Connect Azure Boards to GitHub

03/04/2019 • 8 minutos para ler • Colaboradores

Neste artigo

Azure Boards | Azure DevOps Server 2019

By connecting your Azure Boards project with GitHub.com repositories, you support linking between GitHub commits and pull requests to work items. You can use GitHub for software development while using Azure Boards to plan and track your work.

> **Observação**
>
> We recommend that you use the Azure Boards app for GitHub to configure and manage your connections to GitHub.com. The app provides a more streamlined configuration experience and has the advantage of authenticating and operating as the app rather than an individual. Once you have configured your connection, you can manage the connected repositories either from Azure Boards or GitHub.com.

When you make the connection from GitHub.com, the list of repositories correspond to ones that you allow the Azure Boards app to access. When you make the connection from Azure Boards, the list of repositories correspond to some or all of the repos from the first selection that you make to associate to a particular project. In the first instance, you can limit what the app can access overall, and limit what a particular project can access or split the management of work across different Azure Boards projects.

> **Observação**
>
> Azure Boards and Azure DevOps Services support integration with GitHub.com and GitHub Enterprise Server repositories.
>
> Azure DevOps Server 2019 supports integration with GitHub Enterprise Server repositories.

Prerequisites

- You must connect to an Azure Boards or Azure DevOps project. If you don't have a project yet, create one.
- You must be a member of the Project Administrators group and the project's Contributors group. If you

created the project, then you have permissions.
- You must be an administrator or owner of the GitHub repository you'll be connecting to.

> **Importante**
>
> You can connect to multiple GitHub repositories so long as you are an administrator for those repositories.

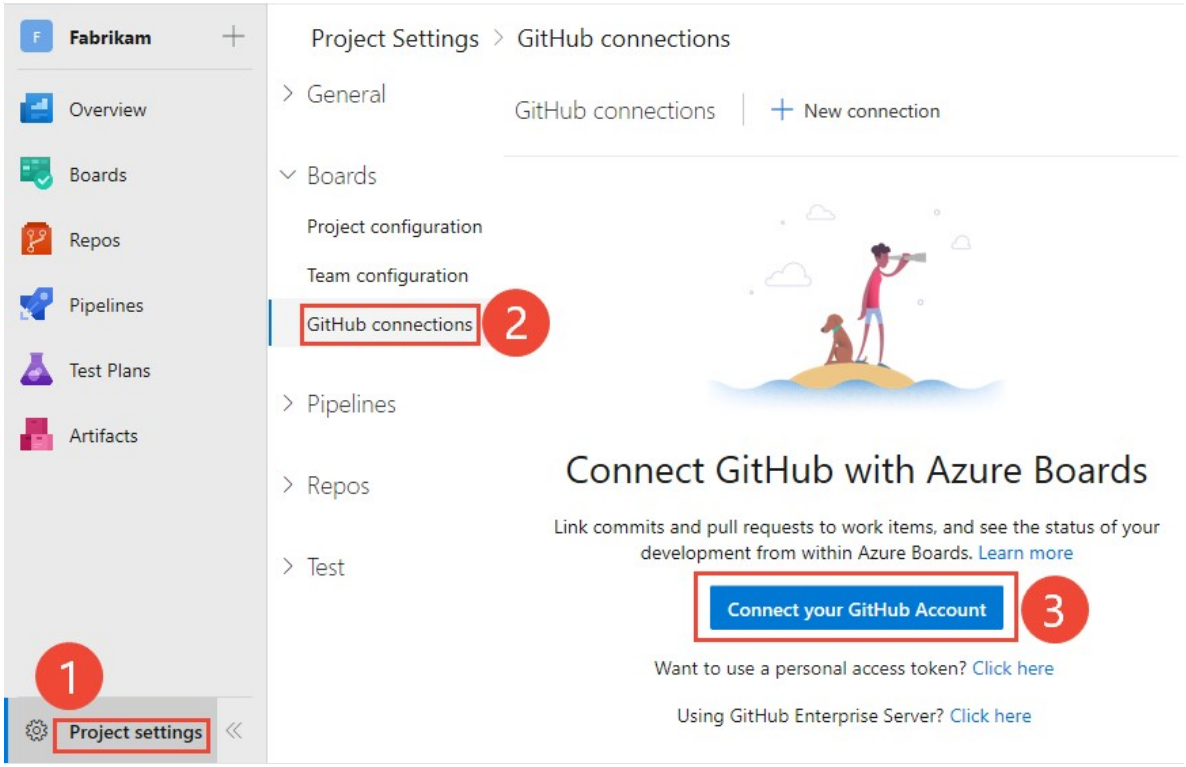
Authentication options

Depending on the platform you work from to connect to GitHub, you have a choice of the credentials you use. The recommended authentication method in all instances is **OAuth**.

Platform	GitHub.com	GitHub Enterprise Server
Azure DevOps Services	• OAuth (preferred, no registration steps) • Personal access token (PAT)	• OAuth (preferred, registration required) • PAT • Username plus password
Azure DevOps Server	• Not supported	• OAuth (preferred, registration required) • PAT • Username plus password

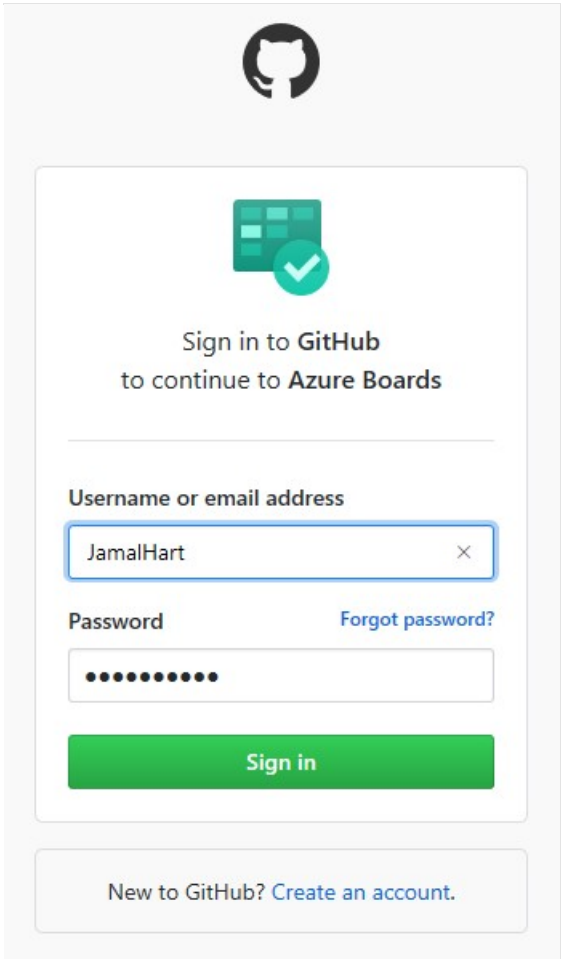
Add a GitHub connection using OAuth

1. Sign into Azure Boards.
2. Choose (1) **Project Settings**, choose (2) **GitHub connections** and then (3) **Connect your GitHub Account**.



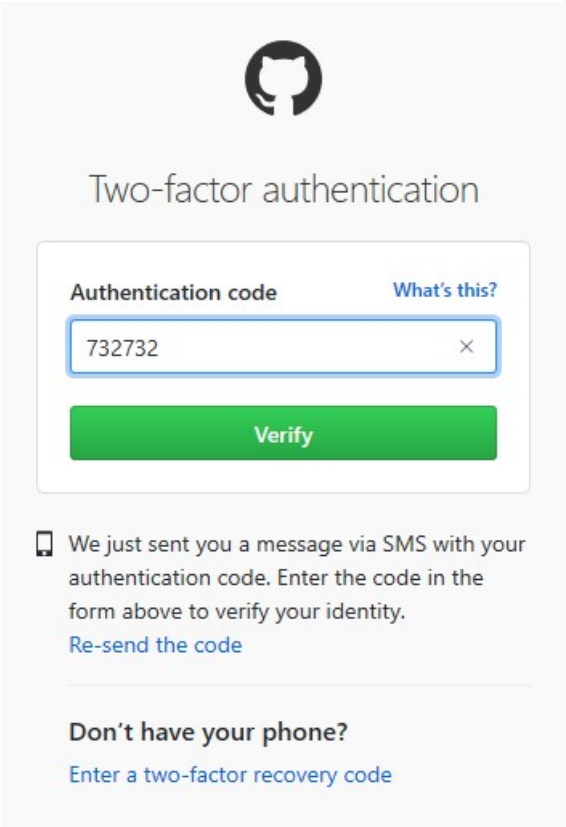
If connecting using PAT, see Add a GitHub connection using PAT. If connecting to a GitHub Enterprise Server, see Register Azure DevOps in GitHub as an OAuth App.

3. Enter your GitHub account credentials. Choose an account for which you are an administrator for the repositories you want to connect to.

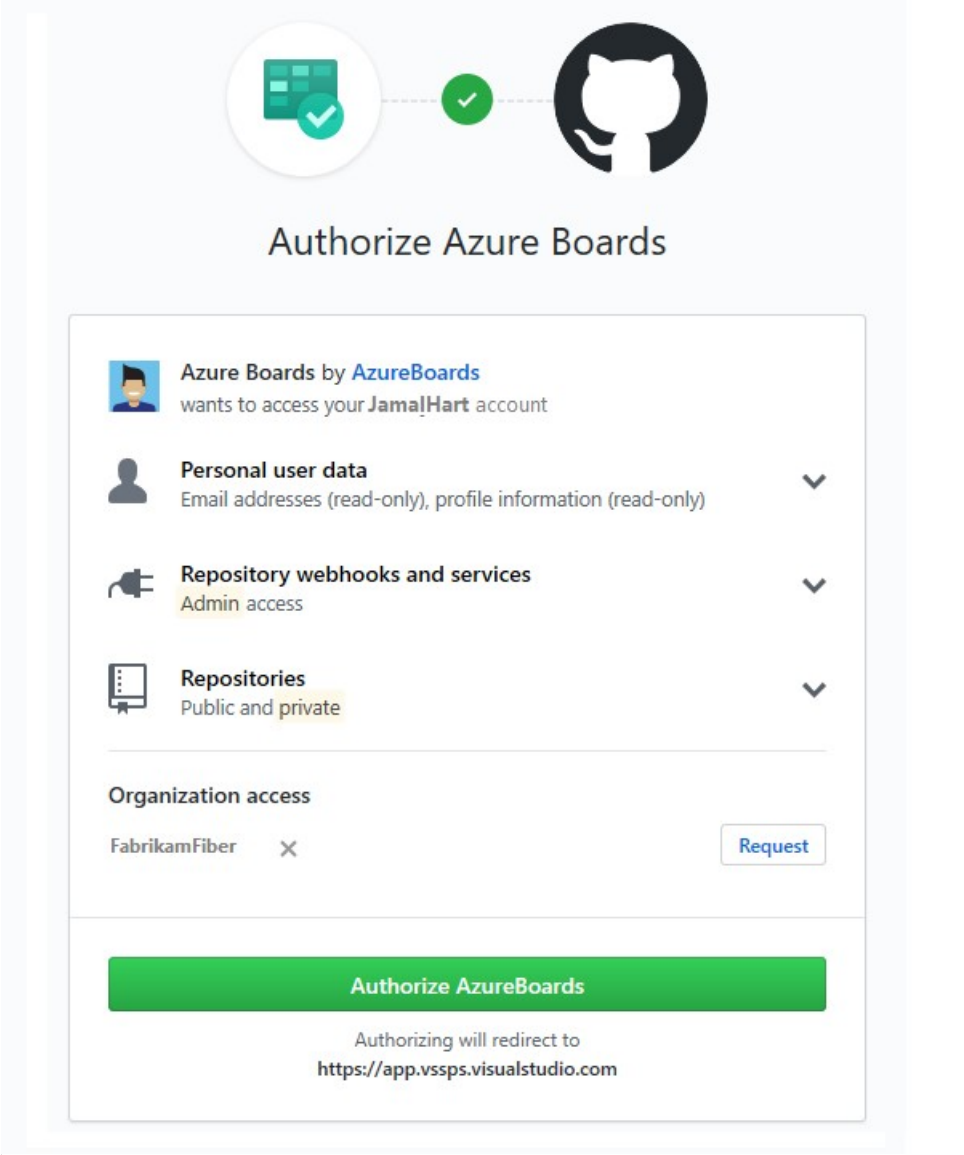


4. If you have enabled two-factor authentication, enter the authentication code that GitHub sent you and

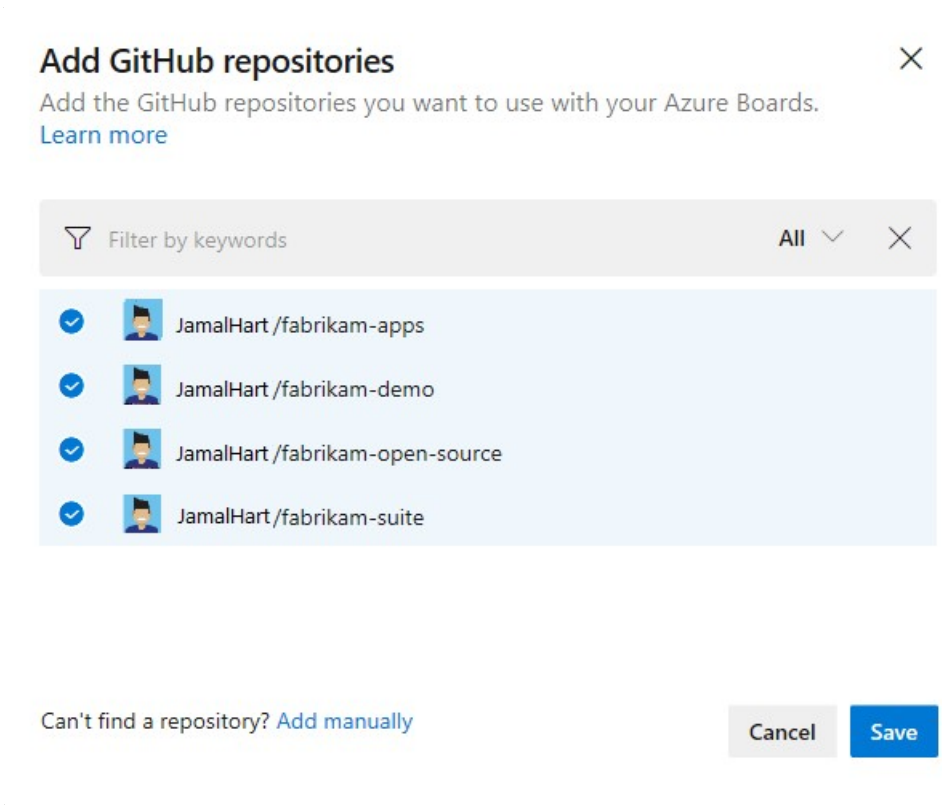
choose **Verify**.



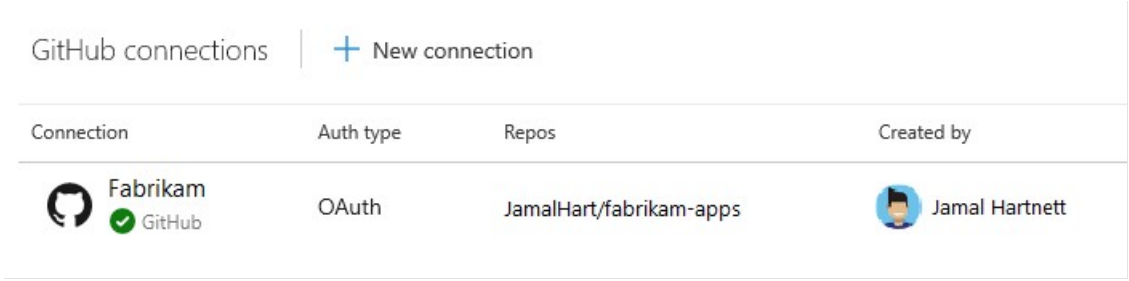
5. Review the GitHub authorization dialog that appears which indicates the information you'll allow Azure Boards to access from GitHub. Choose **Authorize AzureBoards** when ready.



6. In the Add GitHub repositories dialog, you'll see the list of repositories for which you are an administrator.



Check the ones that you want to add and then choose **Save**. When done, you should see the new connection with the selected repository listed.



Dica

We recommend that you only connect a GitHub repo to projects defined in a single Azure DevOps organization. Connecting the same GitHub repo to projects defined in two or more Azure DevOps organizations can lead to unexpected AB# mention linking. For details, see **Troubleshoot GitHub & Azure Boards integration**.

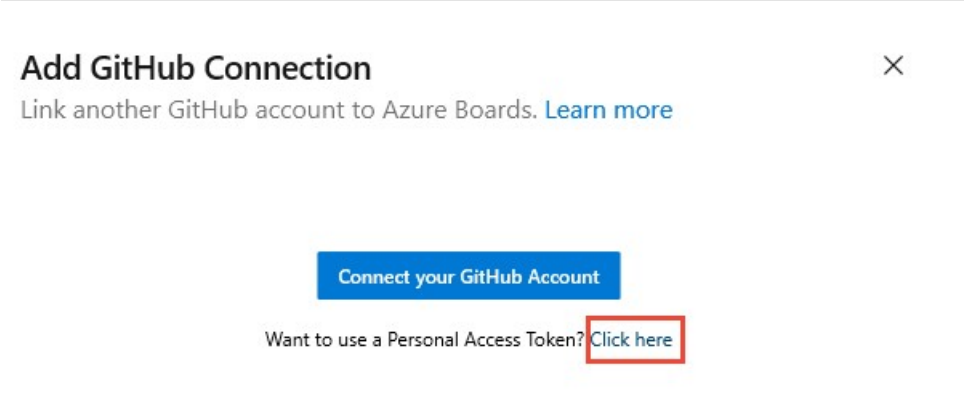
Add a GitHub connection using PAT

We recommend that you use OAuth to connect to your GitHub repository. However, if you need to use a PAT, you can by following these procedures.

Dica

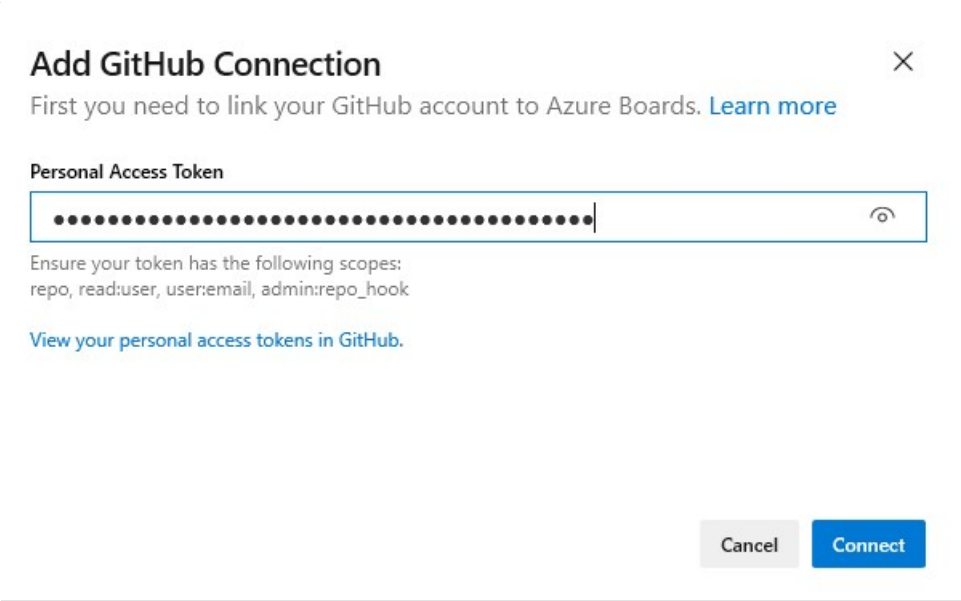
When creating your GitHub PAT, make sure that you include these scopes:
`repo, read:user, user:email, admin:repo_hook` .

1. To choose a PAT when connecting a GitHub repository, choose + **New Connection** and then choose the **Click here** link.



To create a GitHub PAT, go to GitHub Developer Settings>Personal access tokens.

2. Enter the PAT and choose **Connect**.

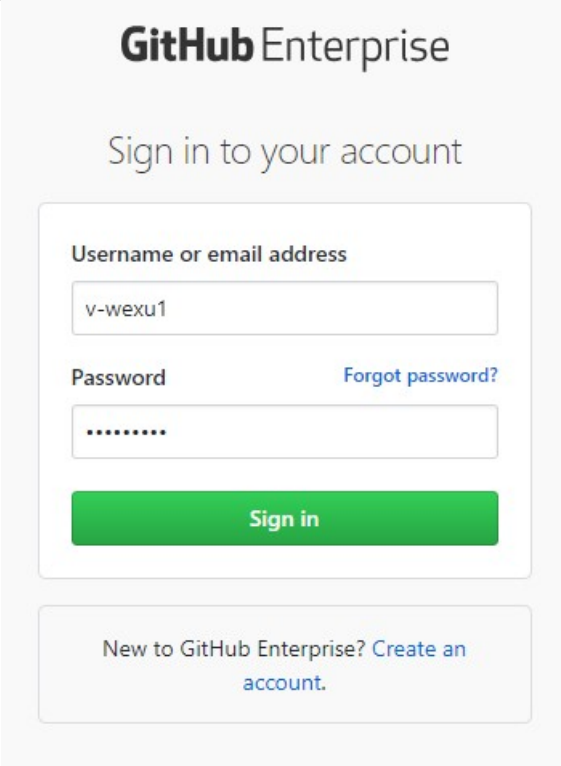


Register Azure DevOps in GitHub as an OAuth App

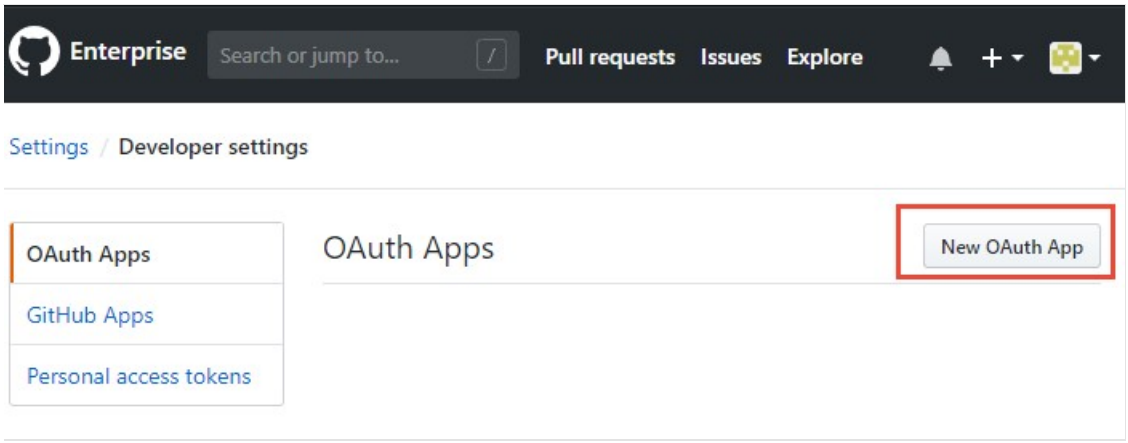
If you plan to use OAuth to connect Azure DevOps Services or Azure DevOps Server with your GitHub Enterprise Server, you first need to register the application as an OAuth App. For details, see Creating an OAuth App.

Register Azure DevOps Services

1. Sign into the web portal for your GitHub Enterprise server.



2. Open **Settings>Developer settings>Oauth Apps** and choose **New OAuth App**.



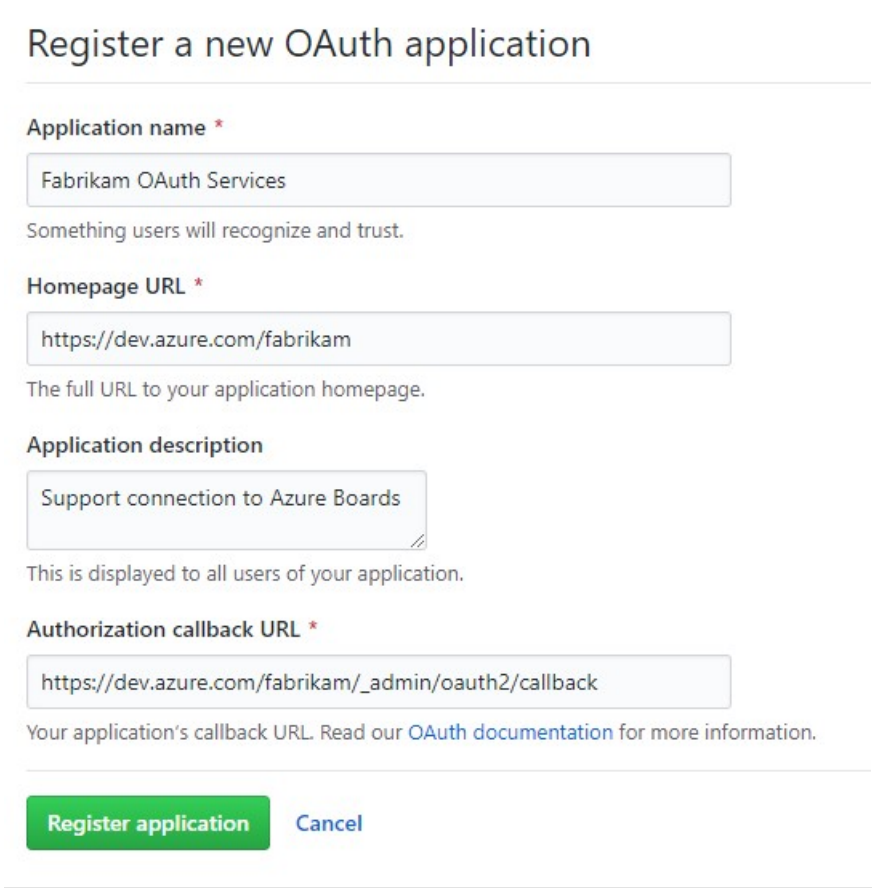
3. Fill out the form to register your Azure DevOps Server application.

   For the **Homepage URL**, specify the **Organization URL** of your organization.
   For the **Authorization callback URL**, use the following pattern to construct the URL.

   ```
   {Azure DevOps Services Organization URL}/_admin/oauth2/callback
   ```

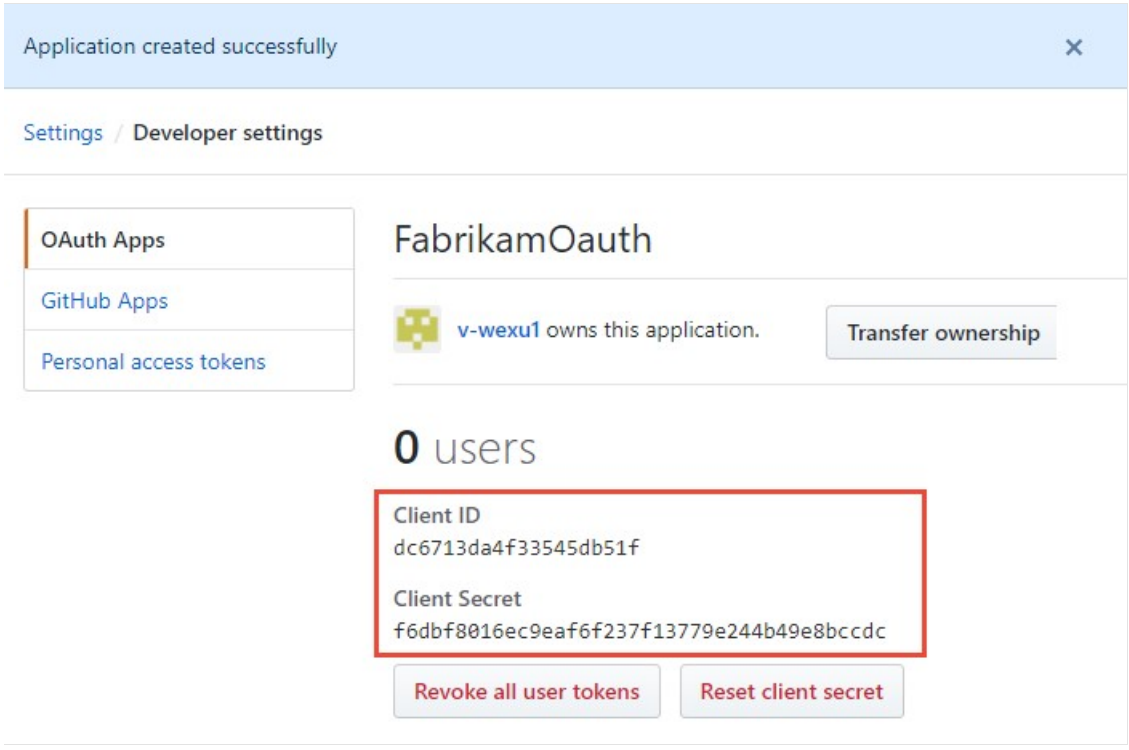
   For example:

   ```
   http://dev.azure.com/fabrikam/_admin/oauth2/callback
   ```



4. Choose **Register application**.

5. Upon success, you'll see a page that provides the **Client ID** and **Client Secret** for your registered OAuth application.

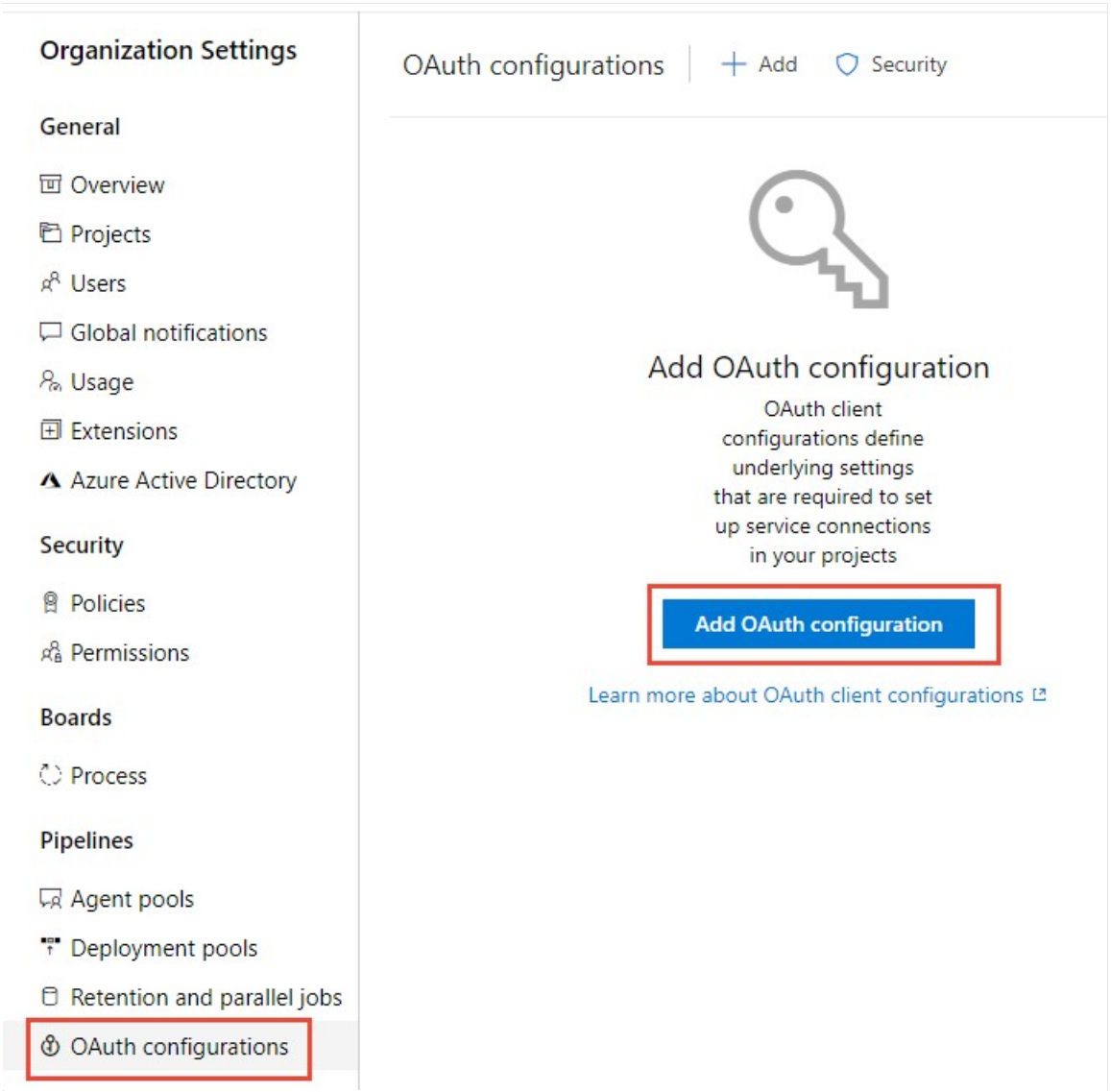


Register your OAuth configuration in Azure DevOps Services

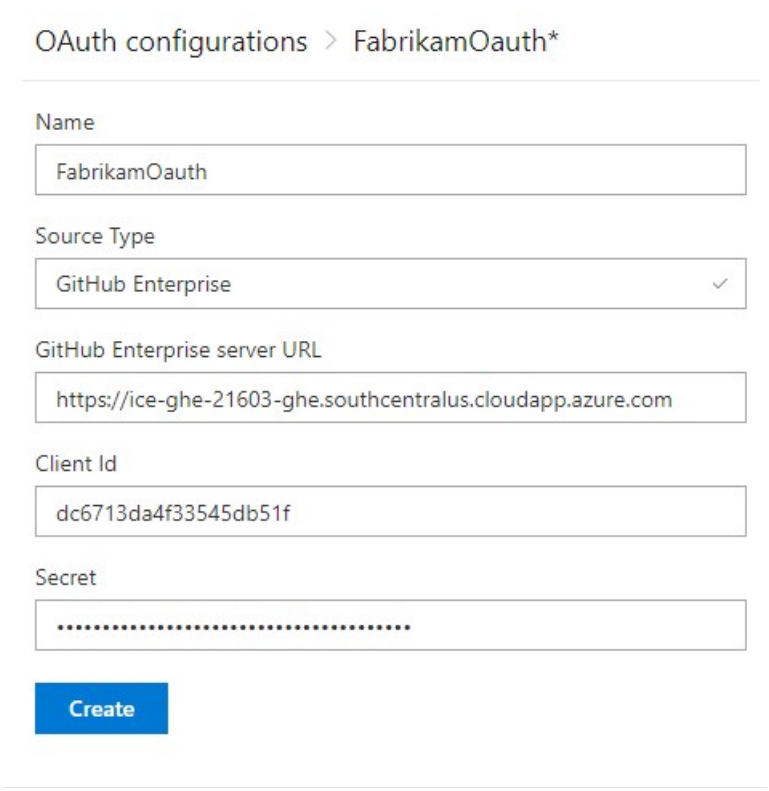
1. Sign into the web portal for Azure DevOps Services.

2. Add the GitHub Enterprise Oauth configuration to your organization.

3. Open **Organization settings>Oauth configurations**, and choose **Add Oauth configuration**.



4. Fill in the form that appears, and then choose **Create**.



Connect Azure DevOps Services to GitHub Enterprise Server

ⓘ Importante

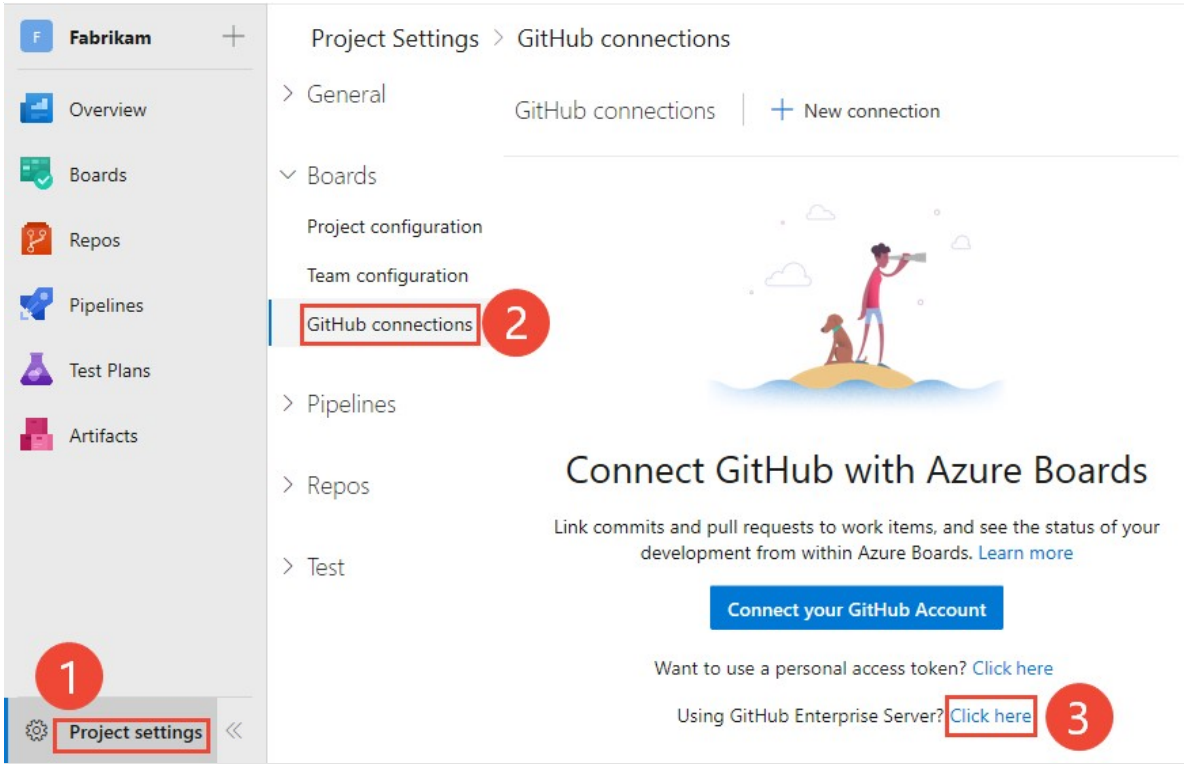
To connect Azure DevOps Services to your GitHub Enterprise Server, your GitHub Enterprise Server must be sufficiently accessible from the Internet. Make sure Azure DNS can resolve your GitHub Enterprise Server name and your firewall allows access from Azure Data Center IP addresses. To determine the IP address range, see **Microsoft Azure Datacenter IP Ranges**. A common error message encountered when connectivity issues exist is:

The remote name could not be resolved: 'github-enterprise-server.contoso.com'

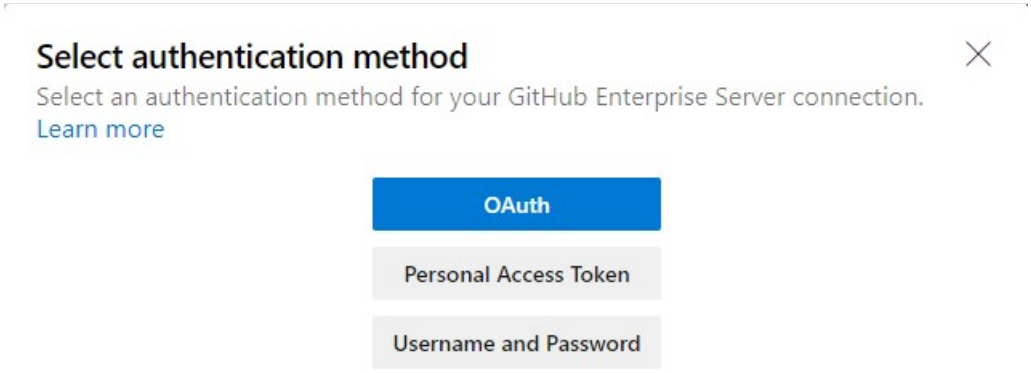
If you encounter this error, check that your server is accessible. For more information, see **Azure DNS FAQ**.

1. Choose the Azure DevOps logo to open **Projects**, and then choose the Azure Boards project you want to configure to connect to your GitHub Enterprise repositories.

2. Choose (1) **Project Settings**, choose (2) **GitHub connections** and then (3) **Click here** to connect to your GitHub Enterprise organization.



Choose from one of the following options—**OAuth**, **Personal Access Token**, **Username and Password**—based on the credentials you've chosen.

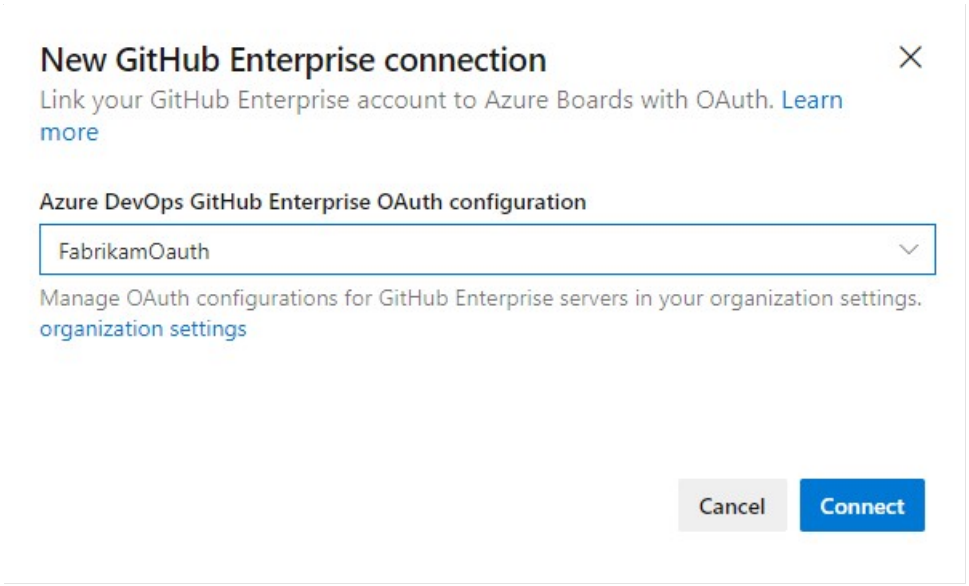


To create a PAT, see Creating a personal access token.

> **Dica**
>
> When creating your GitHub Enterprise Server PAT, make sure that you include these scopes: `repo, admin:repo_hook, read:user, user:email`.

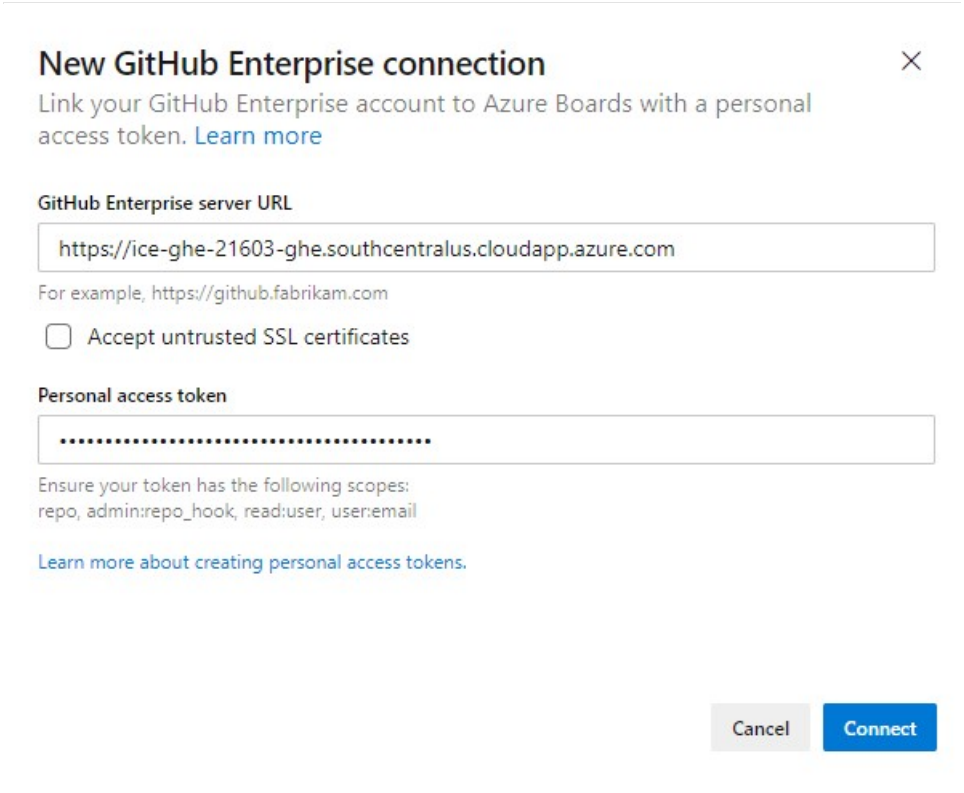
Connect using OAuth

Choose the configuration that you set up in Step 4 of Register your OAuth configuration in Azure DevOps Services, and then choose **Connect**.



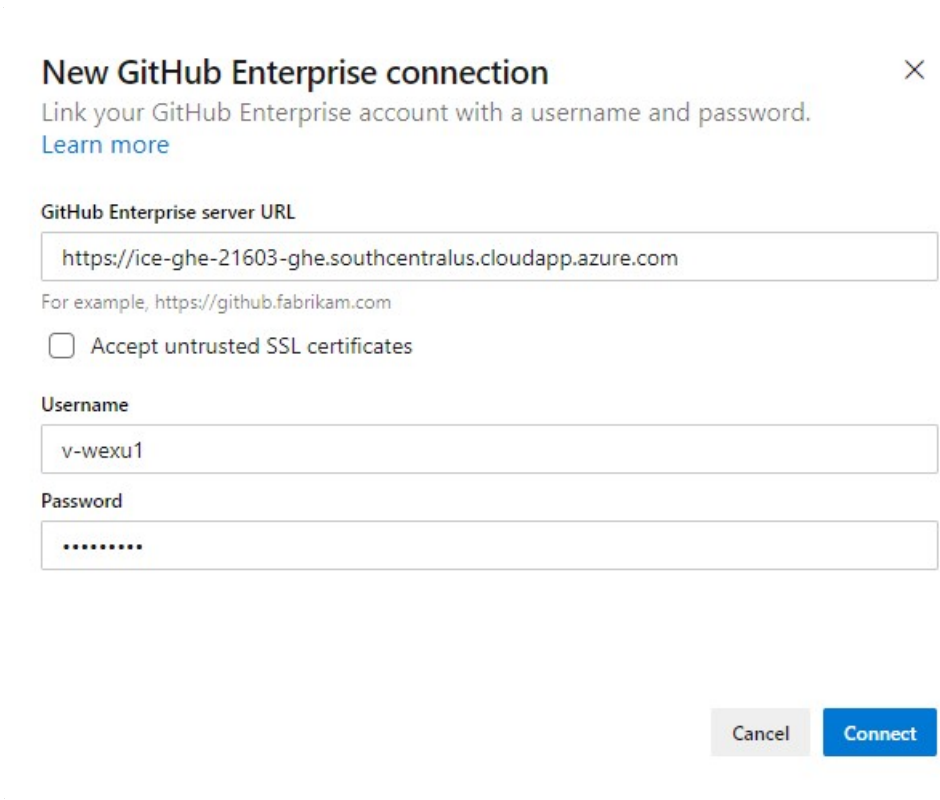
Connect using a Personal Access Token

Enter the URL for your GitHub Enterprise server and the **Personal access token** credentials recognized by that server. And then choose **Connect**.

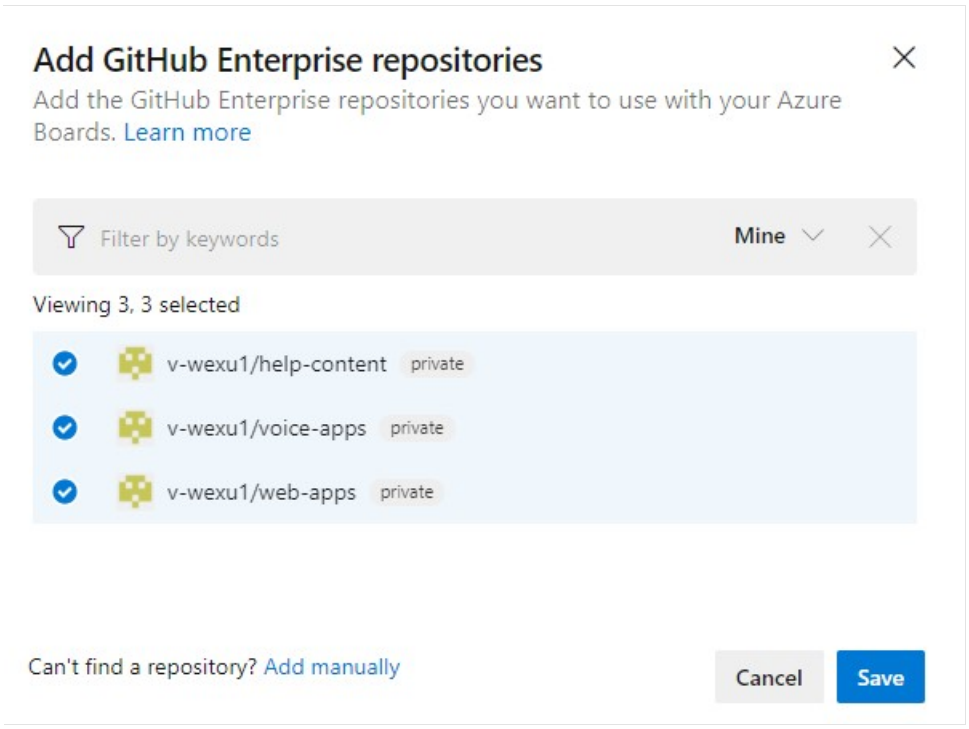


Connect using a Username and Password

Enter the URL for your GitHub Enterprise server and the administrator account credentials recognized by that server. And then choose **Connect**.

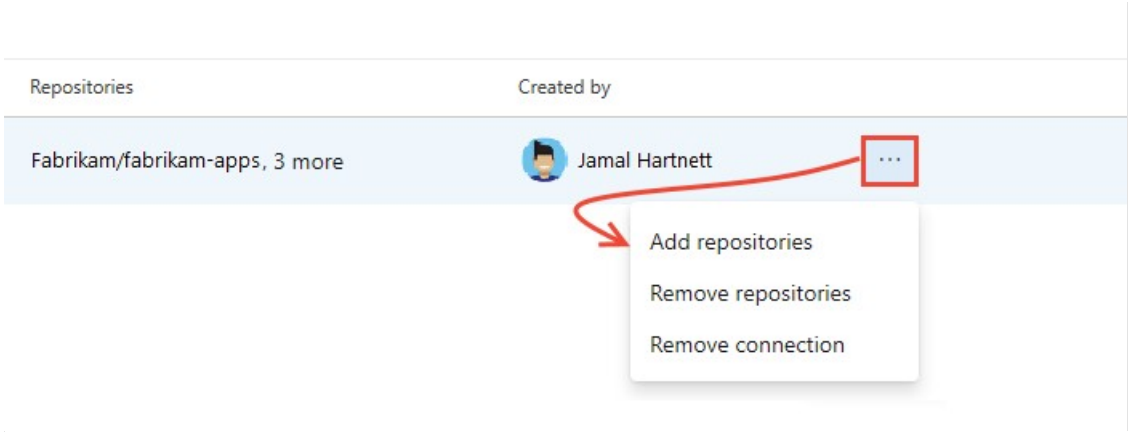


3. The dialog lists all repositories for which you have GitHub administration rights. You can toggle between **Mine** and **All** to determine if others appear, and then check the ones that you want to add. Choose **Save** when done.



Add or remove repositories, or remove a connection

1. To add or remove repositories, open the ··· actions icon for the connection and choose **Add** repositories or **Remove** repositories from the menu.



2. To remove all repositories and the connection, choose the **Remove connection** option. Then, choose **Remove** to confirm.

Resolve connection issues

See Troubleshoot GitHub repository connection.

Try this next

Link GitHub commits and pull requests to work items

Related articles

- What is Azure Boards?
- Install and configure the Azure Boards app for GitHub
- Configure status badges to add to GitHub README files
- Troubleshoot GitHub & Azure Boards integration