada 15 minutos. Ou você pode executar `git config credential.helper 'store --file ~/.git-credentials'` para nunca serem solicitados novamente. O Git armazena suas credenciais em texto nítido em um arquivo sem formatação em seu diretório inicial. Para obter mais informações, consulte [Ferramentas do GiT – armazenamento de credenciais](#) no site do Git.

Depois que o AWS CodeStar detectar o push, ele instruirá o CodePipeline a usar o CodeBuild e o AWS SAM para reconstruir e reimplantar o serviço da web. Você pode acompanhar o progresso da implantação na página Pipeline.

O AWS SAM fornece à nova função o nome `awscodestar-my-sam-project-lambda>Hello-RANDOM_ID`, em que:

- `my-sam-project` é o ID do projeto.
- `Hello` é o ID da função conforme especificado no arquivo `template.yaml`.
- **`RANDOM_ID`** é um ID aleatório que o AWS SAM atribui à função para fins de exclusividade.

## Etapa 6: Testar o serviço da web aprimorado

Nesta etapa, você testará o serviço da web aprimorado que o AWS CodeStar criou e implantou com base na lógica que você adicionou na etapa anterior.

1. Com seu projeto ainda aberto no console do AWS CodeStar, na barra de navegação, selecione Pipeline.
2. Assegure-se de ter executado o pipeline novamente e de que Bem-sucedido apareça nos estágios de Origem, Criação, e Implantação antes de continuar. Isso pode demorar vários minutos.

**Note**

Se Failed (Com falha) for exibido para qualquer uma das fases, consulte o seguinte para obter ajuda na solução do problema:

- No estágio Origem, consulte [Solucionar problemas do AWS CodeCommit](#) no Guia do usuário do AWS CodeCommit.
- No estágio Criar, consulte [Solucionar problemas do AWS CodeBuild](#) no Guia do usuário do AWS CodeBuild.
- No estágio Implantar, consulte [Solucionar problemas do AWS CloudFormation](#) no Guia do usuário do AWS CloudFormation.
- Para outros problemas, consulte [Solução de problemas AWS CodeStar](#).

### 3. Escolha Novo aplicativo

Na nova guia que é aberta no navegador da web, o serviço da web exibe a seguinte saída de resposta:

```
{"output": "Hello World", "timestamp": "2017-08-30T15:53:42.682839"}
```

4. Na caixa de endereço da guia, adicione o caminho **/hello/** e o nome ao final do URL (por exemplo, `https://API_ID.execute-api.REGION_ID.amazonaws.com/Prod/hello/YOUR_FIRST_NAME`) e pressione Enter.

Se o seu nome for Mary, o serviço da web exibirá a saída de resposta a seguir:

```
{"output": "Hello Mary"}
```

## Etapa 7: Adicionar um teste de unidade ao serviço da web

Nesta etapa, você usa a estação de trabalho local para adicionar um teste que o AWS CodeStar executa no serviço da web. Esse teste substitui os testes manuais que você fez anteriormente.

1. Na estação de trabalho local, vá até o diretório que contém o repositório de código-fonte clonado.
2. Nesse diretório, crie um arquivo chamado `hello_test.py`. Adicione o seguinte código e salve-o em seguida.

```
from hello import handler

def test_hello_handler():

    event = {
        'pathParameters': {
            'name': 'testname'
        }
    }

    context = {}

    expected = {
        'body': '{"output": "Hello testname"}',
        'headers': {
            'Content-Type': 'application/json'
        },
        'statusCode': 200
    }

    assert handler(event, context) == expected
```

Esse teste verifica se o resultado da função do Lambda está no formato esperado. Se a resposta for sim, o teste será bem-sucedido. Caso contrário, o teste falhará.

3. No mesmo diretório, abra o arquivo `buildspec.yml`. Substitua o conteúdo do arquivo pelo seguinte código e, em seguida, salve o arquivo.

```
version: 0.2

phases:
  install:
    runtime-versions:
      python: 3.7

    commands:
      - pip install pytest
```

```
# Upgrade AWS CLI to the latest version
- pip install --upgrade awscli

pre_build:
  commands:
    - pytest

build:
  commands:
    # Use AWS SAM to package the application by using AWS CloudFormation
    - aws cloudformation package --template template.yml --s3-bucket
    $S3_BUCKET --output-template template-export.yml

    # Do not remove this statement. This command is required for AWS CodeStar
    projects.
    # Update the AWS Partition, AWS Region, account ID and project ID in the
    project ARN on template-configuration.json file so AWS CloudFormation can tag
    project resources.
    - sed -i.bak 's/\${PARTITION}\$/'${PARTITION}'/g;s/\${AWS_REGION}
    \$/'${AWS_REGION}'/g;s/\${ACCOUNT_ID}\$/'${ACCOUNT_ID}'/g;s/\${PROJECT_ID}\
    \$/'${PROJECT_ID}'/g' template-configuration.json

artifacts:
  type: zip
  files:
    - template-export.yml
    - template-configuration.json
```

Essa especificação de criação instrui o CodeBuild a instalar o pytest, a estrutura de teste do Python, em seu ambiente de produção. O CodeBuild usa o pytest para executar o teste de unidade. O resto da especificação de criação é o mesmo de antes.

#### 4. Use o Git para enviar essas alterações para o repositório remoto.

```
git add .

git commit -m "Added hello_test.py and updated buildspec.yml."

git push
```

## Etapa 8: Visualizar os resultados do teste de unidade

Nesta etapa, você verá se o teste de unidade foi bem-sucedido ou não.

1. Com seu projeto ainda aberto no console do AWS CodeStar, na barra de navegação, selecione Pipeline.
2. Certifique-se de que o pipeline tenha funcionado novamente antes de continuar. Isso pode demorar vários minutos.

Se o teste de unidade foi bem-sucedido, Succeeded (Bem-sucedido) será exibido para a fase Build (Criar).

3. Para visualizar os detalhes do resultado de teste de unidade, no estágio Criar, escolha o link CodeBuild.
4. No console do CodeBuild, na página Criar projeto: my-sam-project, em Histórico de criação, escolha o link na coluna Execução da criação da tabela.
5. Na página my-sam-project:**BUILD\_ID**, em Build logs (Criar logs), escolha o link View entire log (Exibir todo o log).
6. No console do Amazon CloudWatch Logs, procure um resultado de teste semelhante ao seguinte na saída do log. No resultado do teste a seguir, o teste foi bem-sucedido:

```
...
===== test session starts =====
platform linux2 -- Python 2.7.12, pytest-3.2.1, py-1.4.34, pluggy-0.4.0
rootdir: /codebuild/output/src123456789/src, inifile:
collected 1 item

hello_test.py .

===== 1 passed in 0.01 seconds =====
...
```

Se o teste falhou, deve haver detalhes na saída do log para ajudá-lo a solucionar a falha.

## Etapa 9: Limpeza

Nesta etapa, você limpa o projeto para evitar custos contínuos desse projeto.

Se você deseja continuar a usar esse projeto, pode ignorar essa etapa, mas sua conta da AWS poderá continuar a ser cobrada.

1. Com seu projeto ainda aberto no console do AWS CodeStar, na barra de navegação, selecione Configurações.
2. Em Detalhes do projeto, Selecione Excluir projeto.
3. Insira **delete**, mantenha a caixa Excluir recursos selecionada e escolha Excluir.

#### Important

Se você desmarcar essa caixa, o registro do projeto será excluído do AWS CodeStar, mas muitos dos recursos da AWS do projeto serão mantidos. A conta da AWS pode continuar sendo cobrada.

Se ainda houver um bucket do Amazon S3 criado pelo AWS CodeStar para esse projeto, siga estas etapas para excluí-lo:

1. Abra o console do Amazon S3 em <https://console.aws.amazon.com/s3/>.
2. Na lista de buckets, escolha o ícone ao lado de aws-codestar-**REGION\_ID**-**ACCOUNT\_ID**-my-sam-project--pipe, em que:
  - **REGION\_ID** é o ID da região da AWS do projeto recém-excluído.
  - **ACCOUNT\_ID** é o ID da conta da AWS.
  - my-sam-project é o ID do projeto que você acabou de excluir.
3. Escolha Empty Bucket (Esvaziar bucket). Digite o nome do bucket e escolha Confirm (Confirmar).
4. Escolha Excluir bucket. Digite o nome do bucket e escolha Confirm (Confirmar).

## Próximas etapas

Agora que você concluiu este tutorial, sugerimos analisar os seguintes recursos:

- O tutorial do [Conceitos básicos do AWS CodeStar](#) usa um projeto que cria e implanta um aplicativo web baseado em Node.js em execução em uma instância do Amazon EC2.
- [Modelos de projetos do AWS CodeStar](#) descreve outros tipos de projeto que você pode criar.



- [Como trabalhar com equipes AWS CodeStar](#) mostra como outras pessoas podem ajudá-lo a trabalhar em seus projetos.

## Tutorial: Criar um projeto no AWS CodeStar com a AWS CLI

Este tutorial mostra como usar a AWS CLI para criar um projeto do AWS CodeStar com o código-fonte de exemplo e um modelo de cadeia de ferramentas de exemplo. O AWS CodeStar provisiona a infraestrutura do AWS e os recursos do IAM especificados em um modelo de cadeia de ferramentas do AWS CloudFormation. O projeto gerencia os recursos da cadeia de ferramentas para criar e implantar o código-fonte.

O AWS CodeStar usa o AWS CloudFormation para compilar e implantar o código de exemplo. Esse código de exemplo cria um serviço da web hospedado no AWS Lambda e que pode ser acessado por meio do Amazon API Gateway.

Pré-requisitos:

- Siga as etapas em [Configuração AWS CodeStar](#).
- Você deve ter criado um bucket de armazenamento do Amazon S3. Neste tutorial, você faz upload do código-fonte de exemplo e do modelo da cadeia de ferramentas para esse local.

### Note

A conta da AWS pode ser cobrada pelos custos relacionados a este tutorial, inclusive os serviços da AWS usados pelo AWS CodeStar. Para obter mais informações, consulte [Preços do AWS CodeStar](#).

Tópicos

- [Etapa 1: Fazer download e analisar o código-fonte de exemplo](#)
- [Etapa 2: Fazer download do modelo da cadeia de ferramentas de exemplo](#)
- [Etapa 3: Testar o modelo da cadeia de ferramentas no AWS CloudFormation](#)
- [Etapa 4: Fazer upload do código-fonte e do modelo da cadeia de ferramentas](#)
- [Etapa 5: Criar um projeto no AWS CodeStar](#)

## Etapa 1: Fazer download e analisar o código-fonte de exemplo

Para este tutorial, existe um arquivo zip disponível para download. Ele contém o código-fonte de exemplo de um [aplicativo de exemplo](#) Node.js na plataforma de computação Lambda. Quando o código-fonte é colocado no repositório, a pasta e os arquivos são exibidos conforme mostrado:

```
tests/  
app.js  
buildspec.yml  
index.js  
package.json  
README.md  
template.yml
```

Os seguintes elementos do projeto são representados no código-fonte de exemplo:

- `tests/`: testes de unidade configurados para o projeto CodeBuild do projeto. Essa pasta está incluída no código de exemplo, mas não é obrigatória para criar um projeto.
- `app.js`: código-fonte do aplicativo do projeto.
- `buildspec.yml`: as instruções de compilação para o estágio de compilação do recurso CodeBuild. Esse arquivo é obrigatório para um modelo de cadeia de ferramentas com um recurso do CodeBuild.
- `package.json`: as informações de dependências do código-fonte do aplicativo.
- `README.md`: o arquivo readme do projeto incluído em todos os projetos AWS CodeStar. Esse arquivo está incluído no código de exemplo, mas não é obrigatório para criar um projeto.
- `template.yml`: o arquivo de modelo da infraestrutura ou o arquivo de modelo SAM incluído em todos os projetos AWS CodeStar. Isso é diferente de `template.yml` da cadeia de ferramentas que você enviará mais adiante neste tutorial. Esse arquivo está incluído no código de exemplo, mas não é obrigatório para criar um projeto.

## Etapa 2: Fazer download do modelo da cadeia de ferramentas de exemplo

O modelo da cadeia de ferramentas de exemplo fornecido para este tutorial cria um repositório (CodeCommit), um pipeline (CodePipeline) e um contêiner de compilações (CodeBuild), além de usar o AWS CloudFormation para implantar o código-fonte em uma plataforma Lambda. Além desses recursos, também há perfis do IAM que você pode usar para definir o escopo das permissões do ambiente do runtime, um bucket do Amazon S3 que o CodePipeline usa para armazenar os

artefatos da implantação e uma regra do CloudWatch Events usada para acionar implantações de pipeline ao enviar código para o repositório. Para se alinhar com [melhores práticas do IAM da AWS](#), restrinja as políticas das funções da cadeia de ferramentas definidas neste exemplo.

Faça download e descompacte o modelo do AWS CloudFormation de amostra no formato [YAML](#).

Quando você executar o comando `create-project` posteriormente no tutorial, este modelo criará os recursos personalizados da cadeia de ferramentas no AWS CloudFormation. Para obter mais informações sobre os recursos criados neste tutorial, consulte os seguintes tópicos no Guia do usuário do AWS CloudFormation:

- O recurso [AWS::CodeCommit::Repository](#) do AWS CloudFormation cria um repositório do CodeCommit.
- O recurso do AWS CloudFormation [AWS::CodeBuild::Project](#) cria um projeto de compilação do CodeBuild.
- O recurso do AWS CloudFormation [AWS::CodeDeploy::Application](#) cria um aplicativo do CodeDeploy.
- O recurso [AWS::CodePipeline::Pipeline](#) do AWS CloudFormation cria um pipeline do CodePipeline.
- O recurso do AWS CloudFormation [AWS::S3::Bucket](#) cria o bucket de artefatos do seu pipeline.
- O recurso do AWS CloudFormation [AWS::S3::BucketPolicy](#) cria a política de bucket de artefatos para o bucket de artefatos do seu pipeline.
- O recurso do AWS CloudFormation [AWS::IAM::Role](#) cria a função de trabalhador do CodeBuild IAM que dá permissões AWS CodeStar para gerenciar seu projeto de compilação do CodeBuild.
- O recurso do AWS CloudFormation [AWS::IAM::Role](#) cria a função de trabalhador do CodePipeline IAM que dá ao AWS CodeStar permissões para criar seu pipeline.
- O recurso do AWS CloudFormation [AWS::IAM::Role](#) cria a função de trabalhador do AWS CloudFormation IAM que dá ao AWS CodeStar permissões para criar sua pilha de recursos.
- O recurso do AWS CloudFormation [AWS::IAM::Role](#) cria a função de trabalhador do AWS CloudFormation IAM que dá ao AWS CodeStar permissões para criar sua pilha de recursos.
- O recurso do AWS CloudFormation [AWS::IAM::Role](#) cria a função de trabalhador do AWS CloudFormation IAM que dá ao AWS CodeStar permissões para criar sua pilha de recursos.
- O recurso do AWS CloudFormation [AWS::Events::Rule](#) cria a regra do CloudWatch Events que monitora seu repositório em busca de eventos push.
- O recurso do AWS CloudFormation [AWS::IAM::Role](#) cria o perfil do IAM do CloudWatch Events.

## Etapa 3: Testar o modelo da cadeia de ferramentas no AWS CloudFormation

Para fazer upload do modelo da cadeia de ferramentas, você pode testar o modelo da cadeia de ferramentas no AWS CloudFormation e solucionar todos os erros.

1. Salve o modelo atualizado no computador local e abra o console do AWS CloudFormation. Selecione Create Stack (Criar pilha). Seus novos recursos devem ser exibidos na lista.
2. Veja a pilha em busca de eventuais erros de criação.
3. Depois que o teste for concluído, exclua a pilha.

### Note

Verifique se você excluiu a pilha e todos os recursos criados no AWS CloudFormation. Caso contrário, ao criar o projeto, você poderá encontrar erros para nomes de recursos já em uso.

## Etapa 4: Fazer upload do código-fonte e do modelo da cadeia de ferramentas

Para criar um projeto do AWS CodeStar, você deve primeiro empacotar o código-fonte em um arquivo .zip e colocá-lo no Amazon S3. O AWS CodeStar inicializa o repositório com esse conteúdo. Você especifica esse local no arquivo de entrada ao executar o comando para criar o projeto na AWS CLI.

Você também deve fazer upload do arquivo `toolchain.yml` e colocá-lo no Amazon S3. Você especifica esse local no arquivo de entrada ao executar o comando para criar o projeto na AWS CLI.

Para fazer upload do código-fonte e do modelo da cadeia de ferramentas

1. A estrutura do arquivo de exemplo a seguir mostra os arquivos de origem e o modelo da cadeia de ferramentas prontos para serem compactados e enviados. O código de exemplo inclui o arquivo `template.yml`. Lembre-se de que esse arquivo é diferente do arquivo `toolchain.yml`.

```
ls
src toolchain.yml
```

```
ls src/  
README.md    app.js        buildspec.yml  index.js      package.json  
template.yml  tests
```

2. Crie o .zip dos arquivos de código-fonte.

```
cd src; zip -r "../src.zip" *; cd ../
```

3. Use o comando cp e inclua os arquivos como parâmetros.

Os comandos a seguir fazem upload do arquivo .zip e toolchain.yml para o Amazon S3.

```
aws s3 cp src.zip s3://MyBucket/src.zip  
aws s3 cp toolchain.yml s3://MyBucket/toolchain.yml
```

Para configurar o bucket do Amazon S3 a fim de compartilhar o código-fonte

- Como está armazenando o código-fonte e a cadeia de ferramentas no Amazon S3, você pode usar as políticas de bucket do Amazon S3 e as ACLs de objeto para garantir que outros usuários do IAM ou contas do AWS possam criar projetos com base nas amostras. O AWS CodeStar garante que qualquer usuário que crie um projeto personalizado tenha acesso à cadeia de ferramentas e à fonte que deseja usar.

Para permitir que qualquer pessoa use a amostra, execute os seguintes comandos:

```
aws s3api put-object-acl --bucket MyBucket --key toolchain.yml --acl public-read  
aws s3api put-object-acl --bucket MyBucket --key src.zip --acl public-read
```

## Etapa 5: Criar um projeto no AWS CodeStar

Use estas etapas para criar o projeto.

### Important

Verifique se você configurou a região do AWS preferida na AWS CLI. O projeto é criado na região do AWS configurada na AWS CLI.

## 1. Execute o comando `create-project` e inclua o parâmetro `--generate-cli-skeleton`:

```
aws codestar create-project --generate-cli-skeleton
```

Os dados formatados JSON aparecem na saída. Copie os dados para um arquivo (por exemplo, *input.json*) em um local no computador local ou em uma instância na qual a AWS CLI esteja instalada. Modifique os dados copiados da maneira a seguir e salve os resultados. Esse arquivo de entrada é configurado para um projeto chamado MyProject com um nome de bucket myBucket.

- Certifique-se de fornecer o parâmetro `roleArn`. Para modelos personalizados, como o exemplo de modelo deste tutorial, é necessário fornecer uma função. Essa função deve ter permissões para criar todos os recursos especificados em [Etapa 2: Fazer download do modelo da cadeia de ferramentas de exemplo](#).
- Verifique se você forneceu o parâmetro `ProjectId` em `stackParameters`. O modelo de exemplo fornecido para este tutorial exige esse parâmetro.

```
{
  "name": "MyProject",
  "id": "myproject",
  "description": "Sample project created with the CLI",
  "sourceCode": [
    {
      "source": {
        "s3": {
          "bucketName": "MyBucket",
          "bucketKey": "src.zip"
        }
      },
      "destination": {
        "codeCommit": {
          "name": "myproject"
        }
      }
    }
  ],
  "toolchain": {
    "source": {
      "s3": {
```

```

        "bucketName": "MyBucket",
        "bucketKey": "toolchain.yml"
    }
},
"roleArn": "role_ARN",
"stackParameters": {
    "ProjectId": "myproject"
}
}
}

```

2. Alterne para o diretório que contenha o arquivo que você acabou de salvar e execute o comando `create-project` novamente. Inclua o parâmetro `--cli-input-json`.

```
aws codestar create-project --cli-input-json file://input.json
```

3. Se bem-sucedido, dados semelhantes aos seguintes serão exibidos na saída:

```

{
  "id": "project-ID",
  "arn": "arn"
}

```

- A saída contém informações sobre o novo projeto:
    - O valor `id` representa o ID do projeto.
    - O valor `arn` representa o ARN do projeto.
4. Use o comando `describe-project` para verificar o status da criação do projeto. Inclua o parâmetro `--id`.

```
aws codestar describe-project --id <project_ID>
```

Dados semelhantes aos seguintes são exibidos na saída:

```

{
  "name": "MyProject",
  "id": "myproject",
  "arn": "arn:aws:codestar:us-east-1:account_ID:project/myproject",
  "description": "",
  "createdTimeStamp": 1539700079.472,

```

```
"stackId": "arn:aws:cloudformation:us-east-1:account_ID:stack/awscodestar-  
myproject/stack-ID",  
  "status": {  
    "state": "CreateInProgress"  
  }  
}
```

- A saída contém informações sobre o novo projeto:
  - O valor `id` representa o ID do projeto exclusivo.
  - O valor `state` representa o status da criação do projeto, como `CreateInProgress` ou `CreateComplete`.

Embora o projeto esteja sendo criado, você pode [adicionar membros à equipe](#) ou [configurar o acesso](#) ao repositório do projeto na linha de comando ou de seu IDE favorito.

## Tutorial: Criar um projeto de habilidade do Alexa no AWS CodeStar

O AWS CodeStar é um serviço de desenvolvimento baseado em nuvem no AWS que fornece as ferramentas necessárias para desenvolver, compilar e implantar rapidamente aplicativos no AWS. Com o AWS CodeStar, é possível configurar toda a sua cadeia de ferramentas de entrega contínua em alguns minutos, o que permite agilizar o lançamento de código. Os modelos de projeto de habilidade do Alexa no AWS CodeStar permitem que você crie uma simples habilidade "Hello World" do Alexa diretamente de sua conta do AWS com apenas alguns cliques. Os modelos também criam um pipeline de implantação básico para você começar a usar um fluxo de integração contínua (CI) para o desenvolvimento de habilidades.

Os principais benefícios da criação de habilidades do Alexa com o AWS CodeStar são a possibilidade de você começar a usar o desenvolvimento de habilidades no AWS e conectar sua conta de desenvolvedor da Amazon ao projeto para implantar habilidades para o estágio de desenvolvimento diretamente do AWS. Você também pode obter um pipeline de implantação (CI) pronto para uso com um repositório com todo o código-fonte para o projeto. Você pode configurar esse repositório com seu IDE preferencial para criar habilidades com as ferramentas que já conhece.

### Pré-requisitos


- Crie uma conta de desenvolvedor da Amazon acessando <https://developer.amazon.com>. O cadastro é gratuito. Esta conta é proprietária de suas habilidades do Alexa.



- Se você ainda não tiver uma conta da AWS, use o procedimento a seguir para criar uma.


Para cadastrar-se na AWS

1. Abra <https://aws.amazon.com/> e selecione Criar uma conta AWS.

 Note

Se você fez login no AWS Management Console usando credenciais do Usuário raiz da conta da AWS, escolha Sign in to a different account (Fazer login em uma conta diferente). Se você fez login no console usando as credenciais do IAM, selecione Fazer login usando credenciais da Usuário raiz da conta da AWS. Em seguida, escolha Criar uma nova conta do AWS.

2. Siga as instruções online.

 Important

Depois de criar o projeto de habilidade do Alexa, faça todas as edições somente no repositório do projeto. Recomendamos que você não edite essa habilidade diretamente usando nenhuma outra ferramenta do Alexa Skills Kit (ASK), como a CLI do ASK ou o console do desenvolvedor do ASK. Essas ferramentas não estão integradas ao repositório do projeto. Usá-las fará com que a habilidade e o código de repositório percam a sincronia.

## Etapa 1: Criar o projeto e conectar sua conta de desenvolvedor da Amazon

Neste tutorial, você cria uma habilidade usando o Node.js em execução no AWS Lambda. A maioria das etapas são as mesmas para outros idiomas, embora o nome da habilidade seja diferente. Consulte o arquivo README.md no repositório do projeto para obter detalhes específicos do modelo de projeto que você escolher.

1. Faça login no AWS Management Console e então abra o console do AWS CodeStar em <https://console.aws.amazon.com/codestar/>.
2. Escolha a região da AWS onde você deseja criar o projeto e seus recursos. O runtime da skill da Alexa está disponível nas seguintes regiões do AWS:
  - Ásia-Pacífico (Tóquio)

- UE (Irlanda)
  - Leste dos EUA (N. da Virgínia)
  - Oeste dos EUA (Oregon)
3. Escolha Create project (Criar projeto).
  4. Na página Choose a project template (Escolher um modelo de projeto):
    - a. Em Categoria de aplicativo, selecione Alexa Skill.
    - b. Em Linguagem de programação, selecione Node.js.
  5. Escolha a caixa que contém suas seleções.
  6. Em Project name (Nome do projeto), digite o nome do projeto (por exemplo, **My Alexa Skill**). Se você usar um nome diferente, certifique-se de usá-lo ao longo de todo este tutorial. O AWS CodeStar escolhe um identificador relacionado para o Project ID (ID do projeto) do projeto (por exemplo, minha-habilidade-do-alexa). Se vir outro ID de projeto, certifique-se de usá-lo durante todo este tutorial.
  7. Selecione AWS CodeCommit para o repositório neste tutorial e não altere o valor Repository name (Nome do repositório).
  8. Selecione Connect Amazon developer account (Conectar conta de desenvolvedor da Amazon) para vincular sua conta de desenvolvedor da Amazon para hospedar a habilidade. Se você não tiver uma conta de desenvolvedor da Amazon, crie uma conta e conclua primeiro o registro com a [Amazon Developers](#).
  9. Faça login com suas credenciais de desenvolvedor da Amazon. Escolha Permitir e, em seguida, escolha Confirmar para concluir a conexão.
  10. Se você tiver vários IDs de fornecedor associados à sua conta de desenvolvedor da Amazon, selecione aquele que deseja usar para este projeto. Certifique-se de usar uma conta com a função de administrador ou desenvolvedor atribuída.
  11. Escolha Next (Próximo).
  12. (Opcional) Se esta for a primeira vez que você usa o AWS CodeStar nessa região da AWS, insira o nome de exibição e o endereço de e-mail que o AWS CodeStar deve usar para o usuário do IAM. Escolha Next (Próximo).
  13. Aguarde enquanto o AWS CodeStar cria o projeto. Isso pode demorar vários minutos. Não continue até ver o banner provisionado do projeto.

## Etapa 2: Testar sua habilidade no Simulador do Alexa

Na primeira etapa, o AWS CodeStar criou uma habilidade para você e a implantou no estágio de desenvolvimento de habilidades do Alexa. Em seguida, você testará a habilidade no Simulador do Alexa.

1. Em seu projeto no console do AWS CodeStar, escolha Exibir aplicativo. Uma nova guia será aberta no Simulador do Alexa.
2. Faça login com suas credenciais de desenvolvedor da Amazon da conta com a qual você se conectou ao seu projeto na Etapa 1.
3. Em Test (Testar), selecione Development (Desenvolvimento) para habilitar os testes.
4. Digite `ask hello node hello`. O nome de invocação padrão para a habilidade é `hello node`.
5. Sua habilidade deve responder `Hello World!`.

Quando a habilidade estiver habilitada no Simulador do Alexa, você também poderá invocá-la em um dispositivo habilitado para o Alexa que estiver registrado em sua conta de desenvolvedor da Amazon. Para testar a habilidade em um dispositivo, diga Alexa, peça ao nó "hello" para dizer "hello".

Para obter mais informações sobre o Simulador do Alexa, consulte [Testar sua habilidade no console do desenvolvedor](#).

## Etapa 3: Explorar os recursos do projeto

Como parte da criação do projeto, o AWS CodeStar também criou os recursos do AWS em seu nome. Esses recursos incluem um repositório do projeto usando o CodeCommit, um pipeline de implantação usando o CodePipeline e uma função AWS Lambda. Você pode acessar esses recursos na barra de navegação. Por exemplo, escolher Repositório mostra detalhes sobre o repositório do CodeCommit. Você pode ver o status de implantação do pipeline na página Pipeline. Você pode visualizar uma lista completa de recursos do AWS criados como parte de seu projeto selecionando Visão geral na barra de navegação à esquerda. Essa lista inclui links para cada recurso.

## Etapa 4: Fazer uma alteração na resposta da habilidade

Nesta etapa, faça uma pequena alteração na resposta de sua habilidade para compreender o ciclo de iteração.

1. Na barra de navegação, escolha Repositório. Escolha o link em Nome do repositório e o repositório do seu projeto será aberto em uma nova guia ou janela. Esse repositório contém a especificação de compilação (buildspec.yml), a pilha de aplicativos do AWS CloudFormation (template.yml), o arquivo readme e o código-fonte de sua habilidade no [formato do pacote da habilidade \(estrutura do projeto\)](#).
2. Navegue até o arquivo lambda > personalizar > index.js (no caso do Node.js.). Esse arquivo contém o código de manuseio da solicitação, que usa o [ASK SDK](#).
3. Escolha Editar.
4. Substitua a string Hello World! na linha 24 pela string Hello. How are you?.
5. Role para baixo até o final do arquivo. Insira o nome do autor e o endereço de e-mail, bem como uma mensagem opcional de confirmação.
6. Selecione Commit changes (Confirmar alterações) para confirmar as alterações ao repositório.
7. Retorne ao projeto em AWS CodeStar e confira a página do Pipeline. Você agora deverá ver a implantação do pipeline.
8. Quando a implantação do pipeline for concluída, teste a habilidade de novamente no Simulador do Alexa. A habilidade deve responder Hello. How are you?.

## Etapa 5: Configurar sua estação de trabalho local para se conectar ao repositório do projeto

Anteriormente, foi feita uma pequena alteração no código-fonte diretamente do console do CodeCommit. Nesta etapa, você configura o repositório do projeto com sua estação de trabalho local, para que possa editar e gerenciar o código na linha de comando ou no seu IDE favorito. As etapas a seguir explicam como configurar as ferramentas de linha de comando.

1. Navegue até o painel do projeto no AWS CodeStar, se necessário.
2. Na barra de navegação, escolha IDE.
3. Em Acessar o código do seu projeto, exibir as instruções abaixo da interface da linha de comando.
4. Siga as instruções para concluir as tarefas a seguir:
  - a. Instale o Git em sua estação de trabalho local a partir de um site, como [Git Downloads](#).
  - b. Instale a CLI do AWS. Para obter informações, consulte [Instalar a interface de linha de comando da AWS](#).
  - c. Configure a CLI do AWS com sua chave de acesso do usuário do IAM e a chave secreta. Para obter informações, consulte [Configurar a CLI do AWS](#).

- d. Clone o repositório do CodeCommit do projeto em sua estação de trabalho local. Para obter instruções, consulte [Conectar-se a um repositório do CodeCommit](#).

## Próximas etapas

Este tutorial mostrou como começar com uma habilidade básica. Para continuar sua jornada de desenvolvimento de habilidades, consulte os recursos a seguir.

- Entenda os conceitos básicos de uma habilidade assistindo a [Como o Alexa funciona](#) e outros vídeos no canal de desenvolvedores do Alexa no YouTube.
- Entenda os vários componentes da sua habilidade, examinando a documentação do [formato de pacote da habilidade](#), os [esquemas do manifesto da habilidade](#), e os [esquemas do modelo de interação](#).
- Transforme sua ideia em uma habilidade examinando a documentação do [Alexa Skills Kit](#) e os [ASK SDKs](#).

## Tutorial: Criar um projeto com um repositório de origem do GitHub

Com o AWS CodeStar, você pode configurar seu repositório para criar, revisar e mesclar pull requests com sua equipe de projeto.

Neste tutorial, você cria um projeto com um exemplo de código-fonte do aplicativo web em um repositório do GitHub, um pipeline que implanta suas alterações e instâncias do EC2 em que seu aplicativo está hospedado na nuvem. Depois que seu projeto for criado, este tutorial mostra como criar e mesclar uma pull request do GitHub que faz uma alteração na página inicial do seu aplicativo web.

### Tópicos

- [Etapa 1: Criar o projeto e criar seu repositório do GitHub](#)
- [Etapa 2: exibir seu código-fonte](#)
- [Etapa 3: Criar uma pull request do GitHub](#)

## Etapa 1: Criar o projeto e criar seu repositório do GitHub

Nesta etapa, use o console para criar seu projeto e criar uma conexão com seu novo repositório do GitHub. Para acessar seu repositório do GitHub, você cria um recurso de conexão que o AWS

CodeStar usa para gerenciar a autorização com o GitHub. Quando o projeto é criado, seus recursos adicionais são provisionados para você.

1. Faça login no AWS Management Console e então abra o console do AWS CodeStar em <https://console.aws.amazon.com/codestar/>.
2. Escolha a região da AWS onde você deseja criar o projeto e seus recursos.
3. Na página AWS CodeStar, escolha Criar um projeto.
4. Na página Escolha um modelo de projeto, marque as caixas de seleção Aplicativo Web, Node.js e Amazon EC2. Em seguida, escolha entre os modelos disponíveis para esse conjunto de opções.

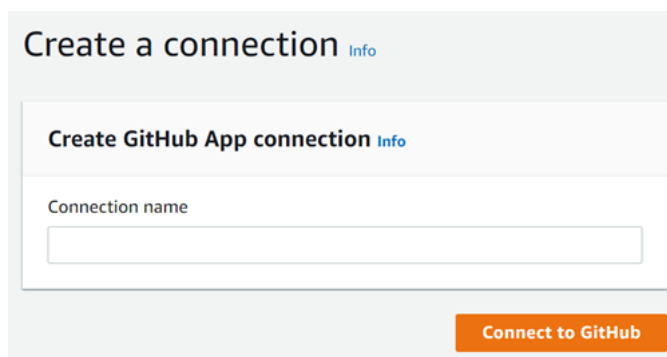
Para obter mais informações, consulte [Modelos de projetos do AWS CodeStar](#).

5. Escolha Next (Próximo).
6. Em Project name (Nome do projeto), digite o nome do projeto (por exemplo, **MyTeamProject**). Se você escolher outro nome, certifique-se de usá-lo durante todo este tutorial.
7. Em Repositório do projeto, escolha GitHub.
8. Se você escolheu o GitHub, precisará escolher ou criar um recurso de conexão. Se você tiver uma conexão existente, escolha-a no campo de pesquisa. Caso contrário, você criará uma nova conexão aqui. Escolha Conectar-se ao GitHub.

A página Criar uma conexão é exibida.

#### Note

Para criar uma conexão, você deve ter uma conta do GitHub. Se você estiver criando uma conexão para uma organização, você deve ser o proprietário da organização.



Create a connection [Info](#)

Create GitHub App connection [Info](#)


Connection name

[Connect to GitHub](#)

- a. Em Criar conexão do App GitHub, em Nome da conexão, insira um nome para sua conexão. Escolha Conectar-se ao GitHub.


A página de Conectar ao GitHub exibe e mostra o campo Apps do GitHub

- b. Em GitHub apps (Aplicações do GitHub), escolha uma instalação de aplicação ou Install a new app (Instalar uma nova aplicação) para criar uma.

 Note

Você instala uma aplicação para todas as suas conexões com um provedor específico. Se você já instalou a aplicação AWS Connector para GitHub, selecione-a e ignore esta etapa.

- c. Na página de instalação do Connect para GitHub, selecione a conta na qual deseja instalar a aplicação.

 Note

Se você instalou a aplicação anteriormente, poderá escolher Configure (Configurar) para prosseguir para uma página de modificação para a instalação da aplicação ou usar o botão Back (Voltar) para retornar ao console.

- d. Se a página Confirme a senha para continuar for exibida, digite sua senha do GitHub e escolha Entrar.
- e. Na página Instalar o AWS Connector para GitHub, mantenha os valores padrão e selecione Instalar.
- f. Na página Conectar ao GitHub o ID de instalação da sua nova instalação é exibido em Aplicativos do GitHub.

Depois que a conexão for criada com sucesso, na página de criação do projeto do CodeStar, a mensagem Pronto para conectar é exibida.


### Note

É possível visualizar sua conexão em Configurações no console do Developer Tools. Para obter mais informações, consulte o tópico de [Conceitos básicos sobre conexões](#).

Select a repository provider


CodeCommit

Use a new AWS CodeCommit repository for your project.



GitHub

Use a new GitHub source repository for your project (requires an existing GitHub account).



**The GitHub repository provider now uses CodeStar Connections**

To use a GitHub repository in CodeStar, create a connection. The connection will use GitHub Apps to access your repository. Use the following options to choose an existing connection or create a new one. [Learn more](#)

Connection

Choose an existing connection or create a new one and then return to this task.

or

✔ **Ready to connect**

Your Github connection is ready for use.

Repository owner

The owner of the new repository. This can be a personal GitHub account or a GitHub organization.

Repository name

The name of the new repository.

Repository description

An optional description of the new repository.

Public

- g. Em Proprietário do repositório, escolha a organização do GitHub ou a conta do GitHub pessoal.
- h. Para Repository name (Nome do repositório), aceite o nome do repositório do GitHub padrão ou digite um outro.
- i. Escolha Público ou Privado.



**Note**

Se você desejar usar o AWS Cloud9 como ambiente de desenvolvimento, será necessário selecionar um repositório público.

- j. (Opcional) Em Repository description (Descrição do repositório), digite uma descrição para o repositório do GitHub.
9. Configure suas instâncias do Amazon EC2 na Configuração do Amazon EC2 se seu projeto estiver implantado em instâncias do Amazon EC2 e você quiser fazer alterações. Por exemplo, você pode escolher entre os tipos de instância disponíveis para o projeto.

Em Par de chaves, escolha o par de chaves do Amazon EC2 que você criou em [Etapa 4: criar um par de chaves do Amazon EC2 para projetos do AWS CodeStar](#). Selecione Eu reconheço que tenho acesso ao arquivo de chave privada.

10. Escolha Next (Próximo).
11. Analise os recursos e os detalhes da configuração.
12. Escolha Avançar ou Criar projeto. (A opção exibida depende do modelo do projeto.)

Aguarde alguns minutos enquanto seu projeto é criado.

13. Depois que seu projeto for criado, escolha Exibir aplicativo para visualizar seu aplicativo web.

## Etapa 2: exibir seu código-fonte

Nesta etapa, você visualiza seu código-fonte e as ferramentas que você pode usar para seu repositório de origem.

1. Na barra de navegação de seu projeto, escolha Repositório.

Para ver uma lista de commits no GitHub, escolha Exibir commits. Isso abre seu histórico de commits no GitHub.

Para ver os problemas, escolha a guia Problemas do seu projeto. Para criar um novo problema no GitHub, escolha Criar problema no GitHub. Isso abre o formulário de problema do repositório no GitHub.

2. Na guia Repositório, escolha o link em Nome do repositório e o repositório do seu projeto será aberto em uma nova guia ou janela. Esse repositório contém o código-fonte do seu projeto.

## Etapa 3: Criar uma pull request do GitHub

Nesta etapa, você faz uma pequena alteração no código-fonte e cria uma pull request.

1. No GitHub, crie uma nova ramificação de atributos no seu repositório. Escolha o campo suspenso da ramificação principal e insira uma nova ramificação no campo chamado `feature-branch`. Escolha Criar nova ramificação. A ramificação é criada e verificada para você.
2. No GitHub, faça uma alteração na ramificação `feature-branch`. Abra a pasta pública e abra o arquivo `index.html`.
3. No console AWS CodeStar, em Pull requests, para criar uma pull request no GitHub, escolha Criar pull request. Isso abre o formulário de problema do repositório no GitHub. No GitHub, selecione o ícone de lápis para editar o arquivo.

Depois de `Congratulations!`, adicione a string `Well done, <name>!` e substitua `<name>` pelo seu nome. Escolha Commit changes (Confirmar alterações). A alteração está comprometida com sua ramificação de atributos.

4. No console do AWS CodeStar, escolha seu projeto. Escolha a guia Repositório. Em Solicitações pull, escolha Criar solicitação pull.

O formulário é aberto no GitHub. Deixe o galho principal no galho base. Em Comparar com, escolha sua ramificação de atributos. Exiba a diferença.

5. No GitHub, escolha Create pull request. Uma pull request chamada Update index.html é criada.
6. No console do AWS CodeStar, visualize o novo pull request. Escolha Mesclar alterações para confirmar as alterações no repositório e mesclar a pull request com a ramificação principal do seu repositório.
7. Retorne ao projeto em AWS CodeStar e confira a página do Pipeline. Você agora deverá ver a implantação do pipeline.
8. Depois que seu projeto for criado, escolha Exibir aplicativo para visualizar seu aplicativo web.

# Modelos de projetos do AWS CodeStar

Os modelos de projeto do AWS CodeStar permitem começar com um aplicativo de exemplo e implantá-lo usando os recursos do AWS criados para oferecer suporte ao projeto de desenvolvimento. Quando você escolhe um modelo de projeto do AWS CodeStar, o tipo de aplicativo, a linguagem de programação e a plataforma de computação são provisionados para você. Depois de criar projetos com aplicativos web, serviços web, habilidades Alexa e páginas estáticas da Web, você poderá substituir o aplicativo de exemplo pelos próprios.

Depois que o AWS CodeStar cria o projeto, você pode modificar os recursos do AWS que oferecem suporte à entrega do aplicativo. O AWS CodeStar funciona com AWS CloudFormation para permitir que você use o código para criar serviços de suporte e servidores/plataformas de tecnologia sem servidor na nuvem. O AWS CloudFormation permite que você modele toda a infraestrutura em um arquivo de texto.

## Tópicos

- [Arquivos de projeto e recursos do AWS CodeStar](#)
- [Primeiros passos: escolher um modelo de projeto](#)
- [Como fazer alterações no projeto do AWS CodeStar](#)

## Arquivos de projeto e recursos do AWS CodeStar

Um projeto do AWS CodeStar é uma combinação de código-fonte e os recursos criados para implantar o código. A coleção de recursos que ajudam a compilar, lançar e implantar o código são chamados de recursos da cadeia de ferramentas. Na criação do projeto, um modelo do AWS CloudFormation provisiona os recursos da cadeia de ferramentas em um pipeline de integração/implantação contínuas (CI/CD).

Você pode usar o AWS CodeStar para criar projetos de duas formas, dependendo do seu nível de experiência com a criação de recursos do AWS:

- Quando você usa o console para criar um projeto, o AWS CodeStar cria sua cadeia de ferramentas de recursos, incluindo seu repositório e preenche seu repositório com exemplos de código de aplicativo e arquivos de projeto. Use o console para configurar rapidamente projetos de amostra com base em um conjunto de opções de projeto pré-configurado.

- Ao usar a CLI para criar um projeto, forneça o modelo do AWS CloudFormation que cria os recursos da cadeia de ferramentas e o código-fonte do aplicativo. Use a CLI para permitir que o AWS CodeStar crie seu projeto com base no modelo e, em seguida, preencher seu repositório com seu código de exemplo.

Um projeto do AWS CodeStar fornece um ponto único de gerenciamento. Você pode usar o assistente Criar projeto no console para configurar um projeto de exemplo. Em seguida, você pode usá-lo como uma plataforma de colaboração para gerenciar permissões e recursos para a equipe. Para obter mais informações, consulte [O que é o AWS CodeStar?](#). Quando você usa o console para criar um projeto, o código-fonte é fornecido como um código de exemplo e os recursos da cadeia de ferramentas CI/CD são criados para você

Quando você cria um projeto no console, o AWS CodeStar provisiona os seguintes recursos:

- Um repositório de código no GitHub ou no CodeCommit.
- No repositório do projeto, um arquivo README .md que dá detalhes dos arquivos e diretórios.
- No repositório do projeto, um arquivo `template.yml` que armazena a definição da pilha do tempo de execução do aplicativo. Você pode usar esse arquivo para adicionar ou modificar recursos do projeto que não sejam recursos da cadeia de ferramentas, como recursos do AWS usados para notificações, suporte ao banco de dados, monitoramento e rastreamento.
- Os serviços e os recursos do AWS criados com o pipeline, como o bucket de artefatos do Amazon S3, o Amazon CloudWatch Events e os perfis de serviço relacionados.
- Um aplicativo de exemplo funcional com código-fonte completo e um endpoint HTTP público.
- Um recurso de computação do AWS, com base no tipo de modelo de projeto do AWS CodeStar:
  - Uma função Lambda.
  - Uma instância do Amazon EC2.
  - Um ambiente do AWS Elastic Beanstalk.
- A partir de 6 de dezembro de 2018 (horário do Pacífico):
  - Um limite de permissões que é uma política do IAM especializada para controlar o acesso aos recursos do projeto. Por padrão, o limite de permissões é anexado às funções no projeto de exemplo. Para obter mais informações, consulte [Limite de permissões do IAM para funções de operador](#).

- Um perfil do IAM do AWS CloudFormation para criar recursos do projeto usando o AWS CloudFormation que inclui permissões para todos os recursos com suporte do AWS CloudFormation, incluindo perfis do IAM.
- Uma função do IAM para cadeia de ferramentas.
- Funções de execução do Lambda definidas na pilha de aplicativos, que você pode modificar.
- Antes de 6 de dezembro de 2018 (horário do Pacífico):
  - Uma função do IAM para o AWS CloudFormation para a criação de recursos do projeto com suporte para um conjunto limitado de recursos do AWS CloudFormation.
  - Um perfil do IAM para criação de um recurso do CodePipeline.
  - Um perfil do IAM para criação de um recurso do CodeBuild.
  - Um perfil do IAM para criação de um recurso do CodeDeploy, se aplicável ao tipo do projeto.
  - Um perfil do IAM para criação do aplicativo web do Amazon EC2, se aplicável ao tipo do projeto.
  - Um perfil do IAM para criar um recurso do CloudWatch Events.
  - Uma função de execução para o Lambda modificada dinamicamente para incluir um conjunto parcial de recursos.

O projeto inclui páginas de detalhes que mostram o status e contêm links para gerenciamento de equipe, links para instruções de configuração para IDEs ou o repositório e um histórico de confirmação de código-fonte é alterado no repositório. Você também pode selecionar ferramentas para se conectar com ferramentas de rastreamento de problemas externos, como o Jira.

## Primeiros passos: escolher um modelo de projeto

Quando escolhe um projeto do AWS CodeStar no console, você está escolhendo de um conjunto de opções pré-configuradas com código de exemplo e recursos para dar rapidamente os primeiros passos. Essas opções são chamadas de modelos de projeto. Cada modelo de projeto do AWS CodeStar consiste em uma linguagem de programação, um tipo de aplicativo e uma plataforma de computação. A combinação que você seleciona determina o modelo de projeto.

## Escolha uma plataforma de computação de modelo

Cada modelo configura um dos seguintes tipos de plataforma de computação:

- Ao escolher um projeto do AWS Elastic Beanstalk, você implanta em um ambiente do AWS Elastic Beanstalk em instâncias do Amazon Elastic Compute Cloud na nuvem.

- Quando você escolhe um projeto do Amazon EC2, o AWS CodeStar cria instâncias do Linux EC2 para hospedar o aplicativo na nuvem. Os membros da equipe do projeto podem acessar as instâncias e a equipe usa o par de chaves que você fornece para o SSH em suas instâncias do Amazon EC2. O AWS CodeStar também tem um SSH gerenciado que usa permissões do membro da equipe para gerenciar conexões de pares de chaves.
- Quando você escolhe AWS Lambda, o AWS CodeStar cria um ambiente de tecnologia sem servidor acessado por meio do Amazon API Gateway, sem instâncias ou servidores para manter.

## Escolha um tipo de aplicativo de modelo

Cada modelo configura um dos seguintes tipos de aplicativo:

- Serviço web

Um serviço web é usado em tarefas executadas em segundo plano, como chamar APIs. Depois que o AWS CodeStar cria o projeto de serviço da web de exemplo, você poderá escolher o URL do endpoint para ver a saída "Hello World", mas o uso principal desse tipo de aplicativo não é como uma interface de usuário (UI). Os modelos de projeto do AWS CodeStar nessa categoria oferecem suporte ao desenvolvimento em Ruby, Java, ASP.NET, PHP, Node.js e muito mais.

- Aplicativo Web

Um aplicativo web tem uma interface do usuário. Depois que o AWS CodeStar cria o projeto de aplicativo web de exemplo, você pode escolher o URL do endpoint para ver um aplicativo web interativo. Os modelos de projeto do AWS CodeStar nessa categoria oferecem suporte ao desenvolvimento em Ruby, Java, ASP.NET, PHP, Node.js e muito mais.

- Página da web estática

Escolha esse modelo se você quiser um projeto para um site HTML. Os modelos de projeto do AWS CodeStar nessa categoria oferecem suporte ao desenvolvimento em HTML5.

- Habilidade do Alexa

Selecione esse modelo se quiser um projeto para uma habilidade do Alexa com uma função do AWS Lambda. Ao criar o projeto de habilidade, o AWS CodeStar retornará um nome do recurso da Amazon (ARN) que pode ser usado como um endpoint do serviço. Para obter mais informações, consulte [Hospedar uma habilidade personalizada como uma função do AWS Lambda](#).

**Note**

As funções do Lambda para as habilidades do Alexa são compatíveis somente nas regiões Leste dos EUA (Norte da Virgínia), Oeste dos EUA (Oregon), UE (Irlanda) e Ásia-Pacífico (Tóquio).

- Regra do Config

Escolha esse modelo caso queira um projeto para uma regra do AWS Config que permite que você automatize regras em recursos do AWS em sua conta. A função retorna um ARN que você pode usar como um endpoint de serviço para a regra.

## Escolha um modelo de linguagem de programação

Ao escolher um modelo de projeto, você seleciona uma linguagem de programação, como Ruby, Java, ASP.NET, PHP, Node.js e muito mais.

## Como fazer alterações no projeto do AWS CodeStar

Você pode atualizar o projeto modificando:

- Código de exemplo e recursos de linguagem de programação para o aplicativo.
- Os recursos que compõem a infraestrutura na qual o aplicativo é armazenado e implantado (sistemas operacionais, aplicativos e serviços de suporte, parâmetros de implantação, e a plataforma de computação em nuvem). Você pode modificar recursos de aplicativos no arquivo `template.yml`. Este é o arquivo AWS CloudFormation que modela o ambiente de tempo de execução do aplicativo.

**Note**

Se estiver trabalhando com um projeto do AWS CodeStar de habilidades do Alexa, não é possível fazer alterações na habilidade fora do repositório de origem do AWS CodeStar (CodeCommit ou GitHub). Se você editar a habilidade no portal do desenvolvedor do Alexa,

talvez a alteração não esteja visível no repositório de origem e as duas versões ficarão fora de sincronia.

## Alterar o código-fonte do aplicativo e enviar alterações por push

Para modificar o código-fonte de exemplo, scripts e outros arquivos de origem do aplicativo, edite os arquivos no repositório de origem por:

- Usar o modo de edição no CodeCommit ou no GitHub.
- Abrir o projeto em um IDE, como AWS Cloud9.
- Clonar o repositório localmente e confirmar e enviar as alterações. Para obter mais informações, consulte [Etapa 4: Confirmar uma alteração](#).

## Alterar recursos do aplicativo com o arquivo `template.yml`

Em vez de modificar manualmente um recurso de infraestrutura, use AWS CloudFormation para modelar e implantar os recursos de tempo de execução do aplicativo.

Você pode modificar ou adicionar um recurso de aplicativo, como uma função do Lambda, na pilha de tempo de execução, editando o arquivo `template.yml` no repositório do projeto. Você pode adicionar qualquer recurso que esteja disponível como um recurso do AWS CloudFormation.

Para alterar o código ou as configurações de uma função do AWS Lambda, consulte [Adicionar um recurso a um projeto](#).

Modifique o arquivo `template.yml` no repositório do projeto para adicionar o tipo de recursos do AWS CloudFormation que são recursos de aplicativos. Quando você adiciona um recurso de aplicativo à seção `Resources` do arquivo `template.yml`, o AWS CloudFormation e o AWS CodeStar criam o recurso para você. Para obter uma lista de recursos do AWS CloudFormation e suas propriedades obrigatórias, consulte [Referência de tipos de recurso do AWS](#). Para obter mais informações, consulte este exemplo em [Etapa 1: editar a função de operador do CloudFormation no IAM](#).

O AWS CodeStar permite que você implemente as melhores práticas ao configurar a modelagem e o ambiente de tempo de execução do aplicativo.



## Como gerenciar permissões para alterar recursos do aplicativo

Quando você usa o AWS CloudFormation para adicionar recursos de aplicativo de tempo de execução, como uma função do Lambda, a função de operador do AWS CloudFormation pode usar as permissões que ela já tem. Para alguns recursos de aplicativo em tempo de execução, você deve ajustar manualmente as permissões da função de operador do AWS CloudFormation antes de editar o arquivo `template.yml`.

Para obter um exemplo de como alterar as permissões da função de operador do AWS CloudFormation, consulte [Etapa 5: adicionar permissões de recursos com uma política em linha](#).

# Melhores práticas do AWS CodeStar

O AWS CodeStar é integrado a uma série de produtos e serviços. As seções a seguir descrevem as melhores práticas para o AWS CodeStar e os produtos e serviços correspondentes.

## Tópicos

- [Melhores práticas de segurança para recursos do AWS CodeStar](#)
- [Melhores práticas a fim de configurar versões para dependências](#)
- [Monitoramento e registro em log das melhores práticas para recursos do AWS CodeStar](#)

## Melhores práticas de segurança para recursos do AWS CodeStar

Você deve aplicar patches regularmente e revisar as melhores práticas de segurança para as dependências usadas pelo aplicativo. Use essas melhores práticas de segurança para atualizar o código de exemplo e manter o projeto em um ambiente de produção:

- Acompanhe anúncios de segurança e atualizações contínuas para a estrutura.
- Para implantar o projeto, siga as melhores práticas desenvolvidas para a estrutura.
- Analise as dependências da estrutura regularmente e atualize conforme necessário.
- Cada modelo do AWS CodeStar contém instruções de configuração para a linguagem de programação. Consulte o arquivo README .md no repositório de origem do projeto.
- Como prática recomendada para isolar os recursos do projeto, gerencie o acesso com privilégio mínimo aos recursos do AWS usando uma estratégia de várias contas, conforme apresentado em [Segurança em AWS CodeStar](#).

## Melhores práticas a fim de configurar versões para dependências

O código-fonte de exemplo no projeto do AWS CodeStar usa dependências listadas no arquivo package .json no repositório de origem. Como melhores práticas, defina sempre as dependências a fim de apontar para uma versão específica. Isso é conhecido como fixar a versão. Não recomendamos definir a versão como latest porque isso pode introduzir alterações que podem travar o aplicativo sem aviso prévio.

# Monitoramento e registro em log das melhores práticas para recursos do AWS CodeStar

Você pode usar atributos de registro na AWS para determinar as ações que os usuários executaram em sua conta e os atributos que foram usados. Os arquivos de log mostram:

- A data e hora das ações.
- O endereço IP de origem de uma ação.
- As ações que falharam devido a permissões inadequadas.

O AWS CloudTrail pode ser usado para registrar em log chamadas de API do AWS e eventos relacionados realizados por uma conta do AWS ou em nome dessa conta. Para obter mais informações, consulte [Log de chamadas de API do AWS CodeStar com o AWS CloudTrail](#).

# Como trabalhar com projetos no AWS CodeStar

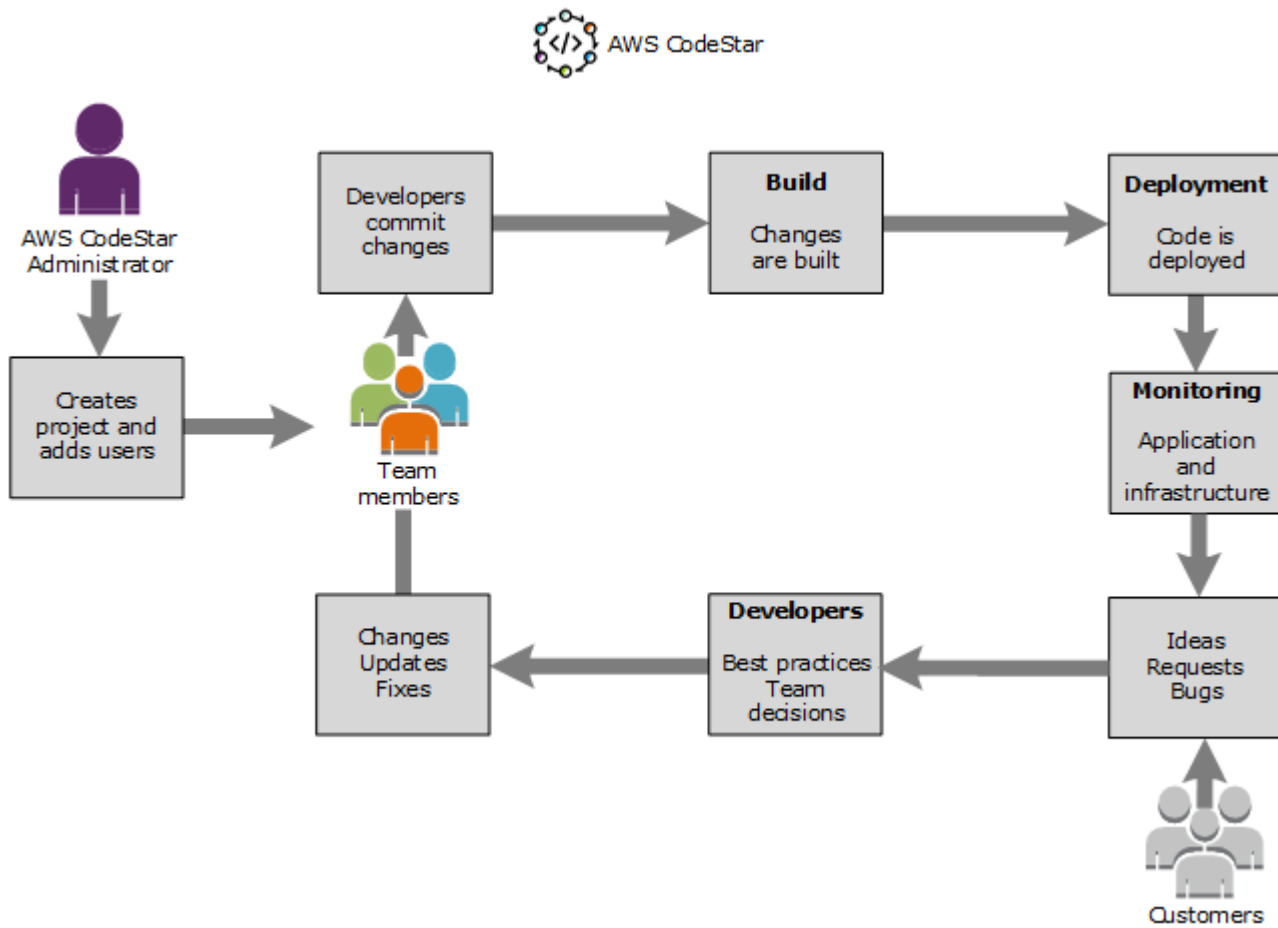
Quando usa um modelo de projeto do AWS CodeStar, você pode criar rapidamente um projeto que já está configurado com os recursos necessários incluindo:

- Repositório de código-fonte
- Ambiente de compilação
- Recursos de hospedagem e compilação
- Linguagem de programação

O modelo inclui até exemplos de código-fonte para que você possa começar a trabalhar com o projeto imediatamente.

Depois que você tiver um projeto, você poderá adicionar ou remover recursos, personalizar o painel do projeto e monitorar o andamento.

O diagrama a seguir mostra um fluxo de trabalho básico em um projeto do AWS CodeStar.



O fluxo de trabalho básico no diagrama mostra um desenvolvedor com a política do `AWSCodeStarFullAccess` aplicada que cria um projeto e adiciona os membros da equipe ao projeto. Juntos, eles escrevem, compilam, testam e implantam código. O painel do projeto fornece ferramentas que podem ser usadas em tempo real para visualizar a atividade do aplicativo e monitorar as compilações, o fluxo do código por meio do pipeline de implantação e muito mais. A equipe usa o bloco de wiki da equipe para compartilhar informações, práticas recomendadas e links. Eles integram o software de rastreamento de problemas do software para ajudar a acompanhar o andamento e as tarefas. À medida que os clientes fornecem solicitações e comentários, a equipe adiciona essas informações ao projeto e as integra no planejamento e no desenvolvimento do projeto. À medida que o projeto cresce, a equipe adiciona mais membros à equipe para oferecer suporte à base de código.

# Criar um projeto no AWS CodeStar

Para usar o console do AWS CodeStar para criar um projeto. Se você usar um modelo de projeto, ele configurará os recursos necessários para você. O modelo também inclui código de exemplo que você pode usar para começar a codificação.

Para criar um projeto, faça login no AWS Management Console com um usuário do IAM que tenha a política `AWSCodeStarFullAccess` ou permissões equivalentes. Para obter mais informações, consulte [Configuração AWS CodeStar](#).

## Note

Você deve concluir as etapas no [Configuração AWS CodeStar](#) antes que possa concluir os procedimentos neste tópico.

## Tópicos

- [Criar um projeto no AWS CodeStar \(console\)](#)
- [Criar um projeto no AWS CodeStar \(AWS CLI\)](#)

## Criar um projeto no AWS CodeStar (console)

Use o console do AWS CodeStar para criar um projeto.

Para criar um projeto no AWS CodeStar

1. Faça login no AWS Management Console e então abra o console do AWS CodeStar em <https://console.aws.amazon.com/codestar/>.

Verifique se você está conectado à região da AWS onde você deseja criar o projeto e seus recursos. Por exemplo, para criar um projeto em no Leste dos EUA (Ohio), verifique se selecionou essa região do AWS. Para obter informações sobre as regiões da AWS onde o AWS CodeStar está disponível, consulte [Regiões e endpoints](#) na Referência geral da AWS.

2. Na página AWS CodeStar, escolha Criar um projeto.
3. Na página Escolher um modelo de projeto, escolha o tipo de projeto na lista de modelos de projeto do AWS CodeStar. Você pode usar a barra de filtro estreitar suas opções. Por exemplo, para um projeto de aplicativo web escrito em Node.js que será implantado em instâncias do


Amazon EC2, selecione as caixas Aplicativo de web, Node.js, e Amazon EC2. Em seguida, escolha entre os modelos disponíveis para esse conjunto de opções.

Para obter mais informações, consulte [Modelos de projetos do AWS CodeStar](#).

4. Escolha Next (Próximo).
5. No campo de entrada de texto Nome do projeto, insira um nome para o projeto, como *Meu primeiro projeto*. Em ID do projeto, a ID para o projeto é derivada desse nome do projeto, mas é limitada a 15 caracteres.

Por exemplo, a ID padrão de um projeto chamado *Meu primeiro projeto* é *my-first-projec*. O ID do projeto é a base dos nomes de todos os recursos associados ao projeto. O AWS CodeStar usa esse ID do projeto como parte do URL para o repositório de códigos, bem como para os nomes de funções de acesso e políticas de segurança relacionados no IAM. Depois que o projeto for criado, o ID do projeto não pode ser alterado. Para editar a ID do projeto antes de criar o projeto, em ID do projeto, insira a ID que você deseja usar.

Para obter informações sobre os limites em nomes e IDs de projetos, consulte [Limites no AWS CodeStar](#).

 Note

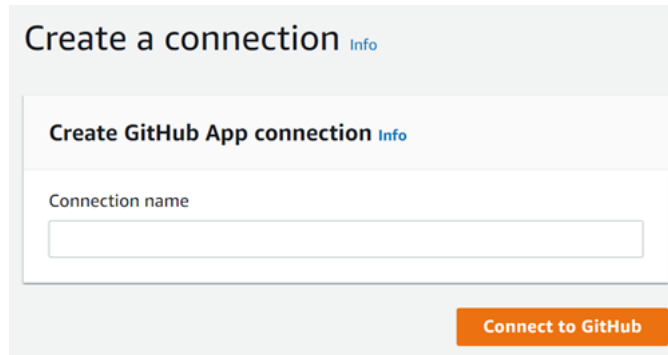
Os IDs dos projetos devem ser exclusivos para a conta da AWS em uma região da AWS.

6. Escolha o provedor do repositório, AWS CodeCommit ou GitHub.
7. Se você tiver escolhido AWS CodeCommit, para Repository name (Nome do repositório), aceite o nome do repositório do AWS CodeCommit padrão ou digite um outro. Em seguida, avance para a etapa 9.
8. Se você escolheu o GitHub, precisará escolher ou criar um recurso de conexão. Se você tiver uma conexão existente, escolha-a no campo de pesquisa. Caso contrário, crie uma nova conexão agora. Escolha Conectar-se ao GitHub.

A página Criar uma conexão é exibida.

**Note**

Para criar uma conexão, você deve ter uma conta do GitHub. Se você estiver criando uma conexão para uma organização, você deve ser o proprietário da organização.



- Em Criar conexão do App GitHub, no campo de entrada de texto Nome da conexão, insira um nome para sua conexão. Escolha Conectar-se ao GitHub.

A página de Conectar ao GitHub exibe e mostra o campo Apps do GitHub

- Em GitHub apps (Aplicações do GitHub), escolha uma instalação de aplicação ou Install a new app (Instalar uma nova aplicação) para criar uma.

**Note**

Você instala uma aplicação para todas as suas conexões com um provedor específico. Se você já instalou a aplicação AWS Connector para GitHub, selecione-a e ignore esta etapa.

- Na página de instalação do Connect para GitHub, selecione a conta na qual deseja instalar a aplicação.


**Note**

Se você instalou a aplicação anteriormente, poderá escolher Configure (Configurar) para prosseguir para uma página de modificação para a instalação da aplicação ou usar o botão Back (Voltar) para retornar ao console.



- d. Se a página Confirme a senha para continuar for exibida, digite sua senha do GitHub e escolha Entrar.
- e. Na página Instalar o AWS Connector para GitHub, mantenha os valores padrão e selecione Instalar.
- f. Na página Conexão ao GitHub o ID de conexão da sua nova instalação é exibido em Aplicações do GitHub.

Depois que a conexão for criada, na página de criação do projeto do CodeStar, a mensagem Pronto para conectar é exibida.


 Note

É possível visualizar sua conexão em Configurações no console do Developer Tools. Para obter mais informações, consulte o tópico de [Conceitos básicos sobre conexões](#).

Select a repository provider


CodeCommit

Use a new AWS CodeCommit repository for your project.



GitHub

Use a new GitHub source repository for your project (requires an existing GitHub account).



**i The GitHub repository provider now uses CodeStar Connections**

To use a GitHub repository in CodeStar, create a connection. The connection will use GitHub Apps to access your repository. Use the following options to choose an existing connection or create a new one. [Learn more](#)

Connection

Choose an existing connection or create a new one and then return to this task.

or

✔ **Ready to connect**

Your Github connection is ready for use.

Repository owner

The owner of the new repository. This can be a personal GitHub account or a GitHub organization.

[Redacted]
▼

Repository name

The name of the new repository.

cs-dk-gh

Repository description

An optional description of the new repository.

Public

- g. Em Proprietário do repositório, escolha a organização do GitHub ou a conta do GitHub pessoal.
- h. Para Repository name (Nome do repositório), aceite o nome do repositório do GitHub padrão ou digite um outro.
- i. Escolha Público ou Privado.

**i Note**

Para usar o AWS Cloud9 como ambiente de desenvolvimento, será necessário selecionar um repositório público.

- j. (Opcional) Em Repository description (Descrição do repositório), digite uma descrição para o repositório do GitHub.

**Note**

Se você escolher um modelo de projeto Alexa Skill, precisará conectar uma conta de desenvolvedor da Amazon. Para obter mais informações sobre como trabalhar com projetos do Alexa Skill, consulte o [Tutorial: Criar um projeto de habilidade do Alexa no AWS CodeStar](#).

9. Se seu projeto estiver implantado em instâncias do Amazon EC2 e você quiser fazer alterações, configure suas instâncias do Amazon EC2 em Configuração do Amazon EC2. Por exemplo, você pode escolher entre os tipos de instância disponíveis para o projeto.

**Note**

Os diferentes tipos de instância do Amazon EC2 fornecem diferentes níveis de potência de computação e podem ter custos associados diferentes. Para obter mais informações, consulte [Tipos de Instância do Amazon EC2](#) e [Preços do Amazon EC2](#).

Se tiver mais de uma nuvem privada virtual (VPC) ou várias sub-redes criadas no Amazon Virtual Private Cloud, você também poderá escolher a VPC e a sub-rede a serem utilizadas. No entanto, se escolher um tipo de instância do Amazon EC2 não compatível com instâncias dedicadas, você não poderá escolher uma VPC cuja localização de instância esteja definida como Dedicada.

Para obter mais informações, consulte [O que é o Amazon VPC?](#) e [Noções básicas de instâncias dedicadas](#).

Em Par de chaves, escolha o par de chaves do Amazon EC2 que você criou em [Etapa 4: criar um par de chaves do Amazon EC2 para projetos do AWS CodeStar](#). Selecione Eu reconheço que tenho acesso ao arquivo de chave privada.

10. Escolha Next (Próximo).
11. Analise os recursos e os detalhes da configuração.
12. Escolha Avançar ou Criar projeto. (A opção exibida depende do modelo do projeto.)

Pode levar alguns minutos para criar o projeto, incluindo o repositório.

13. Depois que seu projeto tiver um repositório, você poderá usar a página Repositório para configurar o acesso a ele. Use os links nas Próximas etapas para configurar um IDE, configurar o controle de problemas ou adicionar membros da equipe ao seu projeto.

Embora o projeto esteja sendo criado, você pode [adicionar membros à equipe](#) ou [configurar o acesso](#) ao repositório do projeto na linha de comando ou de seu IDE favorito.

## Criar um projeto no AWS CodeStar (AWS CLI)

Um projeto do AWS CodeStar é uma combinação de código-fonte e os recursos criados para implantar o código. A coleção de recursos que ajudam a compilar, lançar e implantar o código são chamados de recursos da cadeia de ferramentas. Na criação do projeto, um modelo do AWS CloudFormation provisiona os recursos da cadeia de ferramentas em um pipeline de integração/implantação contínuas (CI/CD).

Quando você usa o console para criar um projeto, o modelo da cadeia de ferramentas é criado para você. Ao usar a AWS CLI para criar um projeto, você cria o modelo da cadeia de ferramentas que cria os recursos da cadeia de ferramentas.

Uma cadeia de ferramentas completa requer os seguintes recursos recomendados:

1. Um repositório do CodeCommit ou do GitHub que contenha seu código-fonte.
2. Um pipeline do CodePipeline configurado para ouvir alterações no repositório.
  - a. Quando você usa o CodeBuild para executar testes de unidade ou de integração, recomendamos adicionar um estágio de compilação ao pipeline para criar artefatos de compilação.
  - b. Recomendamos adicionar um estágio de implantação ao pipeline que usa CodeDeploy ou AWS CloudFormation para implantar o artefato de compilação e o código-fonte para a infraestrutura do runtime.

### Note

Como o CodePipeline requer pelo menos dois estágios em um pipeline, e o primeiro estágio deve ser o estágio de origem, adicione um estágio de compilação ou implantação como o segundo estágio.

As cadeias de ferramentas do AWS CodeStar são definidas como um [modelo do CloudFormation](#).

Para um tutorial que aborda essa tarefa e configura recursos de exemplo, consulte [Tutorial: Criar um projeto no AWS CodeStar com a AWS CLI](#).

### Pré-requisitos:

Ao criar um projeto, você fornece os parâmetros a seguir em um arquivo de entrada. Se eles não forem fornecidos, o AWS CodeStar criará um projeto vazio.

- **Código-fonte.** Se esse parâmetro estiver incluído na solicitação, você também deverá incluir um modelo da cadeia de ferramentas.
  - O código-fonte deve incluir o código do aplicativo necessários para executar o projeto.
  - O código-fonte deve incluir todos os arquivos de configuração necessários, como um `buildspec.yml` para um projeto do CodeBuild ou um `appspec.yml` para uma implantação do CodeDeploy.
  - Você pode incluir itens opcionais no código-fonte, como um README ou um `template.yml` para recursos do AWS que não sejam da cadeia de ferramentas.
- **Modelo da cadeia de ferramentas.** O modelo da cadeia de ferramentas provisiona os recursos do AWS e os perfis do IAM a serem gerenciados para o projeto.
- **Locais de origem.** Se especificar o código-fonte e um modelo da cadeia de ferramentas para o projeto, você deverá fornecer um local. Faça upload dos arquivos de origem e o modelo da cadeia de ferramentas para o bucket do Amazon S3. O AWS CodeStar recupera os arquivos e os utiliza para criar o projeto.

#### Important

Lembre-se de configurar sua região do AWS preferida na AWS CLI. O projeto é criado na região do AWS configurada na AWS CLI.

1. Execute o comando `create-project` e inclua o parâmetro `--generate-cli-skeleton`:

```
aws codestar create-project --generate-cli-skeleton
```


Os dados formatados JSON aparecem na saída. Copie os dados para um arquivo (por exemplo, `input.json`) em um local no computador local ou em uma instância na qual a AWS CLI esteja instalada. Modifique os dados copiados da maneira a seguir e salve os resultados.

```
{
  "name": "project-name",
  "id": "project-id",
  "description": "description",
  "sourceCode": [
    {
      "source": {
        "s3": {
          "bucketName": "s3-bucket-name",
          "bucketKey": "s3-bucket-object-key"
        }
      },
      "destination": {
        "codeCommit": {
          "name": "codecommit-repository-name"
        },
        "gitHub": {
          "name": "github-repository-name",
          "description": "github-repository-description",
          "type": "github-repository-type",
          "owner": "github-repository-owner",
          "privateRepository": true,
          "issuesEnabled": true,
          "token": "github-personal-access-token"
        }
      }
    }
  ],
  "toolchain": {
    "source": {
      "s3": {
        "bucketName": "s3-bucket-name",
        "bucketKey": "s3-bucket-object-key"
      }
    },
    "roleArn": "service-role-arn",
    "stackParameters": {
      "KeyName": "key-name"
    }
  },
  "tags": {
    "KeyName": "key-name"
  }
}
```

```
}
```

Substitua o seguinte:

- *project-name*: necessário. O nome amigável deste projeto do AWS CodeStar.
- *project-id*: necessário. O ID do projeto do AWS CodeStar.

 Note

Você deve ter um ID de projeto exclusivo ao criar um projeto. Você receberá um erro se enviar um arquivo de entrada com um ID de projeto já existente.

- *description*: opcional. A descrição desse projeto do AWS CodeStar.
- *sourceCode*: opcional. As informações de configuração do código-fonte fornecido para o projeto. Atualmente, somente um objeto `sourceCode` é compatível. Cada objeto `sourceCode` contém informações sobre o local onde o código-fonte é recuperado pelo AWS CodeStar, e o destino onde o código-fonte é preenchido.
- *source*: obrigatório. Ele define o local onde você fez upload do código-fonte. A única origem compatível é Amazon S3. O AWS CodeStar recupera o código-fonte e o inclui no repositório depois que o projeto é criado.
  - *S3*: opcional. O local do Amazon S3 do código-fonte.
    - *bucket-name*: o bucket que contém o código-fonte.
    - *bucket-key*: o prefixo do bucket e a chave de objeto que aponta para o arquivo `.zip` que contém o código-fonte (por exemplo, `src.zip`).
- *destination*: opcional. Os locais de destino onde o código-fonte deve ser preenchido quando o projeto é criado. Os destinos compatíveis do código-fonte são o CodeCommit e o GitHub.

Você só pode fornecer uma destas duas opções:

- *codeCommit*: o único atributo obrigatório é o nome do repositório do CodeCommit que deve conter o código-fonte. Esse repositório deve estar no modelo da cadeia de ferramentas.

**Note**

Para CodeCommit, você deve fornecer o nome do repositório que definiu na pilha da cadeia de ferramentas. O AWS CodeStar inicializa esse repositório com o código-fonte que você forneceu no Amazon S3.

- **github**: este objeto representa informações necessárias para criar o repositório do GitHub e propagá-lo com o código-fonte. Se você escolher um repositório do GitHub, os seguintes valores serão necessários.

**Note**

Para o GitHub, não é possível especificar um repositório do GitHub existente. O AWS CodeStar cria um para você e preenche esse repositório com o código-fonte carregado no Amazon S3. O AWS CodeStar usa as seguintes informações para criar seu repositório no GitHub.

- **name**: obrigatório. O nome do repositório do GitHub.
- **description**: obrigatório. A descrição do repositório do GitHub.
- **type**: obrigatório. O tipo de repositório do GitHub. Os valores válidos são User ou Organization.
- **owner**: obrigatório. O nome de usuário do GitHub do proprietário do repositório. Se o repositório do GitHub precisar ser de uma organização do GitHub, forneça o nome da organização.
- **privateRepository**: obrigatório. Se você quiser que esse repositório seja privado ou público. Os valores válidos são true ou false.
- **issuesEnabled**: obrigatório. Se você deseja habilitar problemas no GitHub com esse repositório. Os valores válidos são true ou false.
- **token**: opcional. Este é um token de acesso pessoal que o AWS CodeStar usa para acessar a conta do GitHub. Esse token deve conter os seguintes escopos: repo, user e admin:repo\_hook. Para recuperar um token de acesso pessoal do GitHub, consulte [Criação de um token de acesso pessoal para a linha de comando](#) no site do GitHub.



**Note**

Se você usa a CLI para criar um projeto com um repositório de origem do GitHub, o AWS CodeStar usa seu token para acessar o repositório por meio de aplicativos OAuth. Se você usa o console para criar um projeto com um repositório de origem do GitHub, o AWS CodeStar usa um recurso de conexão, que acessa o repositório com os aplicativos do GitHub.

- **toolchain**: informações sobre a cadeia de ferramentas CI/CD a ser configurada quando o projeto for criado. Isso inclui o local em que você fez upload do modelo da cadeia de ferramentas. O modelo cria uma pilha do AWS CloudFormation que contém seus recursos da cadeia de ferramentas. Isso também inclui as substituições de parâmetro do AWS CloudFormation para referência e a função a ser usada para criar a pilha. O AWS CodeStar recupera o modelo e usa o AWS CloudFormation para executar o modelo.
- **source**: obrigatório. A localização do seu modelo de conjunto de ferramentas. O Amazon S3 é o único local de origem compatível.
- **S3**: opcional. O local do Amazon S3 onde você fez upload do modelo da cadeia de ferramentas.
  - **bucket-name**: o nome do bucket do Amazon S3.
  - **bucket-key**: o prefixo do bucket e a chave de objeto que aponta para o arquivo .yaml ou .json que contém o modelo da cadeia de ferramentas (por exemplo, files/toolchain.yaml).
- **stackParameters**: opcional. Contém pares de chave-valor a serem passados para o AWS CloudFormation. Esses serão os parâmetros, se houver, para o modelo da cadeia de ferramentas fazer referência na configuração.
- **role**: opcional. A função usada para criar os recursos da cadeia de ferramentas na conta. A função é necessária da seguinte maneira:
  - Se a função não for fornecida, o AWS CodeStar usará a função de serviço padrão criada para a conta se a cadeia de ferramentas for um modelo de início rápido do AWS CodeStar. Se a função de serviço não existir na conta, você poderá criar uma. Para obter mais informações, consulte [Etapa 2: Criar a função AWS CodeStar de serviço](#).
  - Você deverá fornecer a função se estiver carregando e usando seu próprio modelo de cadeia de ferramentas personalizado. Você pode criar uma função com base

na função de serviço e na declaração de política do AWS CodeStar. Para obter um exemplo dessa declaração de política, consulte [Política AWSCodeStarServiceRole](#).

- **tags**: opcional. As tags anexadas ao projeto do AWS CodeStar.

**Note**

Essas tags não são anexadas a recursos contidos no projeto.

2. Alterne para o diretório que contenha o arquivo que você acabou de salvar e execute o comando `create-project` novamente. Inclua o parâmetro `--cli-input-json`.

```
aws codestar create-project --cli-input-json file://input.json
```

3. Se bem-sucedido, dados semelhantes aos seguintes serão exibidos na saída:

```
{
  "id": "project-ID",
  "arn": "arn"
}
```

- A saída contém informações sobre o novo projeto:
  - O valor `id` representa o ID do projeto.
  - O valor `arn` representa o ARN do projeto.

4. Use o comando `describe-project` para verificar o status da criação do projeto. Inclua o parâmetro `--id`.

```
aws codestar describe-project --id <project_ID>
```

Dados semelhantes aos seguintes são exibidos na saída:

```
{
  "name": "MyProject",
  "id": "myproject",
  "arn": "arn:aws:codestar:us-east-1:account_ID:project/myproject",
  "description": "",
  "createdTimeStamp": 1539700079.472,
  "stackId": "arn:aws:cloudformation:us-east-1:account_ID:stack/awscodestar-myproject/stack-ID",
  "status": {
```

```
    "state": "CreateInProgress"  
  }  
}
```

- A saída contém informações sobre o novo projeto:
  - O valor `state` representa o status da criação do projeto, como `CreateInProgress` ou `CreateComplete`.

Embora o projeto esteja sendo criado, você pode [adicionar membros à equipe](#) ou [configurar o acesso](#) ao repositório do projeto na linha de comando ou de seu IDE favorito.

## Usar um IDE com o AWS CodeStar

Quando você integra um IDE com o AWS CodeStar, pode continuar a gravar e desenvolver código no ambiente de sua preferência. As alterações feitas são incluídas no projeto do AWS CodeStar cada vez que você se confirmar e enviar código por push.

The screenshot shows an IDE interface with a code editor on the left and a commit message dialog on the right. The code editor displays the following HTML code:

```
48     <nav class="website-nav">
49         <ul>
50             <li><a class="home-link" href="https://aws.amazon.com/">
51             <li><a href="https://aws.amazon.com/what-is-cloud-comput
52             <li><a href="https://aws.amazon.com/solutions/">Services
53             <li><a href="https://aws.amazon.com/contact-us/">Contact
54         </ul>
55     </nav>
56 </header>
57
58     <div class="message">
59         <a class="twitter-link" href="http://twitter.com/home/?status=I
60         <div class="text">
61             <h1>Congratulations!</h1>
62             <h2>You just created a Node.js web application</h2>
63             <h3>And I made a change in Eclipse!</h3>
64         </div>
65     </div>
66 </div>
67
68 <footer>
69     <p class="footer-contents">Designed and developed with <a href="http
```

The commit message dialog shows the following information:

- Commit Message: Updated index.html with a new h3
- Author: Mary Major <mary\_major@example.com>
- Committer: Mary Major <mary\_major@example.com>
- Buttons: Commit and Push..., Commit

The IDE interface also shows a Task List panel on the right, an Outline panel with the message "An outline is not available.", and a bottom toolbar with various icons and a "Filter files" input field.

## Tópicos

- [Usar o AWS Cloud9 com a AWS CodeStar](#)
- [Usar o Eclipse com o AWS CodeStar](#)
- [Use o Visual Studio com o AWS CodeStar](#)

## Usar o AWS Cloud9 com a AWS CodeStar

Você pode usar o AWS Cloud9 para fazer alterações no código e desenvolver software em um projeto do AWS CodeStar. O AWS Cloud9 é um IDE online, que você acessa no navegador da web. O IDE oferece uma experiência de edição de código completa com suporte para várias linguagens de

programação e depuradores de tempo de execução, bem como um terminal integrado. Em segundo plano, uma instância do Amazon EC2 hospeda um ambiente de desenvolvimento do AWS Cloud9. Esse ambiente fornece o IDE do AWS Cloud9 e o acesso aos arquivos de código do projeto do AWS CodeStar. Para obter mais informações, consulte o Guia do usuário do [AWS Cloud9](#).

Você pode usar o console do AWS CodeStar ou o console do AWS Cloud9 para criar ambientes de desenvolvimento do AWS Cloud9 para projetos que armazenam o código no CodeCommit. Para projetos do AWS CodeStar que armazenam o código no GitHub, você só pode usar o console do AWS Cloud9. Este tópico descreve como usar ambos os consoles.

Para usar o AWS Cloud9, você precisa:

- Um usuário do IAM que tenha sido adicionado como membro da equipe a um projeto do AWS CodeStar.
- Se o projeto do AWS CodeStar armazena seu código-fonte no CodeCommit, as credenciais do AWS para o usuário do IAM.

## Tópicos

- [Criar um ambiente do AWS Cloud9 para um projeto](#)
- [Abrir um ambiente do AWS Cloud9 para um projeto](#)
- [Compartilhar um ambiente do AWS Cloud9 com um membro da equipe do projeto](#)
- [Excluir um ambiente do AWS Cloud9 de um projeto](#)
- [Usar o GitHub com o AWS Cloud9](#)
- [Recursos adicionais](#)

## Criar um ambiente do AWS Cloud9 para um projeto

Siga estas etapas para criar um ambiente de desenvolvimento do AWS Cloud9 para um projeto do AWS CodeStar.

1. Siga as etapas em [Criar um projeto](#) se deseja criar um novo projeto.
2. Abra o projeto no console do AWS CodeStar. Na barra de navegação, escolha IDE. Escolha Criar ambiente e use as etapas a seguir.

**⚠ Important**

Se o projeto estiver em uma região do AWS onde o AWS Cloud9 não for compatível, você não verá opções do AWS Cloud9 na aba IDE na barra de navegação. No entanto, você pode usar o console do AWS Cloud9 para criar um ambiente de desenvolvimento, abrir o novo ambiente e conectá-lo ao repositório do AWS CodeCommit do projeto. Ignore as etapas a seguir e consulte [Criar um ambiente](#), [Abrir um ambiente](#) e [Exemplo do AWS CodeCommit](#) no Guia do usuário do AWS Cloud9. Para obter a lista das regiões da AWS compatíveis, consulte [AWS Cloud9](#) no Referência geral da Amazon Web Services.

Em Criar ambiente do AWS Cloud9, personalize os padrões do projeto.


1. Para alterar o tipo padrão de instância do Amazon EC2 para hospedar o ambiente, para Instance type (tipo de instância), selecione o tipo de instância.
2. O AWS Cloud9 usa a Amazon Virtual Private Cloud (Amazon VPC) em sua conta do AWS para se comunicar com a instância. Dependendo de como o Amazon VPC está configurado na conta do AWS, execute uma das ações a seguir.

A conta tem uma VPC com, pelo menos, uma sub-rede nessa VPC?	A VPC que deseja que o AWS Cloud9 use é a VPC padrão da conta?	A VPC possui uma única sub-rede?	Faça o seguinte
Não	—	—	Se não houver nenhuma VPC, crie uma. Expanda Network settings (Configurações de rede). Em Network (VPC) (Rede [VPC]), escolha Create VPC (Criar VPC) e siga as instruções na página. Para obter mais informações, consulte <a href="#">Criar um Amazon</a>

A conta tem uma VPC com, pelo menos, uma sub-rede nessa VPC?	A VPC que deseja que o AWS Cloud9 use é a VPC padrão da conta?	A VPC possui uma única sub-rede?	Faça o seguinte
			<p><a href="#">VPC para o AWS Cloud9</a> no Guia do usuário do AWS Cloud9.</p> <p>Se existe uma VPC mas não há sub-rede, crie uma. Expanda Network settings (Configurações de rede). Em Network (VPC) (Rede [VPC]), escolha Create subnet (Criar sub-rede) e siga as instruções. Para obter mais informações, consulte <a href="#">Criar uma sub-rede para o AWS Cloud9</a> no Guia do usuário AWS Cloud9.</p>
Sim	Sim	Sim	Avance diretamente para a etapa 4 deste procedimento. (O AWS Cloud9 usa a VPC padrão com a única sub-rede.)
Sim	Sim	Não	Em Subnet (Sub-rede), selecione a sub-rede que você deseja que o AWS Cloud9 use na VPC padrão pré-selecionada.
Sim	Não	Yes ou No	Em Network (VPC) (Rede (VPC)), selecione a VPC que deseja que o AWS Cloud9 use. Em Subnet (Sub-rede), selecione a sub-rede que você deseja que o AWS Cloud9 use nessa VPC.

Para obter mais informações, consulte Configurações do [Amazon VPC para ambientes de desenvolvimento do AWS Cloud9](#) no Guia do usuário do AWS Cloud9.

3. Insira um nome de ambiente e, opcionalmente, adicione uma descrição do ambiente.

 Note

Os nomes do ambiente devem ser exclusivos por usuário.

4. Para alterar o período padrão após o qual o AWS Cloud9 desliga o ambiente quando não foi usado, expanda Configurações econômicas e altere a configuração.
5. Selecione Create environment (Criar ambiente).

Para abrir o ambiente, consulte [Abrir um ambiente do AWS Cloud9 para um projeto](#).

Você pode usar essas etapas para criar mais de um ambiente para um projeto. Por exemplo, convém usar um ambiente para trabalhar em uma parte do código e outro ambiente para trabalhar na mesma parte do código com configurações diferentes.

## Abrir um ambiente do AWS Cloud9 para um projeto

Siga estas etapas para abrir um ambiente de desenvolvimento do AWS Cloud9 criado para um projeto do AWS CodeStar.

1. Com o projeto aberto no console do AWS CodeStar, na barra de navegação, escolha IDE.

 Important

Se o código-fonte do projeto estiver armazenado no GitHub, você não verá IDE na barra de navegação. No entanto, você pode usar o console do AWS Cloud9 para abrir um ambiente existente. Ignore o restante deste procedimento e consulte [Abertura de um ambiente](#) no Guia do usuário do AWS Cloud9 e [Usar o GitHub com o AWS Cloud9](#).

2. Em Your AWS Cloud9 environments (Os ambientes do AC9) ou Shared AWS Cloud9 environments (Ambientes do AC9 compartilhados), escolha Open IDE (Abrir IDE) para o ambiente que você deseja abrir.

Você pode usar o IDE do AWS Cloud9 para iniciar o trabalho com código no repositório do AWS CodeCommit do projeto imediatamente. Para obter mais informações, consulte [A janela do ambiente, O editor, as guias e os painéis](#) e [O terminal](#) no Guia do usuário do AWS Cloud9 e [Comandos do Git básicos](#) no Guia do usuário do AWS CodeCommit.



## Compartilhar um ambiente do AWS Cloud9 com um membro da equipe do projeto

Depois de criar um ambiente de desenvolvimento do AWS Cloud9 para um projeto do AWS CodeStar, você poderá convidar outros usuários na conta da AWS, inclusive membros da equipe do projeto, para acessar esse mesmo ambiente. Isso é especialmente útil na programação de pares, em que dois programadores aproveitam a codificação e dão orientações sobre o mesmo código por meio do compartilhamento de tela ou ainda sentados na mesma estação de trabalho. Os membros do ambiente podem usar o IDE do AWS Cloud9 compartilhado para ver alterações feitas no código de cada membro realçado no editor de códigos e para conversar por texto com outros membros durante a codificação.

Adicionar um membro da equipe a um projeto não permite automaticamente que esse membro participe de qualquer ambientes de desenvolvimento do AWS Cloud9 do projeto. Para convidar um membro da equipe do projeto para acessar um ambiente para um projeto, você precisa determinar a função de acesso do membro ambiente correta, aplicar políticas gerenciadas do AWS para o usuário e convidá-lo para o ambiente. Para obter mais informações, consulte [Sobre funções de acesso de membro do ambiente](#) e [Convidar um usuário do IAM para o ambiente](#) no Guia do usuário do AWS Cloud9.

Quando você convida um membro da equipe do projeto para acessar um ambiente de um projeto, o console do AWS CodeStar exibe o ambiente para esse membro da equipe. O ambiente é exibido na lista Ambientes compartilhados da guia IDE no console do AWS CodeStar do projeto. Para exibir essa lista, peça para o membro da equipe abrir o projeto no console e escolher IDE na barra de navegação.

### Important

Se o código-fonte do projeto estiver armazenado no GitHub, você não verá IDE na barra de navegação. No entanto, você pode usar o console do AWS Cloud9 para convidar outros usuários na conta da AWS, inclusive membros da equipe do projeto, para acessar um ambiente. Para isso, consulte [Usar o GitHub com o AWS Cloud9](#) neste guia e consulte [Sobre funções de acesso de membro do ambiente](#) e [Convidar um usuário do IAM para o ambiente](#) no Guia do usuário do AWS Cloud9.

Você também pode convidar um usuário que não seja um membro da equipe do projeto para acessar um ambiente. Por exemplo, convém que um usuário trabalhe no código de um projeto, mas não tenha acesso a esse projeto. Para convidar esse tipo de usuário, consulte [Sobre funções de](#)

[acesso de membro do ambiente](#) e [Convidar um usuário do IAM para o ambiente](#) no Guia do usuário do AWS Cloud9. Quando você convida um usuário que não seja um membro da equipe do projeto para acessar um ambiente de um projeto, esse usuário pode usar o console do AWS Cloud9 para acessar o ambiente. Para obter mais informações, consulte [Abrir um ambiente](#) no Guia do usuário do AWS Cloud9.

## Excluir um ambiente do AWS Cloud9 de um projeto

Quando você exclui um projeto e todos os recursos da AWS do AWS CodeStar, todos os ambientes de desenvolvimento do AWS Cloud9 relacionados que foram criados com o console do AWS CodeStar também são excluídos e não podem ser recuperados. Você pode excluir um ambiente de desenvolvimento de um projeto sem excluir o projeto.

1. Com o projeto aberto no console do AWS CodeStar, na barra de navegação, escolha IDE.

### Important

Se o código-fonte do projeto estiver armazenado no GitHub, você não verá IDE na barra de navegação. Porém, você pode usar o console do AWS Cloud9 para excluir um ambiente de desenvolvimento. Ignore o restante deste procedimento e consulte [Exclusão de um ambiente](#) no Guia do usuário do AWS Cloud9.

2. Escolha o ambiente que você deseja excluir nos ambientes do Cloud9 e escolha Excluir
3. Insira **delete** para confirmar a exclusão para o ambiente de desenvolvimento e então selecione Excluir.

### Warning

Você não poderá recuperar um ambiente de desenvolvimento depois que ele tiver sido excluído. Todas as alterações de código não confirmadas no ambiente são perdidas.

## Usar o GitHub com o AWS Cloud9

Para projetos do AWS CodeStar que tenham o código-fonte armazenado no GitHub, o console do AWS CodeStar não oferece suporte ao trabalho com ambientes de desenvolvimento do AWS Cloud9 diretamente. Porém, você pode usar o console do AWS Cloud9 para trabalhar com código-fonte em repositórios do GitHub.

1. Use o console do AWS Cloud9 para criar um ambiente de desenvolvimento do AWS Cloud9. Para obter mais informações, consulte [Criação de um ambiente](#) no Guia do usuário do AWS Cloud9.
2. Use o console do AWS Cloud9 para abrir o ambiente de desenvolvimento. Para obter mais informações, consulte [Abertura de um ambiente](#) no Guia do usuário do AWS Cloud9.
3. No IDE, use uma sessão de terminal para se conectar ao repositório do GitHub (um processo conhecido como a clonagem). Se uma sessão de terminal não estiver em execução, na barra de menus no IDE, escolha Window, New Terminal (Janela, novo terminal). Para usar os comandos para clonar o repositório do GitHub, consulte [Clonar um repositório](#) no site de Ajuda do GitHub.

Para navegar até a página principal do repositório do GitHub, com o projeto aberto no console do AWS CodeStar, na barra de navegação lateral, escolha Code (Código).

4. Use a janela Environment (Ambiente) e as guias do editor no IDE para exibir, alterar e salvar o código. Para obter mais informações, consulte [A janela de ambiente](#) e [O editor, as guias e os painéis](#) no Guia do usuário do AWS Cloud9.
5. Use o Git na sessão de terminal no IDE para enviar alterações feitas no código para o repositório e obter alterações feitas no código periodicamente de outros do repositório. Para obter mais informações, consulte [Enviar um repositório remoto](#) e [Buscar um repositório remoto](#) no site de Ajuda do GitHub. Para obter comandos do Git, consulte [Git Cheatsheet](#) no site de Ajuda do GitHub.

#### Note

Para evitar que o Git solicite suas credenciais de login do GitHub sempre que enviar ou receber código do repositório, você pode usar um assistente de credenciais. Para obter mais informações, consulte [Armazenar em cache a senha do GitHub no Git](#) no site de Ajuda do GitHub.

## Recursos adicionais

Para obter mais informações sobre como usar o AWS Cloud9, consulte o seguinte no Guia do usuário do AWS Cloud9:

- [Tutorial](#)
- [Trabalhar com ambientes](#)
- [Trabalhar com a IDE](#)

- [Amostras](#)

## Usar o Eclipse com o AWS CodeStar

Você pode usar o Eclipse para fazer alterações no código e desenvolver software em um projeto do AWS CodeStar. Você pode editar o código do projeto do AWS CodeStar com o Eclipse e, em seguida, confirmar e enviar por push as alterações para o repositório de origem para o projeto do AWS CodeStar.

### Note

As informações neste tópico se aplicam somente a projetos do AWS CodeStar que armazenam seu código-fonte no CodeCommit. Se o seu projeto do AWS CodeStar armazenar o código-fonte no GitHub, você poderá usar uma ferramenta como o EGit para Eclipse. Para obter mais informações, consulte a [Documentação do EGit](#) no site do EGit.

Se o projeto do AWS CodeStar armazena seu código-fonte no CodeCommit, você deve instalar uma versão do AWS Toolkit for Eclipse que ofereça suporte ao AWS CodeStar. Você também deve ser membro da equipe de projeto do AWS CodeStar com a função de proprietário ou de colaborador.

Para usar o Eclipse, você também precisa:

- Um usuário do IAM que tenha sido adicionado a um projeto do AWS CodeStar como membro da equipe.
- Se o projeto do AWS CodeStar armazenar seu código-fonte no CodeCommit, [credenciais do Git](#) (credenciais de login) para o usuário do IAM.
- Permissões suficientes para instalar o Eclipse e o AWS Toolkit for Eclipse no computador local.

### Tópicos

- [Etapa 1: Instalar o AWS Toolkit for Eclipse](#)
- [Etapa 2: Importar o projeto do AWS CodeStar no Eclipse](#)
- [Etapa 3: Editar o código do projeto do AWS CodeStar no Eclipse](#)

## Etapa 1: Instalar o AWS Toolkit for Eclipse

O Toolkit for Eclipse é um pacote de software que você pode adicionar ao Eclipse. Ele é instalado e gerenciado da mesma forma como outros pacotes de software no Eclipse. O toolkit do AWS CodeStar é incluído como parte do Toolkit para o Eclipse.

Instalar o kit de ferramentas para Eclipse com o módulo do AWS CodeStar

1. Instale o Eclipse no seu computador local. As versões suportadas do Eclipse incluem Luna, Mars e Neon.
2. Baixe e instale o Toolkit do Eclipse. Para obter mais informações, consulte o [AWS Toolkit for Eclipse Guia de conceitos básicos do](#).
3. No Eclipse, escolha Ajuda e, em seguida, escolha Instalar novo software.
4. Em Software disponível, escolha Adicionar.
5. Em Adicionar repositório, escolha Arquivo, navegue para o local onde você salvou o arquivo .zip e abra o arquivo. Deixe o Nome em branco e, em seguida, clique em OK.
6. Em Available Software (Software disponível), escolha Select all (Selecionar tudo) para selecionar AWS Core Management Tools e Developer Tools (Ferramentas de desenvolvedor), e escolha Next (Avançar).
7. Em Detalhes da instalação, escolha Avançar.
8. Em Analisar licenças, analise os contratos de licença. Escolha I accept the terms of the license agreement (Aceito os termos do contrato de licença) e escolha Finish (Concluir). Reinicie o Eclipse.

## Etapa 2: Importar o projeto do AWS CodeStar no Eclipse

Depois de ter instalado o Toolkit for Eclipse, você pode importar projetos do AWS CodeStar e editar, confirmar e enviar por push o código do IDE.

### Note

Você pode adicionar vários projetos do AWS CodeStar a um único espaço de trabalho no Eclipse, mas você deve atualizar as credenciais do projeto ao mudar de um projeto para outro.

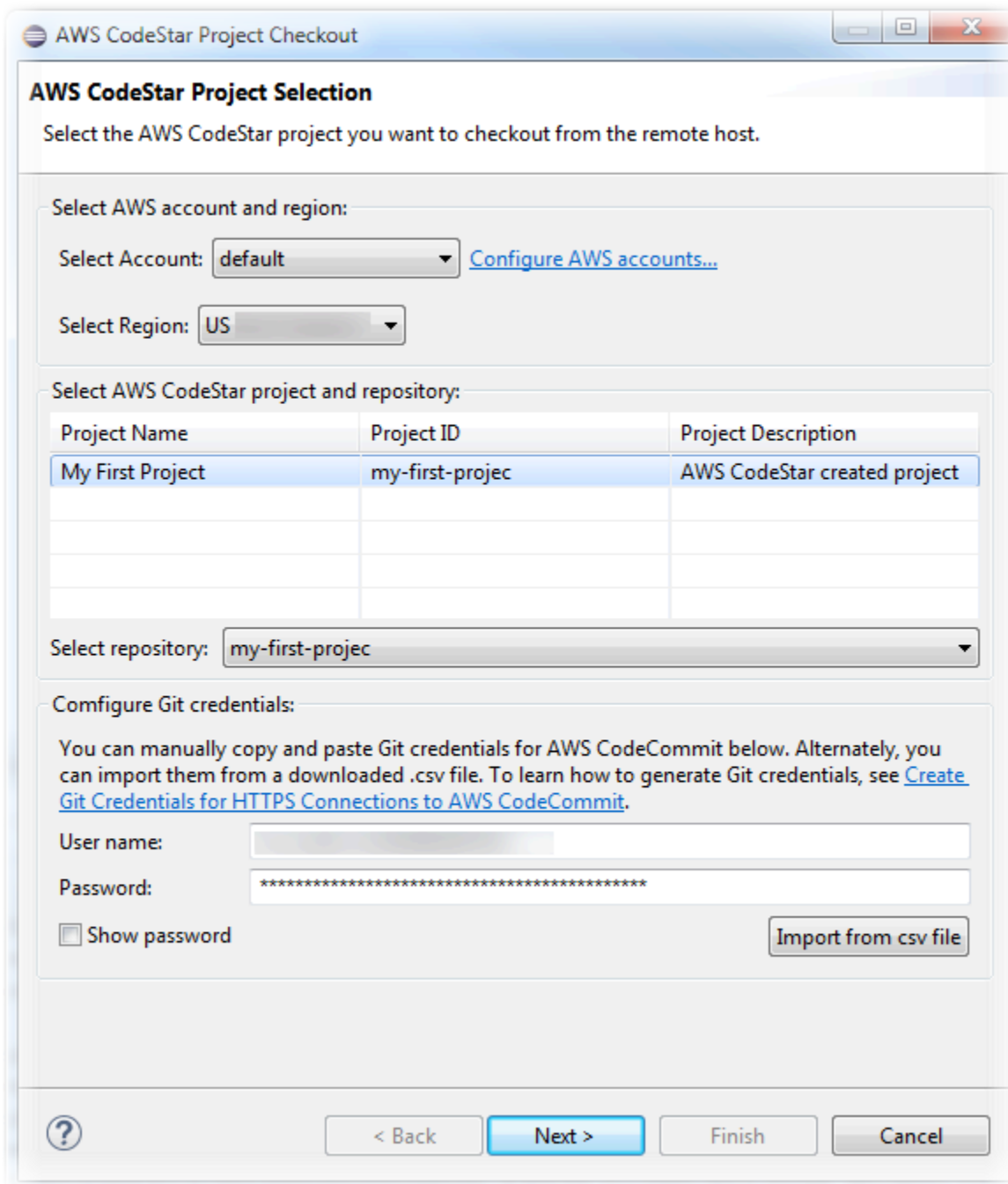
## Para importar um projeto do AWS CodeStar

1. No menu da AWS, escolha Import AWS CodeStar Project (Importar projeto do ACS). Como alternativa, escolha Arquivo e, em seguida, escolha Importar. Em Select (Selecionar), expanda AWS e escolha AWS CodeStar Project (Projeto do ACS).

Escolha Next (Próximo).

2. Em Seleção de projeto do AWS CodeStar, escolha seu perfil e a região da AWS em que o projeto do está hospedado. Se você não tiver um perfil do AWS configurado com uma chave de acesso e uma chave secreta no computador, escolha Configurar contas do AWS e siga as instruções.

Em Select AWS CodeStar project and repository (Selecionar projeto e repositório do ACS), escolha seu projeto do AWS CodeStar na lista. Em Configurar credenciais do Git, insira as credenciais de login que você gerou para acessar o repositório do projeto. (Se você não tiver as credenciais do Git, consulte [Conceitos básicos](#).) Escolha Next (Próximo).



3. Por padrão, todas as ramificações do repositório do projeto estão selecionadas. Se você não desejar importar uma ou mais ramificações, limpe as caixas e, em seguida, escolha Avançar.
4. No Local Destination (Destino local), escolha um destino onde o assistente de importação cria o repositório local em seu computador e, em seguida, escolha Finish (Concluir).
5. Em Project Explorer (Explorador de projeto), expanda a árvore do projeto para procurar os arquivos no projeto do AWS CodeStar.

## Etapa 3: Editar o código do projeto do AWS CodeStar no Eclipse

Depois de importar um projeto do AWS CodeStar em um espaço de trabalho do Eclipse, você pode editar o código do projeto, salvar as alterações e confirmar e enviar por push o código para o repositório de origem do projeto. Este é o mesmo processo que você segue para qualquer repositório Git que usa o plug-in EGit para o Eclipse. Para obter mais informações, consulte o [Guia do usuário do EGit](#) no site do Eclipse.

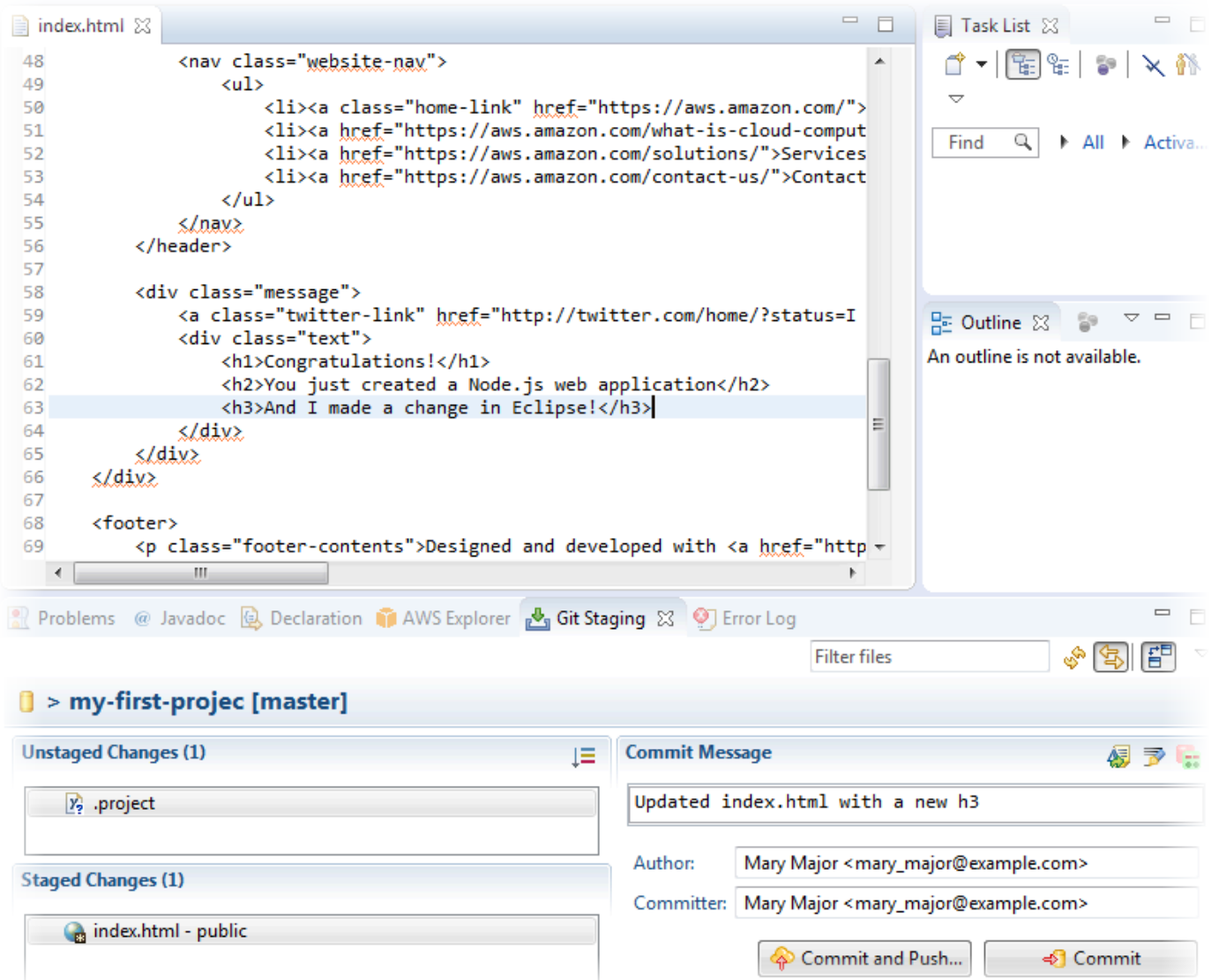
Para editar o código do projeto e fazer a primeira confirmação para o repositório de origem de um projeto do AWS CodeStar

1. Em Project Explorer (Explorador de projeto), expanda a árvore do projeto para procurar os arquivos no projeto do AWS CodeStar.
2. Edite um ou mais arquivos de código e salve as alterações.
3. Quando estiver pronto para confirmar as alterações, abra o menu de contexto desse arquivo, escolha Equipe e, em seguida, escolha Confirmar.

Você poderá ignorar essa etapa, se a janela Git Staging (Preparação do Git) já estiver aberta na visualização do projeto.

4. Em Git Staging (Preparação do Git), prepare as alterações movendo os arquivos alterados para Staged Changes (Alterações preparadas). Digite uma mensagem de confirmação em Commit Message (Mensagem de confirmação) e, em seguida, escolha Commit and Push (Confirmar e enviar por push).





Para visualizar a implantação das alterações do código, volte para o painel de controle de seu projeto. Para obter mais informações, consulte [Etapa 3: Visualizar o projeto](#).

## Use o Visual Studio com o AWS CodeStar

Você pode usar o Visual Studio para fazer alterações no código e desenvolver software em um projeto do AWS CodeStar.

### Note

O Visual Studio para Mac não oferece suporte ao AWS Toolkit, portanto, ele não pode ser usado com o AWS CodeStar.

As informações neste tópico se aplicam somente a projetos do AWS CodeStar que armazenam seu código-fonte no CodeCommit. Se o seu projeto do AWS CodeStar armazenar o código-fonte no GitHub, você poderá usar uma ferramenta como o GitHub Extension para Visual Studio. Para obter mais informações, consulte a página [Visão geral](#) no site do GitHub Extension para Visual Studio e [Conceitos básicos do GitHub para Visual Studio](#) no site do GitHub.

Para usar o Visual Studio para editar o código no repositório de origem para um projeto do AWS CodeStar, você deve instalar uma versão do AWS Toolkit for Visual Studio que ofereça suporte ao AWS CodeStar. Você deve ser membro da equipe de projeto do AWS CodeStar com a função de proprietário ou colaborador.

Para usar o Visual Studio, você também precisa:

- Um usuário do IAM que tenha sido adicionado a um projeto do AWS CodeStar como membro da equipe.
- As credenciais do AWS para o usuário do IAM (por exemplo, sua chave de acesso e sua chave secreta).
- Permissões suficientes para instalar o Visual Studio e o AWS Toolkit for Visual Studio no computador local.

O Toolkit for Visual Studio é um pacote de software que você pode adicionar ao Visual Studio. Ele é instalado e gerenciado da mesma forma como outros pacotes de software no Visual Studio.

Para instalar o Toolkit for Visual Studio com o módulo do AWS CodeStar e configurar o acesso ao repositório do projeto

1. Instale o Visual Studio no computador local.
2. Faça o download e instale o Toolkit for Visual Studio e salve o arquivo .zip em uma pasta ou diretório local. Na página Getting Started with the AWS Toolkit for Visual Studio (Conceitos básicos do TVS), digite ou importe as credenciais da AWS e escolha Save and Close (Salvar e fechar).
3. No Visual Studio, abra o Team Explorer. Em Provedores de serviços hospedados, localize CodeCommit e selecione Conectar.
4. Em Gerenciar conexões, escolha Clonar. Escolha o repositório do seu projeto e a pasta no seu computador local na qual você deseja clonar o repositório e escolha OK.

5. Se você for solicitado a criar credenciais do Git, escolha Yes. O toolkit tenta criar credenciais em seu nome. Salve o arquivo de credenciais em um local seguro. Essa é a única oportunidade que você terá de salvar essas credenciais. Se o kit de ferramentas não puder criar credenciais em seu nome, ou se você escolher Não, você deve criar e fornecer suas próprias credenciais do Git. Para obter mais informações, consulte [Para configurar o computador para confirmar alterações \(usuário do IAM\)](#) ou siga as instruções online.

Quando tiver concluído a clonagem do projeto, você estará pronto para começar a editar o código no Visual Studio e confirmar e enviar suas alterações por push ao repositório do projeto no CodeCommit.

## Alterar recursos da AWS em um projeto do AWS CodeStar

Depois de criar um projeto no AWS CodeStar, você poderá alterar o conjunto padrão de recursos da AWS que o AWS CodeStar adiciona ao projeto.

### Alterações de recursos compatíveis

A tabela a seguir lista as alterações compatíveis em recursos da AWS em um projeto do AWS CodeStar.

Alteração	Observações
Adicione um estágio ao AWS CodePipeline.	Consulte <a href="#">Adicionar um estágio ao AWS CodePipeline</a> .
Alterar as configurações do ambiente do Elastic Beanstalk.	Consulte <a href="#">Alterar as configurações de ambiente do AWS Elastic Beanstalk</a> .
Altere o código ou as configurações de uma função do AWS Lambda, o perfil do IAM ou sua API no Amazon API Gateway.	Consulte <a href="#">Alterar uma função AWS Lambda no código-fonte</a> .
Adicione um recurso a um projeto do AWS Lambda e expanda permissões para criar e acessar o novo recurso.	Consulte <a href="#">Adicionar um recurso a um projeto</a> .

Alteração	Observações
Adicione mudança de tráfego com o CodeDeploy para uma função do AWS Lambda.	Consulte <a href="#">Mudar tráfego para um projeto do AWS Lambda</a> .
Adicionar suporte ao AWS X-Ray	Consulte <a href="#">Habilitar rastreamento para um projeto</a> .
Edite o arquivo buildspec.yml do projeto para adicionar uma fase de compilação de teste de unidade a ser executada pelo AWS CodeBuild.	Consulte <a href="#">Etapa 7: Adicionar um teste de unidade ao serviço da web</a> no tutorial de Projeto sem servidor.
Adicione sua própria função do IAM ao projeto.	Consulte <a href="#">Adicionar uma função do IAM a um projeto</a> .
Alterar uma definição de perfil do IAM.	Para funções definidas na pilha de aplicativos. Você não pode alterar funções definidas na cadeia de ferramentas ou em pilhas do AWS CloudFormation.
Modifique o projeto do Lambda para adicionar um endpoint.	
Modifique o projeto do EC2 para adicionar um endpoint.	
Modifique o projeto do Elastic Beanstalk para adicionar um endpoint.	
Edite o projeto para adicionar um estágio de produção e um endpoint.	Consulte <a href="#">Adicionar um estágio e uma endpoint de produção a um projeto</a> .
Use parâmetros SSM com segurança em um projeto do AWS CodeStar.	Consulte <a href="#">the section called “Use com segurança os parâmetros SSM em um projeto AWS CodeStar”</a> .

As seguintes alterações não são compatíveis.

- Alterne para outro destino de implantação (por exemplo, implantar no AWS Elastic Beanstalk em vez de AWS CodeDeploy).
- Adicione um nome amigável de endpoint da web.
- Altere o nome do repositório do CodeCommit (para um projeto do AWS CodeStar conectado ao CodeCommit).
- Para um projeto do AWS CodeStar conectado ao GitHub, desconecte o repositório do GitHub e, em seguida, reconecte o repositório a esse projeto ou conecte qualquer outro repositório a esse projeto. Você pode usar o console do (não o console do AWS CodeStar) para desconectar e reconectar ao GitHub em uma fase Origem de um pipeline. No entanto, se você reconectar a fase Source a outro repositório do GitHub e, em seguida, no painel do AWS CodeStar para o projeto, as informações nos blocos Repositório e Problemas podem estar incorretas ou desatualizadas. Desconectar o repositório do GitHub não remove as informações desse repositório do histórico de confirmação e dos blocos de chamados do GitHub no painel do AWS CodeStar para o painel do projeto. Para remover essas informações, use o site do GitHub para desativar o acesso ao GitHub do projeto do AWS CodeStar. Para revogar o acesso, no site do GitHub, use a seção Authorized OAuth Apps (Aplicativos OAuth autorizados) da página de configurações de seu perfil de conta do GitHub.
- Desconecte o repositório do CodeCommit (para um projeto do AWS CodeStar conectado ao CodeCommit) e, em seguida, reconecte o repositório a esse projeto ou conecte qualquer outro repositório a esse projeto.

## Adicionar um estágio ao AWS CodePipeline

Você pode adicionar um novo estágio a um pipeline criado pelo AWS CodeStar em um projeto. Para obter mais informações, consulte [Editar um pipeline no AWS CodePipeline](#) no Guia do usuário do AWS CodePipeline.

### Note

Se o novo estágio depender de qualquer recurso da AWS que o AWS CodeStar não criou, o pipeline poderá ser interrompido. Isso ocorre porque o perfil do IAM que o AWS CodeStar criou para o AWS CodePipeline talvez não tenha acesso a esses recursos por padrão. Para tentar conceder acesso ao AWS CodePipeline aos recursos do AWS que o AWS CodeStar não criou, você pode alterar o perfil do IAM que o AWS CodeStar criou. Isso não é

compatível porque o AWS CodeStar pode remover suas alterações no perfil do IAM quando executa verificações regulares de atualização no projeto.

## Alterar as configurações de ambiente do AWS Elastic Beanstalk

Você pode alterar as configurações de um ambiente do Elastic Beanstalk criado pelo AWS CodeStar em um projeto. Por exemplo, talvez você queira alterar o ambiente do Elastic Beanstalk padrão no projeto do AWS CodeStar de Instância única para Balanceamento de carga. Para fazer isso, edite o arquivo `template.yml` no repositório do projeto. Você também pode precisar alterar permissões para as funções de trabalho do projeto. Depois de enviar a alteração do modelo, o AWS CodeStar e o AWS CloudFormation provisionarão os recursos para você.

Para obter mais informações sobre como editar o arquivo `template.yml`, consulte [Alterar recursos do aplicativo com o arquivo `template.yml`](#). Para obter mais informações sobre como personalizar ambientes do Elastic Beanstalk, consulte [Console de Gerenciamento do Ambiente do AWS Elastic Beanstalk](#) no Guia do Desenvolvedor do AWS Elastic Beanstalk.

## Alterar uma função AWS Lambda no código-fonte

Você pode alterar o código ou as configurações de uma função do Lambda ou seu perfil do IAM ou Gateway API, que o AWS CodeStar cria em um projeto. Para fazer isso, recomendamos usar o AWS Serverless Application Model (AWS SAM) (Modelo de aplicação de tecnologia sem servidor) com o arquivo `template.yaml` no repositório do C do projeto. Esse arquivo `template.yaml` define o nome da função, o handler, o runtime, o perfil do IAM e a API no Gateway API. Para obter mais informações, consulte [Como criar aplicativos sem servidor usando o AWS SAM](#) no site do GitHub.

## Habilitar rastreamento para um projeto

O AWS X-Ray oferece rastreamento, que você pode usar para analisar o comportamento de desempenho de aplicativos distribuídos (por exemplo, latências em tempos de resposta). Depois de adicionar rastreamentos ao projeto do AWS CodeStar, você poderá usar o console do AWS X-Ray para ver visualizações de aplicativos e tempos de resposta.

### Note

Você pode usar estas etapas para os seguintes projetos, criados com as seguintes alterações de suporte do projeto:

- Qualquer projeto do Lambda.
- Para projetos do Amazon EC2 ou do Elastic Beanstalk criados depois de 3 de agosto de 2018, o AWS CodeStar provisionou um arquivo `/template.yml` no repositório do projeto.

Cada modelo do AWS CodeStar inclui um arquivo do AWS CloudFormation que modela as dependências de runtime do AWS do aplicativo, como tabelas de banco de dados e funções do Lambda. Esse arquivo é armazenado no repositório de origem no arquivo `/template.yml`.

Você pode modificar esse arquivo para adicionar rastreamento, adicionando o recurso AWS X-Ray à seção `Resources`. Em seguida, você modificar as permissões do IAM do projeto para permitir que o AWS CloudFormation crie o recurso. Para obter mais informações sobre elementos de modelo e formatação, consulte [Referência de tipos de recurso do AWS](#).

Essas são as etapas de alto nível a serem seguidas para personalizar o modelo.

1. [Etapa 1: Editar a função de operador no IAM para rastreamento](#)
2. [Etapa 2: modificar o arquivo `template.yml` para rastreamento](#)
3. [Etapa 3: confirmar e enviar a alteração do modelo para rastreamento](#)
4. [Etapa 4: monitorar a atualização da pilha do AWS CloudFormation para rastreamento](#)

## Etapa 1: Editar a função de operador no IAM para rastreamento

Você deve entrar como administrador para realizar as etapas 1 e 4. Essa etapa mostra um exemplo de permissões de edição para um projeto do Lambda.

### Note

Você pode ignorar essa etapa se seu projeto tiver sido provisionado com uma política de limite de permissões.

Para projetos criados depois de 6 de dezembro de 2018, horário do Pacífico, o AWS CodeStar terá provisionado seu projeto com uma política de limite de permissões.

1. Faça login no AWS Management Console e abra o console do AWS CodeStar em <https://console.aws.amazon.com/codestar/>.

2. Crie um projeto ou escolha um projeto existente com um `template.yml` file e abra a página Project resources (Recursos do projeto).
3. Em Project Resources (Recursos do projeto), localize a função do IAM criada para a função CodeStarWorker/Lambda na lista de recursos. O nome da função segue este formato: `role/CodeStarWorker-Project_name-lambda-Function_name`. Escolha o ARN da função.
4. A função é aberta no console do IAM. Escolha Attach policies (Anexar políticas). Procure a política `AWSXrayWriteOnlyAccess`, marque a caixa ao lado dela e escolha Attach Policy (Anexar política).

## Etapa 2: modificar o arquivo `template.yml` para rastreamento

1. Abra o console do AWS CodeStar em <https://console.aws.amazon.com/codestar/>.
2. Escolha o projeto com tecnologia sem servidor e abra a página Code (Código). No nível mais alto do repositório, localize e edite o arquivo `template.yml`. Em Resources, cole o recurso na seção Properties.

Tracing: Active

Este exemplo mostra um modelo modificado:

```
Resources:
  GetHelloWorld:
    Type: AWS::Serverless::Function
    Properties:
      Handler: index.get
      Runtime: nodejs4.3
      Tracing: Active # Enable X-Ray tracing for the function
    Role:
      Fn::ImportValue:
        !Join ['-', [!Ref 'ProjectId', !Ref 'AWS::Region', 'LambdaTrustRole']]
    Events:
      GetEvent:
        Type: Api
        Properties:
          Path: /
          Method: get
```

## Etapa 3: confirmar e enviar a alteração do modelo para rastreamento

- Confirme e envie as alterações no arquivo `template.yml`.



**Note**

Isso inicia o pipeline. Se você confirmar as alterações antes de atualizar as permissões do IAM, o pipeline será iniciado, a pilha do AWS CloudFormation encontrará erros, e a atualização da pilha será revertida. Se isso acontecer, corrija as permissões e reinicie o pipeline.

## Etapa 4: monitorar a atualização da pilha do AWS CloudFormation para rastreamento

1. A atualização da pilha do AWS CloudFormation começa quando o pipeline do projeto inicia o estágio de implantação. Para ver o status da atualização da pilha, no painel do AWS CodeStar, escolha o estágio do AWS CloudFormation no pipeline.

Se a atualização da pilha no AWS CloudFormation retornar erros, consulte diretrizes da solução de problemas em [AWS CloudFormation: criação da pilha revertida para permissões ausentes](#). Caso as permissões da função de operador não sejam encontradas, edite a política anexada à função de operador Lambda do projeto. Consulte [Etapa 1: Editar a função de operador no IAM para rastreamento](#).

2. Use o painel para visualizar a conclusão bem-sucedida do pipeline. O rastreamento já está habilitado no aplicativo.
3. Verifique se o rastreamento está habilitado exibindo os detalhes da função no console do Lambda.
4. Escolha o endpoint do aplicativo do projeto. Essa interação com o aplicativo é rastreada. Você pode visualizar informações de rastreamento no console do AWS X-Ray.

Trace list					
ID	Age	Method	Response	Response time	URL
...315e2d41	4.7 min		200	270 ms	
...88c0c37c	12.8 sec		200	23.0 ms	

## Adicionar um recurso a um projeto

Cada modelo do AWS CodeStar de todos os projetos é fornecido com um arquivo do AWS CloudFormation que modela as dependências de runtime do AWS do aplicativo, como tabelas de

banco de dados e funções do Lambda. Isso é armazenado no repositório de origem no arquivo `/template.yml`.

### Note

Você pode usar estas etapas para os seguintes projetos, criados com as seguintes alterações de suporte do projeto:

- Qualquer projeto do Lambda.
- Para projetos do Amazon EC2 ou do Elastic Beanstalk criados depois de 3 de agosto de 2018, o AWS CodeStar provisionou um arquivo `/template.yml` no repositório do projeto.

Você pode modificar esse arquivo adicionando recursos do AWS CloudFormation à seção `Resources`. Modificar o arquivo `template.yml` permite que o AWS CodeStar e o AWS CloudFormation adicione o novo recurso ao projeto. Alguns recursos exigem adicionar outras permissões à política da função de operador do CloudFormation do projeto. Para obter mais informações sobre elementos de modelo e formatação, consulte [Referência de tipos de recurso do AWS](#).

Depois de determinar quais recursos você deve adicionar ao projeto, estas serão as etapas de alto nível a serem seguidas para personalizar um modelo. Para obter uma lista de recursos do AWS CloudFormation e suas propriedades obrigatórias, consulte [Referência de tipos de recurso do AWS](#).

1. [Etapa 1: editar a função de operador do CloudFormation no IAM](#) (se necessário)
2. [Etapa 2: modificar o arquivo `template.yml`](#)
3. [Etapa 3: confirmar e enviar a alteração do modelo](#)
4. [Etapa 4: monitorar a atualização da pilha do AWS CloudFormation](#)
5. [Etapa 5: adicionar permissões de recursos com uma política em linha](#)

Use as etapas nesta seção para modificar o modelo de projeto do AWS CodeStar a fim de adicionar um recurso e expandir as permissões da função de operador do CloudFormation do projeto no IAM. Neste exemplo, o recurso [AWS::SQS::Queue](#) é adicionado ao arquivo `template.yml`. A alteração inicia uma resposta automática no AWS CloudFormation, que adiciona uma fila do Amazon Simple Queue Service ao projeto.

## Etapa 1: editar a função de operador do CloudFormation no IAM

Você deve entrar como administrador para realizar as etapas 1 e 5.

### Note

Você pode ignorar essa etapa se seu projeto tiver sido provisionado com uma política de limite de permissões.

Para projetos criados depois de 6 de dezembro de 2018, horário do Pacífico, o AWS CodeStar terá provisionado seu projeto com uma política de limite de permissões.

1. Faça login no AWS Management Console e então abra o console do AWS CodeStar em <https://console.aws.amazon.com/codestar/>.
2. Crie um projeto ou escolha um projeto existente com um `template.yml` file e abra a página Project resources (Recursos do projeto).
3. Em Project Resources (Recursos do projeto), localize a função do IAM criada para a função CodeStarWorker/AWS CloudFormation na lista de recursos. O nome da função segue este formato: `role/CodeStarWorker-Project_name-CloudFormation`.
4. A função é aberta no console do IAM. Na guia Permissions (Permissões), em Inline Policies (Políticas em linha), expanda a linha da política da função de serviço e escolha Edit Policy (Editar política).
5. Escolha a guia JSON para editar a política.

### Note

A política anexada à função de operador é `CodeStarWorkerCloudFormationRolePolicy`.

6. No campo JSON, adicione a declaração de política a seguir no elemento Statement.

```
{
  "Action": [
    "sqs:CreateQueue",
    "sqs>DeleteQueue",
    "sqs:GetQueueAttributes",
    "sqs:SetQueueAttributes",
    "sqs:ListQueues",
```

```

    "sqs:GetQueueUrl"
  ],
  "Resource": [
    "*"
  ],
  "Effect": "Allow"
}

```

7. Escolha Review policy (Revisar política) para garantir que a política não contenha erros e selecione Save changes (Salvar alterações).

## Etapa 2: modificar o arquivo template.yml

1. Abra o console do AWS CodeStar em <https://console.aws.amazon.com/codestar/>.
2. Escolha o projeto com tecnologia sem servidor e abra a página Code (Código). No nível superior do repositório, anote o local de `template.yml`.
3. Use um IDE, o console ou a linha de comando no repositório local para editar o arquivo `template.yml` no repositório. Cole o recurso na seção Resources. Neste exemplo, quando o texto a seguir é copiado, ele adiciona a seção Resources.

```

Resources:
  TestQueue:
    Type: AWS::SQS::Queue

```

Este exemplo mostra um modelo modificado:

```

Resources:
  HelloWorld:
    Type: AWS::Serverless::Function
    Properties:
      Handler: index.handler
      Runtime: python3.6
      Role:
        Fn::ImportValue:
          !Join ['-', [!Ref 'ProjectId', !Ref 'AWS::Region', 'LambdaTrustRole']]
    Events:
      GetEvent:
        Type: Api
        Properties:
          Path: /
          Method: get
      PostEvent:
        Type: Api
        Properties:
          Path: /
          Method: post
  TestQueue:
    Type: AWS::SQS::Queue

```

## Etapa 3: confirmar e enviar a alteração do modelo

- Confirme e envie as alterações no arquivo `template.yml` salvas na etapa 2.

### Note

Isso inicia o pipeline. Se você confirmar as alterações antes de atualizar as permissões do IAM, o pipeline será iniciado e a pilha do AWS CloudFormation encontrará erros, o que fará a atualização da pilha ser revertida. Se isso acontecer, corrija as permissões e reinicie o pipeline.

## Etapa 4: monitorar a atualização da pilha do AWS CloudFormation

1. Quando o pipeline do projeto inicia o estágio de implantação, a atualização da pilha do AWS CloudFormation começa. Você pode escolher o estágio do AWS CloudFormation no pipeline no painel AWS CodeStar para ver a atualização da pilha.

Solucionar problemas:

A atualização da pilha falhará se as permissões de recursos necessárias não forem encontradas. Veja o status da falha na visualização do painel do AWS CodeStar do pipeline do projeto.

Escolha o link CloudFormation no estágio de implantação do pipeline para solucionar problemas da falha no console do AWS CloudFormation. No console, na lista Events (Eventos), escolha o projeto para visualizar detalhes de criação da pilha. Há uma mensagem com os detalhes da falha. Neste exemplo, a permissão `sqs:CreateQueue` não foi encontrada.

▶ 08:37:11 UTC-0700	UPDATE_ROLLBACK_COMPLETE	AWS::CloudFormation::Stack	awscodestar-dk-sqs-red-lambda	
08:37:11 UTC-0700	DELETE_COMPLETE	AWS::SQS::Queue	TestQueue	
▶ 08:37:09 UTC-0700	UPDATE_ROLLBACK_COMPLETE_CLEANUP_IN_PROGRESS	AWS::CloudFormation::Stack	awscodestar-dk-sqs-red-lambda	
▶ 08:37:06 UTC-0700	UPDATE_COMPLETE	AWS::Lambda::Function	HelloWorld	
▶ 08:37:03 UTC-0700	UPDATE_ROLLBACK_IN_PROGRESS	AWS::CloudFormation::Stack	awscodestar-dk-sqs-red-lambda	The following resource(s) failed to create: [TestQueue]. The following resource(s) failed to update: [HelloWorld].
▶ 08:37:02 UTC-0700	UPDATE_FAILED	AWS::Lambda::Function	HelloWorld	Resource update cancelled
08:37:01 UTC-0700	CREATE_FAILED	AWS::SQS::Queue	TestQueue	API: sqs:CreateQueue Access to the resource https://sqs.us-west-2.amazonaws.com/ is denied.
08:37:01 UTC-0700	CREATE_IN_PROGRESS	AWS::SQS::Queue	TestQueue	

Adicione todas as permissões ausentes editando a política anexada à função de operador AWS CloudFormation do projeto. Consulte [Etapa 1: editar a função de operador do CloudFormation no IAM](#).

2. Após a execução bem-sucedida do pipeline, os recursos são criados na pilha do AWS CloudFormation. Na lista Resources (Recursos) no AWS CloudFormation, visualize os recursos criados para o projeto. Neste exemplo, a fila TestQueue está listada na seção Resources (Recursos).

O URL da fila está disponível em AWS CloudFormation. O URL da fila segue este formato:

```
https://{REGION_ENDPOINT}/queue.|api-domain|/{YOUR_ACCOUNT_NUMBER}/  
{YOUR_QUEUE_NAME}
```

Para obter mais informações, consulte [Enviar uma mensagem do Amazon SQS](#), [Receber uma mensagem de uma fila do Amazon SQS](#) e [Excluir uma mensagem de uma fila do Amazon SQS](#).

## Etapa 5: adicionar permissões de recursos com uma política em linha

Conceda a membros da equipe acesso ao novo recurso adicionando a política em linha à função do usuário. Nem todos os recursos exigem que você adicione permissões. Para realizar as etapas a seguir, você deve entrar no console como um usuário raiz, um usuário do administrador na conta ou um usuário do IAM ou um usuário federado com a política gerenciada AdministratorAccess ou equivalente.

Para usar o editor de políticas JSON para criar uma política

1. Faça login no AWS Management Console e abra o console do IAM em <https://console.aws.amazon.com/iam/>.
2. No painel de navegação à esquerda, escolha Políticas.

Se essa for a primeira vez que escolhe Políticas, a página Bem-vindo às políticas gerenciadas será exibida. Escolha Get Started.

3. Na parte superior da página, escolha Create policy (Criar política).
4. Na seção Editor de políticas, escolha a opção JSON.
5. Insira o seguinte documento de política JSON:

```
{  
  "Action": [  
    "sqs:CreateQueue",  
    "sqs>DeleteQueue",  
    "sqs:GetQueueAttributes",
```

```
"sqs:SetQueueAttributes",
"sqs:ListQueues",
"sqs:GetQueueUrl"
],
"Resource": [
  "*"
],
"Effect": "Allow"
}
```

## 6. Escolha Next (Próximo).

### Note

Você pode alternar entre as opções de editor Visual e JSON a qualquer momento. Porém, se você fizer alterações ou escolher Avançar no editor Visual, o IAM poderá reestruturar a política a fim de otimizá-la para o editor visual. Para obter mais informações, consulte [Reestruturação de política](#) no Manual do usuário do IAM.

7. Na página Revisar e criar, insira um Nome de política e uma Descrição (opcional) para a política que você está criando. Revise Permissões definidas nessa política para ver as permissões que são concedidas pela política.
8. Escolha Create Policy (Criar política) para salvar sua nova política.

## Adicionar uma função do IAM a um projeto

Desde 6 de dezembro de 2018, horário do Pacífico você pode definir suas próprias funções e políticas na pilha do aplicativo (template.yml). Para reduzir os riscos de escalonamento de privilégios e ações destrutivas, você deve definir o limite de permissões específicas ao projeto para cada entidade do IAM que criar. Se você tiver um projeto do Lambda com várias funções, uma melhor prática é criar um perfil do IAM para cada função.

Para adicionar uma função do IAM ao projeto

1. Edite o arquivo `template.yml` de seu projeto.
2. Na seção `Resources:`, adicione o recurso do IAM usando o formato no exemplo a seguir:

```
SampleRole:
```

```
Description: Sample Lambda role
Type: AWS::IAM::Role
Properties:
  AssumeRolePolicyDocument:
    Statement:
      - Effect: Allow
        Principal:
          Service: [lambda.amazonaws.com]
        Action: sts:AssumeRole
  ManagedPolicyArns:
    - arn:aws:iam::aws:policy/service-role/AWSLambdaBasicExecutionRole
  PermissionsBoundary: !Sub 'arn:${AWS::Partition}:iam::${AWS::AccountId}:policy/CodeStar_${ProjectId}_PermissionsBoundary'
```

3. Libere suas alterações por meio do pipeline e verifique o sucesso.

## Adicionar um estágio e uma endpoint de produção a um projeto

Use os procedimentos desta seção para adicionar um novo estágio de produção (Prod) ao pipeline e um estágio de aprovação manual entre os estágios de implantação e produção do pipeline. Isso cria uma pilha de recursos adicionais quando o pipeline do projeto é executado.

### Note

Você pode usar esses procedimentos se:

- Para projetos criados depois de 3 de agosto de 2018, o AWS CodeStar terá provisionado seu projeto do Amazon EC2, do Elastic Beanstalk ou do Lambda com um arquivo do `/template.yml` no repositório do projeto.
- Para projetos criados depois de 6 de dezembro de 2018, horário do Pacífico, o AWS CodeStar terá provisionado seu projeto com uma política de limite de permissões.

Todos os projetos do AWS CodeStar usam um arquivo de modelo do AWS CloudFormation que modela as dependências de runtime do AWS do aplicativo, como instâncias do Linux e funções do Lambda. O arquivo `/template.yml` é armazenado em seu repositório de origem.

No arquivo `/template.yml`, use o parâmetro `Stage` para adicionar uma pilha de recursos para um novo estágio no pipeline do projeto.



Stage:

Type: String

Description: The name for a project pipeline stage, such as Staging or Prod, for which resources are provisioned and deployed.

Default: ''

O parâmetro Stage é aplicado a todos os recursos nomeados com o ID de projeto referenciado no recurso. Por exemplo, o seguinte nome de função é um recurso nomeado no modelo:

```
RoleName: !Sub 'CodeStar-${ProjectId}-WebApp${Stage}'
```

## Pré-requisitos

Use as opções do modelo no console do AWS CodeStar para criar um projeto.

Certifique-se de que o usuário do IAM tenha as permissões a seguir:

- iam:PassRole na função AWS CloudFormation do projeto.
- iam:PassRole na função da cadeia de ferramentas do projeto.
- cloudformation:DescribeStacks
- cloudformation:ListChangeSets

Somente para projetos do Elastic Beanstalk ou do Amazon EC2:

- codedeploy:CreateApplication
- codedeploy:CreateDeploymentGroup
- codedeploy:GetApplication
- codedeploy:GetDeploymentConfig
- codedeploy:GetDeploymentGroup
- elasticloadbalancing:DescribeTargetGroups

## Tópicos

- [Etapa 1: Criar um novo grupo de implantação no CodeDeploy \(somente projetos do Amazon EC2\)](#)
- [Etapa 2: Adicionar um novo estágio de pipeline para o estágio de produção](#)

- [Etapa 3: Adicionar um estágio de aprovação manual](#)
- [Etapa 4: Enviar uma alteração por push e monitorar a atualização da pilha do AWS CloudFormation](#)

## Etapa 1: Criar um novo grupo de implantação no CodeDeploy (somente projetos do Amazon EC2)

Você escolhe o aplicativo do CodeDeploy e, em seguida, adiciona um novo grupo de implantação associado à nova instância.

### Note

Se seu projeto for um projeto do Lambda ou do Elastic Beanstalk, você poderá ignorar essa etapa.

1. Abra o console do CodeDeploy em <https://console.aws.amazon.com/codedeploy>.
2. Escolha o aplicativo CodeDeploy que foi gerado para seu projeto quando ele foi criado no AWS CodeStar.
3. Em Deployment groups, escolha Create deployment group.
4. Em Deployment group name (Nome do grupo de implantação), insira **<project-id>-prod-Env**.
5. Em Service role (Função de serviço), escolha a função de operador de cadeia de ferramentas para o projeto do AWS CodeStar.
6. Em Deployment type (Tipo de implantação), selecione In-place (No local).
7. Em Environment configuration (Configuração do ambiente), escolha a guia Amazon EC2 Instances (Instâncias do Amazon EC2).
8. No grupo de tags, em Key (Chave), escolha `aws:cloudformation:stack-name`. Em Value (Valor), escolha `awscodestar-<projectid>-infrastructure-prod` (a pilha a ser criada para a ação GenerateChangeSet).
9. Em Deployment settings (Configurações da implantação), escolha `CodeDeployDefault.AllAtOnce`.
10. Limpe Choose a load balancer (Escolher um load balancer).
11. Selecione Create deployment group (Criar grupo de implantação).

Agora, o segundo grupo de implantação foi criado.

## Etapa 2: Adicionar um novo estágio de pipeline para o estágio de produção

Adicione um estágio com o mesmo conjunto de ações de implantação como o estágio de implantação do projeto. Por exemplo, o novo estágio de produção para um projeto do Amazon EC2 deve ter as mesmas ações que o estágio de implantação criado para o projeto.

Para copiar parâmetros e campos do estágio de implantação

1. No painel do projeto do AWS CodeStar, escolha Detalhes do pipeline para abrir o pipeline no console do CodePipeline.
2. Escolha Editar.
3. No estágio de implantação, escolha Edit stage (Editar estágio).
4. Escolha o ícone de edição na ação GenerateChangeSet. Anote os valores dos seguintes campos. Você usa esses valores ao criar a nova ação.
  - Nome da stack
  - Change set name (Nome do conjunto de alterações)
  - Modelo
  - Configuração do modelo
  - Input artifacts (Artefatos de entrada)
5. Expanda Advanced (Avançado) e, em Parameters (Parâmetros), copie os parâmetros de seu projeto. Você cola esses parâmetros em sua nova ação. Por exemplo, copie os parâmetros mostrados aqui em formato JSON:

- Projetos Lambda:

```
{
  "ProjectId": "MyProject"
}
```

- Projetos do Amazon EC2:

```
{
  "ProjectId": "MyProject",
```

```

    "InstanceType": "t2.micro",
    "WebAppInstanceProfile": "awscodestar-MyProject-WebAppInstanceProfile-EXAMPLEY5VSFS",
    "ImageId": "ami-EXAMPLE1",
    "KeyPairName": "my-keypair",
    "SubnetId": "subnet-EXAMPLE",
    "VpcId": "vpc-EXAMPLE1"
  }

```

- **Projetos do Elastic Beanstalk**


```

{
  "ProjectId": "MyProject",
  "InstanceType": "t2.micro",
  "KeyPairName": "my-keypair",
  "SubnetId": "subnet-EXAMPLE",
  "VpcId": "vpc-EXAMPLE",
  "SolutionStackName": "64bit Amazon Linux 2018.03 v3.0.5 running Tomcat 8 Java 8",
  "EBTrustRole": "CodeStarWorker-myproject-EBService",
  "EBInstanceProfile": "awscodestar-myproject-EBInstanceProfile-11111EXAMPLE"
}

```

6. No painel de edição do estágio, escolha **Cancel** (Cancelar).


Para criar uma ação `GenerateChangeSet` em seu novo estágio de produção

 **Note**

Depois de adicionar a nova ação e enquanto ainda estiver no modo de edição, se você reabrir a nova ação para edição, alguns dos campos poderão não ser exibidos. Você também poderá ver o seguinte: A pilha `stack-name` não existe. Esse erro não impede que você salve o pipeline. No entanto, para restaurar os campos ausentes, é necessário excluir a nova ação e adicioná-la novamente. Depois de salvar e executar o pipeline, a pilha é reconhecida e o erro não aparece novamente.

1. Se o pipeline ainda não estiver exibido, no painel do projeto do AWS CodeStar, escolha **Pipeline Details** (Detalhes do pipeline) para abrir o pipeline no console.
2. Escolha **Editar**.

3. Na parte inferior do diagrama, escolha + Add stage (+ Adicionar estágio).
4. Insira um nome de estágio (por exemplo, **Prod**) e escolha + Add action group (+ Adicionar grupo de ações).
5. Em Action name (Nome da ação), informe um nome (por exemplo, **GenerateChangeSet**).
6. Em Action provider, escolha AWS CloudFormation.
7. Em Action mode (Modo de ação), escolha Create or replace a change set (Criar ou substitui um conjunto de alterações).
8. Em Stack name (Nome da pilha), digite um novo nome para a pilha do AWS CloudFormation que deve ser criada por essa ação. Comece com um nome idêntico ao nome da pilha de implantação e, em seguida, adicione **-prod**:
  - Projetos Lambda: `awscodestar-<project_name>-lambda-prod`
  - Projetos do Amazon EC2 ou do Elastic Beanstalk: `awscodestar-<project_name>-infrastructure-prod`


 Note

Para que não haja falha, o nome da pilha deve começar exatamente com **awscodestar-<project\_name>-**.

9. Em Change set name (Nome do conjunto de alterações), insira o mesmo nome do conjunto de alterações fornecido no estágio de implantação existente (por exemplo, **pipeline-changeset**).
10. Em Input artifacts (Artefatos de entrada), escolha o artefato de criação.
11. Em Template (Modelo), insira o mesmo nome do modelo de alterações fornecido no estágio de implantação existente (por exemplo, **<project-ID>-BuildArtifact::template.yml**).
12. Em Template configuration (Configuração de modelo), insira o mesmo arquivo de configuração do modelo de alterações fornecido no estágio de implantação existente (por exemplo, **<project-ID>-BuildArtifact::template-configuration.json**).
13. Em Capabilities (Recursos), escolha `CAPABILITY_NAMED_IAM`.
14. Em Role name (Nome da função), escolha o nome da função de operador do AWS CloudFormation de seu projeto.


15. Expanda **Advanced** (Avançado) e, em **Parameters** (Parâmetros), cole os parâmetros de seu projeto. Inclua o parâmetro **Stage**, mostrado aqui no formato JSON, para um projeto do Amazon EC2:

```
{  
  
  "ProjectId": "MyProject",  
  "InstanceType": "t2.micro",  
  "WebAppInstanceProfile": "awscodestar-MyProject-WebAppInstanceProfile-  
EXAMPLEY5VSFS",  
  "ImageId": "ami-EXAMPLE1",  
  "KeyPairName": "my-keypair",  
  "SubnetId": "subnet-EXAMPLE",  
  "VpcId": "vpc-EXAMPLE1",  
  "Stage": "Prod"  
}
```

 **Note**

Certifique-se de colar todos os parâmetros do projeto, não apenas os novos parâmetros ou os parâmetros que você deseja alterar.

16. Escolha **Save** (Salvar).
17. No painel do AWS CodePipeline, selecione **Save pipeline change** (Salvar alteração do pipeline) e, em seguida, **Save change** (Salvar alteração).

 **Note**


Pode ser exibida uma mensagem notificando você sobre a exclusão e adição de recursos de detecção de alterações. Reconheça a mensagem e continue na próxima etapa deste tutorial.

Visualize seu pipeline atualizado.

Para criar uma ação **ExecuteChangeSet** no novo estágio de produção

1. Se ainda não estiver visualizando o pipeline, no painel do projeto do AWS CodeStar, escolha **Pipeline Details** (Detalhes do pipeline) para abrir o pipeline no console.

2. Escolha Editar.
3. No novo estágio de produção, depois da nova ação GenerateChangeSet, escolha +Add action group (+Adicionar grupo de ações).
4. Em Action name (Nome da ação), informe um nome (por exemplo, **ExecuteChangeSet**).
5. Em Action provider, escolha AWS CloudFormation.
6. Em Action mode (Modo de ação), escolha Execute a change set (Executar um conjunto de alterações).
7. Em Stack Name (Nome da pilha), digite o novo nome para a pilha do AWS CloudFormation que você informou na ação GenerateChangeSet (por exemplo, **awscodestar-`<project-ID>`-infrastructure-prod**).
8. Em Alterar nome do conjunto, insira o mesmo nome do conjunto usado no estágio de implantação (por exemplo, **pipeline-changeset**).
9. Escolha Done (Concluído).
10. No painel do AWS CodePipeline, selecione Save pipeline change (Salvar alteração do pipeline) e, em seguida, Save change (Salvar alteração).

 Note

Pode ser exibida uma mensagem notificando você sobre a exclusão e adição de recursos de detecção de alterações. Reconheça a mensagem e continue na próxima etapa deste tutorial.

Visualize seu pipeline atualizado.

Para criar uma ação de implantação do CodeDeploy no novo estágio de produção (somente projetos do Amazon EC2)

1. Depois das novas ações no estágio de produção, selecione + Action (+ Ação).
2. Em Action name (Nome da ação), informe um nome (por exemplo, **Deploy**).
3. Em Action provider, selecione AWS CodeDeploy.
4. Em Nome do aplicativo, escolha o nome do aplicativo CodeDeploy para seu projeto.
5. Em Grupo de implantação, escolha o nome do novo grupo de implantação do CodeDeploy que você criou na etapa 2.

6. Em Input artifacts (Artefatos de entrada), escolha o mesmo artefato de criação usado no estágio existente.
7. Escolha Done (Concluído).
8. No painel do AWS CodePipeline, selecione Save pipeline change (Salvar alteração do pipeline) e, em seguida, Save change (Salvar alteração). Visualize seu pipeline atualizado.

### Etapa 3: Adicionar um estágio de aprovação manual

Como uma melhor prática, adicione um estágio de aprovação manual na frente do novo estágio de produção.

1. No canto superior esquerdo, selecione Edit (Editar).
2. No diagrama do pipeline, entre os estágios de implantação e de produção, escolha + Add stage (+ Adicionar estágio).
3. Em Edit stage (Editar estágio), digite um nome de estágio (por exemplo, **Approval**) e escolha + Add action group (+ Adicionar grupo de ação).
4. Em Action name (Nome da ação), informe um nome (por exemplo, **Approval**).
5. Em Tipo de aprovação, selecione Aprovação manual.
6. (Opcional) Em Configuration (Configuração), em SNS Topic ARN (ARN do tópico do SNS), escolha o tópico do SNS que você criou e no qual se inscreveu.
7. Escolha Add Action.
8. No painel do AWS CodePipeline, selecione Save pipeline change (Salvar alteração do pipeline) e, em seguida, Save change (Salvar alteração). Visualize seu pipeline atualizado.
9. Para enviar suas alterações e iniciar uma compilação do pipeline, selecione Release change (Liberar alteração) e, depois, Release (Liberar).

### Etapa 4: Enviar uma alteração por push e monitorar a atualização da pilha do AWS CloudFormation

1. Enquanto seu pipeline está em execução, você pode usar as etapas aqui para acompanhar a criação da pilha e do endpoint para sua nova etapa.
2. Quando o pipeline iniciar o estágio de implantação, a atualização da pilha do AWS CloudFormation será iniciada. Você pode escolher o estágio do AWS CloudFormation no



pipeline no painel do AWS CodeStar para ver a notificação da atualização da pilha. Para visualizar detalhes de criação da pilha, no console, escolha o projeto na lista Events (Eventos).

3. Após a conclusão bem-sucedida do pipeline, os recursos são criados na pilha do AWS CloudFormation. No console do AWS CloudFormation, escolha a pilha da infraestrutura de seu projeto. Os nomes de pilhas seguem este formato:

- Projetos Lambda: `awscodestar-<project_name>-lambda-prod`
- Projetos do Amazon EC2 ou do Elastic Beanstalk: `awscodestar-<project_name>-infrastructure-prod`

Na lista Resources (Recursos) no console do AWS CloudFormation, visualize o recurso criado para o projeto. Neste exemplo, a nova instância do Amazon EC2 é exibida na seção Recursos.

4. Acesse o endpoint do estágio de produção:

- Para um projeto do Elastic Beanstalk, abra a nova pilha no console do AWS CloudFormation e expanda Recursos. Escolha o aplicativo do Elastic Beanstalk. O link abre no console do Elastic Beanstalk. Escolha Environments (Ambientes). Escolha a URL em URL para abrir o endpoint em um navegador.
- Para um projeto do Lambda, abra a nova pilha no console do AWS CloudFormation e expanda Recursos. Escolha o recurso do API Gateway. O link é aberto no console do API Gateway. Escolha Stages (Estágios). Escolha a URL em Invoke URL (Invocar URL) para abrir o endpoint em um navegador.
- Para um projeto do Amazon EC2, escolha a nova instância do Amazon EC2 na lista de recursos do projeto no console do AWS CodeStar. O link é aberto na página Instância do console do Amazon EC2. Escolha a guia Description (Descrição), copie a URL em Public DNS (IPv4) (DNS público (IPv4)) e abra a URL em um navegador.

5. Verifique se a alteração foi implantada.

## Use com segurança os parâmetros SSM em um projeto AWS CodeStar

Vários clientes armazenam segredos, como credenciais, em parâmetros do [Repositório de parâmetros do Gerenciador de Sistemas](#). Agora você pode usar esses parâmetros com segurança em um AWS CodeStar projeto. Por exemplo, talvez você queira usar parâmetros SSM em sua especificação de compilação para CodeBuild ou ao definir recursos do aplicativo em sua pilha de conjuntos de ferramentas (template.yml).

Para usar parâmetros de SSM em um CodeStar projeto da AWS, você deve marcar manualmente os parâmetros com o ARN do CodeStar projeto da AWS. Você também deve fornecer permissões apropriadas à função de trabalhador da CodeStar cadeia de ferramentas da AWS para acessar os parâmetros que você marcou.

## Antes de começar

- [Crie um novo](#) ou identifique um parâmetro do Systems Manager existente que contenha as informações que deseja acessar.
- Identifique qual CodeStar projeto da AWS você deseja usar ou [crie um novo projeto](#).
- Anote o ARN do CodeStar projeto. Ele é semelhante a: `arn:aws:codestar:region-id:account-id:project/project-id`.

## Marque um parâmetro com o ARN do CodeStar projeto da AWS

Consulte [Marcação de parâmetros do gerenciador de sistemas](#) para obter instruções passo a passo.

1. Em Chave, insira `awscodestar:projectArn`.
2. Em Valor, insira o ARN do projeto de CodeStar: `arn:aws:codestar:region-id:account-id:project/project-id`
3. Selecione Save (Salvar).

Agora você pode fazer referência ao parâmetro SSM em seu arquivo `template.yml`. Se quiser usá-lo com uma função de operador da cadeia de ferramentas, será necessário conceder permissões adicionais.

Conceda permissões para usar parâmetros marcados em seu conjunto de ferramentas de CodeStar projetos da AWS

### Note

Estas etapas se aplicam somente a projetos criados após 6 de dezembro de 2018, horário do Pacífico.

1. Abra o painel do CodeStar projeto da AWS para o projeto que você deseja usar.

2. Clique em Project (Projeto) para visualizar a lista de recursos criados e localize a função de operador da cadeia de ferramentas. É um recurso do IAM com um nome no formato: `role/CodeStarWorker-project-id-ToolChain`.
3. Clique no ARN para abri-lo no console do IAM.
4. Localize o ToolChainWorkerPolicy e expanda-o, se necessário.
5. Clique em Edit Policy (Editar política).
6. Em Action:, adicione a seguinte linha:  

```
ssm:GetParameter*
```
7. Clique em Review policy (Revisar política) e clique em Save changes (Salvar alterações).


Para projetos criados antes de 6 de dezembro de 2018, horário do Pacífico será necessário adicionar as seguintes permissões às funções de operador para cada serviço.

```
{
  "Action": [
    "ssm:GetParameter*"
  ],
  "Resource": "*",
  "Effect": "Allow",
  "Condition": {
    "StringEquals": {
      "ssm:ResourceTag/awscodestar:projectArn": "arn:aws:codestar:region-id:account-id:project/project-id"
    }
  }
}
```

## Mudar tráfego para um projeto do AWS Lambda

O AWS CodeDeploy oferece suporte a implantações de versão de função para funções do AWS Lambda nos projetos com tecnologia sem servidor do AWS CodeStar. Uma implantação do AWS Lambda muda o tráfego recebido de uma função do Lambda existente para uma versão de função do Lambda atualizada. Convém testar uma função do Lambda atualizada implantando uma versão separada e revertendo a implantação para a primeira versão, se necessário.

Use as etapas nesta seção para modificar o modelo de projeto do AWS CodeStar e atualizar as permissões do IAM das funções CodeStarWorker. Essa tarefa inicia uma resposta automática no AWS CloudFormation que cria funções do AWS Lambda com alias e instrui o AWS CodeDeploy a mudar o tráfego para um ambiente atualizado.

 Note

Conclua estas etapas somente se você tiver criado seu projeto do AWS CodeStar antes de 12 de dezembro de 2018.

O AWS CodeDeploy tem três opções de implantação que permitem mudar o tráfego para versões da função do AWS Lambda no aplicativo:

- **Canário:** o tráfego é deslocado em dois incrementos. Você pode escolher entre opções do canary predefinidas que especificam a porcentagem de tráfego deslocada para a versão da função do Lambda atualizada no primeiro incremento e no intervalo, em minutos, antes que o tráfego restante seja deslocado no segundo incremento.
- **Linear:** o tráfego é deslocado em incrementos iguais com um número igual de minutos entre cada incremento. Você pode escolher entre opções lineares predefinidas que especificam a porcentagem de tráfego deslocado em cada incremento e o número de minutos entre cada incremento. O tráfego é deslocado em incrementos iguais com um número igual de minutos entre cada incremento. Você pode escolher entre opções lineares predefinidas que especificam a porcentagem de tráfego deslocado em cada incremento e o número de minutos entre cada incremento.
- **All-at-once (Tudo de uma vez):** todo o tráfego é deslocado da função do Lambda original para a versão da função do Lambda atualizada de uma única vez.

#### Tipo de preferência de implantação

Canary10Percent30Minutes

Canary10Percent5Minutes

Canary10Percent10Minutes

Canary10Percent15Minutes

## Tipo de preferência de implantação

Linear10PercentEvery10Minutes

Linear10PercentEvery1Minute

Linear10PercentEvery2Minutes

Linear10PercentEvery3Minutes

AllAtOnce

Para obter mais informações sobre implantações do AWS CodeDeploy em uma plataforma de computação do AWS Lambda, consulte [Implantações em uma plataforma de computação do AWS Lambda](#).

Para obter mais informações sobre o AWS SAM, consulte [AWS Serverless Application Model \(AWS SAM\)](#) (Modelo de aplicação de tecnologia sem servidor) no GitHub.

Pré-requisitos:

Quando você cria um projeto com tecnologia sem servidor, selecione qualquer modelo com a plataforma de computação Lambda. Você deve entrar como administrador para realizar as etapas de 4 a 6.

Etapa 1: modificar o modelo SAM para adicionar parâmetros de implantação da versão do AWS Lambda

1. Abra o console do AWS CodeStar em <https://console.aws.amazon.com/codestar/>.
2. Crie um projeto ou escolha um projeto existente com um arquivo `template.yml` e abra a página Code (Código). No nível mais alto do repositório, observe o local do modelo SAM chamado `template.yml` a ser modificado.
3. Abra o arquivo `template.yml` no IDE ou o repositório local. Copie o texto a seguir para adicionar uma seção `Globals` ao arquivo. O texto de exemplo neste tutorial escolhe a opção `Canary10Percent5Minutes`.

```
Globals:
  Function:
    AutoPublishAlias: live
```

```
DeploymentPreference:
  Enabled: true
  Type: Canary10Percent5Minutes
```

Este exemplo mostra um modelo modificado após a adição da seção `Globals`:

```
AWSTemplateFormatVersion: 2010-09-09
Transform:
- AWS::Serverless-2016-10-31
- AWS::CodeStar

Parameters:
  ProjectId:
    Type: String
    Description: CodeStar projectId used to associate new resources to team members

Globals:
  Function:
    AutoPublishAlias: live
    DeploymentPreference:
      Enabled: true
      Type: Canary10Percent5Minutes

Resources:
  HelloWorld:
    Type: AWS::Serverless::Function
    Properties:
      Handler: index.handler
      Runtime: python3.6
      Role:
        Fn::ImportValue:
          !Join ['-', [!Ref 'ProjectId', !Ref 'AWS::Region', 'LambdaTrustRole']]
    Events:
```

Para obter mais informações, consulte o guia de referência [Seções globais](#) dos modelos SAM.

Etapa 2: editar a função do AWS CloudFormation para adicionar permissões


1. Faça login no AWS Management Console e abra o console do AWS CodeStar em <https://console.aws.amazon.com/codestar/>.

#### Note

Você deve fazer login no AWS Management Console usando as credenciais associadas ao usuário do IAM criado ou identificado em [Configuração AWS CodeStar](#). Esse usuário deve ter a política gerenciada pela AWS denominada **AWSCodeStarFullAccess** anexada.

2. Escolha o projeto com tecnologia sem servidor existente e abra a página Project resources (Recursos do projeto).
3. Em Resources (Recursos), escolha a função do IAM criada para a função CodeStarWorker/AWS CloudFormation. A função é aberta no console do IAM.

4. Na aba Permissions, em Inline Policies, na linha da sua política da função de serviço, selecione Edit Policy. Escolha a guia JSON para editar a política no formato JSON.

 Note

Sua função de serviço é chamada CodeStarWorkerCloudFormationRolePolicy.

5. No campo JSON, adicione as declarações de política dentro do elemento Statement. Substitua os espaços reservados *region* e *id* pela sua região e o ID da sua conta.

```
{
  "Action": [
    "s3:GetObject",
    "s3:GetObjectVersion",
    "s3:GetBucketVersioning"
  ],
  "Resource": "*",
  "Effect": "Allow"
},
{
  "Action": [
    "s3:PutObject"
  ],
  "Resource": [
    "arn:aws:s3:::codepipeline*"
  ],
  "Effect": "Allow"
},
{
  "Action": [
    "lambda:*"
  ],
  "Resource": [
    "arn:aws:lambda:region:id:function:*"
  ],
  "Effect": "Allow"
},
{
  "Action": [
    "apigateway:*"
  ],
  "Resource": [
```

```

    "arn:aws:apigateway:region::*"
  ],
  "Effect": "Allow"
},
{
  "Action": [
    "iam:GetRole",
    "iam:CreateRole",
    "iam>DeleteRole",
    "iam:PutRolePolicy"
  ],
  "Resource": [
    "arn:aws:iam::id:role/*"
  ],
  "Effect": "Allow"
},
{
  "Action": [
    "iam:AttachRolePolicy",
    "iam>DeleteRolePolicy",
    "iam:DetachRolePolicy"
  ],
  "Resource": [
    "arn:aws:iam::id:role/*"
  ],
  "Effect": "Allow"
},
{
  "Action": [
    "iam:PassRole"
  ],
  "Resource": [
    "*"
  ],
  "Effect": "Allow"
},
{
  "Action": [
    "codedeploy:CreateApplication",
    "codedeploy>DeleteApplication",
    "codedeploy:RegisterApplicationRevision"
  ],
  "Resource": [
    "arn:aws:codedeploy:region:id:application:*"
  ]
}

```



```

    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "codedeploy:CreateDeploymentGroup",
      "codedeploy:CreateDeployment",
      "codedeploy>DeleteDeploymentGroup",
      "codedeploy:GetDeployment"
    ],
    "Resource": [
      "arn:aws:codedeploy:region:id:deploymentgroup:*"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "codedeploy:GetDeploymentConfig"
    ],
    "Resource": [
      "arn:aws:codedeploy:region:id:deploymentconfig:*"
    ],
    "Effect": "Allow"
  }
}

```

6. Selecione Review policy (Revisar política) para garantir que a política não contenha erros. Quando a política não tiver erros, selecione Save changes (Salvar alterações).

Etapa 3: confirmar e enviar a alteração do modelo para iniciar a migração da versão do AWS Lambda

1. Confirme e envie as alterações no arquivo `template.yml` salvas na etapa 1.

#### Note

Isso inicia o pipeline. Se você confirmar as alterações antes de atualizar as permissões do IAM, o pipeline será iniciado, a pilha do AWS CloudFormation encontrará erros, e a atualização da pilha será revertida. Se isso acontecer, reinicie o pipeline depois que as permissões forem corrigidas.

2. A atualização da pilha AWS CloudFormation começa quando o pipeline do projeto inicia o estágio Implantar. Para ver a notificação de atualização da pilha quando a implantação é iniciada, no painel do AWS CodeStar, selecione o estágio do AWS CloudFormation no pipeline.

Durante a atualização de pilha, o AWS CloudFormation atualiza automaticamente os recursos do projeto da seguinte forma:

- O AWS CloudFormation processa o arquivo `template.yml` criando as funções do Lambda com alias, os ganchos de eventos e os recursos.
- O AWS CloudFormation chama o Lambda para criar a nova versão da função.
- O AWS CloudFormation cria um arquivo AppSpec e chama o AWS CodeDeploy para migrar o tráfego.

Para obter mais informações sobre como publicar funções do Lambda com o alias no SAM, consulte a referência de modelo [Modelo de Aplicação de tecnologia sem servidor do AWS](#). Para obter mais informações sobre ganchos de eventos e recursos no arquivo AppSpec do AWS CodeDeploy, consulte a [Seção de 'recursos' do AppSpec \(Somente implantações do AWS Lambda\)](#) e a [seção de 'ganchos' do AppSpec para uma implantação do AWS Lambda](#).

3. Após a conclusão bem-sucedida do pipeline, os recursos são criados na pilha do AWS CloudFormation. Na página Project (Projeto), na lista Project Resources (Recursos do projeto), veja o aplicativo do AWS CodeDeploy, o grupo de implantação do AWS CodeDeploy e os recursos de função do serviço AWS CodeDeploy para o projeto.
4. Para criar uma nova versão, faça uma alteração na função do Lambda no repositório. A nova implantação começa e migra o tráfego de acordo com o tipo de implantação indicado no modelo do SAM. Para visualizar o status do tráfego que está sendo migrado para a nova versão, na página Project (Projeto), na lista Project Resources (Recursos do projeto), escolha a implantação do AWS CodeDeploy.
5. Para visualizar detalhes sobre cada revisão, em Revisions (Revisões), escolha o link para o grupo de implantação do AWS CodeDeploy.
6. No diretório de trabalho local, você pode fazer alterações na função do AWS Lambda e confirmar a alteração feita no repositório do projeto. O AWS CloudFormation oferece suporte ao AWS CodeDeploy durante o gerenciamento da próxima revisão da mesma forma. Para obter mais informações sobre como reimplantar, parar ou reverter uma implantação do Lambda, consulte [Implantações em uma plataforma de computação do AWS Lambda](#).

# Transição do projeto do AWS CodeStar para Produção

Depois de criar o aplicativo usando um projeto do AWS CodeStar e ver o que o AWS CodeStar fornece, você talvez queira fazer a transição do projeto para uso em produção. Uma maneira de fazer isso é replicar os recursos do AWS do aplicativo fora do AWS CodeStar. Você ainda precisará de um repositório, um projeto de compilação, um pipeline e uma implantação, mas, em vez de o AWS CodeStar criá-los para você, você os recriará usando o AWS CloudFormation.

## Note

Pode ser útil criar ou visualizar um projeto semelhante usando primeiro um dos inícios rápidos do AWS CodeStar e usar isso como um modelo para seu próprio projeto para certificar-se de incluir os recursos e as políticas necessários.

Um projeto do AWS CodeStar é uma combinação de código-fonte e os recursos criados para implantar o código. A coleção de recursos que ajudam a compilar, lançar e implantar o código são chamados de recursos da cadeia de ferramentas. Na criação do projeto, um modelo do AWS CloudFormation provisiona os recursos da cadeia de ferramentas em um pipeline de integração/implantação contínuas (CI/CD).

Quando você usa o console para criar um projeto, o modelo da cadeia de ferramentas é criado para você. Ao usar a AWS CLI para criar um projeto, você cria o modelo da cadeia de ferramentas que cria os recursos da cadeia de ferramentas.

Uma cadeia de ferramentas completa requer os seguintes recursos recomendados:

1. Um repositório do CodeCommit ou do GitHub que contenha seu código-fonte.
2. Um pipeline do CodePipeline configurado para ouvir alterações no repositório.
  - a. Quando você usa o AWS CodeBuild para executar testes de unidade ou de integração, recomendamos adicionar um estágio de compilação ao pipeline para criar artefatos de compilação.
  - b. Recomendamos adicionar um estágio de implantação ao pipeline que usa CodeDeploy ou AWS CloudFormation para implantar o artefato de compilação e o código-fonte para a infraestrutura do runtime.

**Note**

Como o CodePipeline requer pelo menos dois estágios em um pipeline, e o primeiro estágio deve ser o estágio de origem, adicione um estágio de compilação ou implantação como o segundo estágio.

## Tópicos

- [Criar um repositório do GitHub](#)

## Criar um repositório do GitHub

Crie um repositório do GitHub definindo-o no modelo da cadeia de ferramentas. Certifique-se de que já tenha criado um local para um arquivo ZIP contendo o código-fonte, para que o código possa ser carregado no repositório. Além disso, é necessário já ter criado um token de acesso pessoal no GitHub para que a AWS possa se conectar ao GitHub em seu nome. Além do token de acesso pessoal para GitHub, também é necessário ter a permissão `s3:GetObject` para o objeto Code que você passa.

Para especificar um repositório do GitHub, adicione um código como este ao seu modelo de cadeia de ferramentas no AWS CloudFormation.

```
GitHubRepo:
  Condition: CreateGitHubRepo
  Description: GitHub repository for application source code
  Properties:
    Code:
      S3:
        Bucket: MyCodeS3Bucket
        Key: MyCodeS3BucketKey
    EnableIssues: true
    IsPrivate: false
    RepositoryAccessToken: MyGitHubPersonalAccessToken
    RepositoryDescription: MyAppCodeRepository
    RepositoryName: MyAppSource
    RepositoryOwner: MyGitHubUserName
  Type: AWS::CodeStar::GitHubRepository
```

Esse código especifica as seguintes informações:

- O local do código que você deseja incluir, que deve ser um bucket do Amazon S3.
- Se você deseja habilitar problemas no repositório do GitHub.
- Se o repositório do GitHub é privado.
- O token de acesso pessoal do GitHub que você criou.
- Uma descrição, um nome e um proprietário para o repositório que você está criando.

Para obter detalhes completos sobre quais informações especificar, consulte [AWS::CodeStar::GitHubRepository](#) no Guia do usuário do AWS CloudFormation.

## Como trabalhar com tags de projetos no AWS CodeStar

Você pode associar tags com projetos no AWS CodeStar. As tags podem ajudá-lo a gerenciar seus projetos. Por exemplo, convém adicionar uma tag com uma chave de Release e um valor de Beta a qualquer projeto no qual sua organização esteja trabalhando para uma versão beta.

### Adicionar uma tag a um projeto

1. Com o projeto aberto no console do AWS CodeStar, no painel de navegação lateral, escolha Configurações.
2. Em Tags, escolha Editar.
3. Em Chave, insira um nome para a tag. Em Value (Valor), digite o valor da tag.
4. Opcional: Escolha Adicionar tag para adicionar mais tags.
5. Quando terminar de adicionar tags, escolha Salvar.

### Remover uma tag de um projeto

1. Com o projeto aberto no console do AWS CodeStar, no painel de navegação lateral, escolha Configurações.
2. Em Tags, escolha Editar.
3. Em Tags, encontre a tag que deseja remover e selecione Remover tag.
4. Escolha Save (Salvar).

## Obter uma lista de tags de um projeto

Use a AWS CLI para executar o comando `list-tags-for-project` do AWS CodeStar, especificando o nome do projeto:

```
aws codestar list-tags-for-project --id my-first-projec
```

Se bem-sucedido, uma lista de tags aparecerá na saída, semelhante à seguinte:

```
{
  "tags": {
    "Release": "Beta"
  }
}
```

## Excluir um projeto do AWS CodeStar

Se você não precisar mais de um projeto, você pode excluir o projeto e seus recursos, para que ele não seja mais cobrado na AWS. Quando você exclui um projeto, todos os membros da equipe são removidos desse projeto. As funções do projeto são removidas dos usuários do IAM, mas seus perfis de usuário no AWS CodeStar não são alterados. Também é possível usar o console do AWS CodeStar ou a AWS CLI para excluir um projeto. Excluir um projeto requer a função de serviço do AWS CodeStar, `aws-codestar-service-role`, que deve ser não modificada e deve poder ser assumida pelo AWS CodeStar.

### Important

A exclusão de um projeto no AWS CodeStar não pode ser desfeita. Por padrão, todos os recursos da AWS para o projeto são excluídos de sua conta da AWS, incluindo:

- O repositório do CodeCommit do projeto juntamente com qualquer coisa armazenada no repositório.
- As funções do projeto do AWS CodeStar e as políticas do IAM associadas configuradas para o projeto e seus recursos.
- Quaisquer instâncias do Amazon EC2 criadas para o projeto.
- O aplicativo de implantação e os recursos associados, como:
  - Um aplicativo do CodeDeploy e os grupos de implantação associados.

- Uma função do AWS Lambda e as APIs do API Gateway associadas.
- Um aplicativo do AWS Elastic Beanstalk e o ambiente associado.
- O pipeline de implantação contínua do projeto no CodePipeline.
- As pilhas do AWS CloudFormation associadas ao projeto.
- Todos os ambientes de desenvolvimento do AWS Cloud9 criados com o console do AWS CodeStar. Todas as alterações de código não confirmadas nos ambientes são perdidas.

Para excluir todos os recursos do projeto junto com o projeto, marque a caixa de seleção Excluir recursos. Se você desmarcar essa opção, o projeto será excluído no AWS CodeStar e as funções do projeto que permitem o acesso a esses recursos serão excluídas no IAM, mas todos os outros recursos serão mantidos. Você pode continuar a ser cobrado por esses recursos na AWS. Se decidir que não deseja mais um ou mais desses recursos, você deverá excluí-los manualmente. Para obter mais informações, consulte [Exclusão do projeto: um AWS CodeStar projeto foi excluído, mas os recursos ainda existem](#).

Se você decidir manter os recursos ao excluir um projeto, como uma prática recomendada, copie a lista de recursos na página de detalhes do projeto. Dessa forma, você tem um registro de todos os recursos que você manteve, embora o projeto não exista mais.

## Tópicos

- [Excluir um projeto no AWS CodeStar \(console\)](#)
- [Excluir um projeto no AWS CodeStar \(AWS CLI\)](#)

## Excluir um projeto no AWS CodeStar (console)

Também é possível usar o console do AWS CodeStar para excluir um projeto.

Para excluir um projeto no AWS CodeStar

1. Abra o console do AWS CodeStar em <https://console.aws.amazon.com/codestar/>.
2. Selecione Projetos no painel de navegação.
3. Selecione o projeto que deseja excluir e selecione Excluir.

Ou abra o projeto e escolha Configurações do painel de navegação na esquerda do console. Na página de detalhes do projeto, escolha Excluir projeto.

4. Na Página de confirmação de exclusão, insira excluir. Mantenha a opção Excluir recursos selecionada se quiser excluir recursos do projeto. Escolha Delete (Excluir).

A exclusão de um projeto pode demorar vários minutos. Depois de ser excluído, o projeto não aparece mais na lista de projetos no console do AWS CodeStar.

 Important

Se o seu projeto usa recursos fora da AWS (por exemplo, um repositório do GitHub ou chamados no Atlassian JIRA), esses recursos não serão excluídos, mesmo se você marcar a caixa de seleção.

O projeto não poderá ser excluído se alguma política gerenciada do AWS CodeStar tiver sido anexada manualmente a funções que não sejam usuários do IAM. Se tiver anexado as políticas gerenciadas do projeto a uma função do usuário federado, você deverá desanexar a política para excluir o projeto. Para obter mais informações, consulte [???](#).

## Excluir um projeto no AWS CodeStar (AWS CLI)

Você pode usar a AWS CLI para excluir um projeto.

Para excluir um projeto no AWS CodeStar

1. Em um terminal (Linux, macOS, ou Unix) ou prompt de comando (Windows), execute o comando `delete-project` incluindo o nome do projeto. Por exemplo, para excluir um projeto com o ID *my-2nd-project*:

```
aws codestar delete-project --id my-2nd-project
```

Esse comando retorna uma saída semelhante à seguinte:

```
{
  "projectArn": "arn:aws:codestar:us-east-2:111111111111:project/my-2nd-project"
}
```

Os projetos não são excluídos imediatamente.

2. Execute o comando `describe-project`, incluindo o nome do projeto. Por exemplo, para verificar o status de um projeto com o ID *my-2nd-project*:



```
aws codestar describe-project --id my-2nd-project
```

se o projeto ainda não foi excluído, esse comando retorna uma saída semelhante à seguinte:

```
{
  "name": "my project",
  "id": "my-2nd-project",
  "arn": "arn:aws:codestar:us-west-2:123456789012:project/my-2nd-project",
  "description": "My second CodeStar project.",
  "createdTimeStamp": 1572547510.128,
  "status": {
    "state": "CreateComplete"
  }
}
```

Se o projeto for excluído, esse comando retornará uma saída semelhante à seguinte:

```
An error occurred (ProjectNotFoundException) when calling the DescribeProject
operation: The project ID was not found: my-2nd-project. Make sure that the
project ID is correct and then try again.
```

3. Execute o comando `list-projects` e verifique se o projeto excluído não aparece mais na lista de projetos associados à sua conta da AWS.

```
aws codestar list-projects
```

# Como trabalhar com equipes AWS CodeStar

Depois de criar um projeto de desenvolvimento, conceda acesso a outros usuários, para que vocês possam trabalhar em conjunto. No AWS CodeStar, cada projeto tem uma equipe de projeto. Um usuário pode pertencer a vários projetos do AWS CodeStar e ter diferentes funções do AWS CodeStar (e, portanto, diferentes permissões) em cada um. No console do AWS CodeStar, os usuários veem todos os projetos associados à conta da AWS, mas só podem exibir e trabalhar em projetos nos quais são membros da equipe.

Os membros da equipe podem escolher um nome amigável para si mesmos. Eles também podem adicionar um endereço de e-mail para que outros membros da equipe possam entrar em contato com eles. Os membros da equipe que não são proprietários não podem alterar sua função do AWS CodeStar para o projeto.

Cada projeto no AWS CodeStar tem três funções:

Funções e permissões em um projeto do AWS CodeStar

Função	Visualizar o painel e o status do projeto	Adicionar/remover/acessar recursos do projeto	Adicionar/remover membros da equipe	Excluir projeto
Proprietário	x	x	x	x
Colaborador	x	x		
Visualizador	x			

- **Proprietário:** pode adicionar e remover outros membros da equipe, contribuir com o código em um repositório do projeto se o código estiver armazenado no CodeCommit, conceder ou negar a outros membros da equipe acesso remoto a qualquer instância do Amazon EC2 executada no Linux associada ao projeto, configurar o painel do projeto e excluir o projeto.
- **Colaborador:** pode adicionar e remover recursos do painel, como um bloco do JIRA, contribuir com código no repositório do projeto se o código estiver armazenado no CodeCommit e interagir totalmente com o painel. Não pode adicionar ou remover membros da equipe, conceder ou negar acesso remoto a recursos ou excluir o projeto. Essa é a função que você deve escolher para a maioria dos membros da equipe.

- Visualizador: pode visualizar o painel do projeto, o código se estiver armazenado no CodeCommit e, nos blocos do painel, o estado do projeto e seus recursos.

**⚠ Important**

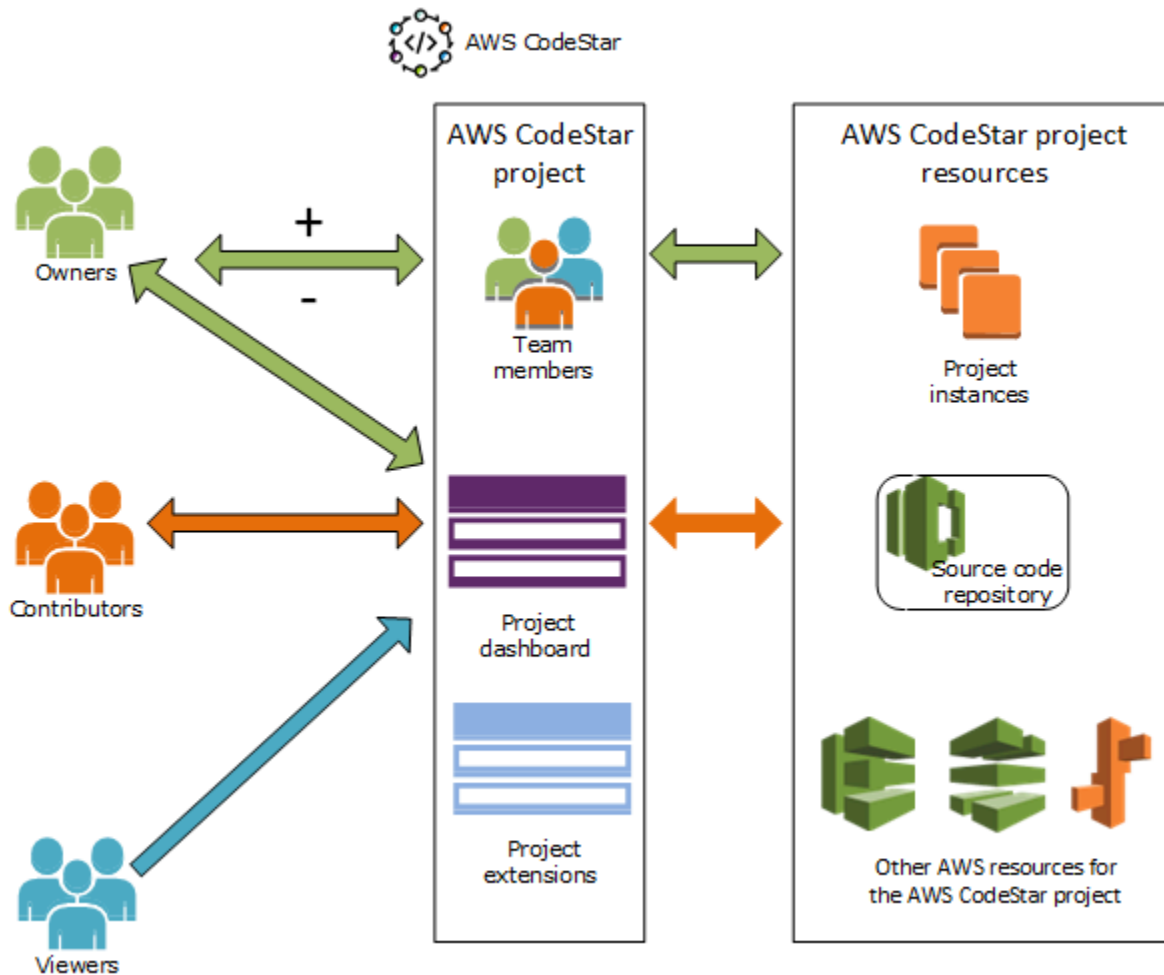
Se o seu projeto usa recursos fora da AWS (por exemplo, um repositório do GitHub ou chamados no Atlassian JIRA), o acesso a esses recursos é controlado pelo provedor de recursos, não pelo AWS CodeStar. Para obter mais informações, consulte a documentação do provedor do recurso.

Qualquer pessoa que tenha acesso a um projeto do AWS CodeStar pode usar o console do AWS CodeStar para acessar recursos que estão fora da AWS mas estão relacionados a esse projeto.

O AWS CodeStar não permite automaticamente que os membros da equipe do projeto participem de ambientes de desenvolvimento relacionados ao AWS Cloud9 para um projeto. Para permitir que um membro da equipe participe de um ambiente compartilhado, consulte [Compartilhar um ambiente do AWS Cloud9 com um membro da equipe do projeto](#).

Uma política do IAM é associada a cada função do projeto. Esta política é personalizada para o projeto para refletir seus recursos. Para obter mais informações sobre essas políticas, consulte [Exemplos de políticas baseadas em identidade para o AWS CodeStar](#).

O diagrama a seguir mostra a relação entre cada função e um projeto do AWS CodeStar.



## Tópicos

- [Adicionar membros da equipe a um projeto do AWS CodeStar](#)
- [Gerenciar permissões para membros da equipe do AWS CodeStar](#)
- [Remover membros da equipe de um projeto do AWS CodeStar](#)

## Adicionar membros da equipe a um projeto do AWS CodeStar

Se você tiver a função de proprietário em um projeto do AWS CodeStar ou tiver a política de `AWSCodeStarFullAccess` aplicada a seu usuário do IAM, poderá adicionar outros usuários do IAM à equipe do projeto. Esse é um processo simples que aplica uma função do AWS CodeStar (proprietário, colaborador e visualizador) ao usuário. Essas funções são por projeto e personalizadas. Por exemplo, um membro da equipe colaborador no projeto A pode ter permissões para recursos diferentes das de um membro da equipe colaborador no projeto B. Um membro da equipe pode ter

apenas uma função em um projeto. Depois que você adiciona um membro da equipe, ele ou ela pode interagir imediatamente com o projeto no nível definido pela função.

Os benefícios das funções do AWS CodeStar e da associação à equipe incluem:

- Você não precisa configurar manualmente as permissões no IAM para os membros da equipe.
- Pode alterar facilmente o nível de acesso de um membro da equipe a um projeto.
- Os usuários podem acessar projetos no console do AWS CodeStar apenas se forem membros da equipe.
- O acesso do usuário a um projeto é definido pela função.

Para obter mais informações sobre equipes e funções do AWS CodeStar, consulte [Como trabalhar com equipes AWS CodeStar](#) e [Como trabalhar com seu perfil de usuário do AWS CodeStar](#).

Para adicionar uma função a um usuário para um projeto, você deve definir a função proprietário do AWS CodeStar para o projeto ou aplicar a política `AWSCodeStarFullAccess`.

#### Important

A inclusão de um membro da equipe não afeta o acesso desse membro da equipe aos recursos que estão fora da AWS (por exemplo, um repositório do GitHub ou chamados no Atlassian JIRA). Essas permissões de acesso são controladas pelo provedor de recursos, não pelo AWS CodeStar. Para obter mais informações, consulte a documentação do provedor do recurso.

Qualquer pessoa que tenha acesso a um projeto do AWS CodeStar pode usar o console do AWS CodeStar para acessar recursos que estão fora do AWS mas estão relacionados a esse projeto.

Adicionar um membro da equipe a um projeto não permite automaticamente que esse membro participe de qualquer ambiente de desenvolvimento do AWS Cloud9 do projeto. Para permitir que um membro da equipe participe de um ambiente compartilhado, consulte [Compartilhar um ambiente do AWS Cloud9 com um membro da equipe do projeto](#).

A concessão de acesso de usuário federado a um projeto envolve anexar manualmente a política de proprietário, colaborador ou visualizador do AWS CodeStar gerenciada à função assumida pelo usuário federado. Para obter mais informações, consulte [Acesso de usuário federado ao AWS CodeStar](#).

## Tópicos

- [Adicionar um membro da equipe \(Console\)](#)
- [Adicionar e exibir membros da equipe \(AWS CLI\)](#)

## Adicionar um membro da equipe (Console)

Você pode usar o console do AWS CodeStar para adicionar um membro da equipe ao projeto. Se um usuário do IAM já existir para a pessoa que você deseja adicionar, você poderá adicionar o usuário do IAM. Do contrário, você poderá criar um usuário do IAM para essa pessoa ao adicioná-los ao projeto.


Para adicionar um membro da equipe a um projeto do AWS CodeStar (console)

1. Abra o console do AWS CodeStar em <https://console.aws.amazon.com/codestar/>.
2. Selecione Projetos no painel de navegação e escolha seu projeto.
3. No painel lateral de navegação do projeto, escolha Equipe.
4. Na página Membros da equipe, escolha Adicionar membro da equipe.
5. Em Escolher usuário, proceda de uma das seguintes maneiras:
  - Se um usuário do IAM já existir para a pessoa que você deseja adicionar, escolha o usuário IAM na lista.

### Note

Os usuários que já foram adicionados a outro projeto do AWS CodeStar aparecerão na lista de Usuários existentes do AWS CodeStar.


Em Função do projeto, escolha a função do AWS CodeStar (proprietário, colaborador ou visualizador) para esse usuário. Essa é uma função em nível de projeto do AWS CodeStar que só pode ser alterada por um proprietário do projeto. Quando aplicada a um usuário do IAM, a função fornece todas as permissões necessárias para acessar os recursos do projeto do AWS CodeStar. Ela aplica as políticas necessárias para criar e gerenciar as credenciais do Git para o código armazenado no CodeCommit no IAM ou para fazer upload de chaves SSH do Amazon EC2 para o usuário no IAM.

 Important

Você não pode fornecer ou alterar as informações de nome de exibição ou de e-mail de um usuário do IAM, a menos que você tenha feito login no console com esse usuário. Para obter mais informações, consulte [Gerenciar informações de exibição de seu perfil de usuário do AWS CodeStar](#).

Escolha Adicionar membro da equipe.

- Se um usuário do IAM não existir para a pessoa que você deseja adicionar ao projeto, escolha Criar novo usuário do IAM;. Você será redirecionado para o console do IAM, onde poderá criar um novo usuário do IAM. Consulte [Como criar usuários do IAM](#) no guia do usuário do IAM para obter mais informações. Depois de criar seu usuário do IAM, retorne ao console do AWS CodeStar, atualize a lista de usuários e escolha o usuário do IAM que você criou na lista suspensa. Digite o nome de exibição do AWS CodeStar, o endereço de e-mail e a função do projeto que você deseja aplicar a esse novo usuário e escolha Adicionar membro da equipe.

 Note

Para facilitar o gerenciamento, pelo menos um usuário deve receber a função Proprietário do projeto.

6. Envie ao novo membro da equipe as seguintes informações:

- Informações sobre a conexão de seu projeto do AWS CodeStar.
- Se o código-fonte estiver armazenado no CodeCommit, [instruções para configurar o acesso com as credenciais do Git](#) ao repositório do CodeCommit de seus computadores locais.
- Informações sobre como o usuário pode gerenciar seu nome de exibição, endereço de e-mail e chave SSH pública do Amazon EC2, conforme descrito em [Como trabalhar com seu perfil de usuário do AWS CodeStar](#).
- Uma senha avulsa e informações de conexão, se o usuário for novo no AWS e você tiver criado um usuário do IAM para essa pessoa. A senha expira na primeira vez em que o usuário entra. O usuário deve escolher uma nova senha.

## Adicionar e exibir membros da equipe (AWS CLI)

Você pode usar a AWS CLI para adicionar membros à equipe do projeto. Você também pode visualizar informações sobre todos os membros da equipe do projeto.

Para adicionar um membro da equipe

1. Abra uma janela de terminal ou de comando.
2. Execute o comando `associate-team-member` com os parâmetros `--project-id`, `-user-arn` e `--project-role`. Você também pode especificar se o usuário tem acesso remoto às instâncias do projeto incluindo os parâmetros `--remote-access-allowed` ou `--no-remote-access-allowed`. Por exemplo:

```
aws codestar associate-team-member --project-id my-first-projec --user-arn
arn:aws:iam:111111111111:user/Jane_Doe --project-role Contributor --remote-access-
allowed
```

Esse comando não retorna uma saída.

Para visualizar todos os membros da equipe (AWS CLI)

1. Abra uma janela de terminal ou de comando.
2. Execute o comando `list-team-members` com o parâmetro `--project-id`. Por exemplo:

```
aws codestar list-team-members --project-id my-first-projec
```

Esse comando retorna uma saída semelhante à seguinte:

```
{
  "teamMembers": [
    {"projectRole": "Owner", "remoteAccessAllowed": true, "userArn": "arn:aws:iam::111111111111:use
Mary_Major"},
    {"projectRole": "Contributor", "remoteAccessAllowed": true, "userArn": "arn:aws:iam::1111111111
Jane_Doe"},
    {"projectRole": "Contributor", "remoteAccessAllowed": true, "userArn": "arn:aws:iam::1111111111
John_Doe"},
  ]
}
```



```
{ "projectRole": "Viewer", "remoteAccessAllowed": false, "userArn": "arn:aws:iam::111111111111:u
John_Stiles" }
]
```

## Gerenciar permissões para membros da equipe do AWS CodeStar

Você altera permissões para os membros da equipe alterando suas funções do AWS CodeStar. Cada membro da equipe pode ser atribuído a apenas uma função em um projeto do AWS CodeStar, mas muitos usuários podem ser atribuídos à mesma função. Você não pode usar o console do AWS CodeStar ou a AWS CLI para gerenciar permissões.

### Important

Para alterar uma função para um membro da equipe, você deve definir a função de proprietário do AWS CodeStar a esse projeto ou aplicar a política `AWSCodeStarFullAccess`.

A alteração das permissões de um membro da equipe não afeta o acesso desse membro da equipe aos recursos que estão fora da AWS, (por exemplo, um repositório GitHub ou chamados no Atlassian JIRA). Essas permissões de acesso são controladas pelo provedor de recursos, não pelo AWS CodeStar. Para obter mais informações, consulte a documentação do provedor do recurso.

Qualquer pessoa que tenha acesso a um projeto do AWS CodeStar pode usar o console do AWS CodeStar para acessar recursos que estão fora da AWS mas estão relacionados a esse projeto.

Alterar a função de um membro da equipe de um projeto não permite nem evita automaticamente que o membro participe de ambientes de desenvolvimento do AWS Cloud9 do projeto. Para permitir ou evitar que um membro da equipe participe de um ambiente compartilhado, consulte [Compartilhar um ambiente do AWS Cloud9 com um membro da equipe do projeto](#).

Você também pode conceder permissões aos usuários para acessar remotamente qualquer instância do Amazon EC2 associada ao projeto. Depois de conceder essa permissão, o usuário deve fazer upload de uma chave SSH pública que será associada a seu perfil de usuário do AWS

CodeStar em todos os projetos da equipe. Para conectar-se com êxito às instâncias do Linux, o usuário deve ter o SSH configurado e a chave privada no computador local.

## Tópicos

- [Gerenciar permissões da equipe \(console\)](#)
- [Gerenciar permissões da equipe \(AWS CLI\)](#)

## Gerenciar permissões da equipe (console)

Você pode usar o console do AWS CodeStar para gerenciar as funções de membros da equipe. Você também pode gerenciar se os membros da equipe têm acesso remoto às instâncias do Amazon EC2 associadas ao projeto.

Para alterar a função de um membro da equipe

1. Abra o console do AWS CodeStar em <https://console.aws.amazon.com/codestar/>.
2. Selecione Projetos no painel de navegação e escolha seu projeto.
3. No painel lateral de navegação do projeto, escolha Equipe.
4. Na página Membros da equipe, selecione o membro da equipe e escolha Editar.
5. Em Função do projeto, escolha a função do AWS CodeStar (proprietário, colaborador ou visualizador) que você deseja conceder a esse usuário.

Para obter mais informações sobre as funções do AWS CodeStar e suas permissões, consulte [Como trabalhar com equipes AWS CodeStar](#).

Escolha Editar membro da equipe.

Para conceder permissões de acesso remoto a um membro da equipe para instâncias do Amazon EC2

1. Abra o console do AWS CodeStar em <https://console.aws.amazon.com/codestar/>.
2. Selecione Projetos no painel de navegação e escolha seu projeto.
3. No painel lateral de navegação do projeto, escolha Equipe.
4. Na página Membros da equipe, selecione o membro da equipe e escolha Editar.
5. Selecione Permitir acesso SSH a instâncias do projeto e, em seguida, escolha Editar membro da equipe.

6. (Opcional) Notifique os membros da equipe que eles devem fazer upload de uma chave SSH pública para seus usuários do AWS CodeStar, se ainda não o tiveram feito. Para obter mais informações, consulte [Adicione uma chave pública ao seu perfil de AWS CodeStar usuário](#).

## Gerenciar permissões da equipe (AWS CLI)

Você pode usar a AWS CLI para gerenciar a função do projeto atribuída a um membro da equipe. Você pode usar os mesmos comandos da AWS CLI para gerenciar se os membros da equipe têm acesso remoto às instâncias do Amazon EC2 associadas ao projeto.

Para gerenciar as permissões para um membro da equipe

1. Abra uma janela de terminal ou de comando.
2. Execute o comando `update-team-member` com os parâmetros `--project-id`, `--user-arn` e `--project-role`. Você também pode especificar se o usuário tem acesso remoto às instâncias do projeto incluindo os parâmetros `--remote-access-allowed` ou `--no-remote-access-allowed`. Por exemplo, para atualizar a função do projeto de um usuário do IAM chamado `John_Doe` e alterar suas permissões para visualizador sem acesso remoto às instâncias do Amazon EC2 do projeto:

```
aws codestar update-team-member --project-id my-first-projec --user-arn
arn:aws:iam:111111111111:user/John_Doe --project-role Viewer --no-remote-access-
allowed
```

Esse comando retorna uma saída semelhante à seguinte:

```
{
  "projectRole": "Viewer",
  "remoteAccessAllowed": false,
  "userArn": "arn:aws:iam::111111111111:user/John_Doe"
}
```

## Remover membros da equipe de um projeto do AWS CodeStar

Depois de remover um usuário de um projeto do AWS CodeStar, o usuário ainda aparecerá no histórico de confirmação do repositório do projeto, mas não terá mais acesso ao repositório do CodeCommit ou a qualquer outro recurso do projeto, como o pipeline do projeto. (A exceção a

essa regra é um usuário do IAM que tem outras políticas aplicadas que concedem acesso a esses recursos.) O usuário não pode acessar o painel do projeto, e o projeto deixa de ser exibido na lista de projetos que o usuário vê no painel do AWS CodeStar. Você pode usar o console do AWS CodeStar ou a AWS CLI para remover membros da equipe de seu projeto.

### Important

Embora a remoção de um membro da equipe de um projeto negue o acesso remoto às instâncias do Amazon EC2 do projeto, ele não fecha nenhuma sessão SSH ativa do usuário. A remoção de um membro da equipe não afeta o acesso desse membro da equipe aos recursos que estão fora da AWS (por exemplo, um repositório GitHub ou chamados no Atlassian JIRA). Essas permissões de acesso são controladas pelo provedor de recursos, não pelo AWS CodeStar. Para obter mais informações, consulte a documentação do provedor do recurso.

A remoção de um membro da equipe de um projeto não exclui automaticamente esse membro da equipe do ambiente de desenvolvimento do AWS Cloud9 nem impede que o membro participe de ambientes de desenvolvimento relacionados ao AWS Cloud9 para os quais tenham sido convidados. Para excluir um ambiente de desenvolvimento, consulte [Excluir um ambiente do AWS Cloud9 de um projeto](#). Para evitar que um membro da equipe participe de um ambiente compartilhado, consulte [Compartilhar um ambiente do AWS Cloud9 com um membro da equipe do projeto](#).

Para remover um membro da equipe de um projeto, você deve ter a função de proprietário do AWS CodeStar para esse projeto ou ter a política `AWSCodeStarFullAccess` aplicada à sua conta.

### Tópicos

- [Remover membros da equipe \(console\)](#)
- [Remover membros da equipe \(AWS CLI\)](#)

## Remover membros da equipe (console)

Você pode usar o console do AWS CodeStar para remover membros da equipe de seu projeto.

### Remover um membro da equipe de um projeto

1. Abra o console do AWS CodeStar em <https://console.aws.amazon.com/codestar/>.

2. Selecione Projetos no painel de navegação e escolha seu projeto.
3. No painel lateral de navegação do projeto, escolha Equipe.
4. Na página Membros da equipe, selecione o membro da equipe e escolha Remover.

## Remover membros da equipe (AWS CLI)

Você pode usar a AWS CLI para remover membros da equipe de seu projeto.

Para remover um membro da equipe

1. Abra uma janela de terminal ou de comando.
2. Execute o comando `disassociate-team-member` com os parâmetros `--project-id` e `-user-arn`. Por exemplo:

```
aws codestar disassociate-team-member --project-id my-first-projec --user-arn
arn:aws:iam:111111111111:user/John_Doe
```

Esse comando retorna uma saída semelhante à seguinte:

```
{
  "projectId": "my-first-projec",
  "userArn": "arn:aws:iam::111111111111:user/John_Doe"
}
```

# Como trabalhar com seu perfil de usuário do AWS CodeStar

Seu perfil de usuário do AWS CodeStar está associado a seu usuário do IAM. Esse perfil contém um nome de exibição e um endereço de e-mail que é usado em todos os projetos do AWS CodeStar que pertencem a você. Você pode fazer upload de uma chave pública SSH a ser associada a seu perfil. Essa chave pública é parte do par de chaves pública/privada do SSH que você usa ao conectar-se a instâncias do Amazon EC2 associadas aos projetos do AWS CodeStar que pertencem a você.

## Note

As informações nestes tópicos abrangem apenas o perfil de usuário do AWS CodeStar. Se o seu projeto usa recursos fora da AWS (por exemplo, um repositório GitHub ou chamados no Atlassian JIRA), esses provedores de recursos podem usar os próprios perfis de usuário, que podem ter configurações diferentes. Para obter mais informações, consulte a documentação do provedor do recurso.

## Tópicos

- [Gerenciar informações de exibição de seu perfil de usuário do AWS CodeStar](#)
- [Adicione uma chave pública ao seu perfil de AWS CodeStar usuário](#)

## Gerenciar informações de exibição de seu perfil de usuário do AWS CodeStar

Você pode usar o console do AWS CodeStar ou a AWS CLI para alterar o nome de exibição e o endereço de e-mail no perfil do usuário. Um perfil de usuário não é específico do projeto. Ele é associado ao usuário do IAM e é aplicado em todos os projetos do AWS CodeStar pertencentes a uma região do AWS. Se pertencer a projetos em mais de uma região da AWS, você terá perfis de usuário separados.

Você pode gerenciar seu próprio perfil de usuário somente no console do AWS CodeStar. Se tiver a política `AWSCodeStarFullAccess`, você poderá usar a AWS CLI para exibir e gerenciar outros perfis.

 Note


As informações neste tópico abrangem apenas o perfil de usuário do AWS CodeStar. Se o seu projeto usa recursos fora da AWS (por exemplo, um repositório GitHub ou chamados no Atlassian JIRA), esses provedores de recursos podem usar os próprios perfis de usuário, que podem ter configurações diferentes. Para obter mais informações, consulte a documentação do provedor do recurso.

## Tópicos

- [Gerenciar seu perfil de usuário \(console\)](#)
- [Gerenciar perfis de usuário \(AWS CLI\)](#)

## Gerenciar seu perfil de usuário (console)

Você pode gerenciar seu perfil de usuário no console do AWS CodeStar navegando para qualquer projeto no qual você seja membro da equipe e alterar suas informações de perfil. Como os perfis de usuário são específicos do usuário e não específicos ao projeto, as alterações em seu perfil de usuário aparecerão em cada projeto no qual uma região da AWS você é membro da equipe.

 Important

Para usar o console a fim de alterar as informações de exibição de um usuário, você deve fazer login como esse usuário do IAM. Nenhum outro usuário, mesmo aqueles que têm a função de proprietário no AWS CodeStar para um projeto ou têm a política `AWSCodeStarFullAccess` aplicada, poderá alterar suas informações de exibição.

Para alterar as informações de exibição em todos os projetos em uma região da AWS

1. Abra o console do AWS CodeStar em <https://console.aws.amazon.com/codestar/>.
2. Escolha Projetos no painel de navegação e escolha um projeto em que você seja membro da equipe.
3. No painel lateral de navegação do projeto, escolha Equipe.
4. Na página Membros da equipe, escolha o usuário do IAM e, em seguida, escolha Editar.

5. Edite o nome de exibição, o endereço de e-mail, ou os dois, e, em seguida, escolha Editar membro da equipe.

 Note

Um nome de exibição e um endereço de e-mail são necessários. Para obter mais informações, consulte [Limites no AWS CodeStar](#).

## Gerenciar perfis de usuário (AWS CLI)

Você pode usar a AWS CLI para criar e gerenciar seu perfil de usuário no AWS CodeStar. Você também pode usar a AWS CLI para visualizar as informações de seu perfil de usuário e para visualizar todos os perfis de usuário configurados para sua conta da AWS em uma região da AWS.

Verifique se seu perfil do AWS está configurado para a região em que você deseja criar, gerenciar ou visualizar perfis de usuário.

Para criar um novo usuário

1. Abra uma janela de terminal ou de comando.
2. Execute o comando `create-user-profile` com os parâmetros `user-arn`, `display-name` e `email-address`. Por exemplo:

```
aws codestar create-user-profile --user-arn arn:aws:iam:111111111111:user/John_Stiles --display-name "John Stiles" --email-address "john_stiles@example.com"
```

Esse comando retorna uma saída semelhante à seguinte:

```
{
  "createdTimestamp":1.491439687681E9,
  "displayName":"John Stiles",
  "emailAddress":"john.stiles@example.com",
  "lastModifiedTimestamp":1.491439687681E9,
  "userArn":"arn:aws:iam::111111111111:user/Jane_Doe"
}
```



## Para visualizar informações de exibição

1. Abra uma janela de terminal ou de comando.
2. Execute o comando `describe-user-profile` com o parâmetro `user-arn`. Por exemplo:

```
aws codestar describe-user-profile --user-arn arn:aws:iam:111111111111:user/
Mary_Major
```

Esse comando retorna uma saída semelhante à seguinte:

```
{
  "createdTimestamp":1.490634364532E9,
  "displayName":"Mary Major",
  "emailAddress":"mary.major@example.com",
  "lastModifiedTimestamp":1.491001935261E9,
  "sshPublicKey":"EXAMPLE=",
  "userArn":"arn:aws:iam::111111111111:user/Mary_Major"
}
```

## Para alterar suas informações de exibição

1. Abra uma janela de terminal ou de comando.
2. Execute o comando `update-user-profile` com o parâmetro `user-arn` e os parâmetros de perfil que você deseja alterar, como os parâmetros `display-name` ou `email-address`. Por exemplo, se um usuário com o nome de exibição Jane Doe quiser alterar seu nome de exibição para Jane Mary Doe:

```
aws codestar update-user-profile --user-arn arn:aws:iam:111111111111:user/Jane_Doe
--display-name "Jane Mary Doe"
```

Esse comando retorna uma saída semelhante à seguinte:

```
{
  "createdTimestamp":1.491439687681E9,
  "displayName":"Jane Mary Doe",
  "emailAddress":"jane.doe@example.com",
  "lastModifiedTimestamp":1.491442730598E9,
  "sshPublicKey":"EXAMPLE1",
  "userArn":"arn:aws:iam::111111111111:user/Jane_Doe"
```

```
}
```

Para listar todos os perfis de usuários em uma região da AWS em sua conta da AWS

1. Abra uma janela de terminal ou de comando.
2. Execute o comando `aws codestar list-user-profiles`. Por exemplo:

```
aws codestar list-user-profiles
```

Esse comando retorna uma saída semelhante à seguinte:

```
{
  "userProfiles": [
    {
      "displayName": "Jane Doe",
      "emailAddress": "jane.doe@example.com",
      "sshPublicKey": "EXAMPLE1",
      "userArn": "arn:aws:iam::111111111111:user/Jane_Doe"
    },
    {
      "displayName": "John Doe",
      "emailAddress": "john.doe@example.com",
      "sshPublicKey": "EXAMPLE2",
      "userArn": "arn:aws:iam::111111111111:user/John_Doe"
    },
    {
      "displayName": "Mary Major",
      "emailAddress": "mary.major@example.com",
      "sshPublicKey": "EXAMPLE=",
      "userArn": "arn:aws:iam::111111111111:user/Mary_Major"
    },
    {
      "displayName": "John Stiles",
      "emailAddress": "john.stiles@example.com",
      "sshPublicKey": "",
      "userArn": "arn:aws:iam::111111111111:user/John_Stiles"
    }
  ]
}
```

# Adicione uma chave pública ao seu perfil de AWS CodeStar usuário

Você pode fazer upload de uma chave SSH pública como parte do par de chaves pública/privada que você cria e gerencia. Você pode usar esse par de chaves SSH pública/privada para acessar instâncias do Amazon EC2 que executam o Linux. Se um proprietário de projeto concedeu a você permissão de acesso remoto, você poderá acessar somente as instâncias associadas ao projeto. Você pode usar o AWS CodeStar console ou AWS CLI gerenciar sua chave pública.

## Important

O proprietário AWS CodeStar do projeto pode conceder aos proprietários, colaboradores e visualizadores do projeto acesso SSH às instâncias do Amazon EC2 para o projeto, mas somente o indivíduo (proprietário, colaborador ou visualizador) pode definir a chave SSH. Para isso, o usuário deve fazer login como o proprietário, colaborador ou o visualizador individual.

AWS CodeStar não gerencia chaves SSH para AWS Cloud9 ambientes.

## Tópicos

- [Gerenciar a chave pública \(console\)](#)
- [Gerenciar sua chave pública \(AWS CLI\)](#)
- [Conectar-se à instância do Amazon EC2 com sua chave privada](#)

## Gerenciar a chave pública (console)

Embora você não possa gerar um par de chaves pública-privada no console, você pode criar um localmente e depois adicioná-lo ou gerenciá-lo como parte do seu perfil de usuário por meio do AWS CodeStar console.

Para gerenciar a chave SSH pública

1. Em um terminal ou janela do emulador Bash, execute o comando `ssh-keygen` para gerar um par de chaves SSH pública/privada no computador local. Você pode gerar uma chave em qualquer formato permitido pelo Amazon EC2. Para obter informações sobre os formatos aceitáveis, consulte [Importar sua própria chave pública no Amazon EC2](#). De forma ideal, gere

uma chave que seja SSH-2 RSA, no formato OpenSSH e contenha 2048 bits. A chave pública é armazenada em um arquivo com a extensão .pub.

2. Abra o AWS CodeStar console em <https://console.aws.amazon.com/codestar/>.

Escolha um projeto em que você é membro da equipe.

3. No painel de navegação, escolha Equipe.
4. Na página Membros da equipe, localize o nome de seu usuário do IAM e então escolha Editar.
5. Na página Editar membro da equipe, em Acesso remoto, habilite Permitir acesso SSH às instâncias do projeto.
6. Na caixa Chave pública SSH, cole a chave pública e escolha Editar membro da equipe.

#### Note

Você pode alterar sua chave pública excluindo a chave antiga nesse campo e colando a nova chave. Você pode excluir uma chave pública excluindo o conteúdo desse campo e escolhendo Editar membro da equipe.

Quando altera ou exclui uma chave pública você está alterando seu perfil de usuário. Essa não é uma alteração por projeto. Como sua chave está associada a seu perfil, ela é alterada (ou é excluída) em todos os projetos em que você tiver recebido acesso remoto.

A exclusão de sua chave pública remove seu acesso às instâncias do Amazon EC2 que executam o Linux em todos os projetos em que você recebeu acesso remoto. No entanto, ela não fecha nenhuma sessão SSH aberta que usa essa chave. Feche todas as sessões abertas.

## Gerenciar sua chave pública (AWS CLI)

Você pode usar o AWS CLI para gerenciar sua chave pública SSH como parte do seu perfil de usuário.

Para gerenciar a chave pública

1. Em um terminal ou janela do emulador Bash, execute o comando `ssh-keygen` para gerar um par de chaves SSH pública/privada no computador local. Você pode gerar uma chave em qualquer formato permitido pelo Amazon EC2. Para obter informações sobre os formatos aceitáveis, consulte [Importar sua própria chave pública no Amazon EC2](#). De forma ideal, gere

uma chave que seja SSH-2 RSA, no formato OpenSSH e contenha 2048 bits. A chave pública é armazenada em um arquivo com a extensão `.pub`.

2. Para adicionar ou alterar sua chave pública SSH em seu perfil de AWS CodeStar usuário, execute o `update-user-profile` comando com o `--ssh-public-key` parâmetro. Por exemplo: .

```
aws codestar update-user-profile --user-arn arn:aws:iam:111111111111:user/Jane_Doe
--ssh-key-id EXAMPLE1
```

Esse comando retorna uma saída semelhante à seguinte:

```
{
  "createdTimestamp":1.491439687681E9,
  "displayName":"Jane Doe",
  "emailAddress":"jane.doe@example.com",
  "lastModifiedTimestamp":1.491442730598E9,
  "sshPublicKey":"EXAMPLE1",
  "userArn":"arn:aws:iam::111111111111:user/Jane_Doe"
}
```

## Conectar-se à instância do Amazon EC2 com sua chave privada

Verifique se você já criou um par de chaves do Amazon EC2. Adicione sua chave pública ao seu perfil de usuário em AWS CodeStar. Para criar um par de chaves, consulte [Etapa 4: criar um par de chaves do Amazon EC2 para projetos do AWS CodeStar](#) Para adicionar a chave pública a seu perfil de usuário, consulte as instruções anteriormente neste tópico.

Para se conectar a uma instância de Linux do Amazon EC2 usando a chave privada

1. Com seu projeto aberto no AWS CodeStar console, no painel de navegação, escolha Projeto.
2. Em Recursos do projeto, escolha o link ARN na linha em que o Tipo é Amazon EC2 e o Nome começa com Instância.
3. No Amazon EC2 console, selecione Conectar.
4. Siga as instruções na caixa de diálogo Connect To Your Instance (Conectar à sua instância).

Para o nome do usuário, use `ec2-user`. Se usar o nome de usuário incorreto, você não poderá conectar-se à instância.

---

Para obter mais informações, consulte os seguintes recursos no Guia do usuário do Amazon EC2.

- [Conexão à sua instância do Linux utilizando SSH](#)
- [Conexão com a instância do Linux no Windows utilizando PuTTY](#)
- [Conectando-se à sua instância Linux usando MindTerm](#)

# Segurança em AWS CodeStar

A segurança para com a nuvem na AWS é a nossa maior prioridade. Como cliente da AWS, você se contará com um datacenter e uma arquitetura de rede criados para atender aos requisitos das organizações com as maiores exigências de segurança.

A segurança é uma responsabilidade compartilhada entre a AWS e você. O [modelo de responsabilidade compartilhada](#) descreve isso como segurança da nuvem e segurança na nuvem:

- **Segurança da nuvem:** a AWS é responsável pela proteção da infraestrutura que executa produtos da AWS na Nuvem AWS. A AWS também fornece serviços que podem ser usados com segurança. Auditores externos testam e verificam regularmente a eficácia da nossa segurança como parte dos [Programas de conformidade da AWS](#). Para saber mais sobre os programas de conformidade que se aplicam ao AWS CodeStar, consulte [Serviços da AWS no escopo por programa de conformidade](#).
- **Segurança na nuvem:** sua responsabilidade é determinada pelo serviço da AWS que você usa. Você também é responsável por outros fatores, incluindo a confidencialidade de seus dados, os requisitos da sua empresa e as leis e regulamentos aplicáveis.

Esta documentação ajuda a entender como aplicar o modelo de responsabilidade compartilhada ao usar o AWS CodeStar. Os tópicos a seguir mostram como configurar o AWS CodeStar para atender aos seus objetivos de segurança e conformidade. Saiba também como usar outros produtos da AWS que ajudam a monitorar e proteger os recursos do AWS CodeStar.

Ao criar políticas personalizadas e usar limites de permissão no AWS CodeStar, garanta o acesso com privilégio mínimo concedendo somente as permissões necessárias para realizar uma tarefa e definindo o escopo das permissões para os recursos específicos. Para impedir que membros de outros projetos acessem recursos em seu projeto, conceda aos membros da organização permissões separadas para cada projeto do AWS CodeStar. Como prática recomendada, crie uma conta de projeto para cada membro e, em seguida, atribua acesso baseado em funções a essa conta.

Por exemplo, você pode usar um serviço como o AWS Control Tower com o AWS Organizations para provisionar contas para cada função de desenvolvedor em um grupo de DevOps. Em seguida, você pode atribuir permissões a essas contas. As permissões gerais se aplicam à conta, mas o usuário tem acesso limitado aos recursos fora do projeto.

Para obter mais informações sobre como gerenciar o acesso com privilégio mínimo aos recursos do AWS usando uma estratégia de várias contas, consulte a [estratégia de várias contas da AWS para sua zona de pouso](#) no Guia do usuário do AWS Control Tower.

## Tópicos

- [Proteção de dados no AWS CodeStar](#)
- [Identity and Access Management para o AWS CodeStar](#)
- [Log de chamadas de API do AWS CodeStar com o AWS CloudTrail](#)
- [Validação de Compatibilidade para AWS CodeStar](#)
- [Resiliência no AWS CodeStar](#)
- [Segurança da infraestrutura no AWS CodeStar](#)

## Proteção de dados no AWS CodeStar

O [modelo de responsabilidade compartilhada](#) do AWS aplica-se à proteção de dados no AWS CodeStar. Conforme descrito nesse modelo, a AWS é responsável por proteger a infraestrutura global que executa toda a Nuvem AWS. Você é responsável por manter o controle sobre seu conteúdo hospedado nessa infraestrutura. Você também é responsável pelas tarefas de configuração e gerenciamento de segurança dos Serviços da AWS que usa. Para obter mais informações sobre a privacidade de dados, consulte as [Perguntas frequentes sobre privacidade de dados](#). Para mais informações sobre a proteção de dados na Europa, consulte o artigo [AWS Shared Responsibility Model and GDPR](#) no Blog de segurança da AWS.

Para fins de proteção de dados, recomendamos que você proteja as Conta da AWS credenciais da e configure as contas de usuário individuais com o AWS IAM Identity Center ou o AWS Identity and Access Management (IAM). Dessa maneira, cada usuário receberá apenas as permissões necessárias para cumprir suas obrigações de trabalho. Recomendamos também que você proteja seus dados das seguintes formas:

- Use uma autenticação multifator (MFA [multi-factor authentication]) com cada conta.
- Use SSL/TLS para se comunicar com os atributos da AWS. Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Configure o registro em log das atividades da API e do usuário com o .AWS CloudTrail
- Use AWS as soluções de criptografia da , juntamente com todos os controles de segurança padrão dos Serviços da AWS.



- Use serviços gerenciados de segurança avançada, como o Amazon Macie, que ajuda a localizar e proteger dados sigilosos armazenados no Amazon S3.
- Se você precisar de módulos criptográficos validados pelo FIPS 140-2 ao acessar a AWS por meio de uma interface de linha de comandos ou uma API, use um endpoint do FIPS. Para ter mais informações sobre endpoints do FIPS, consulte [Federal Information Processing Standard \(FIPS\) 140-2](#).

É altamente recomendável que nunca sejam colocadas informações de identificação confidenciais, como endereços de e-mail dos seus clientes, em marcações ou campos de formato livre, como um campo Name (Nome). Isso inclui trabalhar com o CodeStar ou outro Serviços da AWS usando o console, a API, a AWS CLI ou os SDKs da AWS. Quaisquer dados inseridos em tags ou campos de texto de formato livre usados para nomes podem ser usados para logs de faturamento ou de diagnóstico. Se você fornecer um URL para um servidor externo, recomendamos fortemente que não sejam incluídas informações de credenciais no URL para validar a solicitação a esse servidor.

## Criptografia de dados no AWS CodeStar

Por padrão, o AWS CodeStar criptografa as informações armazenadas sobre seu projeto. Tudo além do ID do projeto é criptografado em repouso, como o nome do projeto, a descrição e os e-mails do usuário. Evite colocar informações pessoais em seus IDs de projeto. O AWS CodeStar também criptografa informações em trânsito por padrão. Nenhuma ação por parte do cliente é necessária para a criptografia em repouso ou a criptografia em trânsito.

## Identity and Access Management para o AWS CodeStar

O AWS Identity and Access Management (IAM) é um AWS service (Serviço da AWS) que ajuda o administrador a controlar o acesso aos recursos da AWS de forma segura. Os administradores do IAM controlam quem pode ser autenticado (fazer login) e autorizado (ter permissões) para usar recursos do AWS CodeStar. O IAM é um AWS service (Serviço da AWS) que pode ser usado sem custo adicional.

### Tópicos

- [Público](#)
- [Autenticação com identidades](#)
- [Gerenciamento do acesso usando políticas](#)
- [Como o AWS CodeStar funciona com o IAM](#)

- [Políticas e permissões em nível de projeto do AWS CodeStar](#)
- [Exemplos de políticas baseadas em identidade para o AWS CodeStar](#)
- [Solução de problemas de identidade e acesso do AWS CodeStar Identity](#)

## Público

O uso do AWS Identity and Access Management (IAM) varia dependendo do trabalho que é realizado no AWS CodeStar.

**Usuário do serviço:** se você usa o serviço AWS CodeStar para fazer o trabalho, o administrador fornece as credenciais e as permissões necessárias. À medida que usar mais atributos do AWS CodeStar para fazer seu trabalho, você poderá precisar de permissões adicionais. Entender como o acesso é gerenciado pode ajudá-lo a solicitar as permissões corretas ao seu administrador. Se não for possível acessar um atributo no AWS CodeStar, consulte [Solução de problemas de identidade e acesso do AWS CodeStar Identity](#).

**Administrador do serviço:** se você for o responsável pelos recursos do AWS CodeStar na empresa, provavelmente terá acesso total ao AWS CodeStar. Cabe a você determinar quais funcionalidades e atributos do AWS CodeStar os usuários do serviço devem acessar. Assim, você deve enviar solicitações ao administrador do IAM para alterar as permissões dos usuários de seu serviço. Revise as informações nesta página para entender a Introdução ao IAM. Para saber mais sobre como a empresa pode usar o IAM com o AWS CodeStar, consulte [Como o AWS CodeStar funciona com o IAM](#).

**Administrador do IAM:** se você for um administrador do IAM, talvez queira saber detalhes sobre como é possível criar políticas para gerenciar o acesso ao AWS CodeStar. Para visualizar exemplos de políticas baseadas em identidade do AWS CodeStar que podem ser usadas no IAM, consulte [Exemplos de políticas baseadas em identidade para o AWS CodeStar](#).

## Autenticação com identidades

A autenticação é a forma como você faz login na AWS usando suas credenciais de identidade. É necessário ser autenticado (fazer login na AWS) como o usuário raiz da Usuário raiz da conta da AWS, como usuário do IAM ou assumindo um perfil do IAM.

Você pode fazer login na AWS como uma identidade federada usando credenciais fornecidas por uma fonte de identidades. Os usuários do AWS IAM Identity Center (IAM Identity Center), a

autenticação única da empresa e as suas credenciais do Google ou do Facebook são exemplos de identidades federadas. Quando você faz login como uma identidade federada, o administrador já configurou anteriormente a federação de identidades utilizando perfis do IAM. Quando você acessa a AWS usando a federação, está indiretamente assumindo um perfil.

É possível fazer login no ou no portal de acesso da AWS Management Console dependendo do tipo de usuário que você é. Para obter mais informações sobre como fazer login na AWS, consulte [How to sign in to your Conta da AWS](#) (Como fazer login na conta da) no Início de Sessão da AWS User Guide (Guia do usuário do ).

Se você acessar a AWS programaticamente, a AWS fornecerá um kit de desenvolvimento de software (SDK) e uma interface da linha de comando (CLI) para você assinar criptograficamente as solicitações usando as suas credenciais. Se você não utilizar as ferramentas da AWS, deverá assinar as solicitações por conta própria. Para obter mais informações sobre como usar o método recomendado para assinar solicitações por conta própria, consulte [Assinar AWS solicitações de API da](#) no Guia do usuário do IAM.

Independentemente do método de autenticação usado, também pode ser exigido que você forneça mais informações de segurança. Por exemplo, a AWS recomenda o uso da autenticação multifator (MFA) para aumentar a segurança de sua conta. Para saber mais, consulte [Autenticação multifator](#) no Guia do usuário do AWS IAM Identity Center e [Usar a autenticação multifator \(MFA\) na AWS](#) no Guia do usuário do IAM.

## Usuário raiz da Conta da AWS

Ao criar uma Conta da AWS, você começa com uma identidade de login que tem acesso completo a todos os atributos e Serviços da AWS na conta. Essa identidade, denominada usuário raiz da Conta da AWS, e é acessada por login com o endereço de e-mail e a senha que você usou para criar a conta. É altamente recomendável não utilizar o usuário raiz para tarefas diárias. Proteja as credenciais do usuário raiz e use-as para executar as tarefas que somente ele puder executar. Para obter a lista completa das tarefas que exigem login como usuário raiz, consulte [Tarefas que exigem credenciais de usuário raiz](#) no Guia do usuário do IAM.

## Grupos e usuários do IAM

Um [usuário do IAM](#) é uma identidade dentro da Conta da AWS que tem permissões específicas para uma única pessoa ou aplicação. Sempre que possível, recomendamos contar com credenciais temporárias em vez de criar usuários do IAM com credenciais de longo prazo, como senhas e chaves de acesso. No entanto, se você tiver casos de utilização específicos que exijam credenciais

de longo prazo com usuários do IAM, recomendamos alternar as chaves de acesso. Para obter mais informações, consulte [Altere as chaves de acesso regularmente para casos de uso que exijam credenciais](#) de longo prazo no Guia do usuário do IAM.

Um [grupo do IAM](#) é uma identidade que especifica uma coleção de usuários do IAM. Não é possível fazer login como um grupo. É possível utilizar grupos para especificar permissões para vários usuários de uma vez. Os grupos facilitam o gerenciamento de permissões para grandes conjuntos de usuários. Por exemplo, você pode ter um grupo chamado IAMAdmins e atribuir a esse grupo permissões para administrar atributos do IAM.

Usuários são diferentes de perfis. Um usuário é exclusivamente associado a uma pessoa ou a uma aplicação, mas um perfil pode ser assumido por qualquer pessoa que precisar dele. Os usuários têm credenciais permanentes de longo prazo, mas os perfis fornecem credenciais temporárias. Para saber mais, consulte [Quando criar um usuário do IAM \(em vez de uma função\)](#) no Guia do usuário do IAM.

## Perfis do IAM

Um [perfil do IAM](#) é uma identidade dentro da Conta da AWS que tem permissões específicas. Ele é semelhante a um usuário do IAM, mas não está associado a uma pessoa específica. É possível assumir temporariamente um perfil do IAM no AWS Management Console [alternando perfis](#). É possível assumir um perfil chamando uma operação de API da AWS CLI ou da AWS, ou usando um URL personalizado. Para obter mais informações sobre métodos para o uso de perfis, consulte [Uso de funções do IAM](#) no Guia do usuário do IAM.

Perfis do IAM com credenciais temporárias são úteis nas seguintes situações:

- **Acesso de usuário federado:** para atribuir permissões a identidades federadas, você pode criar um perfil e definir permissões para ele. Quando uma identidade federada é autenticada, essa identidade é associada ao perfil e recebe as permissões definidas pelo mesmo. Para obter mais informações sobre perfis para federação, consulte [Criar uma função para um provedor de identidade de terceiros](#) no Guia do usuário do IAM. Se você usar o IAM Identity Center, deverá configurar um conjunto de permissões. Para controlar o que suas identidades podem acessar após a autenticação, o Centro de identidade do IAM correlaciona o conjunto de permissões a um perfil no IAM. Para obter informações sobre conjuntos de permissões, consulte [Conjuntos de permissões](#) no Guia do usuário do AWS IAM Identity Center.
- **Permissões temporárias para usuários do IAM:** um usuário ou um perfil do IAM pode assumir um perfil do IAM para obter temporariamente permissões diferentes para uma tarefa específica.

- **Acesso entre contas:** é possível usar um perfil do IAM para permitir que alguém (uma entidade principal confiável) em outra conta acesse recursos em sua conta. Os perfis são a principal forma de conceder acesso entre contas. No entanto, alguns Serviços da AWS permitem que você anexe uma política diretamente a um atributo (em vez de usar um perfil como proxy). Para saber a diferença entre perfis e políticas baseadas em atributo para acesso entre contas, consulte [Como os perfis do IAM diferem das políticas baseadas em atributo](#) no Guia do usuário do IAM.
- **Acesso entre serviços:** alguns Serviços da AWS usam atributos em outros Serviços da AWS. Por exemplo, quando você faz uma chamada em um serviço, é comum que esse serviço execute aplicações no Amazon EC2 ou armazene objetos no Amazon S3. Um serviço pode fazer isso usando as permissões do principal de chamada, usando um perfil de serviço ou um perfil vinculado ao serviço.
- **Encaminhamento de sessões de acesso (FAS):** qualquer pessoa que utilizar uma função ou usuário do IAM para realizar ações na AWS é considerada uma entidade principal. Ao usar alguns serviços, você pode executar uma ação que inicia outra ação em um serviço diferente. O recurso FAS utiliza as permissões da entidade principal que chama um AWS service (Serviço da AWS), combinadas às permissões do AWS service (Serviço da AWS) solicitante, para realizar solicitações para serviços downstream. As solicitações de FAS só são feitas quando um serviço recebe uma solicitação que exige interações com outros Serviços da AWS ou com recursos para serem concluídas. Nesse caso, você precisa ter permissões para executar ambas as ações. Para obter detalhes da política ao fazer solicitações de FAS, consulte [Encaminhar sessões de acesso](#).
- **Perfil de serviço:** um perfil de serviço é um [perfil do IAM](#) que um serviço assume para realizar ações em seu nome. Um administrador do IAM pode criar, modificar e excluir um perfil de serviço do IAM. Para obter mais informações, consulte [Criar um perfil para delegar permissões a um AWS service \(Serviço da AWS\)](#) no Guia do usuário do IAM.
- **Função vinculada ao serviço:** uma função vinculada a serviço é um tipo de função de serviço vinculada a um AWS service (Serviço da AWS). O serviço pode assumir o perfil para executar uma ação em seu nome. Os perfis vinculados ao serviço aparecem em sua Conta da AWS e são de propriedade do serviço. Um administrador do IAM pode exibir, mas não pode editar as permissões para perfis vinculados ao serviço.
- **Aplicações em execução no Amazon EC2:** é possível usar um perfil do IAM para gerenciar credenciais temporárias para aplicações em execução em uma instância do EC2 e fazer solicitações da AWS CLI ou da AWS API. É preferível fazer isso e armazenar chaves de acesso na instância do EC2. Para atribuir um perfil da AWS a uma instância do EC2 e disponibilizá-la para todas as suas aplicações, crie um perfil de instância que esteja anexado a ela. Um perfil de

instância contém a perfil e permite que os programas em execução na instância do EC2 obtenham credenciais temporárias. Para mais informações, consulte [Usar um perfil do IAM para conceder permissões a aplicações em execução nas instâncias do Amazon EC2](#) no Guia do usuário do IAM.

Para saber se deseja usar os perfis do IAM, consulte [Quando criar um perfil do IAM \(em vez de um usuário\)](#) no Guia do usuário do IAM.

## Gerenciamento do acesso usando políticas

Você controla o acesso na AWS criando políticas e anexando-as a identidades ou atributos da AWS. Uma política é um objeto na AWS que, quando associado a uma identidade ou atributo, define suas permissões. A AWS avalia essas políticas quando uma entidade principal (usuário, usuário raiz ou sessão de perfil) faz uma solicitação. As permissões nas políticas determinam se a solicitação será permitida ou negada. A maioria das políticas são armazenadas na AWS como documentos JSON. Para obter mais informações sobre a estrutura e o conteúdo de documentos de políticas JSON, consulte [Visão geral das políticas JSON](#) no Guia do usuário do IAM.

Os administradores podem usar AWS as políticas JSON da para especificar quem tem acesso a quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

Por padrão, usuários e funções não têm permissões. Para conceder aos usuários permissão para executar ações nos recursos de que eles precisam, um administrador do IAM pode criar políticas do IAM. O administrador pode então adicionar as políticas do IAM a perfis, e os usuários podem assumir os perfis.

As políticas do IAM definem permissões para uma ação independente do método usado para executar a operação. Por exemplo, suponha que você tenha uma política que permite a ação `iam:GetRole`. Um usuário com essa política pode obter informações de perfil do AWS Management Console, da AWS CLI ou da AWS API.

### Políticas baseadas em identidade

As políticas baseadas em identidade são documentos de políticas de permissões JSON que você pode anexar a uma identidade, como usuário, grupo de usuários ou perfil do IAM. Essas políticas controlam quais ações os usuários e perfis podem realizar, em quais recursos e em que condições. Para saber como criar uma política baseada em identidade, consulte [Criar políticas do IAM](#) no Guia do usuário do IAM.

As políticas baseadas em identidade podem ser categorizadas ainda mais como políticas embutidas ou políticas gerenciadas. As políticas embutidas são anexadas diretamente a um único usuário, grupo ou função. As políticas gerenciadas são políticas independentes que podem ser anexadas a vários usuários, grupos e perfis na Conta da AWS. As políticas gerenciadas incluem políticas gerenciadas pela AWS e políticas gerenciadas pelo cliente. Para saber como escolher entre uma política gerenciada ou uma política em linha, consulte [Escolher entre políticas gerenciadas e políticas em linha](#) no Guia do usuário do IAM.

## Políticas com base em recurso

Políticas baseadas em atributos são documentos de políticas JSON que você anexa a um atributo. São exemplos de políticas baseadas em recursos as políticas de confiança de perfil do IAM e políticas de bucket do Amazon S3. Em serviços compatíveis com políticas baseadas em recurso, os administradores de serviço podem usá-las para controlar o acesso a um recurso específico. Para o recurso ao qual a política está anexada, a política define quais ações uma entidade principal especificado pode executar nesse recurso e em que condições. Você precisa [especificar uma entidade principal](#) em uma política baseada em recurso. As entidades principais podem incluir contas, usuários, perfis, usuários federados ou Serviços da AWS.

Políticas baseadas em atributos são políticas em linha que estão localizadas nesse serviço. Não é possível usar as políticas gerenciadas da AWS do IAM em uma política baseada em atributos.

## Listas de controle de acesso (ACLs)

As listas de controle de acesso (ACLs) controlam quais entidades principais (membros, usuários ou funções da conta) têm permissões para acessar um recurso. As ACLs são semelhantes às políticas baseadas em recurso, embora não usem o formato de documento de política JSON.

Amazon S3, AWS WAF e Amazon VPC são exemplos de serviços compatíveis com ACLs. Para saber mais sobre ACLs, consulte [Visão geral da lista de controle de acesso \(ACL\)](#) no Guia do desenvolvedor do Amazon Simple Storage Service.

## Outros tipos de política

A AWS aceita tipos de política menos comuns. Esses tipos de política podem definir o máximo de permissões concedidas a você pelos tipos de política mais comuns.

- **Limites de permissões:** um limite de permissões é um atributo avançado no qual você define o máximo de permissões que uma política baseada em identidade pode conceder a uma entidade do IAM (usuário ou perfil do IAM). É possível definir um limite de permissões para uma entidade.

As permissões resultantes são a interseção das políticas baseadas em identidade de uma entidade com seus limites de permissões. As políticas baseadas em recurso que especificam o usuário ou perfil no campo `Principal` não são limitadas pelo limite de permissões. Uma negação explícita em qualquer uma dessas políticas substitui a permissão. Para obter mais informações sobre limites de permissões, consulte [Limites de permissões para identidades do IAM](#) no Guia do usuário do IAM.

- Políticas de controle de serviço (SCPs): SCPs são políticas JSON que especificam as permissões máximas para uma organização ou unidade organizacional (UO) no AWS Organizations. O AWS Organizations é um serviço para agrupar e gerenciar centralmente várias Contas da AWS pertencentes à sua empresa. Se você habilitar todos os atributos em uma organização, poderá aplicar políticas de controle de serviço (SCPs) a qualquer uma ou a todas as contas. O SCP limita as permissões para entidades em contas-membro, incluindo cada `.Usuário raiz` da conta da AWS. Para obter mais informações sobre o Organizações e SCPs, consulte [How SCPs work \(Como os SCPs funcionam\)](#) no AWS Organizations Guia do usuário do .
- Políticas de sessão: são políticas avançadas que você transmite como um parâmetro quando cria de forma programática uma sessão temporária para um perfil ou um usuário federado. As permissões da sessão resultante são a interseção das políticas baseadas em identidade do usuário ou do perfil e das políticas de sessão. As permissões também podem ser provenientes de uma política baseada em atributo. Uma negação explícita em qualquer uma dessas políticas substitui a permissão. Para obter mais informações, consulte [Políticas de sessão](#) no Guia do usuário do IAM.

## Vários tipos de política

Quando vários tipos de política são aplicáveis a uma solicitação, é mais complicado compreender as permissões resultantes. Para saber como a AWS determina se deve permitir uma solicitação quando há vários tipos de política envolvidos, consulte [Lógica da avaliação](#) de políticas no Guia do usuário do IAM.

## Como o AWS CodeStar funciona com o IAM

Antes de usar o IAM para gerenciar o acesso ao AWS CodeStar, você precisa saber quais atributos do IAM estão disponíveis para serem usados com o AWS CodeStar. Para ter uma visão geral de como o AWS CodeStar e outros serviços do AWS funcionam com o IAM, consulte [Serviços do AWS compatíveis com o IAM](#) no Guia do usuário do IAM.

### Tópicos



- [Exemplos de políticas baseadas em identidade para o AWS CodeStar](#)
- [Políticas baseadas em recursos do AWS CodeStar](#)
- [Autorização com base nas tags AWS CodeStar](#)
- [Perfis do IAM do AWS CodeStar](#)
- [Acesso do usuário do IAM ao AWS CodeStar](#)
- [Acesso de usuário federado ao AWS CodeStar](#)
- [Usar credenciais temporárias com o AWS CodeStar](#)
- [Funções vinculadas ao serviço](#)
- [Perfis de serviço](#)

## Exemplos de políticas baseadas em identidade para o AWS CodeStar

Com políticas baseadas em identidade do IAM, é possível especificar ações e recursos permitidos ou negados e as condições em que as ações são permitidas ou negadas. O AWS CodeStar cria várias políticas baseadas em identidade em seu nome, que permitem que o AWS CodeStar crie e gerencie recursos dentro do escopo de um projeto do AWS CodeStar. O AWS CodeStar oferece suporte a ações, recursos e chaves de condição específicos. Para conhecer todos os elementos usados em uma política JSON, consulte [Referência de elementos de política JSON do IAM](#) no Guia do usuário do IAM.

### Ações

Os administradores podem usar as políticas JSON da AWS para especificar quem tem acesso a o quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

O elemento `Action` de uma política JSON descreve as ações que você pode usar para permitir ou negar acesso em uma política. As ações de política geralmente têm o mesmo nome que a operação de API da AWS associada. Existem algumas exceções, como ações somente de permissão, que não têm uma operação de API correspondente. Algumas operações também exigem várias ações em uma política. Essas ações adicionais são chamadas de ações dependentes.

Incluem ações em uma política para conceder permissões para executar a operação associada.

As ações de políticas no AWS CodeStar usam o seguinte prefixo antes da ação: `codestar:`. Por exemplo, para permitir que um usuário do IAM especificado edite os atributos de um projeto do AWS CodeStar, como a descrição do projeto, use a seguinte declaração de política:

```
{
  "Version": "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "codestar:UpdateProject"
      ],
      "Resource" : "arn:aws:codestar:us-east-2:project/my-first-projec"
    }
  ]
}
```

As instruções de política devem incluir um elemento `Action` ou `NotAction`. O AWS CodeStar define seu próprio conjunto de ações que descrevem as tarefas que você pode executar com esse serviço.

Para especificar várias ações em uma única instrução, separe-as com vírgulas, como segue:

```
"Action": [
  "codestar:action1",
  "codestar:action2"
```

Você também pode especificar várias ações usando caracteres curinga (\*). Por exemplo, para especificar todas as ações que começam com a palavra `List`, inclua a seguinte ação:

```
"Action": "codestar:List*"
```

Para ver uma lista de ações do AWS CodeStar, consulte [Ações definidas pelo AWS CodeStar](#) no Manual do usuário do IAM.

## Recursos

Os administradores podem usar as políticas de JSON da AWS para especificar quem tem acesso a quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

O elemento de política `Resource` JSON especifica o objeto ou os objetos aos quais a ação se aplica. As instruções devem incluir um elemento `Resource` ou um elemento `NotResource`. Como prática recomendada, especifique um atributo usando seu [Nome do atributo da Amazon \(ARN\)](#). Isso

pode ser feito para ações que oferecem suporte a um tipo de atributo específico, conhecido como permissões em nível de atributo.

Para ações que não oferecem suporte a permissões em nível de recurso, como operações de listagem, use um caractere curinga (\*) para indicar que a instrução se aplica a todos os recursos.

```
"Resource": "*"
```

O recurso do projeto do AWS CodeStar tem o seguinte ARN:

```
arn:aws:codestar:region:account:project/resource-specifier
```

Para obter mais informações sobre o formato de ARNs, consulte [Nomes de recursos da Amazon \(ARNs\)AWS e namespaces de serviços da](#)

Por exemplo, o seguinte especifica o projeto do AWS CodeStar chamado *my-first-projec* registrado na conta do AWS 111111111111 na região AWS us-east-2:

```
arn:aws:codestar:us-east-2:111111111111:project/my-first-projec
```

O seguinte especifica o projeto do AWS CodeStar que começa com o nome *my-proj* registrado na conta do AWS 111111111111 na região AWS us-east-2:

```
arn:aws:codestar:us-east-2:111111111111:project/my-proj*
```

Algumas ações do AWS CodeStar, como para a listagem de projetos, não podem ser executadas em um recurso. Nesses casos, você deve utilizar o caractere curinga (\*).

```
"LisProjects": "*"
```

Para ver uma lista de tipos de recurso do AWS CodeStar e seus ARNs, consulte [Recursos definidos pelo AWS CodeStar](#) no Guia do usuário do IAM. Para saber com quais ações você pode especificar o ARN de cada recurso, consulte [Ações definidas pelo AWS CodeStar](#).

## Chaves de condição

O AWS CodeStar não fornece nenhuma chave de condição específica ao serviço, mas oferece suporte ao uso de algumas chaves de condição globais. Para ver todas as chaves de condição

globais da AWS, consulte [Chaves de contexto de condição globais da AWS](#) no Guia do usuário do IAM.

## Exemplos

Para visualizar exemplos de políticas baseadas em identidade do AWS CodeStar, consulte [Exemplos de políticas baseadas em identidade para o AWS CodeStar](#).

## Políticas baseadas em recursos do AWS CodeStar

O AWS CodeStar não oferece suporte a políticas baseadas em recurso.

## Autorização com base nas tags AWS CodeStar

É possível anexar tags a projetos do AWS CodeStar ou passar tags em uma solicitação para o AWS CodeStar. Para controlar o acesso baseado em tags, forneça informações sobre as tags no [elemento de condição](#) de uma política usando as `codestar:ResourceTag/key-name`, `aws:RequestTag/key-name` ou `aws:TagKeys` chaves de condição. Para obter mais informações sobre marcar com tags recursos do AWS CodeStar, consulte [the section called “Trabalhar com tags de projetos”](#).

Para visualizar um exemplo de política baseada em identidade para limitar o acesso a um projeto do AWS CodeStar com base nas tags desse projeto, consulte [Visualizar projetos do AWS CodeStar com base em tags](#).

## Perfis do IAM do AWS CodeStar

Um [perfil do IAM](#) é uma entidade dentro da sua conta da AWS que tem permissões específicas.

Você pode usar o AWS CodeStar como um [usuário do IAM](#), um usuário federado, o usuário raiz ou uma função assumida. Todos os tipos de usuário com as permissões apropriadas podem gerenciar permissões de projeto para os recursos do AWS, mas o AWS CodeStar gerencia automaticamente permissões do projeto para usuários do IAM. As políticas e perfis do IAM concedem permissões e acesso a esse usuário com base na função do projeto. Você pode usar o console do IAM para criar outras políticas que atribuam o AWS CodeStar e outras permissões a um usuário do IAM.

Por exemplo, você pode permitir que um usuário visualize, mas não altere, um projeto do AWS CodeStar. Neste caso, você adiciona o usuário do IAM a um projeto do AWS CodeStar com a função

de visualizador. Cada projeto do AWS CodeStar tem um conjunto de políticas que ajudam a controlar o acesso ao projeto. Além disso, você pode controlar quais usuários têm acesso ao AWS CodeStar.

O acesso ao AWS CodeStar é tratado de maneira diferente para usuários do IAM e usuários federados. Somente os usuários do IAM podem ser adicionados às equipes. Para conceder aos usuários do IAM permissões para projetos, adicione o usuário à equipe do projeto e atribua uma função a ele. Para conceder aos usuários federados permissões para projetos, você anexa manualmente a política gerenciada da função de projeto do AWS CodeStar à função do usuário federado.

Esta tabela resume as ferramentas disponíveis para cada tipo de acesso.

Recurso de permissões	IAM user (Usuário do IAM)	Usuário federado	Usuário raiz
Gerenciamento de chaves SSH para acesso remoto para projetos do Amazon EC2 e do Elastic Beanstalk	✓		
AWS CodeCommitAcesso à SSH	✓		
Permissões de usuário do IAM gerenciadas pelo AWS CodeStar	✓		
Permissões de projeto gerenciadas manualmente		✓	✓
Os usuários podem ser adicionados ao projeto como membros da equipe	✓		

## Acesso do usuário do IAM ao AWS CodeStar

Quando você adiciona um usuário do IAM a um projeto e escolhe uma função para o usuário, o AWS CodeStar aplica a política adequada automaticamente ao usuário do IAM. Para os usuários do IAM, você não precisa anexar ou gerenciar diretamente políticas ou permissões no IAM. Para obter informações sobre como adicionar um usuário do IAM a um projeto do AWS CodeStar, consulte [Adicionar membros da equipe a um projeto do AWS CodeStar](#). Para obter informações sobre como remover um usuário do IAM de um projeto do AWS CodeStar, consulte [Remover membros da equipe de um projeto do AWS CodeStar](#).

## Anexar uma política em linha a um usuário do IAM

Quando você adiciona um usuário a um projeto, o AWS CodeStar anexa automaticamente a política gerenciada do projeto que corresponde à função do usuário. Você não deve anexar manualmente uma política gerenciada do AWS CodeStar de um projeto a um usuário do IAM. Com exceção da `AWSCodeStarFullAccess`, não recomendamos que você anexe políticas que alteram as permissões de um usuário do IAM em um projeto do AWS CodeStar. Se você decidir criar e anexar suas próprias políticas, consulte [Adicionar e remover permissões de identidade do IAM](#) no Guia do usuário do IAM.

## Acesso de usuário federado ao AWS CodeStar

Em vez de criar um usuário do IAM ou utilizar o usuário raiz, você pode usar identidades de usuário do AWS Directory Service, o diretório de usuário da sua empresa ou um provedor de identidades da web, ou ainda usuários do IAM assumindo funções. Estes são conhecidos como usuários federados.

Conceda aos usuários federados acesso ao seu projeto do AWS CodeStar anexando manualmente as políticas gerenciadas descritas em [Políticas e permissões no nível do projeto do AWS CodeStar](#) à função do IAM do usuário. Você anexará a política de proprietário, colaborador ou visualizador depois que o AWS CodeStar criar os recursos de projeto e as funções do IAM.

### Pré-requisitos:

- Você deve ter criado um provedor de identidade. Por exemplo, é possível configurar um provedor de identidade SAML e configurar a autenticação da AWS por meio do provedor. Para obter mais informações sobre como configurar um provedor de identidade, consulte [Criar provedores de identidade do IAM](#). Para obter mais informações sobre a federação SAML, consulte [Sobre a federação baseada em SAML 2.0](#).
- Você deve ter criado uma função para um usuário federado a ser assumida quando o acesso for solicitado por meio de um [provedor de identidade](#). Uma política de confiança STS deve ser anexada à função que permite que os usuários federados assumam a função. Para obter mais informações, consulte [Usuários federados e funções](#) no Guia do usuário do IAM.
- Você deve ter criado o projeto do AWS CodeStar e saber o ID do projeto.

Para obter mais informações sobre como criar uma função para provedores de identidade, consulte [Criar uma função para um provedor de identidade de terceiros \(federação\)](#).

## Anexar a política gerenciada `AWSCodeStarFullAccess` à função do usuário federado

Conceda a um usuário federado permissões para criar um projeto anexando a política gerenciada `AWSCodeStarFullAccess`. Para realizar essas etapas, você deve entrar no console como um usuário raiz, um usuário do administrador da conta ou um usuário do IAM ou um usuário federado com a política gerenciada `AdministratorAccess` associada ou equivalente.

### Note

Depois que você criar o projeto, as permissões de proprietário do projeto não serão aplicadas automaticamente. Usando uma função com permissões administrativas para a conta, anexe a política gerenciada do proprietário, conforme descrito em [Anexar a política gerenciada de visualizador/colaborador/proprietário do AWS CodeStar à função do usuário federado](#).

1. Abra o console do IAM. No painel de navegação, escolha `Policies`.
2. Digite `AWSCodeStarFullAccess` no campo de pesquisa. O nome da política é exibido, com um tipo de política de Gerenciado pelo AWS. Você pode expandir a política para ver as permissões na declaração de política.
3. Marque o círculo ao lado da política e, em `Policy actions (Ações da política)`, escolha `Attach (Anexar)`.
4. Na página `Summary (Resumo)`, selecione a guia `Attached entities (Entidades anexadas)`. Escolha `Anexar`.
5. Na página `Attach Policy (Anexar política)`, filtre a função do usuário federado no campo de pesquisa. Selecione a caixa ao lado do nome da função e escolha `Attach policy (Anexar política)`. A guia `Attached entities (Entidades anexadas)` mostrará o novo anexo.

## Anexar a política gerenciada de visualizador/colaborador/proprietário do AWS CodeStar à função do usuário federado

Conceda aos usuários federados acesso ao projeto anexando a política gerenciada de proprietário, colaborador ou visualizador do apropriada à função do usuário. A política gerenciada oferece o nível apropriado de permissões. Diferentemente dos usuários do IAM, você precisa anexar manualmente e desanexar políticas gerenciadas para usuários federados. Isso equivale a atribuir permissões de projeto a membros da equipe no AWS CodeStar. Para realizar essas etapas, você deve entrar no console como um usuário raiz, um usuário do administrador da conta ou um usuário do IAM ou um usuário federado com a política gerenciada `AdministratorAccess` associada ou equivalente.

## Pré-requisitos:

- Você deve ter criado uma função ou ter uma função existente assumida pelo usuário federado.
- Você deve saber qual nível de permissões deseja conceder. As políticas gerenciadas anexadas às funções de proprietário, colaborador e visualizador fornecem permissões baseadas na função para o projeto.
- O projeto do AWS CodeStar deve ter sido criado. A política gerenciada não está disponível no IAM até a criação do projeto.

1. Abra o console do IAM. No painel de navegação, escolha Policies.
2. Digite o ID do projeto no campo de pesquisa. O nome da política correspondente ao projeto é exibido, com um tipo de política de Customer managed (Gerenciado pelo cliente). Você pode expandir a política para ver as permissões na declaração de política.
3. Escolha uma dessas políticas gerenciadas. Marque o círculo ao lado da política e, em Policy actions (Ações da política), escolha Attach (Anexar).
4. Na página Summary (Resumo), selecione a guia Attached entities (Entidades anexadas). Escolha Anexar.
5. Na página Attach Policy (Anexar política), filtre a função do usuário federado no campo de pesquisa. Selecione a caixa ao lado do nome da função e escolha Attach policy (Anexar política). A guia Attached entities (Entidades anexadas) mostrará o novo anexo.

Desanexar uma política gerenciada do AWS CodeStar da função do usuário federado.

Para excluir o projeto do AWS CodeStar, você deve desanexar manualmente todas as políticas gerenciadas anexadas a uma função do usuário federado. Para realizar essas etapas, você deve entrar no console como um usuário raiz, um usuário do administrador da conta ou um usuário do IAM ou um usuário federado com a política gerenciada AdministratorAccess associada ou equivalente.

1. Abra o console do IAM. No painel de navegação, escolha Policies.
2. Digite o ID do projeto no campo de pesquisa.
3. Marque o círculo ao lado da política e, em Policy actions (Ações da política), escolha Attach (Anexar).
4. Na página Summary (Resumo), selecione a guia Attached entities (Entidades anexadas).
5. Filtre a função do usuário federado no campo de pesquisa. Escolha Desassociar.



## Anexar uma política gerenciada AWS Cloud9 à função do usuário federado

Se você estiver usando um ambiente de desenvolvimento do AWS Cloud9, conceda acesso a ele aos usuários federados anexando a política gerenciada `AWSCloud9User` à função do usuário. Diferentemente dos usuários do IAM, você precisa anexar manualmente e desanexar políticas gerenciadas para usuários federados. Para realizar essas etapas, você deve entrar no console como um usuário raiz, um usuário do administrador da conta ou um usuário do IAM ou um usuário federado com a política gerenciada `AdministratorAccess` associada ou equivalente.

### Pré-requisitos:

- Você deve ter criado uma função ou ter uma função existente assumida pelo usuário federado.
  - Você deve saber qual nível de permissões deseja conceder:
    - A política gerenciada `AWSCloud9User` permite que o usuário faça o seguinte:
      - Crie os ambientes de desenvolvimento AWS Cloud9 próprios.
      - Obtenha informações sobre os ambientes.
      - Altere as configurações para os ambientes.
    - A política gerenciada `AWSCloud9Administrator` permite que o usuário faça o seguinte para eles ou outros:
      - Crie ambientes.
      - Obtenha informações sobre ambientes.
      - Exclua ambientes.
      - Altere as configurações de ambientes.
1. Abra o console do IAM. No painel de navegação, escolha **Policies**.
  2. Digite o nome da política no campo de pesquisa. A política gerenciada é exibida, com um tipo de política de Gerenciado pelo AWS. Você pode expandir a política para ver as permissões na declaração de política.
  3. Escolha uma dessas políticas gerenciadas. Marque o círculo ao lado da política e, em **Policy actions** (Ações da política), escolha **Attach** (Anexar).
  4. Na página **Summary** (Resumo), selecione a guia **Attached entities** (Entidades anexadas). Escolha **Anexar**.
  5. Na página **Attach Policy** (Anexar política), filtre a função do usuário federado no campo de pesquisa. Escolha a caixa ao lado do nome da função e **Attach policy** (Anexar política). A guia **Attached entities** (Entidades anexadas) mostrará o novo anexo.

Desanexar uma política gerenciada do AWS Cloud9 da função do usuário federado.

Se estiver usando um ambiente de desenvolvimento do AWS Cloud9, você poderá remover o acesso de um usuário federado a ele desanexando a política que concede acesso. Para realizar essas etapas, você deve entrar no console como um usuário raiz, um usuário do administrador da conta ou um usuário do IAM ou um usuário federado com a política gerenciada `AdministratorAccess` associada ou equivalente.

1. Abra o console do IAM. No painel de navegação, escolha `Policies`.
2. Digite o nome do projeto no campo de pesquisa.
3. Marque o círculo ao lado da política e, em `Policy actions (Ações da política)`, escolha `Attach (Anexar)`.
4. Na página `Summary (Resumo)`, selecione a guia `Attached entities (Entidades anexadas)`.
5. Filtre a função do usuário federado no campo de pesquisa. Escolha `Desassociar`.

## Usar credenciais temporárias com o AWS CodeStar

É possível usar credenciais temporárias para fazer login com federação, assumir um perfil do IAM ou assumir um perfil entre contas. As credenciais de segurança temporárias são obtidas chamando operações da API do AWS STS, como [AssumeRole](#) ou [GetFederationToken](#).

O AWS CodeStar oferece suporte ao uso de credenciais temporárias, mas a funcionalidade de membro da equipe do AWS CodeStar não funciona para acesso federado. A funcionalidade de membro da equipe do AWS CodeStar oferece suporte somente à adição de um usuário do IAM como membro da equipe.

## Funções vinculadas ao serviço

[Perfis vinculados ao serviço](#) permitem que os serviços da AWS acessem recursos em outros serviços para concluir uma ação em seu nome. Os perfis vinculados a serviço aparecem na sua conta do IAM e são de propriedade do serviço. Um administrador do pode visualizar, mas não editar, as permissões de funções vinculadas ao serviço.

O AWS CodeStar não oferece suporte a funções vinculadas ao serviço.

## Perfis de serviço

Esse atributo permite que um serviço assumira um [perfil de serviço](#) em seu nome. O perfil permite que o serviço acesse recursos em outros serviços para concluir uma ação em seu nome. Os perfis

de serviço aparecem na sua conta do IAM e são de propriedade da conta. Isso significa que um administrador do pode alterar as permissões para essa função. Porém, fazer isso pode alterar a funcionalidade do serviço.

O AWS CodeStar suporta perfis de serviço. O AWS CodeStar utiliza um perfil de serviço, `aws-codestar-service-role`, ao criar e gerenciar os recursos de seu projeto. Para obter mais informações, consulte [Termos e conceitos de funções](#) no Guia do usuário do IAM.

#### Important

Você deve estar conectado como um usuário administrativo do ou conta raiz para criar essa função de serviço. Para obter mais informações, consulte [Somente primeiro acesso: as credenciais de usuário raiz](#) e [Criar o primeiro usuário e grupo de administrador do IAM](#) no Guia do usuário do IAM.

Essa função é criada na primeira vez que você cria um projeto no AWS CodeStar. A função de serviço atua em seu nome para:

- Criar os recursos escolhidos ao criar um projeto.
- Exibir informações sobre esses recursos no painel de projeto do AWS CodeStar.

Ela também atua em seu nome quando você gerencia os recursos de um projeto. Para obter um exemplo dessa declaração de política, consulte [Política AWSCodeStarServiceRole](#).

Além disso, o AWS CodeStar cria várias funções de serviço específicas do projeto, dependendo do tipo de projeto. O AWS CloudFormation e as funções da cadeia de ferramentas são criadas para cada tipo de projeto.

- As funções do AWS CloudFormation permitem que o AWS CodeStar acesse o AWS CloudFormation a fim de criar e modificar pilhas para seu projeto do AWS CodeStar.
- As funções da cadeia de ferramentas permitem que o AWS CodeStar acesse outros serviços da AWS a fim de criar e modificar recursos para seu projeto do AWS CodeStar.

## Políticas e permissões em nível de projeto do AWS CodeStar

Ao criar um projeto, o AWS CodeStar cria as funções e as políticas do IAM necessárias para gerenciar os recursos do projeto. As políticas se encaixam em três categorias:

- Políticas do IAM para membros da equipe do projeto.
- Políticas do IAM para funções de operador.
- Políticas do IAM para uma função de execução em tempo de execução.

## Políticas do IAM para membros de equipe

Quando você cria um projeto, o AWS CodeStar cria três políticas gerenciadas pelo cliente para o proprietário, o colaborador e o visualizador acessarem o projeto. Todos os projetos do AWS CodeStar contêm políticas do IAM para esses três níveis de acesso. Esses níveis de acesso são específicos ao projeto e definidos por uma política gerenciada pelo IAM com um nome padrão, onde *project-id* é a ID do projeto do AWS CodeStar (por exemplo, *my-first-projec*):

- CodeStar\_*project-id*\_Owner
- CodeStar\_*project-id*\_Contributor
- CodeStar\_*project-id*\_Viewer

### Important

Essas políticas estão sujeitas a alterações pelo AWS CodeStar. Elas não devem ser editadas manualmente. Se você deseja adicionar ou alterar permissões, anexe políticas adicionais ao usuário do IAM.

À medida que você adiciona membros da equipe (usuários do IAM) ao projeto e escolhe seus níveis de acesso, a política correspondente é anexada ao usuário do IAM, concedendo ao usuário o conjunto de permissões apropriado para atuar em recursos do projeto. Na maioria das circunstâncias, você não precisa anexar ou gerenciar diretamente políticas ou permissões no IAM. A anexação manual de uma política de nível de acesso do AWS CodeStar a um usuário do IAM não é recomendada. Se for absolutamente necessário, como um complemento para uma política de nível de acesso do AWS CodeStar, você poderá criar suas próprias políticas gerenciadas ou em linha para aplicar seu próprio nível de permissões a um usuário do IAM.

As políticas têm um escopo rígido para recursos do projeto e ações específicas. À medida que novos recursos são adicionados à pilha da infraestrutura, o AWS CodeStar tenta atualizar as políticas de membros da equipe para incluir permissões para acessar o novo recurso, se o novo recurso for de um dos tipos de recursos compatíveis.

**Note**

As políticas para níveis de acesso em um projeto do AWS CodeStar se aplicam apenas a esse projeto. Isso ajuda a garantir que os usuários possam ver e interagir somente com os projetos do AWS CodeStar para os quais têm permissão, no nível determinado por sua função. Somente os usuários que criam projetos do AWS CodeStar devem ter uma política aplicada que permita acesso a todos os recursos do AWS CodeStar, independentemente do projeto.

Todas as políticas de nível de acesso do AWS CodeStar variam, dependendo dos recursos da AWS associados ao projeto com o qual os níveis de acesso estão associados. Ao contrário de outros serviços da AWS, essas políticas são personalizadas quando o projeto é criado e atualizado à medida que os recursos do projeto mudam. Portanto, não existe uma política gerenciada de proprietário, colaborador ou visualizador canônica.

#### Política da função Proprietário do AWS CodeStar

A política gerenciada pelo cliente `CodeStar_project-id_Owner` permite que um usuário execute todas as ações no projeto do AWS CodeStar sem restrições. Essa é a única política que permite que um usuário adicione ou remova membros da equipe. O conteúdo da política varia, dependendo dos recursos associados ao projeto. Consulte [Política da função de proprietário do AWS CodeStar](#) para ver um exemplo.

Um usuário do IAM com essa política pode executar todas as ações do AWS CodeStar no projeto, mas, ao contrário de um usuário do IAM com a política de `AWSCodeStarFullAccess`, o usuário não pode criar novos projetos. A permissão `codestar:*` é limitada em escopo para um recurso específico (o projeto do AWS CodeStar associado a esse ID de projeto).

#### Política da função Colaborador do AWS CodeStar

A política gerenciada pelo cliente `CodeStar_project-id_Contributor` permite que um usuário colabore com o projeto e altere o painel do projeto, mas não permite que um usuário adicione ou remova membros da equipe. O conteúdo da política varia, dependendo dos recursos associados ao projeto. Consulte [Política de função de colaborador do AWS CodeStar](#) para ver um exemplo.

#### Política da função Visualizador do AWS CodeStar

A política gerenciada pelo cliente `CodeStar_project-id_Viewer` permite que um usuário visualize um projeto no AWS CodeStar, mas não altere seus recursos ou adicione ou remova

membros da equipe. O conteúdo da política varia, dependendo dos recursos associados ao projeto. Consulte [Política da função de visualizador do AWS CodeStar](#) para ver um exemplo.

## Políticas do IAM para funções de operador

Se você criar um projeto do AWS CodeStar depois de 6 de dezembro de 2018, horário do Pacífico, o AWS CodeStar criará duas funções de operador, `CodeStar-project-id-ToolChain` e `CodeStar-project-id-CloudFormation`. A função de operador é uma função do IAM específica ao projeto que o AWS CodeStar cria para passar para um serviço. Ela concede permissões para que o serviço possa criar recursos e executar ações no contexto do projeto do AWS CodeStar. A função de operador da cadeia de ferramentas tem uma relação de confiança estabelecida com os serviços da cadeia de ferramentas, como CodeBuild, CodeDeploy, e CodePipeline. Os membros da equipe do projeto (proprietários e colaboradores) recebem acesso para passar a função de operador para serviços downstream confiáveis. Para obter um exemplo da declaração da política em linha para essa função, consulte [Política de funções dos trabalhadores do Toolchain do AWS CodeStar \(após 6 de dezembro de 2018 no horário do Pacífico\)](#).

A função de operador do CloudFormation inclui permissões para recursos exclusivos compatíveis com o AWS CloudFormation, bem como permissões para criar usuários, funções e políticas do IAM na pilha de aplicativos. Ela também tem uma relação de confiança estabelecida com o AWS CloudFormation. Para reduzir os riscos de escalonamento de privilégios e ações destrutivas, a política de função do AWS CloudFormation inclui uma condição que exige um limite de permissões específicas ao projeto para cada entidade do IAM (usuário ou função) criada na pilha da infraestrutura. Para obter um exemplo da declaração da política em linha para essa função, consulte [Política da função de operador do AWS CloudFormation](#).

Para projetos do AWS CodeStar criados antes de 6 de dezembro de 2018, horário do Pacífico, o AWS CodeStar cria funções de operador individuais para recursos da cadeia de ferramentas, como CodePipeline, CodeBuild e CloudWatch, e também cria uma função de operador para o AWS CloudFormation que oferece suporte a um conjunto limitado de recursos. Cada uma dessas funções tem uma relação de confiança estabelecida com o serviço correspondente. Os membros da equipe do projeto (proprietários e colaboradores) e algumas das outras funções de operador recebem acesso para passar a função para serviços downstream confiáveis. As permissões para as funções de operador são definidas em uma política em linha que é delimitada para um conjunto básico de ações que a função pode executar em um conjunto de recursos do projeto. Essas permissões são estáticas. Elas incluem permissões para recursos incluídos no projeto durante a criação, mas não são atualizadas quando novos recursos são adicionados ao projeto. Para obter exemplos dessas declarações de políticas, consulte.

- [Política de funções do trabalhador do AWS CloudFormation \(antes de 6 de dezembro de 2018 no horário do Pacífico\)](#)
- [Política de funções do trabalhador do AWS CodePipeline \(antes de 6 de dezembro de 2018 no horário do Pacífico\)](#)
- [Política de funções do trabalhador do AWS CodeBuild \(antes de 6 de dezembro de 2018 no horário do Pacífico\)](#)
- [Política de funções de trabalhadores do Amazon CloudWatch Events \(antes de 6 de dezembro de 2018 no horário do Pacífico\)](#)

## Política do IAM para a função de execução

Para projetos criados após 6 de dezembro de 2018, horário do Pacífico, o AWS CodeStar cria uma função de execução genérica para o projeto de exemplo em sua pilha de aplicativos. A função é limitada aos recursos do projeto usando a política de limite de permissões. À medida que expande o projeto de exemplo, você pode criar funções do IAM adicionais, e a política de função do AWS CloudFormation exige que essas funções sejam delimitadas usando o limite de permissão para evitar escalonamento de privilégios. Para obter mais informações, consulte [Adicionar uma função do IAM a um projeto](#).

Para projetos do Lambda criados antes de 6 de dezembro de 2018, horário do Pacífico, o AWS CodeStar cria uma função do Lambda que tem uma política em linha anexada com permissões para atuar em recursos na pilha do AWS SAM do projeto. À medida que novos recursos são adicionados ao modelo do SAM, o AWS CodeStar tenta atualizar a política de função de execução do Lambda para incluir permissões para o novo recurso, se o novo recurso for de um dos tipos de recursos compatíveis.

## Limite de permissões do IAM

Depois de 6 de dezembro de 2018 no horário do Pacífico, quando você cria um projeto, o AWS CodeStar cria uma política gerenciada pelo cliente e atribui essa política [como limite de permissões do IAM aos perfis do IAM no](#) projeto. O AWS CodeStar exige que todas as entidades do IAM criadas na pilha de aplicativos tenham um limite de permissões. Um limite de permissões controla as permissões máximas que a função pode ter, mas não fornece a função com nenhuma permissão. As políticas de permissões definem as permissões para a função. Isso significa que, independentemente de quantas permissões extras forem adicionadas a uma função, qualquer pessoa que use a função não poderá executar mais do que as ações incluídas no limite de permissões. Para obter

informações sobre como as políticas de permissões e os limites de permissões são avaliados, consulte [Lógica de avaliação de políticas](#) no Guia do usuário do IAM.

O AWS CodeStar usa um limite de permissões específico ao projeto para impedir o escalonamento de privilégios a recursos fora do projeto. O limite de permissões do AWS CodeStar inclui ARNs para os recursos do projeto. Para obter um exemplo dessa declaração de política, consulte [Política de limite de permissões do AWS CodeStar](#).

A transformação do AWS CodeStar atualiza essa política quando você adiciona ou remove um recurso compatível do projeto por meio da pilha do aplicativo (`template.yml`).

## Adicionar um limite de permissões do IAM a projetos existentes

Se você tiver um projeto do AWS CodeStar criado antes de 6 de dezembro de 2018, horário do Pacífico, deverá adicionar manualmente um limite de permissão aos perfis do IAM no projeto. Como uma melhor prática, recomendamos usar um limite específico ao projeto que inclua apenas recursos no projeto para impedir o escalonamento de privilégios para recursos fora do projeto. Siga estas etapas para usar o limite de permissões gerenciadas pelo AWS CodeStar que é atualizado à medida que o projeto evolui.

1. Faça login no console do AWS CloudFormation e localize o modelo da pilha da cadeia de ferramentas em seu projeto. Este modelo é chamado `awscodestar-project-id`.
2. Escolha o modelo, selecione Actions (Ações) e View/Edit template in Designer (Visualizar/editar modelo no Designer).
3. Localize a seção Resources e inclua o trecho de código a seguir na parte superior da seção.

```
PermissionsBoundaryPolicy:
  Description: Creating an IAM managed policy for defining the permissions boundary
for an AWS CodeStar project
  Type: AWS::IAM::ManagedPolicy
  Properties:
    ManagedPolicyName: !Sub 'CodeStar_${ProjectId }_PermissionsBoundary'
    Description: 'IAM policy to define the permissions boundary for IAM entities
created in an AWS CodeStar project'
    PolicyDocument:
      Version: '2012-10-17'
      Statement:
        - Sid: '1'
          Effect: Allow
```



```

    Action: ['*']
    Resource:
      - !Sub 'arn:${AWS::Partition}:cloudformation:${AWS::Region}:
${AWS::AccountId}:stack/awscodestar-${ProjectId}-*'

```

Talvez você precise de permissões adicionais do IAM para atualizar a pilha no console do AWS CloudFormation.

4. (Opcional) Para criar perfis do IAM específicas ao aplicativo, execute esta etapa. No console do IAM, atualize a política em linha anexada à função do AWS CloudFormation de seu projeto para incluir o trecho de código a seguir. Talvez você precise recursos adicionais do IAM para atualizar a política.

```

{
  "Action": [
    "iam:PassRole"
  ],
  "Resource": "arn:aws:iam:::{AccountId}:role/CodeStar-{ProjectId}*",
  "Effect": "Allow"
},
{
  "Action": [
    "iam:CreateServiceLinkedRole",
    "iam:GetRole",
    "iam>DeleteRole",
    "iam>DeleteUser"
  ],
  "Resource": "*",
  "Effect": "Allow"
},
{
  "Action": [
    "iam:AttachRolePolicy",
    "iam:AttachUserPolicy",
    "iam:CreateRole",
    "iam:CreateUser",
    "iam>DeleteRolePolicy",
    "iam>DeleteUserPolicy",
    "iam:DetachUserPolicy",
    "iam:DetachRolePolicy",
    "iam:PutUserPermissionsBoundary",

```

```
        "iam:PutRolePermissionsBoundary"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "iam:PermissionsBoundary": "arn:aws:iam::{AccountId}:policy/
CodeStar_{ProjectId}_PermissionsBoundary"
        }
    },
    "Effect": "Allow"
}
```

5. Envie uma alteração por push por meio do pipeline do projeto para que o AWS CodeStar atualize o limite de permissões com as permissões apropriadas.

Para obter mais informações, consulte [Adicionar uma função do IAM a um projeto](#).

## Exemplos de políticas baseadas em identidade para o AWS CodeStar

Por padrão, os usuários e perfis do IAM não têm permissão para criar ou modificar recursos do AWS CodeStar. Eles também não podem executar tarefas usando o AWS Management Console, a AWS CLI ou uma API da AWS. Um administrador deve criar as políticas do IAM que concedam aos usuários e aos perfis permissões para executar operações de API específicas nos recursos especificados que precisam. O administrador deve anexar essas políticas aos usuários ou grupos do IAM que exigem essas permissões.

Para saber como criar uma política baseada em identidade do IAM usando esses exemplos de documentos de política JSON, consulte [Criar políticas na guia JSON](#) no Guia do usuário do IAM.

### Tópicos

- [Melhores práticas de políticas](#)
- [Política AWSCodeStarServiceRole](#)
- [Política AWSCodeStarFullAccess](#)
- [Política da função de proprietário do AWS CodeStar](#)
- [Política de função de colaborador do AWS CodeStar](#)
- [Política da função de visualizador do AWS CodeStar](#)

- [Política de funções dos trabalhadores do Toolchain do AWS CodeStar \(após 6 de dezembro de 2018 no horário do Pacífico\)](#)
- [Política da função de operador do AWS CloudFormation](#)
- [Política de funções do trabalhador do AWS CloudFormation \(antes de 6 de dezembro de 2018 no horário do Pacífico\)](#)
- [Política de funções do trabalhador do AWS CodePipeline \(antes de 6 de dezembro de 2018 no horário do Pacífico\)](#)
- [Política de funções do trabalhador do AWS CodeBuild \(antes de 6 de dezembro de 2018 no horário do Pacífico\)](#)
- [Política de funções de trabalhadores do Amazon CloudWatch Events \(antes de 6 de dezembro de 2018 no horário do Pacífico\)](#)
- [Política de limite de permissões do AWS CodeStar](#)
- [Listar recursos de um projeto](#)
- [Usando o console do AWS CodeStar](#)
- [Permitir que os usuários visualizem suas próprias permissões](#)
- [Atualizar um projeto do AWS CodeStar](#)
- [Adicionar um membro da equipe a um projeto](#)
- [Listar perfis de usuário associados a uma conta do AWS](#)
- [Visualizar projetos do AWS CodeStar com base em tags](#)
- [Atualizações do AWS CodeStar para políticas gerenciadas pela AWS](#)

## Melhores práticas de políticas

As políticas baseadas em identidade determinam se alguém pode criar, acessar ou excluir recursos do AWS CodeStar em sua conta. Essas ações podem incorrer em custos para a Conta da AWS. Ao criar ou editar políticas baseadas em identidade, siga estas diretrizes e recomendações:

- Comece com AWS as políticas gerenciadas pela e avance para as permissões de privilégio mínimo: para começar a conceder permissões a seus usuários e workloads, use as AWS políticas gerenciadas pela que concedem permissões para muitos casos de uso comuns. Elas estão disponíveis na sua Conta da AWS. Recomendamos que você reduza ainda mais as permissões definindo políticas gerenciadas pelo cliente da AWS específicas para seus casos de uso. Para obter mais informações, consulte [Políticas gerenciadas pela AWS](#) ou [Políticas gerenciadas pela AWS para perfis de trabalho](#) no Guia do usuário do IAM.

- Aplique permissões de privilégio mínimo: ao definir permissões com as políticas do IAM, conceda apenas as permissões necessárias para executar uma tarefa. Você faz isso definindo as ações que podem ser executadas em atributos específicos sob condições específicas, também conhecidas como permissões de privilégio mínimo. Para obter mais informações sobre como usar o IAM para aplicar permissões, consulte [Políticas e permissões no IAM](#) no Guia do usuário do IAM.
- Use condições nas políticas do IAM para restringir ainda mais o acesso: você pode adicionar uma condição às políticas para limitar o acesso a ações e atributos. Por exemplo, você pode escrever uma condição de política para especificar que todas as solicitações devem ser enviadas usando SSL. Você também pode usar condições para conceder acesso a ações de serviço, se elas forem usadas por meio de um AWS service (Serviço da AWS) específico, como o AWS CloudFormation. Para obter mais informações, consulte [Elementos de política JSON do IAM: Condition](#) no Guia do usuário do IAM.
- Use o IAM Access Analyzer para validar suas políticas do IAM a fim de garantir permissões seguras e funcionais: o IAM Access Analyzer valida as políticas novas e existentes para que elas sigam a linguagem de política do IAM (JSON) e as práticas recomendadas do IAM. O IAM Access Analyzer oferece mais de 100 verificações de política e recomendações acionáveis para ajudá-lo a criar políticas seguras e funcionais. Para obter mais informações, consulte [Validação de políticas do IAM Access Analyzer](#) no Guia do usuário do IAM.
- Exigir autenticação multifator (MFA): se houver um cenário que exija usuários do IAM ou um usuário raiz em sua Conta da AWS, ative a MFA para obter segurança adicional. Para exigir MFA quando as operações de API forem chamadas, adicione condições de MFA às suas políticas. Para obter mais informações, consulte [Configuração de acesso à API protegido por MFA](#) no Guia do usuário do IAM.

Para obter mais informações sobre as práticas recomendadas do IAM, consulte [Práticas recomendadas de segurança no IAM](#) no Guia do usuário do IAM.

## Política AWSCodeStarServiceRole

A política `aws-codestar-service-role` é anexada à função de serviço que permite que o AWS CodeStar execute ações com outros serviços. Na primeira vez que fizer login no AWS CodeStar, você criará a função de serviço. É necessário criá-la somente uma vez. A política será anexada automaticamente à função de serviço depois que ela for criada.

```
{
  "Version": "2012-10-17",
  "Statement": [
```

```

{
  "Sid": "ProjectEventRules",
  "Effect": "Allow",
  "Action": [
    "events:PutTargets",
    "events:RemoveTargets",
    "events:PutRule",
    "events>DeleteRule",
    "events:DescribeRule"
  ],
  "Resource": [
    "arn:aws:events:*:*:rule/awscodestar-*"
  ]
},
{
  "Sid": "ProjectStack",
  "Effect": "Allow",
  "Action": [
    "cloudformation:*Stack*",
    "cloudformation:CreateChangeSet",
    "cloudformation:ExecuteChangeSet",
    "cloudformation>DeleteChangeSet",
    "cloudformation:GetTemplate"
  ],
  "Resource": [
    "arn:aws:cloudformation:*:*:stack/awscodestar-*",
    "arn:aws:cloudformation:*:*:stack/awseb-*",
    "arn:aws:cloudformation:*:*:stack/aws-cloud9-*",
    "arn:aws:cloudformation:*:aws:transform/CodeStar*"
  ]
},
{
  "Sid": "ProjectStackTemplate",
  "Effect": "Allow",
  "Action": [
    "cloudformation:GetTemplateSummary",
    "cloudformation:DescribeChangeSet"
  ],
  "Resource": "*"
},
{
  "Sid": "ProjectQuickstarts",
  "Effect": "Allow",
  "Action": [

```

```

        "s3:GetObject"
    ],
    "Resource": [
        "arn:aws:s3:::awscodestar-*/*"
    ]
},
{
    "Sid": "ProjectS3Buckets",
    "Effect": "Allow",
    "Action": [
        "s3:*"
    ],
    "Resource": [
        "arn:aws:s3:::aws-codestar-*",
        "arn:aws:s3:::elasticbeanstalk-*"
    ]
},
{
    "Sid": "ProjectServices",
    "Effect": "Allow",
    "Action": [
        "codestar:*",
        "codecommit:*",
        "codepipeline:*",
        "codedeploy:*",
        "codebuild:*",
        "autoscaling:*",
        "cloudwatch:Put*",
        "ec2:*",
        "elasticbeanstalk:*",
        "elasticloadbalancing:*",
        "iam:ListRoles",
        "logs:*",
        "sns:*",
        "cloud9:CreateEnvironmentEC2",
        "cloud9>DeleteEnvironment",
        "cloud9:DescribeEnvironment*",
        "cloud9:ListEnvironments"
    ],
    "Resource": "*"
},
{
    "Sid": "ProjectWorkerRoles",
    "Effect": "Allow",

```

```

    "Action": [
      "iam:AttachRolePolicy",
      "iam:CreateRole",
      "iam>DeleteRole",
      "iam>DeleteRolePolicy",
      "iam:DetachRolePolicy",
      "iam:GetRole",
      "iam:PassRole",
      "iam:GetRolePolicy",
      "iam:PutRolePolicy",
      "iam:SetDefaultPolicyVersion",
      "iam:CreatePolicy",
      "iam>DeletePolicy",
      "iam:AddRoleToInstanceProfile",
      "iam:CreateInstanceProfile",
      "iam>DeleteInstanceProfile",
      "iam:RemoveRoleFromInstanceProfile"
    ],
    "Resource": [
      "arn:aws:iam::*:role/CodeStarWorker*",
      "arn:aws:iam::*:policy/CodeStarWorker*",
      "arn:aws:iam::*:instance-profile/awscodestar-*"
    ]
  },
  {
    "Sid": "ProjectTeamMembers",
    "Effect": "Allow",
    "Action": [
      "iam:AttachUserPolicy",
      "iam:DetachUserPolicy"
    ],
    "Resource": "*",
    "Condition": {
      "ArnEquals": {
        "iam:PolicyArn": [
          "arn:aws:iam::*:policy/CodeStar_*"
        ]
      }
    }
  }
},
{
  "Sid": "ProjectRoles",
  "Effect": "Allow",
  "Action": [

```

```

        "iam:CreatePolicy",
        "iam>DeletePolicy",
        "iam:CreatePolicyVersion",
        "iam>DeletePolicyVersion",
        "iam:ListEntitiesForPolicy",
        "iam:ListPolicyVersions",
        "iam:GetPolicy",
        "iam:GetPolicyVersion"
    ],
    "Resource": [
        "arn:aws:iam::*:policy/CodeStar_*"
    ]
},
{
    "Sid": "InspectServiceRole",
    "Effect": "Allow",
    "Action": [
        "iam:ListAttachedRolePolicies"
    ],
    "Resource": [
        "arn:aws:iam::*:role/aws-codestar-service-role",
        "arn:aws:iam::*:role/service-role/aws-codestar-service-role"
    ]
},
{
    "Sid": "IAMLinkRole",
    "Effect": "Allow",
    "Action": [
        "iam:CreateServiceLinkedRole"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "iam:AWSServiceName": "cloud9.amazonaws.com"
        }
    }
},
{
    "Sid": "DescribeConfigRuleForARN",
    "Effect": "Allow",
    "Action": [
        "config:DescribeConfigRules"
    ],
    "Resource": [

```



```

        "*"
    ]
},
{
    "Sid": "ProjectCodeStarConnections",
    "Effect": "Allow",
    "Action": [
        "codestar-connections:UseConnection",
        "codestar-connections:GetConnection"
    ],
    "Resource": "*"
},
{
    "Sid": "ProjectCodeStarConnectionsPassConnections",
    "Effect": "Allow",
    "Action": "codestar-connections:PassConnection",
    "Resource": "*",
    "Condition": {
        "StringEqualsIfExists": {
            "codestar-connections:PassedToService":
"codepipeline.amazonaws.com"
        }
    }
}
]
}

```

## Política AWSCodeStarFullAccess

Nas instruções [Configuração AWS CodeStar](#), você anexou uma política chamada `AWSCodeStarFullAccess` ao usuário do IAM. Essa declaração de política permite que o usuário realize todas as ações disponíveis no AWS CodeStar com todos os recursos do AWS CodeStar disponíveis associados à conta da AWS. Isso inclui a criação e a exclusão de projetos. O exemplo a seguir é um trecho de uma política `AWSCodeStarFullAccess` representativa. A política real difere dependendo do modelo selecionado ao iniciar um novo projeto do AWS CodeStar.

O AWS CloudFormation exige permissão do `cloudformation::ListStacks` ao chamar `cloudformation::DescribeStacks` sem uma pilha-alvo.

### Detalhes da permissão

Esta política inclui as permissões para fazer o seguinte:

- `ec2`–Recuperar informações sobre instâncias do EC2 para criar um projeto do AWS CodeStar.
- `cloud9`–Recuperar informações sobre ambientes AWS Command Line Interface.
- `cloudformation`–Recuperar informações sobre pilhas de projetos do AWS CodeStar.
- `codestar`–Executar ações dentro de um projeto do AWS CodeStar.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CodeStarEC2",
      "Effect": "Allow",
      "Action": [
        "codestar:*",
        "ec2:DescribeKeyPairs",
        "ec2:DescribeVpcs",
        "ec2:DescribeSubnets",
        "cloud9:DescribeEnvironment*"
      ],
      "Resource": "*"
    },
    {
      "Sid": "CodeStarCF",
      "Effect": "Allow",
      "Action": [
        "cloudformation:DescribeStack*",
        "cloudformation:ListStacks*",
        "cloudformation:GetTemplateSummary"
      ],
      "Resource": [
        "arn:aws:cloudformation:*:*:stack/awscodestar-*"
      ]
    }
  ]
}
```

Talvez você não queira conceder todo esse acesso a todos os usuários. Em vez disso, você pode adicionar permissões no nível do projeto usando funções de projeto gerenciadas pelo AWS CodeStar. As funções concedem níveis específicos de acesso a projetos do AWS CodeStar e são nomeadas da seguinte forma:

- Proprietário
- Colaborador
- Visualizador

## Política da função de proprietário do AWS CodeStar

A política de função de proprietário do AWS CodeStar permite que um usuário execute todas as ações em um projeto do AWS CodeStar sem restrições. O AWS CodeStar aplica a política do CodeStar\_*project-id*\_Owner aos membros da equipe do projeto com o nível de acesso de proprietário.

```

...
{
  "Effect": "Allow",
  "Action": [
    ...
    "codestar:*",
    ...
  ],
  "Resource": [
    "arn:aws:codestar:us-east-2:111111111111:project/project-id",
    "arn:aws:iam::account-id:policy/CodeStar_project-id_Owner"
  ]
},
{
  "Effect": "Allow",
  "Action": [
    "codestar:DescribeUserProfile",
    "codestar:ListProjects",
    "codestar:ListUserProfiles",
    "codestar:VerifyServiceRole",
    ...
  ],
  "Resource": [
    "*"
  ]
},
{
  "Effect": "Allow",
  "Action": [
    "codestar:*UserProfile",

```

```

...
],
"Resource": [
  "arn:aws:iam::account-id:user/user-name"
]
}
...

```

## Política de função de colaborador do AWS CodeStar

A política de função de colaborador do AWS CodeStar permite que um usuário colabore no projeto e altere o painel de controle do projeto. O AWS CodeStar aplica a política `CodeStar_project-id_Contributor` aos membros da equipe do projeto com o nível de acesso de colaborador. Os usuários com acesso de colaborador podem colaborar com o projeto e alterar o painel do projeto, mas não podem adicionar ou remover membros da equipe.

```

...
{
  "Effect": "Allow",
  "Action": [
    ...
    "codestar:Describe*",
    "codestar:Get*",
    "codestar:List*",
    "codestar:PutExtendedAccess",
    ...
  ],
  "Resource": [
    "arn:aws:codestar:us-east-2:111111111111:project/project-id",
    "arn:aws:iam::account-id:policy/CodeStar_project-id_Contributor"
  ]
},
{
  "Effect": "Allow",
  "Action": [
    "codestar:DescribeUserProfile",
    "codestar:ListProjects",
    "codestar:ListUserProfiles",
    "codestar:VerifyServiceRole",
    ...
  ],
  "Resource": [
    "*"
  ]
}
...

```

```

]
},
{
  "Effect": "Allow",
  "Action": [
    "codestar:*UserProfile",
    ...
  ],
  "Resource": [
    "arn:aws:iam::account-id:user/user-name"
  ]
}
...

```

## Política da função de visualizador do AWS CodeStar

A política de função de visualizador do AWS CodeStar permite que um usuário visualize um projeto no AWS CodeStar. O AWS CodeStar aplica a política do CodeStar\_*project-id*\_Viewer aos membros da equipe do projeto com o nível de acesso de visualizador. Os usuários com acesso de visualizador podem visualizar um projeto no AWS CodeStar, mas não podem alterar seus recursos ou adicionar ou remover membros da equipe.

```

...
{
  "Effect": "Allow",
  "Action": [
    ...
    "codestar:Describe*",
    "codestar:Get*",
    "codestar:List*",
    ...
  ],
  "Resource": [
    "arn:aws:codestar:us-east-2:111111111111:project/project-id",
    "arn:aws:iam::account-id:policy/CodeStar_project-id_Viewer"
  ]
},
{
  "Effect": "Allow",
  "Action": [
    "codestar:DescribeUserProfile",
    "codestar:ListProjects",
    "codestar:ListUserProfiles",

```

```

    "codestar:VerifyServiceRole",
    ...
  ],
  "Resource": [
    "*"
  ]
},
{
  "Effect": "Allow",
  "Action": [
    "codestar:*UserProfile",
    ...
  ],
  "Resource": [
    "arn:aws:iam::account-id:user/user-name"
  ]
}
...

```

## Política de funções dos trabalhadores do Toolchain do AWS CodeStar (após 6 de dezembro de 2018 no horário do Pacífico)

Para projetos do AWS CodeStar criados depois de 6 de dezembro de 2018, horário do Pacífico, o AWS CodeStar cria uma política em linha para uma função de operador que cria recursos para seu projeto em outros serviços do do AWS. O conteúdo da política depende do tipo de projeto que está sendo criado. A política a seguir é um exemplo. Para obter mais informações, consulte [Políticas do IAM para funções de operador](#).

```

{
  "Statement": [
    {
      "Action": [
        "s3:GetObject",
        "s3:GetObjectVersion",
        "s3:GetBucketVersioning",
        "s3:PutObject*",
        "codecommit:CancelUploadArchive",
        "codecommit:GetBranch",
        "codecommit:GetCommit",
        "codecommit:GetUploadArchiveStatus",
        "codecommit:GitPull",
        "codecommit:UploadArchive",

```

```
    "codebuild:StartBuild",
    "codebuild:BatchGetBuilds",
    "codebuild:StopBuild",
    "logs:CreateLogGroup",
    "logs:CreateLogStream",
    "logs:PutLogEvents",
    "cloudformation:DescribeStacks",
    "cloudformation:DescribeChangeSet",
    "cloudformation:CreateChangeSet",
    "cloudformation>DeleteChangeSet",
    "cloudformation:ExecuteChangeSet",
    "codepipeline:StartPipelineExecution",
    "lambda:ListFunctions",
    "lambda:InvokeFunction",
    "sns:Publish"
  ],
  "Resource": [
    "*"
  ],
  "Effect": "Allow"
},
{
  "Action": [
    "iam:PassRole"
  ],
  "Resource": [
    "*"
  ],
  "Effect": "Allow"
},
{
  "Action": [
    "kms:GenerateDataKey*",
    "kms:Encrypt",
    "kms:Decrypt"
  ],
  "Resource": [
    "*"
  ],
  "Effect": "Allow"
}
]
```

## Política da função de operador do AWS CloudFormation

Para projetos do AWS CodeStar criados depois de 6 de dezembro de 2018, horário do Pacífico, o AWS CodeStar cria uma política em linha para uma função de operador que cria recursos do AWS CloudFormation para o projeto do seu AWS CodeStar. O conteúdo da política depende do tipo de recursos necessários para o projeto. A política a seguir é um exemplo. Para obter mais informações, consulte [Políticas do IAM para funções de operador](#).

```
{
  {
    "Statement": [
      {
        "Action": [
          "s3:PutObject",
          "s3:GetObject",
          "s3:GetObjectVersion"
        ],
        "Resource": [
          "arn:aws:s3:::aws-codestar-region-id-account-id-project-id",
          "arn:aws:s3:::aws-codestar-region-id-account-id-project-id/*"
        ],
        "Effect": "Allow"
      },
      {
        "Action": [
          "apigateway:DELETE",
          "apigateway:GET",
          "apigateway:PATCH",
          "apigateway:POST",
          "apigateway:PUT",
          "codedeploy:CreateApplication",
          "codedeploy:CreateDeployment",
          "codedeploy:CreateDeploymentConfig",
          "codedeploy:CreateDeploymentGroup",
          "codedeploy>DeleteApplication",
          "codedeploy>DeleteDeployment",
          "codedeploy>DeleteDeploymentConfig",
          "codedeploy>DeleteDeploymentGroup",
          "codedeploy:GetDeployment",
          "codedeploy:GetDeploymentConfig",
          "codedeploy:GetDeploymentGroup",
```



```
"codedeploy:RegisterApplicationRevision",
"codestar:SyncResources",
"config>DeleteConfigRule",
"config:DescribeConfigRules",
"config:ListTagsForResource",
"config:PutConfigRule",
"config:TagResource",
"config:UntagResource",
"dynamodb>CreateTable",
"dynamodb>DeleteTable",
"dynamodb:DescribeContinuousBackups",
"dynamodb:DescribeTable",
"dynamodb:DescribeTimeToLive",
"dynamodb:ListTagsOfResource",
"dynamodb:TagResource",
"dynamodb:UntagResource",
"dynamodb:UpdateContinuousBackups",
"dynamodb:UpdateTable",
"dynamodb:UpdateTimeToLive",
"ec2:AssociateIamInstanceProfile",
"ec2:AttachVolume",
"ec2:CreateSecurityGroup",
"ec2:createTags",
"ec2:DescribeIamInstanceProfileAssociations",
"ec2:DescribeInstances",
"ec2:DescribeSecurityGroups",
"ec2:DescribeSubnets",
"ec2:DetachVolume",
"ec2:DisassociateIamInstanceProfile",
"ec2:ModifyInstanceAttribute",
"ec2:ModifyInstanceCreditSpecification",
"ec2:ModifyInstancePlacement",
"ec2:MonitorInstances",
"ec2:ReplaceIamInstanceProfileAssociation",
"ec2:RunInstances",
"ec2:StartInstances",
"ec2:StopInstances",
"ec2:TerminateInstances",
"events>DeleteRule",
"events:DescribeRule",
"events:ListTagsForResource",
"events:PutRule",
"events:PutTargets",
"events:RemoveTargets",
```

```
"events:TagResource",
"events:UntagResource",
"kinesis:AddTagsToStream",
"kinesis:CreateStream",
"kinesis:DecreaseStreamRetentionPeriod",
"kinesis>DeleteStream",
"kinesis:DescribeStream",
"kinesis:IncreaseStreamRetentionPeriod",
"kinesis:RemoveTagsFromStream",
"kinesis:StartStreamEncryption",
"kinesis:StopStreamEncryption",
"kinesis:UpdateShardCount",
"lambda:CreateAlias",
"lambda:CreateFunction",
"lambda>DeleteAlias",
"lambda>DeleteFunction",
"lambda>DeleteFunctionConcurrency",
"lambda:GetFunction",
"lambda:GetFunctionConfiguration",
"lambda:ListTags",
"lambda:ListVersionsByFunction",
"lambda:PublishVersion",
"lambda:PutFunctionConcurrency",
"lambda:TagResource",
"lambda:UntagResource",
"lambda:UpdateAlias",
"lambda:UpdateFunctionCode",
"lambda:UpdateFunctionConfiguration",
"s3:CreateBucket",
"s3>DeleteBucket",
"s3>DeleteBucketWebsite",
"s3:PutAccelerateConfiguration",
"s3:PutAnalyticsConfiguration",
"s3:PutBucketAcl",
"s3:PutBucketCORS",
"s3:PutBucketLogging",
"s3:PutBucketNotification",
"s3:PutBucketPublicAccessBlock",
"s3:PutBucketVersioning",
"s3:PutBucketWebsite",
"s3:PutEncryptionConfiguration",
"s3:PutInventoryConfiguration",
"s3:PutLifecycleConfiguration",
"s3:PutMetricsConfiguration",
```

```

        "s3:PutReplicationConfiguration",
        "sns:CreateTopic",
        "sns>DeleteTopic",
        "sns:GetTopicAttributes",
        "sns:ListSubscriptionsByTopic",
        "sns:ListTopics",
        "sns:SetSubscriptionAttributes",
        "sns:Subscribe",
        "sns:Unsubscribe",
        "sqs:CreateQueue",
        "sqs>DeleteQueue",
        "sqs:GetQueueAttributes",
        "sqs:GetQueueUrl",
        "sqs:ListQueueTags",
        "sqs:TagQueue",
        "sqs:UntagQueue"
    ],
    "Resource": "*",
    "Effect": "Allow"
},
{
    "Action": [
        "lambda:AddPermission",
        "lambda:RemovePermission"
    ],
    "Resource": [
        "arn:aws:lambda:region-id:account-id:function:awscodestar-*"
    ],
    "Effect": "Allow"
},
{
    "Action": [
        "iam:PassRole"
    ],
    "Resource": [
        "arn:aws:iam::account-id:role/CodeStar-project-id*"
    ],
    "Effect": "Allow"
},
{
    "Condition": {
        "StringEquals": {
            "iam:PassedToService": "codedeploy.amazonaws.com"
        }
    }
}

```

```

    },
    "Action": [
        "iam:PassRole"
    ],
    "Resource": [
        "arn:aws:iam::account-id:role/CodeStarWorker-project-id-CodeDeploy"
    ],
    "Effect": "Allow"
},
{
    "Action": [
        "cloudformation:CreateChangeSet"
    ],
    "Resource": [
        "arn:aws:cloudformation:region-id:aws:transform/Serverless-2016-10-31",
        "arn:aws:cloudformation:region-id:aws:transform/CodeStar"
    ],
    "Effect": "Allow"
},
{
    "Action": [
        "iam:CreateServiceLinkedRole",
        "iam:GetRole",
        "iam>DeleteRole",
        "iam>DeleteUser"
    ],
    "Resource": "*",
    "Effect": "Allow"
},
{
    "Condition": {
        "StringEquals": {
            "iam:PermissionsBoundary": "arn:aws:iam::account-id:policy/CodeStar_project-id_PermissionsBoundary"
        }
    },
    "Action": [
        "iam:AttachRolePolicy",
        "iam:AttachUserPolicy",
        "iam:CreateRole",
        "iam:CreateUser",
        "iam>DeleteRolePolicy",
        "iam>DeleteUserPolicy",
        "iam:DetachUserPolicy",

```

```

        "iam:DetachRolePolicy",
        "iam:PutUserPermissionsBoundary",
        "iam:PutRolePermissionsBoundary"
    ],
    "Resource": "*",
    "Effect": "Allow"
},
{
    "Action": [
        "kms:CreateKey",
        "kms:CreateAlias",
        "kms>DeleteAlias",
        "kms:DisableKey",
        "kms:EnableKey",
        "kms:UpdateAlias",
        "kms:TagResource",
        "kms:UntagResource"
    ],
    "Resource": "*",
    "Effect": "Allow"
},
{
    "Condition": {
        "StringEquals": {
            "ssm:ResourceTag/awscodestar:projectArn":
"arn:aws:codestar:project-id:account-id:project/project-id"
        }
    },
    "Action": [
        "ssm:GetParameter*"
    ],
    "Resource": "*",
    "Effect": "Allow"
}
]
}

```

## Política de funções do trabalhador do AWS CloudFormation (antes de 6 de dezembro de 2018 no horário do Pacífico)

Se seu projeto do AWS CodeStar foi criado antes de 6 de dezembro de 2018 no horário do Pacífico, o AWS CodeStar criou uma política em linha para uma função de trabalhador do AWS CloudFormation. A declaração de política a seguir é um exemplo.

```
{
  "Statement": [
    {
      "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:GetObjectVersion"
      ],
      "Resource": [
        "arn:aws:s3:::aws-codestar-us-east-1-account-id-project-id-pipe",
        "arn:aws:s3:::aws-codestar-us-east-1-account-id-project-id-pipe/*"
      ],
      "Effect": "Allow"
    },
    {
      "Action": [
        "codestar:SyncResources",
        "lambda:CreateFunction",
        "lambda>DeleteFunction",
        "lambda:AddPermission",
        "lambda:UpdateFunction",
        "lambda:UpdateFunctionCode",
        "lambda:GetFunction",
        "lambda:GetFunctionConfiguration",
        "lambda:UpdateFunctionConfiguration",
        "lambda:RemovePermission",
        "lambda:listTags",
        "lambda:TagResource",
        "lambda:UntagResource",
        "apigateway:*",
        "dynamodb:CreateTable",
        "dynamodb>DeleteTable",
        "dynamodb:DescribeTable",
        "kinesis:CreateStream",
        "kinesis>DeleteStream",
        "kinesis:DescribeStream",
```

```

        "sns:CreateTopic",
        "sns>DeleteTopic",
        "sns:ListTopics",
        "sns:GetTopicAttributes",
        "sns:SetTopicAttributes",
        "s3:CreateBucket",
        "s3>DeleteBucket",
        "config:DescribeConfigRules",
        "config:PutConfigRule",
        "config>DeleteConfigRule",
        "ec2:*",
        "autoscaling:*",
        "elasticloadbalancing:*",
        "elasticbeanstalk:*"
    ],
    "Resource": "*",
    "Effect": "Allow"
},
{
    "Action": [
        "iam:PassRole"
    ],
    "Resource": [
        "arn:aws:iam::account-id:role/CodeStarWorker-project-id-Lambda"
    ],
    "Effect": "Allow"
},
{
    "Action": [
        "cloudformation:CreateChangeSet"
    ],
    "Resource": [
        "arn:aws:cloudformation:us-east-1:aws:transform/Serverless-2016-10-31",
        "arn:aws:cloudformation:us-east-1:aws:transform/CodeStar"
    ],
    "Effect": "Allow"
}
]
}

```

## Política de funções do trabalhador do AWS CodePipeline (antes de 6 de dezembro de 2018 no horário do Pacífico)

Se seu projeto do AWS CodeStar foi criado antes de 6 de dezembro de 2018 no horário do Pacífico, o AWS CodeStar criou uma política em linha para uma função de trabalhador do CodePipeline. A declaração de política a seguir é um exemplo.

```
{
  "Statement": [
    {
      "Action": [
        "s3:GetObject",
        "s3:GetObjectVersion",
        "s3:GetBucketVersioning",
        "s3:PutObject"
      ],
      "Resource": [
        "arn:aws:s3::aws-codestar-us-east-1-account-id-project-id-pipe",
        "arn:aws:s3::aws-codestar-us-east-1-account-id-project-id-pipe/*"
      ],
      "Effect": "Allow"
    },
    {
      "Action": [
        "codecommit:CancelUploadArchive",
        "codecommit:GetBranch",
        "codecommit:GetCommit",
        "codecommit:GetUploadArchiveStatus",
        "codecommit:UploadArchive"
      ],
      "Resource": [
        "arn:aws:codecommit:us-east-1-account-id-project-id"
      ],
      "Effect": "Allow"
    },
    {
      "Action": [
        "codebuild:StartBuild",
        "codebuild:BatchGetBuilds",
        "codebuild:StopBuild"
      ],
      "Resource": [
        "arn:aws:codebuild:us-east-1-account-id:project/project-id"
      ]
    }
  ]
}
```



```

    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "cloudformation:DescribeStacks",
      "cloudformation:DescribeChangeSet",
      "cloudformation:CreateChangeSet",
      "cloudformation>DeleteChangeSet",
      "cloudformation:ExecuteChangeSet"
    ],
    "Resource": [
      "arn:aws:cloudformation:us-east-1:account-id:stack/awscodestar-project-id-lambda/*"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "iam:PassRole"
    ],
    "Resource": [
      "arn:aws:iam::account-id:role/CodeStarWorker-project-id-CloudFormation"
    ],
    "Effect": "Allow"
  }
]
}

```

## Política de funções do trabalhador do AWS CodeBuild (antes de 6 de dezembro de 2018 no horário do Pacífico)

Se seu projeto do AWS CodeStar foi criado antes de 6 de dezembro de 2018 no horário do Pacífico, o AWS CodeStar criou uma política em linha para uma função de trabalhador do CodeBuild. A declaração de política a seguir é um exemplo.

```

{
  "Statement": [
    {
      "Action": [
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:PutLogEvents"
      ]
    }
  ]
}

```

```

    ],
    "Resource": "*",
    "Effect": "Allow"
  },
  {
    "Action": [
      "s3:PutObject",
      "s3:GetObject",
      "s3:GetObjectVersion"
    ],
    "Resource": [
      "arn:aws:s3::aws-codestar-us-east-1-account-id-project-id-pipe",
      "arn:aws:s3::aws-codestar-us-east-1-account-id-project-id-pipe/*",
      "arn:aws:s3::aws-codestar-us-east-1-account-id-project-id-app",
      "arn:aws:s3::aws-codestar-us-east-1-account-id-project-id-app/*"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "codecommit:GitPull"
    ],
    "Resource": [
      "arn:aws:codecommit:us-east-1:account-id:project-id"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "kms:GenerateDataKey*",
      "kms:Encrypt",
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:us-east-1:account-id:alias/aws/s3"
    ],
    "Effect": "Allow"
  }
]
}

```

## Política de funções de trabalhadores do Amazon CloudWatch Events (antes de 6 de dezembro de 2018 no horário do Pacífico)

Se seu projeto do AWS CodeStar foi criado antes de 6 de dezembro de 2018 no horário do Pacífico, o AWS CodeStar criou uma política em linha para uma função de trabalhador do CloudWatch Events. A declaração de política a seguir é um exemplo.

```
{
  "Statement": [
    {
      "Action": [
        "codepipeline:StartPipelineExecution"
      ],
      "Resource": [
        "arn:aws:codepipeline:us-east-1:account-id:project-id-Pipeline"
      ],
      "Effect": "Allow"
    }
  ]
}
```

## Política de limite de permissões do AWS CodeStar

Se você criar um projeto do AWS CodeStar após 6 de dezembro de 2018 no horário do Pacífico, o AWS CodeStar criará uma política de limite de permissões para seu projeto. Essa política impede o escalonamento de privilégios a recursos fora do projeto. É uma política dinâmica que é atualizada conforme o projeto evolui. O conteúdo da política depende do tipo de projeto que está sendo criado. A política a seguir é um exemplo. Para obter mais informações, consulte [Limite de permissões do IAM](#).

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "1",
      "Effect": "Allow",
      "Action": [
        "s3:GetObject"
      ],
      "Resource": [
        "arn:aws:s3:::*/AWSLogs/*/Config/*"
      ]
    }
  ]
}
```

```

    ]
  },
  {
    "Sid": "2",
    "Effect": "Allow",
    "Action": [
      "*"
    ],
    "Resource": [
      "arn:aws:codestar:us-east-1:account-id:project/project-id",
      "arn:aws:cloudformation:us-east-1:account-id:stack/awscodestar-project-id-lambda/eefbbf20-c1d9-11e8-8a3a-500c28b4e461",
      "arn:aws:cloudformation:us-east-1:account-id:stack/awscodestar-project-id/4b80b3f0-c1d9-11e8-8517-500c28b236fd",
      "arn:aws:codebuild:us-east-1:account-id:project/project-id",
      "arn:aws:codecommit:us-east-1:account-id:project-id",
      "arn:aws:codepipeline:us-east-1:account-id:project-id-Pipeline",
      "arn:aws:execute-api:us-east-1:account-id:7rlst5mrgi",
      "arn:aws:iam::account-id:role/CodeStarWorker-project-id-CloudFormation",
      "arn:aws:iam::account-id:role/CodeStarWorker-project-id-CloudWatchEventRule",
      "arn:aws:iam::account-id:role/CodeStarWorker-project-id-CodeBuild",
      "arn:aws:iam::account-id:role/CodeStarWorker-project-id-CodePipeline",
      "arn:aws:iam::account-id:role/CodeStarWorker-project-id-Lambda",
      "arn:aws:lambda:us-east-1:account-id:function:awscodestar-project-id-lambda-GetHelloWorld-KFKTXYNH9573",
      "arn:aws:s3::aws-codestar-us-east-1-account-id-project-id-app",
      "arn:aws:s3::aws-codestar-us-east-1-account-id-project-id-pipe"
    ]
  },
  {
    "Sid": "3",
    "Effect": "Allow",
    "Action": [
      "apigateway:GET",
      "config:Describe*",
      "config:Get*",
      "config:List*",
      "config:Put*",
      "logs:CreateLogGroup",
      "logs:CreateLogStream",
      "logs:DescribeLogGroups",
      "logs:PutLogEvents"
    ],
    "Resource": [

```

```
        "*"
    ]
}
]
```

## Listar recursos de um projeto

Neste exemplo, você deseja conceder acesso a um usuário do IAM especificado em sua conta do AWS para listar os recursos de um projeto do AWS CodeStar.

```
{
  "Version": "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "codestar:ListResources",
      ],
      "Resource" : "arn:aws:codestar:us-east-2:project/my-first-projec"
    }
  ]
}
```

## Usando o console do AWS CodeStar

Não são necessárias permissões específicas para acessar o console do AWS CodeStar, mas você não poderá fazer nada útil, a menos que tenha a política do `AWSCoDeStarFullAccess` ou uma das funções de nível de projeto do AWS CodeStar: proprietário, colaborador ou visualizador. Para obter mais informações sobre `AWSCoDeStarFullAccess`, consulte [Política AWSCoDeStarFullAccess](#). Para obter mais informações sobre as políticas de nível de projeto, consulte [Políticas do IAM para membros de equipe](#).

Não é necessário conceder permissões mínimas do console para usuários que fazem chamadas somente à AWS CLI ou à API do AWS. Em vez disso, permita o acesso somente às ações que correspondem à operação da API que você está tentando executar.

## Permitir que os usuários visualizem suas próprias permissões

Este exemplo mostra como é possível criar uma política que permite que os usuários do IAM visualizem as políticas gerenciadas e em linha anexadas a sua identidade de usuário. Essa política

inclui permissões para concluir essa ação no console ou de forma programática usando a AWS CLI ou a AWS API.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupForUser",
        "iam:ListAttachedUserPolicies",
        "iam:ListUserPolicies",
        "iam:GetUser"
      ],
      "Resource": ["arn:aws:iam::*:user/${aws:username}"]
    },
    {
      "Sid": "NavigateInConsole",
      "Effect": "Allow",
      "Action": [
        "iam:GetGroupPolicy",
        "iam:GetPolicyVersion",
        "iam:GetPolicy",
        "iam:ListAttachedGroupPolicies",
        "iam:ListGroupPolicies",
        "iam:ListPolicyVersions",
        "iam:ListPolicies",
        "iam:ListUsers"
      ],
      "Resource": "*"
    }
  ]
}
```

## Atualizar um projeto do AWS CodeStar

Neste exemplo, você deseja conceder acesso a um usuário do IAM especificado em sua conta do AWS para editar os atributos de um projeto do AWS CodeStar, como a descrição do projeto.

```
{
```

```
"Version": "2012-10-17",
"Statement" : [
  {
    "Effect" : "Allow",
    "Action" : [
      "codestar:UpdateProject"
    ],
    "Resource" : "arn:aws:codestar:us-east-2:project/my-first-projec"
  }
]
```

## Adicionar um membro da equipe a um projeto

Neste exemplo, você deseja conceder a um usuário do usuário do IAM especificado a capacidade de adicionar membros da equipe a um projeto do AWS CodeStar com o ID do projeto *my-first-projec*, mas negar explicitamente a esse usuário a capacidade de remover membros da equipe:

```
{
  "Version": "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "codestar:AssociateTeamMember",
      ],
      "Resource" : "arn:aws:codestar:us-east-2:project/my-first-projec"
    },
    {
      "Effect" : "Deny",
      "Action" : [
        "codestar:DisassociateTeamMember",
      ],
      "Resource" : "arn:aws:codestar:us-east-2:project/my-first-projec"
    }
  ]
}
```

## Listar perfis de usuário associados a uma conta do AWS

No exemplo a seguir, você permite que um usuário do IAM que tenha essa política anexada, liste todos os perfis de usuário do AWS CodeStar associados a uma conta do AWS:

```
{
  "Version": "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "codestar:ListUserProfiles",
      ],
      "Resource" : "*"
    }
  ]
}
```

## Visualizar projetos do AWS CodeStar com base em tags

É possível usar condições em sua política baseada em identidade para controlar o acesso aos recursos do AWS CodeStar com base em tags. Este exemplo mostra como é possível criar uma política que permite visualizar um projeto. No entanto, a permissão será concedida somente se a tag `Owner` do projeto tiver o valor do nome de usuário desse usuário. Essa política também concede as permissões necessárias concluir essa ação no console.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ListProjectsInConsole",
      "Effect": "Allow",
      "Action": "codestar:ListProjects",
      "Resource": "*"
    },
    {
      "Sid": "ViewProjectIfOwner",
      "Effect": "Allow",
      "Action": "codestar:GetProject",
      "Resource": "arn:aws:codestar:*:*:project/*",
      "Condition": {
        "StringEquals": {"codestar:ResourceTag/Owner": "${aws:username}"}
      }
    }
  ]
}
```



```

    }
  }
]
}

```

É possível anexar essa política aos usuários do IAM na sua conta. Se um usuário chamado `richard-roe` tentar visualizar um projeto do AWS CodeStar, o projeto deverá ser marcado como `Owner=richard-roe or owner=richard-roe`. Caso contrário, ele terá o acesso negado. A chave da tag de condição `Owner` corresponde a `Owner` e a `owner` porque os nomes das chaves de condição não fazem distinção entre maiúsculas e minúsculas. Para obter mais informações, consulte [IAM JSON Policy Elements: Condition](#) (Elementos da política JSON do IAM: Condição) no Guia do usuário do IAM.

## Atualizações do AWS CodeStar para políticas gerenciadas pela AWS

Visualize detalhes sobre atualizações em políticas gerenciadas pela AWS para o AWS CodeStar desde que esse serviço começou a rastrear essas alterações. Para receber alertas automáticos sobre alterações realizadas nesta página, inscreva-se no feed RSS na página [Histórico de documentos do AWS CodeStar](#).

Alteração	Descrição	Data
<a href="#">Política do AWScodeStarfullAccess — Atualize a política do AWScodeStarfullAccess</a>	A política de acesso do AWS CodeStar foi atualizada. O resultado da política é o mesmo, mas o cloudformation requer ListStacks além de DescribeStacks, o que já é necessário.	24 de março de 2023
<a href="#">Política do AWScodestarServiceRole — Atualize a política do AWScodestarServiceRole</a>	A política do perfil de serviço do AWS CodeStar foi atualizada para corrigir ações redundantes na declaração de política.  A política de perfil de serviço permite que o serviço do AWS	23 de setembro de 2021

Alteração	Descrição	Data
	CodeStar execute ações em seu nome.	
O AWS CodeStar começou a monitorar alterações	O AWS CodeStar começou a monitorar as alterações nas políticas gerenciadas do AWS.	23 de setembro de 2021

## Solução de problemas de identidade e acesso do AWS CodeStar Identity

Use as seguintes informações para ajudar a diagnosticar e corrigir problemas comuns que podem ser encontrados ao trabalhar com o AWS CodeStar e o IAM.

### Tópicos

- [Não tenho autorização para executar uma ação no AWS CodeStar](#)
- [Não estou autorizado a executar iam:PassRole](#)
- [Desejo permitir que pessoas fora da minha conta da AWS acessem meus recursos do AWS CodeStar](#)

### Não tenho autorização para executar uma ação no AWS CodeStar

Se o AWS Management Console informar que você não está autorizado a executar uma ação, entre em contato com o administrador para obter assistência. Seu administrador é a pessoa que forneceu suas credenciais de login.

O exemplo de erro a seguir ocorre quando o usuário mateojackson do IAM tenta usar o console para visualizar detalhes sobre um *widget*, mas não tem as permissões `codestar:GetWidget`.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
codestar:GetWidget on resource: my-example-widget
```

Neste caso, Mateo pede ao administrador para atualizar suas políticas para permitir a ele o acesso ao recurso *my-example-widget* usando a ação `codestar:GetWidget`.

## Não estou autorizado a executar iam:PassRole

Se você receber uma mensagem de erro informando que não tem autorização para executar a ação `iam:PassRole`, as suas políticas deverão ser atualizadas para permitir a passagem de um perfil para o AWS CodeStar.

Alguns Serviços da AWS permitem que você passe um perfil existente para o serviço, em vez de criar um novo perfil de serviço ou perfil vinculado ao serviço. Para fazer isso, é preciso ter permissões para passar o perfil para o serviço.

O erro do exemplo a seguir ocorre quando uma usuária do IAM chamada `marymajor` tenta usar o console para executar uma ação no AWS CodeStar. No entanto, a ação exige que o serviço tenha permissões concedidas por um perfil de serviço. Mary não tem permissões para passar o perfil para o serviço.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

Nesse caso, as políticas de Mary devem ser atualizadas para permitir que ela realize a ação `iam:PassRole`.

Se você precisar de ajuda, entre em contato com seu administrador AWS. Seu administrador é a pessoa que forneceu suas credenciais de login.

## Desejo permitir que pessoas fora da minha conta da AWS acessem meus recursos do AWS CodeStar

Você pode criar um perfil que os usuários de outras contas ou pessoas fora da sua organização possam usar para acessar seus recursos. Você pode especificar quem é confiável para assumir o perfil. Para serviços que oferecem suporte a políticas baseadas em recurso ou listas de controle de acesso (ACLs), você pode usar essas políticas para conceder às pessoas acesso aos seus recursos.

Para saber mais, consulte:

- Para saber se o AWS CodeStar oferece suporte a esses atributos, consulte [Como o AWS CodeStar funciona com o IAM](#).
- Para saber como conceder acesso a seus atributos em todas as Contas da AWS pertencentes a você, consulte [Fornecimento de acesso a um usuário do IAM em outra Conta da AWS pertencente a você](#) no Guia de usuário do IAM.

- Para saber como conceder acesso a seus recursos para Contas da AWS terceirizadas, consulte [Fornecimento de acesso a Contas da AWS pertencentes a terceiros](#) no Guia do usuário do IAM.
- Para saber como conceder acesso por meio da federação de identidades, consulte [Conceder acesso a usuários autenticados externamente \(federação de identidades\)](#) no Guia do usuário do IAM.
- Para saber a diferença entre usar perfis e políticas baseadas em atributos para acesso entre contas, consulte [Como os perfis do IAM diferem de políticas baseadas em atributos](#) no Guia do usuário do IAM.

## Log de chamadas de API do AWS CodeStar com o AWS CloudTrail

O AWS CodeStar é integrado ao AWS CloudTrail, um serviço que fornece um registro das ações realizadas por um usuário, uma função ou um serviço da AWS no AWS CodeStar. O CloudTrail captura as chamadas de API do AWS CodeStar como eventos. As chamadas capturadas incluem as chamadas do console do AWS CodeStar e as chamadas de código para as operações da API do AWS CodeStar. Se você criar uma trilha, poderá habilitar a entrega contínua de eventos do CloudTrail para um bucket do S3, incluindo eventos para o AWS CodeStar. Se você não configurar uma trilha, ainda poderá visualizar os eventos mais recentes no console do CloudTrail em Event history (Histórico de eventos). Ao usar as informações coletadas pelo CloudTrail, é possível determinar a solicitação feita ao AWS CodeStar, o endereço IP do qual a solicitação foi feita, quem a fez e quando ela foi feita, além de outros detalhes.

Para saber mais sobre o CloudTrail, consulte o [Guia do usuário do AWS CloudTrail](#).

### Informações do AWS CodeStar no CloudTrail

O CloudTrail é habilitado em sua conta da AWS quando ela é criada. Quando ocorre uma atividade no AWS CodeStar, essa atividade é registrada em um evento do CloudTrail com outros eventos de produtos da AWS em Event history (Histórico de eventos). Você pode visualizar, pesquisar e baixar os eventos recentes em sua conta da AWS. Para obter mais informações, consulte [Como visualizar eventos com o histórico de eventos do CloudTrail](#).

Para obter um registro contínuo de eventos na conta da AWS, incluindo eventos do AWS CodeStar, crie uma trilha. Por padrão, quando você cria uma trilha no console, ela é aplicada a todas as regiões da AWS. A trilha registra eventos de todas as regiões na partição da AWS e fornece os arquivos de log ao bucket do S3 que você especificar. Você pode configurar outros serviços da AWS para

analisar e atuar mais profundamente sobre os dados de eventos coletados nos logs do CloudTrail. Para obter mais informações, consulte as informações a seguir.

- [Visão geral da criação de uma trilha](#)
- [Serviços e integrações compatíveis com o CloudTrail](#)
- [Configurar notificações do Amazon SNS para o CloudTrail](#)
- [Receber arquivos de log do CloudTrail de várias regiões](#) e [receber arquivos de log do CloudTrail de várias contas](#)

Todas as ações do AWS CodeStar são registradas pelo CloudTrail e são documentadas na [Referência de API do AWS CodeStar](#). Por exemplo, as chamadas para as APIs `DescribeProject`, `UpdateProject` e `AssociateTeamMember` geram entradas nos arquivos de log do CloudTrail.

Cada entrada de log ou evento contém informações sobre quem gerou a solicitação. As informações de identidade ajudam a determinar:

- Se a solicitação foi feita com credenciais de usuário raiz ou do usuário do IAM.
- Se a solicitação foi feita com credenciais de segurança temporárias de uma função ou de um usuário federado.
- Se a solicitação foi feita por outro serviço da AWS.

Para obter mais informações, consulte o [Elemento `userIdentity` do CloudTrail](#).

## Noções básicas das entradas dos arquivos de log do AWS CodeStar

Os arquivos de log do CloudTrail contêm uma ou mais entradas de log. Um evento representa uma única solicitação de qualquer fonte e inclui informações sobre a ação solicitada, a data e a hora da ação, os parâmetros de solicitação e assim por diante. Os arquivos de log do CloudTrail não são um rastreamento de pilha ordenada de chamadas de API pública. Dessa forma, eles não são exibidos em uma ordem específica.

O exemplo a seguir mostra uma entrada do log do CloudTrail que demonstra uma operação do `CreateProject` sendo chamada no AWS CodeStar:

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "AssumedRole",
```

```

"principalId": "AR0AJLIN20F3UBEXAMPLE:role-name",
"arn": "arn:aws:sts::account-ID:assumed-role/role-name/role-session-name",
"accountId": "account-ID",
"accessKeyId": "ASIAJ44LFQS5XEXAMPLE",
"sessionContext": {
  "attributes": {
    "mfaAuthenticated": "false",
    "creationDate": "2017-06-04T23:56:57Z"
  },
  "sessionIssuer": {
    "type": "Role",
    "principalId": "AR0AJLIN20F3UBEXAMPLE",
    "arn": "arn:aws:iam::account-ID:role/service-role/role-name",
    "accountId": "account-ID",
    "userName": "role-name"
  }
},
"invokedBy": "codestar.amazonaws.com"
},
"eventTime": "2017-06-04T23:56:57Z",
"eventSource": "codestar.amazonaws.com",
"eventName": "CreateProject",
"awsRegion": "region-ID",
"sourceIPAddress": "codestar.amazonaws.com",
"userAgent": "codestar.amazonaws.com",
"requestParameters": {
  "clientRequestToken": "arn:aws:cloudformation:region-ID:account-ID:stack/stack-name/additional-ID",
  "id": "project-ID",
  "stackId": "arn:aws:cloudformation:region-ID:account-ID:stack/stack-name/additional-ID",
  "description": "AWS CodeStar created project",
  "name": "project-name",
  "projectTemplateId": "arn:aws:codestar:region-ID::project-template/project-template-name"
},
"responseElements": {
  "projectTemplateId": "arn:aws:codestar:region-ID::project-template/project-template-name",
  "arn": "arn:aws:codestar:us-east-1:account-ID:project/project-ID",
  "clientRequestToken": "arn:aws:cloudformation:region-ID:account-ID:stack/stack-name/additional-ID",
  "id": "project-ID"
},

```

```
"requestID": "7d7556d0-4981-11e7-a3bc-dd5daEXAMPLE",  
"eventID": "6b0d6e28-7a1e-4a73-981b-c8fdbEXAMPLE",  
"eventType": "AwsApiCall",  
"recipientAccountId": "account-ID"  
}
```

## Validação de Compatibilidade para AWS CodeStar

O AWS CodeStar não está no escopo de nenhum programa de conformidade da AWS.

Para obter uma lista dos produtos da AWS no escopo de programas de conformidade específicos, consulte [Produtos da AWS no escopo por programa de conformidade](#). Para obter informações gerais, consulte [Programas de conformidade da AWS](#).

Você pode baixar relatórios de auditoria de terceiros usando o AWS Artifact. Para obter mais informações, consulte [Download de relatórios no AWS Artifact](#).

## Resiliência no AWS CodeStar

A infraestrutura global da AWS é criada com base em regiões da AWS e zonas de disponibilidade. As regiões da AWS fornecem várias zonas de disponibilidade separadas e isoladas fisicamente, as quais são conectadas com baixa latência, alto throughput e redes altamente redundantes. Com as zonas de disponibilidade, você pode projetar e operar aplicações e bancos de dados que executam o failover automaticamente entre as zonas de disponibilidade sem interrupção. As zonas de disponibilidade são mais altamente disponíveis, tolerantes a falhas e escaláveis que uma ou várias infraestruturas de data center tradicionais.

Para obter mais informações sobre regiões e zonas de disponibilidade da AWS, consulte [Infraestrutura global da AWS](#).

## Segurança da infraestrutura no AWS CodeStar

Por ser um serviço gerenciado, o AWS CodeStar é protegido pela segurança da rede global do AWS. Para obter informações sobre serviços de segurança da AWS e como a AWS protege a infraestrutura, consulte [Segurança na Nuvem AWS](#). Para projetar seu ambiente da AWS usando as práticas recomendadas de segurança de infraestrutura, consulte [Proteção de infraestrutura](#) em Pilar segurança: AWS Well-Architected Framework.

Você usa chamadas de API publicadas pelo AWS para acessar o CodeStar por meio da rede. Os clientes devem oferecer suporte para:

- Transport Layer Security (TLS). Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Conjuntos de criptografia com perfect forward secrecy (PFS) como DHE (Ephemeral Diffie-Hellman) ou ECDHE (Ephemeral Elliptic Curve Diffie-Hellman). A maioria dos sistemas modernos, como Java 7 e versões posteriores, comporta esses modos.

Além disso, as solicitações devem ser assinadas usando um ID da chave de acesso e uma chave de acesso secreta associada a uma entidade principal do IAM. Ou você pode usar o [AWS Security Token Service](#) (AWS STS) para gerar credenciais de segurança temporárias para assinar solicitações.

Por padrão, o AWS CodeStar não isola o tráfego de serviço. Os projetos criados usando o AWS CodeStar estão abertos para a Internet pública, a menos que você modifique manualmente as configurações de acesso pelo Amazon EC2, API Gateway ou Elastic Beanstalk. Isso é intencional. É possível modificar as configurações de acesso no Amazon EC2, no API Gateway, ou no Elastic Beanstalk na medida desejada, incluindo impedir todo o acesso à Internet.

O AWS CodeStar não oferece suporte a endpoints da VPC (AWS PrivateLink) por padrão, mas é possível configurar esse suporte diretamente nos recursos do projeto.



## Limites no AWS CodeStar

A tabela a seguir descreve os limites no AWS CodeStar. O AWS CodeStar depende de outros serviços da AWS para recursos de projeto. Alguns desses limites de serviço podem ser alterados. Para obter informações sobre os limites que podem ser alterados, consulte [Limites de serviço da AWS](#).

Número de projetos	Máximo de 333 projetos em uma conta da AWS. O limite real varia de acordo com o nível de outras dependências de serviço (por exemplo, o número máximo de pipelines no CodePipeline permitido para a conta do AWS).
O número de projetos do AWS CodeStar ao qual um usuário do IAM pode pertencer	Máximo de 10 por usuário do IAM individual.
IDs de projetos	<p>Os IDs de projetos devem ser exclusivos em uma conta da AWS. Os IDs de projetos devem ter no mínimo dois caracteres e não podem exceder a 15 caracteres. Os caracteres permitidos incluem:</p> <ul style="list-style-type: none"><li>Letras a a z inclusive.</li><li>Números 0 a 9 inclusive.</li><li>O caractere especial - (sinal de menos).</li></ul> <p>Todos os outros caracteres, como letras maiúsculas, espaços . (ponto), @ (arroba) ou _ (sublinhado), não são permitidos.</p>
Nomes de projetos	Os nomes de projetos não podem ter mais que 100 caracteres de comprimento e não podem começar ou terminar com um espaço vazio.

Descrições de projetos	Qualquer combinação de caracteres entre 0 e 1.024 caracteres. As descrições de projetos são opcionais.
Membros da equipe em um projeto do AWS CodeStar	100
Nome de exibição em um perfil de usuário	Qualquer combinação de caracteres entre 1 e 100 caracteres. Os nomes de exibição devem incluir pelo menos um caractere. Esse caractere não pode ser um espaço. Os nomes de exibição não podem começar ou terminar com um espaço.
Endereço de e-mail em um perfil de usuário	O endereço de e-mail deve incluir uma @ e terminar em uma extensão de domínio válida.
Acesso federado, acesso à conta root ou acesso temporário ao AWS CodeStar	O AWS CodeStar oferece suporte a usuários federados e ao uso de credenciais de acesso temporário. O uso do AWS CodeStar com uma conta root não é recomendado.
Perfis do IAM	Um máximo de 5.120 caracteres em qualquer política gerenciada que seja anexada a um perfil do IAM.

# Solução de problemas AWS CodeStar

As informações a seguir podem ajudar a solucionar problemas comuns no AWS CodeStar.

## Tópicos

- [Falha ao criar o projeto: um projeto não foi criado](#)
- [Criação do projeto: vejo um erro quando tento editar a configuração do Amazon EC2 ao criar um projeto](#)
- [Exclusão do projeto: um AWS CodeStar projeto foi excluído, mas os recursos ainda existem](#)
- [Falha no gerenciamento da equipe: um usuário do IAM não pôde ser adicionado a uma equipe em um AWS CodeStar projeto](#)
- [Falha de acesso: um usuário federado não pode acessar um projeto AWS CodeStar](#)
- [Falha de acesso: um usuário federado não pode acessar ou criar um ambiente AWS Cloud9](#)
- [Falha de acesso: um usuário federado pode criar um AWS CodeStar projeto, mas não pode visualizar os recursos do projeto](#)
- [Problema de função de serviço: a função de serviço não pôde ser criada](#)
- [Problema de função de serviço: a função de serviço não é válida ou está ausente](#)
- [Problema de função do projeto: as verificações do status de AWS Elastic Beanstalk saúde falham em instâncias em um AWS CodeStar projeto](#)
- [Problema de função de projeto: uma função de projeto não é válida ou está ausente](#)
- [Extensões de projetos: não é possível se conectar ao JIRA](#)
- [GitHub: Não consigo acessar o histórico de commits, problemas ou código de um repositório](#)
- [AWS CloudFormation: criação da pilha revertida para permissões ausentes](#)
- [AWS CloudFormation não está autorizado a realizar iam: PassRole na função de execução do Lambda](#)
- [Não é possível criar a conexão para um GitHub repositório](#)

## Falha ao criar o projeto: um projeto não foi criado

**Problema:** quando você tenta criar um projeto, você vê uma mensagem informando que houve falha na criação.

**Correções possíveis:** os motivos mais comuns para a falha são:

- Já existe um projeto com esse ID na sua AWS conta, possivelmente em uma AWS região diferente.
- O usuário do IAM que você usou para fazer login AWS Management Console não tem as permissões necessárias para criar um projeto.
- A função AWS CodeStar de serviço não tem uma ou mais permissões necessárias.
- Você atingiu o limite máximo de um ou mais recursos para um projeto (como o limite de políticas gerenciadas pelo cliente no IAM, buckets do Amazon S3 ou pipelines em). CodePipeline

Antes de criar um projeto, verifique se você tem a política `AWSCodeStarFullAccess` aplicada para o usuário do IAM. Para ter mais informações, consulte [Política AWSCodeStarFullAccess](#).

Ao criar um projeto, verifique se o ID é exclusivo e atende aos requisitos do AWS CodeStar . Certifique-se de marcar a caixa de seleção AWS CodeStar Gostaria de ter permissão para administrar AWS recursos em seu nome.

Para solucionar outros problemas, abra o AWS CloudFormation console, escolha a pilha do projeto que você tentou criar e escolha a guia Eventos. Pode haver mais de uma pilha para um projeto. Os nomes das pilhas começam com `awscodestar-` e são seguidos pelo ID do projeto. As pilhas podem estar sob a visualização de filtro Excluídas. Analise as mensagens de falha nos eventos da pilha e corrija o problema listado como a causa dessas falhas.

## Criação do projeto: vejo um erro quando tento editar a configuração do Amazon EC2 ao criar um projeto

Problema: ao editar as opções de configuração do Amazon EC2 durante a criação do projeto, você vê uma mensagem de erro ou uma opção esmaecida e não pode continuar com a criação do projeto.

Correções possíveis: os motivos mais comuns para uma mensagem de erro são:

- A VPC no modelo do AWS CodeStar projeto (a VPC padrão ou a usada quando a configuração do Amazon EC2 foi editada) tem localização de instância dedicada, e o tipo de instância não é compatível com instâncias dedicadas. Escolha outro tipo de instância ou outro Amazon VPC.
- Sua AWS conta não tem Amazon VPCs. Você pode ter excluído a VPC padrão e não criou nenhuma outra. Abra o console da Amazon VPC em <https://console.aws.amazon.com/vpc/>, escolha Suas VPCs e certifique-se de ter pelo menos uma VPC configurada. Se não tiver, crie uma. Para obter mais informações, consulte [Visão geral do Amazon Virtual Private Cloud](#) no Guia de introdução do Amazon VPC.

- O Amazon VPC não tem nenhuma sub-rede. Escolha outra VPC ou crie uma sub-rede para a VPC. Para obter mais informações, consulte [Noções básicas da VPC e da sub-rede](#).

## Exclusão do projeto: um AWS CodeStar projeto foi excluído, mas os recursos ainda existem

Problema: um AWS CodeStar projeto foi excluído, mas os recursos criados para esse projeto ainda existem. Por padrão, AWS CodeStar exclui os recursos do projeto quando o projeto é excluído. Alguns recursos, como os buckets do Amazon S3, são mantidos, mesmo que o usuário selecione a caixa de seleção Excluir recursos, porque os buckets podem conter dados.

Possíveis correções: abra o [AWS CloudFormation console](#) e encontre uma ou mais AWS CloudFormation pilhas usadas para criar o projeto. Os nomes das pilhas começam com `awscodestar-` e são seguidos pelo ID do projeto. As pilhas podem estar sob a visualização de filtro Excluídas. Analise os eventos associados à pilha para descobrir os recursos criados para o projeto. Abra o console de cada um desses recursos na AWS região em que você criou o AWS CodeStar projeto e exclua manualmente os recursos.

Os recursos do projeto que podem permanecer incluem:

- Um ou mais buckets do projeto no Amazon S3. Ao contrário de outros recursos do projeto, os buckets do projeto no Amazon S3 não são excluídos quando a caixa de seleção Excluir recursos AWS associados junto AWS CodeStar com o projeto é marcada.

Abra o console do Amazon S3 em <https://console.aws.amazon.com/s3/>.

- Um repositório de origem para seu projeto em CodeCommit.

Abra o CodeCommit console em <https://console.aws.amazon.com/codecommit/>.

- Um pipeline para seu projeto em CodePipeline.

Abra o CodePipeline console em <https://console.aws.amazon.com/codepipeline/>.

- Um aplicativo e grupos de implantação associados em CodeDeploy.

Abra o CodeDeploy console em <https://console.aws.amazon.com/codedeploy/>.

- Um aplicativo e ambientes associados no AWS Elastic Beanstalk.

Abra o console do Elastic Beanstalk em <https://console.aws.amazon.com/elasticbeanstalk/>.

- Uma função no AWS Lambda.

Abra o AWS Lambda console em <https://console.aws.amazon.com/lambda/>.

- Uma ou mais APIs no API Gateway.

Abra o console do API Gateway em <https://console.aws.amazon.com/apigateway/>.

- Uma ou mais políticas ou perfis do IAM no IAM.

Faça login AWS Management Console e abra o console do IAM em <https://console.aws.amazon.com/iam/>.

- Uma instância do Amazon EC2.

Abra o console do Amazon EC2 em <https://console.aws.amazon.com/ec2/>.

- Um ou mais ambientes de desenvolvimento em AWS Cloud9.

Para visualizar, acessar e gerenciar ambientes de desenvolvimento, abra o AWS Cloud9 console em <https://console.aws.amazon.com/cloud9/>.

Se seu projeto usa recursos externos AWS (por exemplo, um GitHub repositório ou problemas no Atlassian JIRA), esses recursos não são excluídos, mesmo que a caixa Excluir AWS recursos associados junto com o CodeStar projeto esteja selecionada.

## Falha no gerenciamento da equipe: um usuário do IAM não pôde ser adicionado a uma equipe em um AWS CodeStar projeto

Problema: quando você tenta adicionar um usuário a um projeto, você vê uma mensagem de erro informando que houve falha na adição.

Correções possíveis: o motivo mais comum para esse erro é que o usuário do atingiu o limite de políticas gerenciadas que podem ser aplicadas a um usuário no IAM. Você também pode receber esse erro se não tiver o papel de proprietário no AWS CodeStar projeto em que tentou adicionar o usuário ou se o usuário do IAM não existir ou tiver sido excluído.

Verifique se você está conectado como um usuário proprietário desse AWS CodeStar projeto. Para ter mais informações, consulte [Adicionar membros da equipe a um projeto do AWS CodeStar](#).

Para solucionar outros problemas, abra o console do IAM, escolha o usuário que você tentou adicionar e verifique quantas políticas gerenciadas estão aplicadas ao usuário do IAM.

Para obter mais informações, consulte [Limitações em entidades e objetos do IAM](#). Para obter os limites que podem ser alterados, consulte [Service Limits da AWS](#).

## Falha de acesso: um usuário federado não pode acessar um projeto AWS CodeStar

Problema: um usuário federado não consegue ver projetos no AWS CodeStar console.

Correções possíveis: se você estiver conectado como um usuário federado, verifique se tem a política gerenciada apropriada anexada à função que você assumir para fazer login. Para ter mais informações, consulte [Anexar a política gerenciada de visualizador/colaborador/proprietário do AWS CodeStar à função do usuário federado](#).

Adicione usuários federados ao seu AWS Cloud9 ambiente anexando políticas manualmente. Consulte [Anexar uma política gerenciada AWS Cloud9 à função do usuário federado](#).

## Falha de acesso: um usuário federado não pode acessar ou criar um ambiente AWS Cloud9

Problema: um usuário federado não consegue ver ou criar um AWS Cloud9 ambiente no AWS Cloud9 console.

Correções possíveis: se você estiver conectado como um usuário federado, verifique se tem a política gerenciada apropriada anexada à função do usuário federado.

Você adiciona usuários federados ao seu AWS Cloud9 ambiente anexando manualmente as políticas à função do usuário federado. Consulte [Anexar uma política gerenciada AWS Cloud9 à função do usuário federado](#).

## Falha de acesso: um usuário federado pode criar um AWS CodeStar projeto, mas não pode visualizar os recursos do projeto

Problema: um usuário federado foi capaz de criar um projeto, mas não pode visualizar recursos do projeto, como o pipeline do projeto.

Possíveis correções: Se você anexou a política **AWSCodeStarFullAccess** gerenciada, você tem permissões para criar um projeto em AWS CodeStar. No entanto, para acessar todos os recursos do projeto, você deve anexar a política gerenciada do proprietário.

Depois de AWS CodeStar criar os recursos do projeto, as permissões do projeto para todos os recursos do projeto ficam disponíveis nas políticas gerenciadas pelo proprietário, colaborador e visualizador. Para acessar todos os recursos, você deve anexar manualmente a política de proprietário à função. Consulte [Etapa 3: configurar as permissões do IAM do usuário](#).

## Problema de função de serviço: a função de serviço não pôde ser criada

**Problema:** Ao tentar criar um projeto no AWS CodeStar, você vê uma mensagem solicitando que você crie a função de serviço. Quando você escolhe a opção para criá-la, você vê um erro.

**Possíveis correções:** o motivo mais comum para esse erro é que você está conectado AWS com uma conta que não tem permissões suficientes para criar a função de serviço. Para criar a função de AWS CodeStar serviço (`aws-codestar-service-role`), você deve estar conectado como usuário administrativo ou com uma conta raiz. Faça logout do console e faça login com um usuário do IAM que tenha a política gerenciada `AdministratorAccess` aplicada.

## Problema de função de serviço: a função de serviço não é válida ou está ausente

**Problema:** Ao abrir o AWS CodeStar console, você vê uma mensagem informando que a função AWS CodeStar de serviço está ausente ou não é válida.

**Correções possíveis:** o motivo mais comum para esse erro é que um usuário administrativo editou ou excluiu a função de serviço (`aws-codestar-service-role`). Se a função de serviço foi excluída, você é solicitado a criá-la. Você deve estar conectado como um usuário administrativo ou com uma conta root para criar a função. Se tiver sido editada, a função não será mais válida. Faça login no console do IAM como um usuário administrativo, localize o perfil de serviço na lista de funções e exclua-a. Mude para o AWS CodeStar console e siga as instruções para criar a função de serviço.

## Problema de função do projeto: as verificações do status de AWS Elastic Beanstalk saúde falham em instâncias em um AWS CodeStar projeto

**Problema:** Se você criou um AWS CodeStar projeto que inclui o Elastic Beanstalk antes de 22 de setembro de 2017, as verificações do status de saúde do Elastic Beanstalk podem falhar. Se



Se você não tiver alterado a configuração do Elastic Beanstalk desde que criou o projeto, a verificação de status de integridade falhará e informará um estado cinza. Apesar da falha na verificação de integridade, o aplicativo ainda deve ser executado conforme o esperado. Se você não tiver alterado a configuração do Elastic Beanstalk desde que criou o projeto, a verificação de status de integridade falhará e o aplicativo não executará corretamente.

Correção: um ou mais perfis do IAM estão ausentes nas declarações de política do IAM obrigatórias. Adicione as políticas para as funções afetadas ausentes na sua conta da AWS .

1. Faça login AWS Management Console e abra o console do IAM em <https://console.aws.amazon.com/iam/>.  
(Se você não conseguir fazer isso, consulte o administrador AWS da conta para obter ajuda.)
2. No painel de navegação, escolha Perfis.
3. Na lista de funções, escolha CodeStarWorker- ID do **projeto -EB, em que ID do projeto é a ID de** um dos projetos afetados. (Se você não conseguir encontrar facilmente uma função na lista, insira todo ou parte do nome da função na caixa Search (Pesquisa).)
4. Na guia Permissions, escolha Attach Policy.
5. Na lista de políticas, selecione AWSElasticBeanstalkEnhancedHealthAWSElasticBeanstalkService. (Se você não conseguir encontrar facilmente uma política na lista, insira todo ou parte do nome da política na caixa de Pesquisa.)
6. Escolha Attach Policy.
7. Repita as etapas de 3 a 6 para cada função afetada que tenha um nome seguindo o padrão CodeStarWorker- **ID do projeto -EB**.

## Problema de função de projeto: uma função de projeto não é válida ou está ausente

Problema: quando você tenta adicionar um usuário a um projeto, você vê uma mensagem de erro informando que houve falha na adição porque a política de uma função de projeto está ausente ou não é válida.

Correções possíveis: o motivo mais comum para esse erro é que uma ou mais políticas de projeto foram editadas ou excluídas do IAM. As políticas do projeto são exclusivas AWS CodeStar dos projetos e não podem ser recriadas. O projeto não pode ser usado. Crie um projeto em e AWS

CodeStar, em seguida, migre os dados para o novo projeto. Clone o código do projeto no repositório do projeto não utilizável e envie esse código por push para o novo repositório do projeto. Copie as informações de wiki da equipe do projeto antigo para o novo projeto. Adicione usuários ao novo projeto. Quando tiver certeza de que migrou todos os dados e configurações, exclua o projeto inutilizável.

## Extensões de projetos: não é possível se conectar ao JIRA

**Problema:** Quando você usa a extensão Atlassian JIRA para tentar conectar um AWS CodeStar projeto a uma instância do JIRA, a seguinte mensagem é exibida: "A URL não é uma URL válida do JIRA. Verificar se a URL está correta."

Correções possíveis:

- Verifique se a URL do JIRA está correta e, em seguida, tente a conexão novamente.
- Sua instância auto-hospedada do JIRA pode não estar acessível na internet pública. Entre em contato com o administrador da rede para garantir que sua instância do JIRA possa ser acessada na internet pública e, em seguida, tente a conexão novamente.

## GitHub: Não consigo acessar o histórico de commits, problemas ou código de um repositório

**Problema:** No painel de um projeto que armazena seu código GitHub, os blocos Histórico de confirmações e GitHubProblemas exibem um erro de conexão, ou escolher Abrir em GitHub ou Criar problema nesses blocos exibe um erro.

Causas possíveis:

- Talvez o AWS CodeStar projeto não tenha mais acesso ao GitHub repositório.
- O repositório pode ter sido excluído ou renomeado em GitHub

# AWS CloudFormation: criação da pilha revertida para permissões ausentes

Depois de adicionar um recurso ao arquivo `template.yml`, visualize a atualização da pilha AWS CloudFormation para as mensagens de erro. A atualização da pilha falhará se determinados critérios não forem atendidos (por exemplo, quando as permissões de recursos necessários estão ausentes).

## Note

Em 2 de maio de 2019, atualizamos a política de funções do AWS CloudFormation trabalhador para todos os projetos existentes. Essa atualização reduz o escopo das permissões de acesso concedidas ao seu pipeline do projeto para maior segurança em seus projetos.

Para solucionar problemas, visualize o status da falha na visualização do AWS CodeStar painel do pipeline do seu projeto.

Em seguida, escolha o CloudFormationlink no estágio de implantação do seu pipeline para solucionar a falha no AWS CloudFormation console. Para visualizar os detalhes da criação da pilha, expanda a lista Events (Eventos) do projeto e visualize as mensagens de falha. A mensagem indica que a permissão está ausente. Corrija a política de função de operador do AWS CloudFormation e execute o pipeline novamente.

## AWS CloudFormation não está autorizado a realizar iam: PassRole na função de execução do Lambda

Se você tiver um projeto criado antes do PDT de 6 de dezembro de 2018 que cria funções Lambda, talvez veja AWS CloudFormation um erro como este:

```
User: arn:aws:sts::id:assumed-role/CodeStarWorker-project-id-CloudFormation/  
AWSCloudFormation is not authorized to perform: iam:PassRole on resource:  
arn:aws:iam::id:role/CodeStarWorker-project-id-Lambda (Service: AWSLambdaInternal;  
Status Code: 403; Error Code: AccessDeniedException; Request ID: id)
```

Esse erro ocorre porque sua função de AWS CloudFormation trabalhador não tem permissão para passar uma função para provisionar sua nova função Lambda.

Para corrigir esse erro, você precisará atualizar sua política de função de AWS CloudFormation trabalhador com o seguinte trecho.

```
{
  "Action": [ "iam:PassRole" ],
  "Resource": [
    "arn:aws:iam::account-id:role/CodeStarWorker-project-id-Lambda",
  ],
  "Effect": "Allow"
}
```

Depois de atualizar a política, execute o pipeline novamente.

Como alternativa, é possível usar uma função personalizada para sua função do Lambda adicionando um limite de permissões ao seu projeto, conforme descrito em [Adicionar um limite de permissões do IAM a projetos existentes](#)

## Não é possível criar a conexão para um GitHub repositório

Problema:

Como uma conexão com um GitHub repositório usa o AWS Conector para GitHub, você precisa de permissões de proprietário da organização ou permissões de administrador no repositório para criar a conexão.

Possíveis correções: para obter informações sobre os níveis de permissão de um GitHub repositório, consulte <https://docs.github.com/en/free-pro-team@latest/github/setting-up-and-managing-organization-permissions/organizations-and-teams-permission-levels-for-an>

# Notas de release do Guia do usuário do AWS CodeStar

A tabela a seguir descreve as alterações importantes em cada versão do Guia do usuário do AWS CodeStar. Para receber notificações sobre atualizações dessa documentação, você pode se inscrever em um feed RSS.

Alteração	Descrição	Data
<a href="#">Atualizações de política de acesso</a>	A política de acesso do AWS CodeStar foi atualizada. O resultado da política é o mesmo, mas o cloudformation requer ListStacks além de DescribeStacks, o que já é necessário. Para fazer referência à política atualizada, consulte a <a href="#">Política do AWSCodeStarfullAccess</a> .	24 de março de 2023
<a href="#">Atualizações da política de perfil de serviço</a>	A política de perfil de serviço do AWS CodeStar foi atualizada. Para fazer referência à política atualizada, consulte a <a href="#">Política do AWSCodeStarServiceRole</a> .	23 de setembro de 2021
<a href="#">Use um recurso de conexão para projetos com um repositório de origem do GitHub</a>	Quando você usa o console para criar um projeto no AWS CodeStar com um repositório do GitHub, um recurso de conexão é usado para gerenciar suas ações do GitHub. As conexões usam o GitHub Apps, enquanto a autorização anterior do GitHub usava o OAuth. Para um tutorial que mostra como	27 de abril de 2021

criar um projeto que usa uma conexão com o GitHub, consulte [Tutorial: Criar um projeto com um repositório de origem do GitHub](#). O tutorial também mostra como criar, revisar e mesclar uma pull request para o repositório de origem do seu projeto.

[O AWS CodeStar oferece suporte ao AWS Cloud9 na região Oeste dos EUA \(Norte da Califórnia\)](#)

O AWS CodeStar agora oferece suporte ao AWS Cloud9 na região Oeste dos EUA (Norte da Califórnia). Para obter mais informações, consulte [Configurando o Cloud9](#).

16 de fevereiro de 2021

[Atualize a documentação para refletir a nova experiência do console](#)

Em 12 de agosto de 2020, o serviço do AWS CodeStar mudou para uma nova experiência de usuário no console do AWS. O guia do usuário foi atualizado para corresponder à nova experiência do console.

12 de agosto de 2020

[Os projetos do AWS CodeStar podem ser criados com a CLI do AWS CodeStar](#)

Os projetos do AWS CodeStar podem ser criados com o comando da CLI. O AWS CodeStar cria o projeto e a infraestrutura usando o código-fonte e o modelo da cadeia de ferramentas fornecido. Consulte [Criar um projeto no AWS CodeStar \(AWS CLI\)](#).

24 de outubro de 2018

[Todos os modelos de projeto do AWS CodeStar agora incluem arquivos do AWS CloudFormation para atualizações feitas na infraestrutura](#)

O AWS CodeStar funciona com o AWS CloudFormation para permitir usar código para criar serviços de suporte e servidores ou plataformas sem servidor na nuvem. O arquivo do AWS CloudFormation agora está disponível para todos os tipos de modelo do projeto do (modelos com a plataforma de computação Lambda, EC2 ou Elastic Beanstalk). O arquivo é armazenado em `template.yml` no repositório de origem. Você pode visualizar e modificar o arquivo para adicionar recursos ao projeto. Consulte [Modelos de projetos](#).

3 de agosto de 2018

[As notificações de atualização do Manual do usuário do AWS CodeStar já estão disponíveis por meio de RSS](#)

A versão HTML do Guia do usuário do AWS CodeStar agora oferece suporte a um feed RSS das atualizações que estão documentadas na página Notas de release de atualizações da documentação. O feed RSS inclui as atualizações feitas após 30 de junho de 2018. As atualizações anunciadas anteriormente ainda estão disponíveis na página Notas de release de atualizações da documentação. Use o botão RSS no menu superior do painel para assinar o feed.

30 de junho de 2018

A tabela a seguir descreve as alterações importantes em cada versão do Guia do usuário do AWS CodeStar antes de 30 de junho de 2018.

Alteração	Descrição	Alterado em
O Guia do usuário do AWS CodeStar agora está disponível no GitHub	Este guia agora está disponível no GitHub. Você também pode usar o GitHub para enviar comentários e solicitações de alteração do conteúdo desse guia. Para obter mais informações, selecione o ícone Edit on GitHub (Editar no GitHub) na barra de navegação do guia ou consulte o repositório <a href="https://github.com/awsdocs/aws-codestar-user-guide">awsdocs/aws-codestar-user-guide</a> no site do GitHub.	22 de fevereiro de 2018
Agora, o AWS CodeStar está disponível na Ásia-Pacífico (Seul)	Agora o AWS CodeStar está disponível na região da Ásia-Pacífico (Seul). Para ter mais informações, consulte <a href="#">AWS CodeStar</a> no Referência geral da Amazon Web Services.	14 de fevereiro de 2018



Alteração	Descrição	Alterado em
O AWS CodeStar agora está disponível na Ásia-Pacífico (Tóquio) e Canadá (Central)	O AWS CodeStar agora está disponível nas regiões da Ásia-Pacífico (Tóquio) e do Canadá (Central) Para ter mais informações, consulte <a href="#">AWS CodeStar</a> no Referência geral da Amazon Web Services.	20 de dezembro de 2017
O AWS CodeStar agora oferece suporte para AWS Cloud9	<p>Agora o AWS CodeStar oferece suporte ao uso do AWS Cloud9, um navegador da web, com base em IDE online, para trabalhar com o código do projeto. Para obter mais informações, consulte <a href="#">Usar o AWS Cloud9 com a AWS CodeStar</a>.</p> <p>Para ver uma lista das regiões do AWS compatíveis, consulte <a href="#">AWS Cloud9</a> no Referência geral da Amazon Web Services.</p>	30 de novembro de 2017
O AWS CodeStar agora oferece suporte ao GitHub	O AWS CodeStar agora oferece suporte ao armazenamento de código do projeto no GitHub. Para obter mais informações, acesse <a href="#">Criar um projeto</a> .	12 de outubro de 2017
O AWS CodeStar agora está disponível no Oeste dos EUA (Norte da Califórnia) e Europa (Londres)	O AWS CodeStar agora está disponível nas regiões do Oeste dos EUA (Norte da Califórnia) e da Europa (Londres) Para ter mais informações, consulte <a href="#">AWS CodeStar</a> no Referência geral da Amazon Web Services.	17 de agosto de 2017
O AWS CodeStar agora está disponível nas regiões Ásia-Pacífico (Singapura), Ásia-Pacífico (Sydney) e Europa (Frankfurt).	O AWS CodeStar agora está disponível nas regiões Ásia-Pacífico (Sydney) e Ásia-Pacífico (Singapura) e Europa (Frankfurt). Para ter mais informações, consulte <a href="#">AWS CodeStar</a> no Referência geral da Amazon Web Services.	25 de julho de 2017

Alteração	Descrição	Alterado em
O AWS CloudTrail agora oferece suporte para AWS CodeStar	O AWS CodeStar já foi integrado ao CloudTrail, um serviço que captura chamadas de API feitas pelo ou em nome do AWS CodeStar na sua conta do AWS e oferece os arquivos de log para um bucket do Amazon S3 especificado por você. Para obter mais informações, consulte <a href="#">Log de chamadas de API do AWS CodeStar com o AWS CloudTrail</a> .	14 de junho de 2017
Versão inicial	Esta é a primeira versão do Guia do usuário do AWS CodeStar.	19 de abril de 2017

# Glossário do AWS

Para obter a terminologia mais recente da AWS, consulte o [glossário da AWS](#) na Referência do Glossário da AWS.