

Development - Development Pull Request Checklist

01/03/2024 5:15 pm EST

Description

This is a living document that helps to define the process of what the developer who has created a pull request should do and what the reviewing developers should look for.

Developer Making The Change

Before creating a pull request, you should make sure you have merged all the changesets from the parent branch into your branch and have tested the changes that the merge introduces. This will help ensure there are no merge conflicts when the pull request is completed. To merge in Visual Studio 2017, make sure you are in your branch, click on Team Explorer, click the Home button, click the Branches button, click on the Merge link. It will ask you "Merge from branch:". Pick your branches parent. In the "Into current branch:" text box should be your branch. Leave the "Commit changes after merging" checkbox checked. Click the Merge button. This will create outgoing commits on your branch (locally) from the commits that you didn't have from the parent branch. If there are any merge conflicts, you can resolve them. You can review the outgoing commits in the Sync tab under Team Explorer/Home/Sync. You should rebuild your solution(s) to make sure they still build. If you're not sure if the merged changes affected your changes then re-test your changes. Once you are confident with your changes after the merge, you can use Sync or Push to get those changes back into DevOps. You are now ready to create the pull request. You should add at least one other developer as a reviewer at this time. After the developers have approved the pull request, you should add QA people to the pull request so that they are notified of an approved pull request. Only after the QA people approve the pull request should the pull request be completed.

Owner Of A Feature Or Epic

If you are the owner of a feature or epic, you should routinely merge changes from master into the branch. Handle any merge conflicts and Sync the changes back into DevOps. This way people working on PBIs or bugs underneath your branch will get changes from master when they merge.

Developer Reviewing A Pull Request

As a developer reviewing a pull request you should look for things such as:

- Do the changes complete what was stated in the work item?
- Memory leaks.
- Complicated code without comments.
- Are best practices being used?
- Consider if the changes are in the "right place". For example, should the changes have been made in the AppServer instead of the NEO API?

- Consider how the changes could affect other things and have those other things been accounted for in the change.

If it is a complex change then you might have to get the branch, build and test the changes yourself before approving.

If you have questions/comments/concerns, then make comments in the pull request.

If everything checks out then you can approve the pull request.