Azure Boards Integration with GitHub

**Estimated Time: 20 Minutes**

**Overview:**

**In today’s DevOps world, it is important that you track your work, write your code, and be able to link your code changes back to the reason why you made the change. With the integration between Azure Boards and GitHub, you can do just that. You can link both commits and pull requests in GitHub back to Azure Board work items, such as tasks.**

**In this lab, you will:**

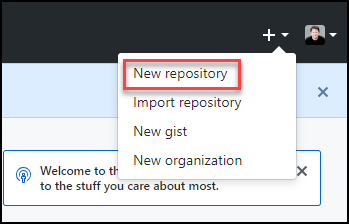
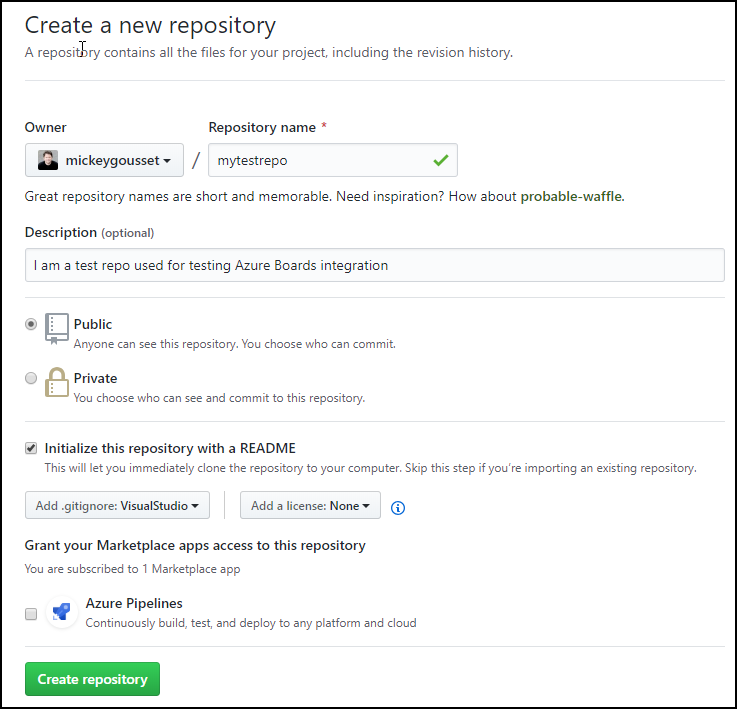
* **Create a new GitHub repo**
* **Create a new Azure DevOps team project**
* **Connect your team project to your GitHub repo**
* **Commit a code change to the GitHub repo, and link the commit to an Azure Boards work item**

|  |
| --- |
| **NOTE: At the time this lab was created, Azure Boards and GitHub integration is in Public Preview and supported only for Azure DevOps Services at this time.** |

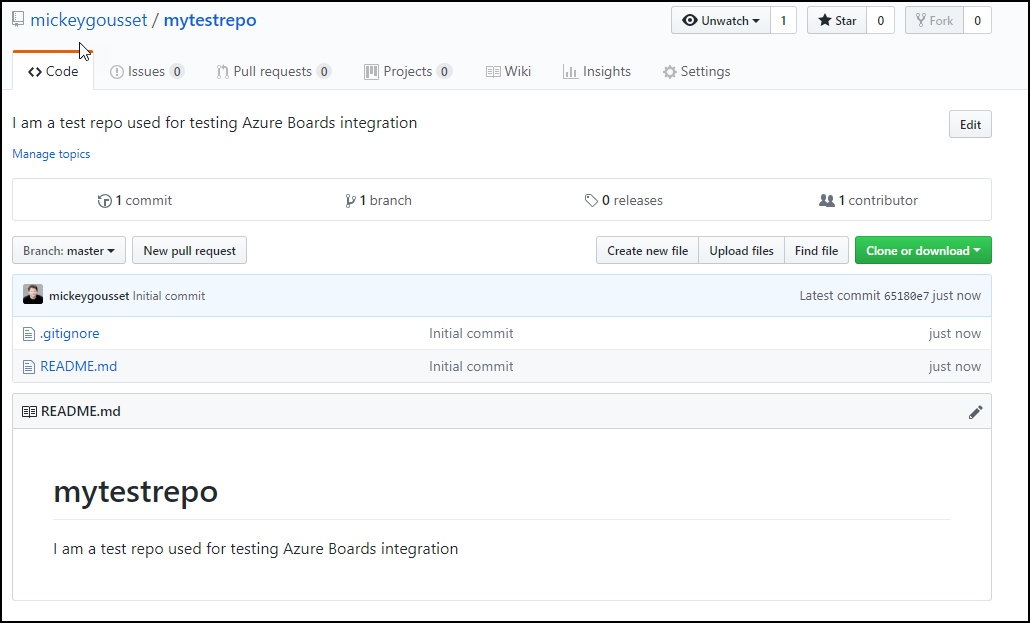
# **Exercise 1: Create a GitHub Repo**

**Estimated Time:** 5 Minutes

Azure Boards integration will work with existing GitHub repos, but for the purposes of this lab, you are going to create a new one.

1. **Open a web browser and navigate to** <https://github.com>**. Log into GitHub if you are not already logged in.**
2. **In the upper-right corner, click the “+”, and select New repository.**
3. **Enter the following information:  
   Repository Name: mytestrepo  
   Description: I am a test repo used for testing Azure Boards integration  
   Select Public  
   Check the box to Initialize this repository with a README  
   Select VisualStudio in the Add .gitignore drop down box  
     
   Click the Create repository button to create the new repo.**

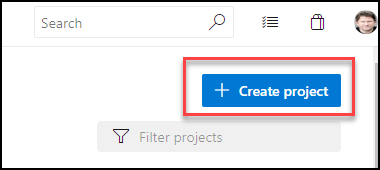
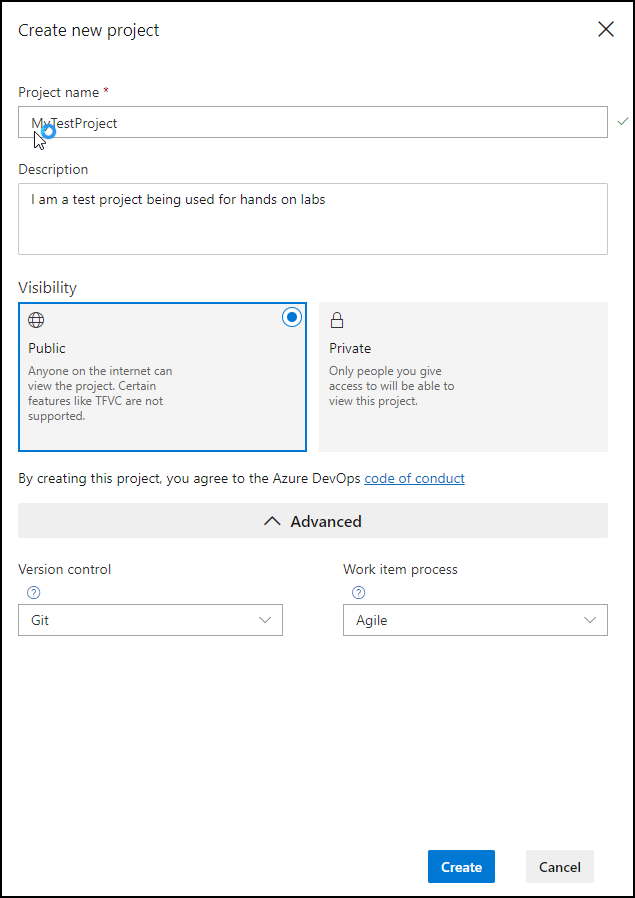
**At this point, you have a new repository in GitHub to use with this lab.**



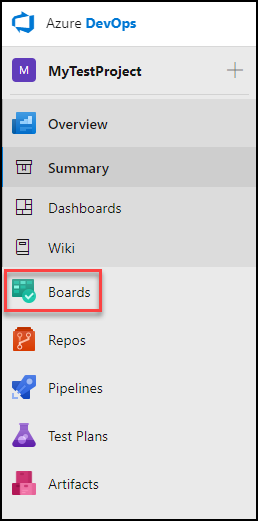
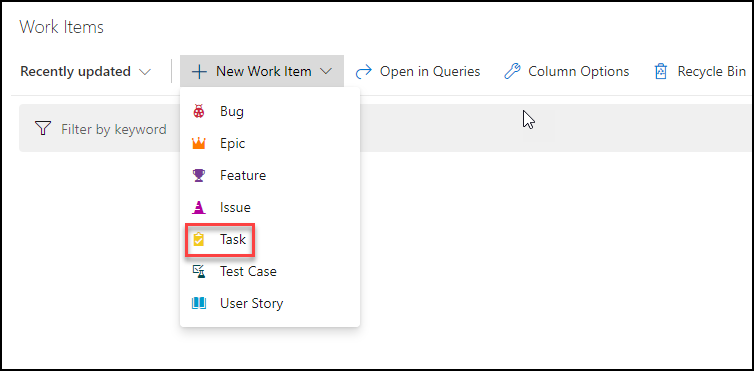
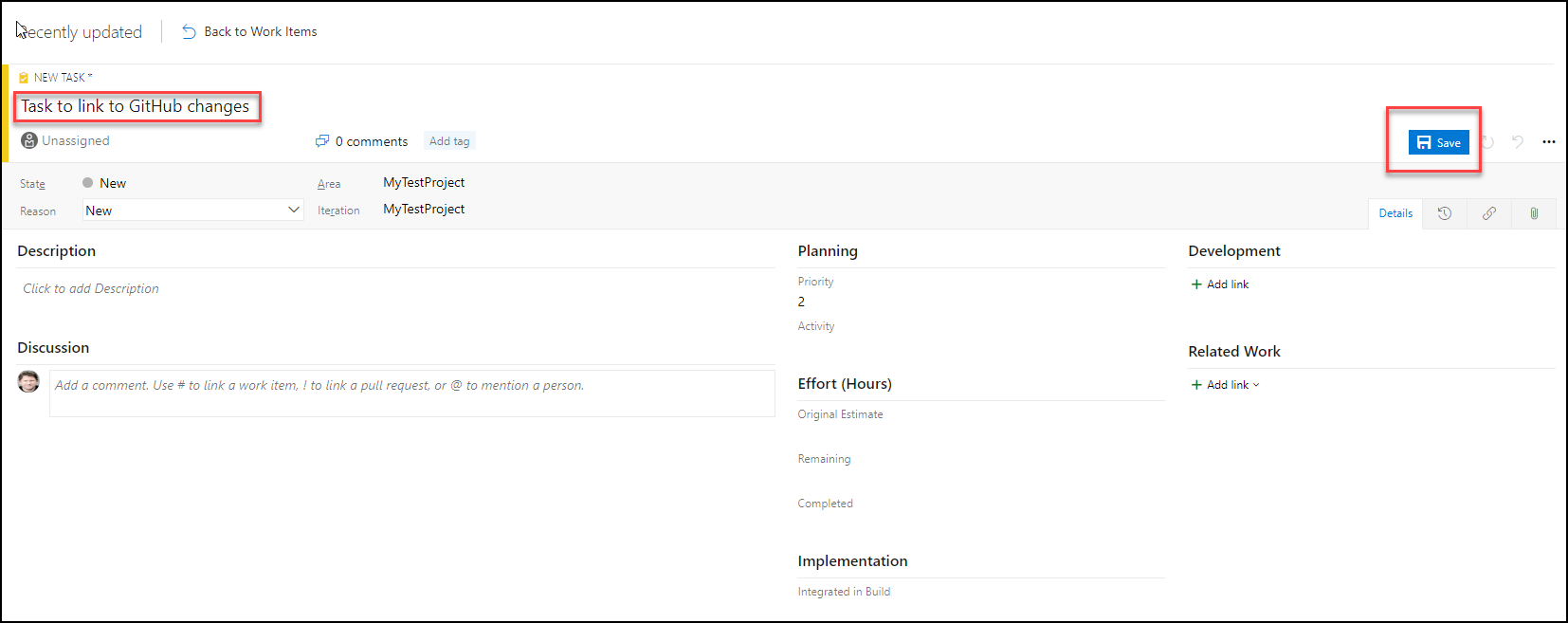
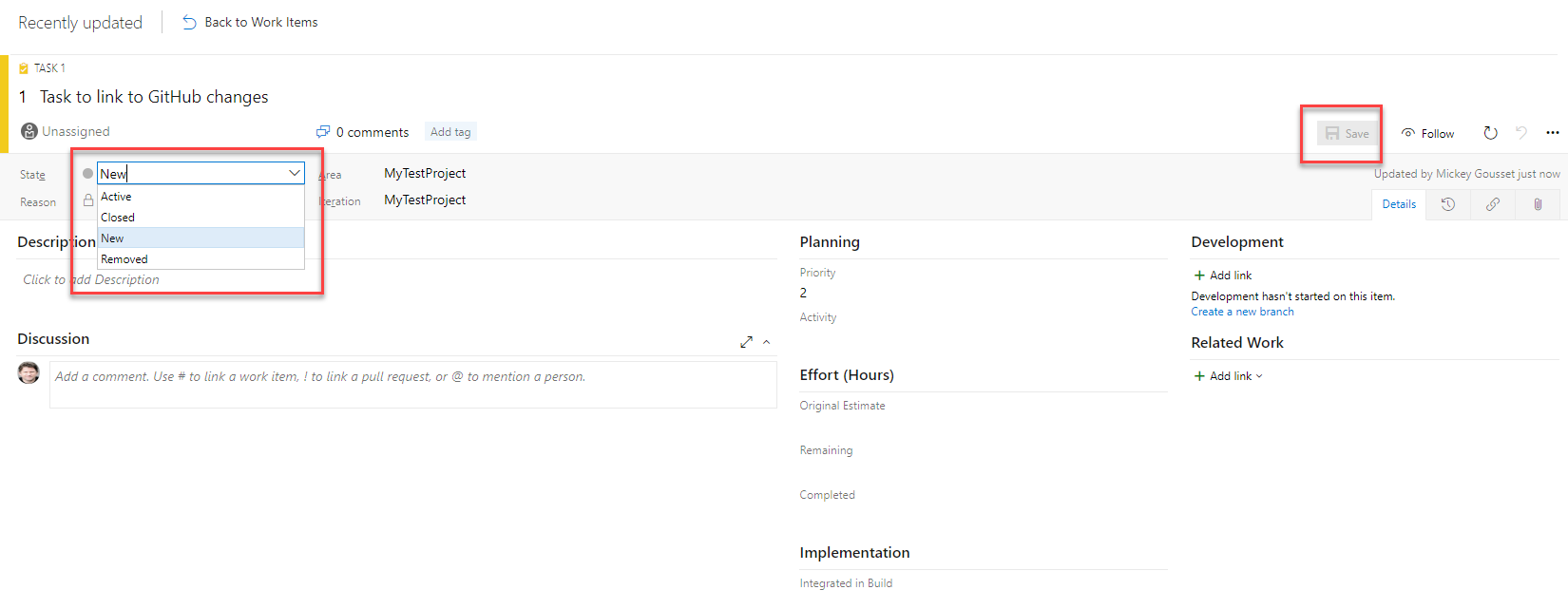
# **Exercise 2: Create an Azure DevOps Team Project**

**Estimated Time:** 5 Minutes

Azure Boards integration will work with existing Azure DevOps team projects, but for the purposes of this lab, you are going to create a new one. This lab assumes you already have an Azure DevOps organization created.

1. **Open a web browser and navigate to http://dev.azure.com/<YourOrgName>.**
2. **In the upper-right corner of the page, click the Create project button.**
3. **Enter the following information:  
   Project name: MyTestProject  
   Description: I am a test project being used for hands on labs  
   Set the project visibility to be Public. (You are doing this to help with future hands on labs. The project can also be a private project)  
     
   Click the Create button to create the new team project.**

**With the team project created, the next step is to create a task work item that you will associate with your GitHub commit in a future step.**

1. **On the left-hand side of the screen, select Boards. This will open to Azure Boards section on work items.**
2. **In the middle section, click the + New Work Item drop down list, and select Task. This will open a new task work item form.**
3. **Enter the following information:  
   Name: Task to link to GitHub changes** **Click the Save button to save changes to the task.**
4. **On the task work item form, change the state from New to Active. Click the Save button to save your changes.**

**Make a note of the work item ID (in the screenshot above, the work item ID is “1”. However, yours may be different). You will need this ID in a future exercise, when you link your changes back to this work item.**

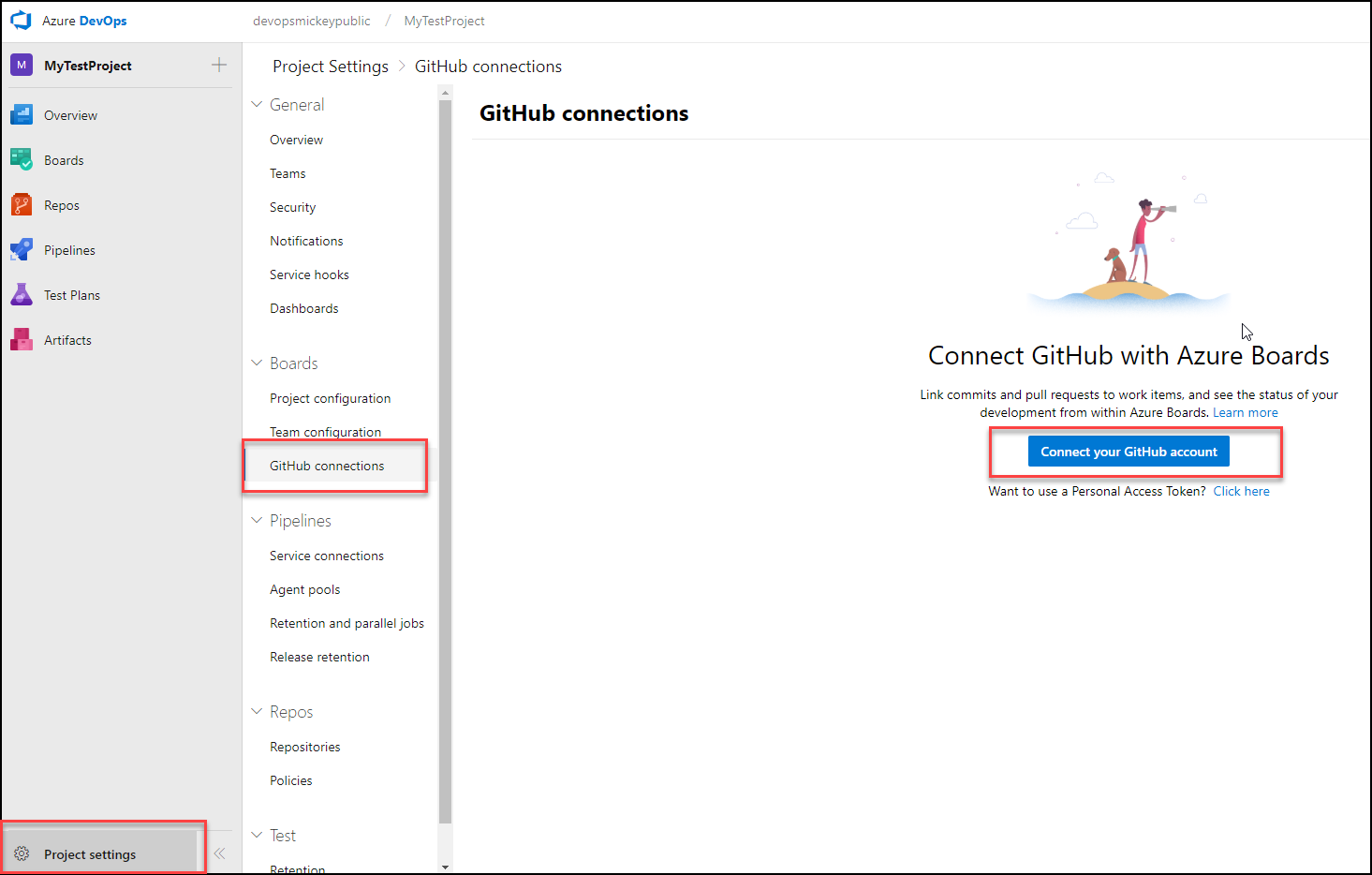
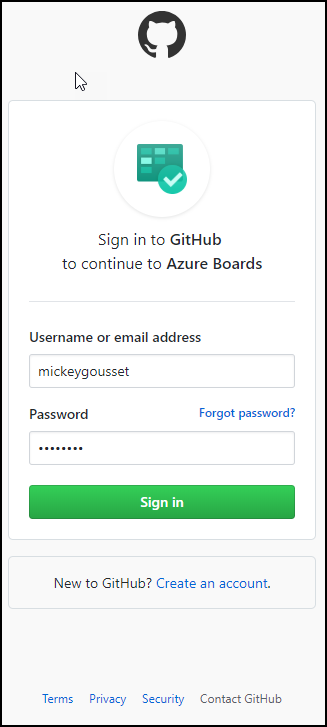
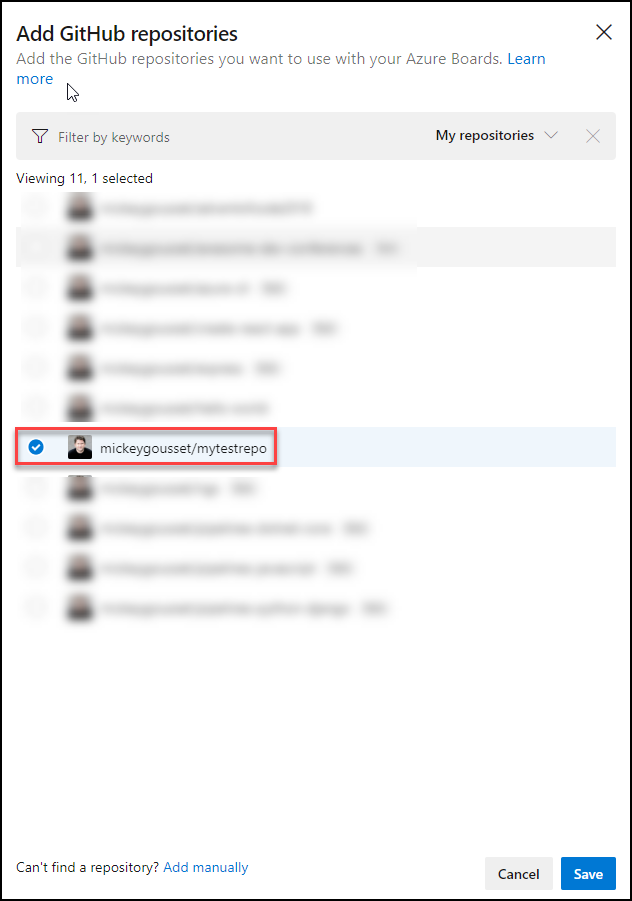
**At this point, you have created a new GitHub repo to store your code, and you have created a new Azure DevOps team project to track your work. You have created a task, representing the changes you want to make to your code.**

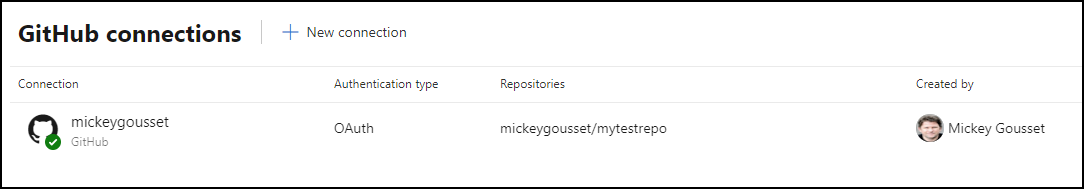
**However, before you can link your commits with your work items, you need to connect your Azure Boards project with your GitHub repository. You’ll learn how to do that in the next lab.**

# **Exercise 3: Connect Your Azure Boards project to GitHub**

**Estimated Time:** 5 Minutes

When you connect your Azure Boards project to your GitHub repositories, you are providing the needed connection to allow linking between GitHub commits/pull requests and Azure Boards work items. This allows you to use GitHub to develop your code, while using Azure Boards for planning and tracking.

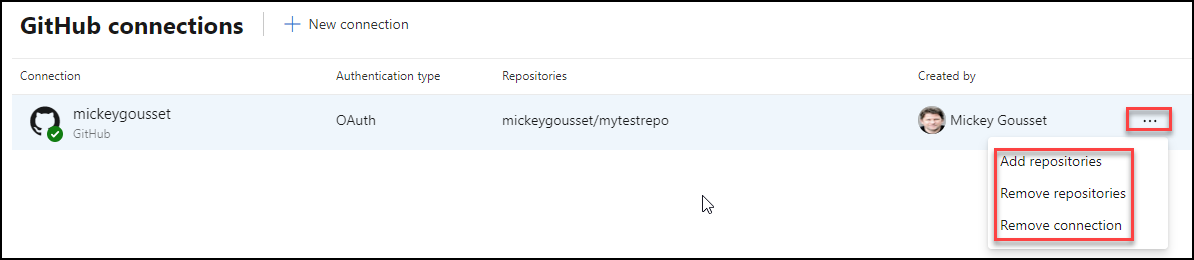
1. **You should still have your web browser, from Exercise 2, open to your team project. In the bottom-left corner, click Project settings.**
2. **In the menu that appears, under Boards, select GitHub Connections.**
3. **In the middle pane, click the Connect your GitHub account button.**
4. **You will need to sign into GitHub. Enter your username and password for GitHub and click the Sign in button.**
5. **On the Add GitHub respositories page, select the mytestrepo that you created in Exercise 1. Click the Save button.**

**You should now see your GitHub connection listed under GitHub connections.**

**If you need to add new connections, you can click the + New Connection button.**

**If you need to modify an existing connection, hover your mouse over the connection. Click on the ellipse “…” that appears on the right, and a menu will open with three options:**

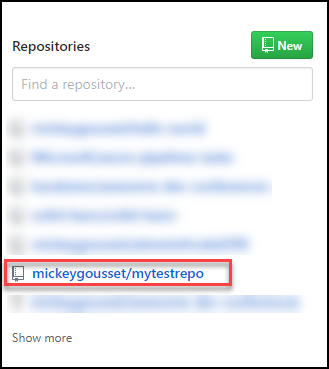
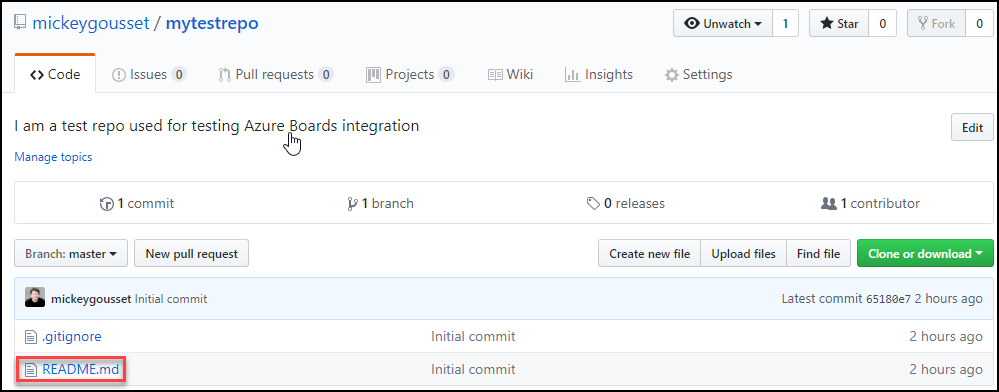
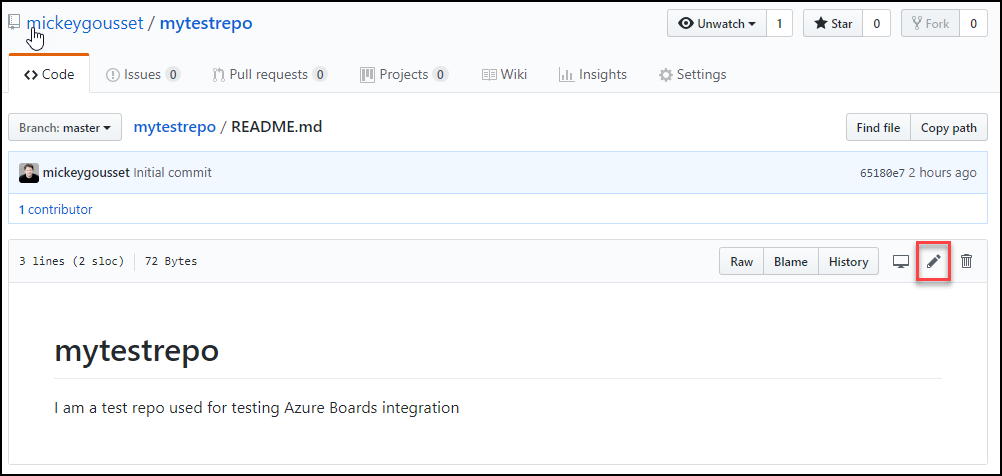
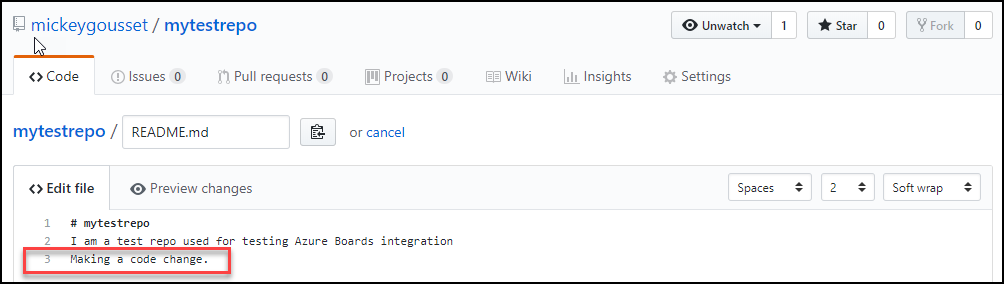
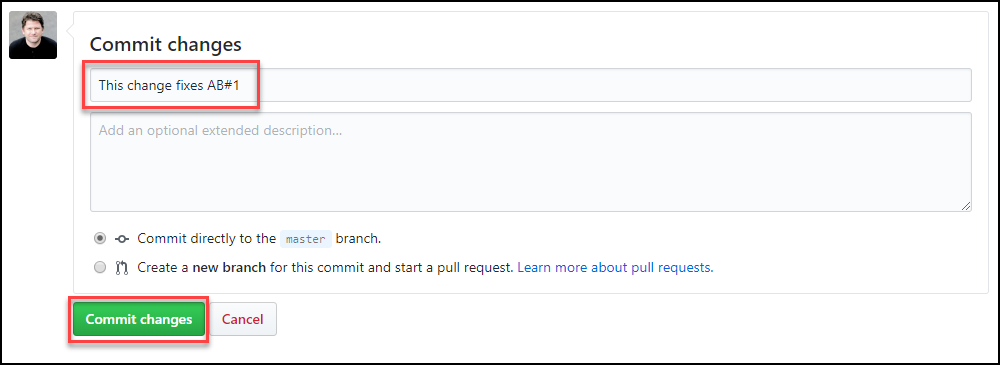
* **Add repositories**
* **Remove repositories**
* **Remove connection**



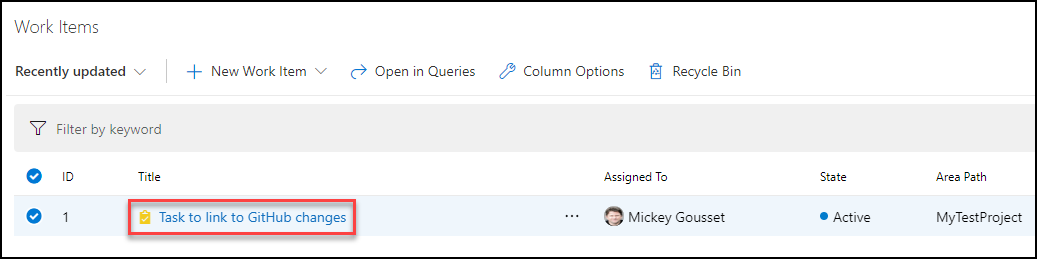
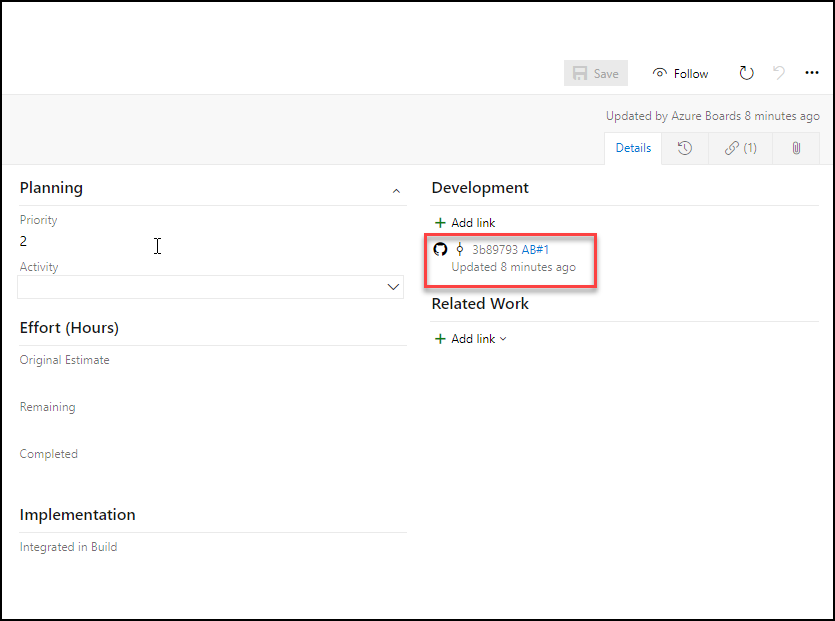
# **Exercise 4: Commit a Change and Link It to Azure Boards Task**

**Estimated Time:** 5 Minutes

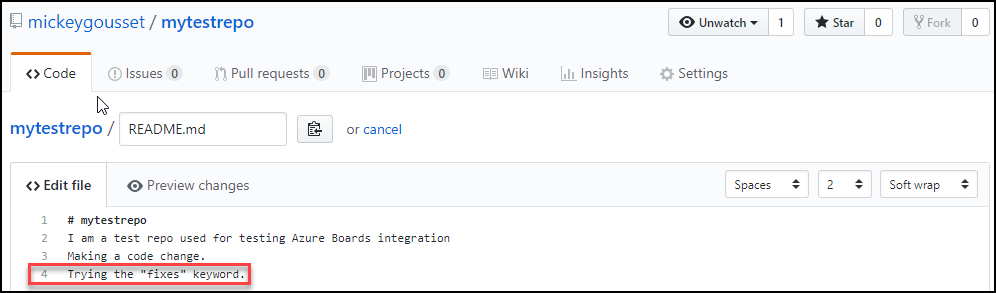
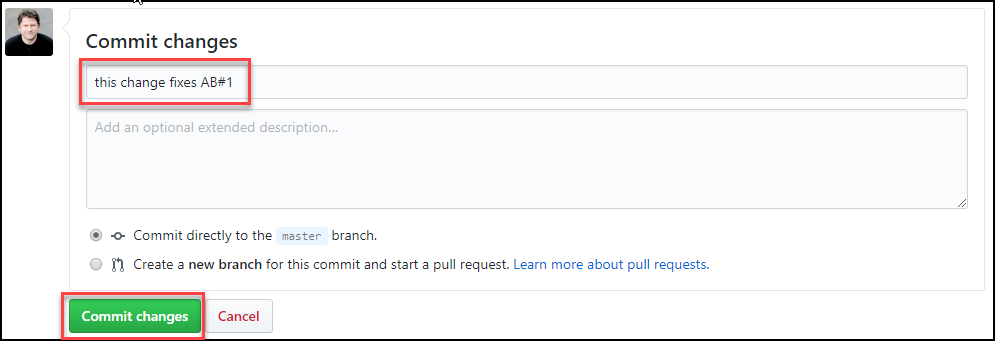
At this point, you have everything in place to enable linking between GitHub Commit/Pull requests and Azure Boards work items. For this exercise, you are going to edit the README.MD file directly in the browser, and commit it back to the repository.

1. **In your web browser, navigate to** <https://github.com>**. On the left-hand side of the screen, click the mytestrepo you created in Exercise 1.**
2. **Click the ReadMe.md file.**
3. **Click the Edit this file button to start editing the file.**
4. **Add the following statement to the file on line 3: Making a code change.**
5. **Scroll to the bottom of the page and enter the following for the commit description: AB#<YourWorkItemId>. Click the Commit changes button to commit your changes.**

**The magic here is in the commit description. AB#1 creates a link between the commit and the work id with ID “1” in Azure Boards. You could have added a more descriptive message, and included AB#<YourWorkItemID> and the linking would still occur. In addition, you can use the keywords fix, fixes, fixed and apply it to the #-mention item to transition the work item to its completed state.**

1. **In your web browser, navigate to your Azure DevOps team project, and click Boards. You should see the task work item you created earlier. Click it to open it.**
2. **On the right-hand side of the work item, under Development, you can see the link from the GitHub commit. Clicking the link will take you to the commit information in GitHub.**
3. **You can also click the Links tab in the work item and see the same information.**

**Now let’s make another edit to the file and try using the keyword “fixes” in the commit message.**

1. **Back in the GitHub repo, click the README.md file, then click the Edit this file icon.**
2. **Add the following statement to the file on line 4: Trying the “fixes” keyword.**
3. **Scroll to the bottom of the page and enter the following for the commit description: this change fixes AB#<YourWorkItemId>. Click the Commit changes button to commit your changes.**
4. **Navigate back to the task work item in Azure Boards. You will see that a second link has been added to the task, and that the task has transitioned from Active** to **Closed**.

**The same AB# mention syntax and keywords will work the same way with pull requests.**