

# Azure DevOps and GitHub

Wolfgang Ofner  
Senior Cloud Architect



# Agenda



Overview Azure DevOps  
Demo Workflow  
Automate Deployments  
Overview GitHub

# Wolfgang Ofner

Freelance Cloud Architect, Toronto, Canada

Focus on Azure, Kubernetes, and DevOps



[programmingwithwolfgang.com](http://programmingwithwolfgang.com)



[wolfgangofner](#)



[@programmingwithwolfgang](#)



# Organization Settings

P ProgrammingWithWolfgang

[+ New project](#)[New organization](#)

## ProgrammingWithWolfgang

[Projects](#) [My work items](#) [My pull requests](#)[Filter projects](#)

M MicroserviceDemo

P Presentation

Y Youtube

A AzureArcDemo

A AzureArcDemo

K K8sAdoAgent

M MicroserviceDemo

P Presentation

W WindowsDeveloperArcDemo

Y Youtube

[What's new](#)[Sprint 241](#)

With this update, we improved the structure of personal access tokens to boost security.

[Organization settings](#)

# Organization Settings

Manage global settings:

- Manage users and their licenses
- Configure billing
- Extensions
- Microsoft Entra
- Processes
- Global settings for repositories

# Access Level

## Basic

- Provides access to all features except Test Plans

# Access Level

## Basic

- Provides access to all features except Test Plans

## Visual Studio Subscriber

- Same features as Basic
- “Free” license
- Visual Studio Enterprise allows for an additional parallel build

# Access Level

## Basic + Test Plans

- Same as Basic plus Test Plans

# Access Level

## Basic + Test Plans

- Same as Basic plus Test Plans

## Stakeholder

- Unlimited free users
- Limited features including no repository or pipeline access

# Billing

Microsoft hosted agents have 1800 minutes / month includes for free

Microsoft hosted agents allow 1 parallel job for free

5 Basic licenses are included for free

# Billing

Billing needs an Azure subscription

- More than 5 Basic licenses
- Additional build minutes
- Additional parallel jobs

# Extensions

Free and paid extensions

Bring additional features to ADO

- Time tracking
- Pipeline tasks
- Integration with other tools, e.g. Jira

# Microsoft Entra

Microsoft Entra, formerly Azure Active Directory  
Used for user and billing management

# Process

Manage existing project processes

- Scrum
- Agile
- CMMI
- Basic

# Process

“Create inherited process” based on an existing process

- Change existing items and workflows
- Create new items and workflows

# Agent Pools

Collection of (pipeline) agents

- Manage groups of agents instead of individual ones
- Group agents based on capabilities → operating system, installed software
- Overview of all executed jobs

# Agent Pools

Microsoft offers one agent in the Azure Pipelines pool for free

Microsoft-hosted agents vs. Self-hosted agents

# Parallel Jobs

Overview of available jobs

Purchase additional jobs

- Microsoft-hosted agent: 40 USD / month
- Self-hosted agent: 15 USD / month

# Project Settings

# Permissions

Configure permissions for users or groups  
(groups are preferred)

- Allow
- Deny
- Not set (has a deny effect)

# Permissions

Permissions can be inherited

Direct permissions win over inherited ones

# Service Hooks

Integrate third party tools with ADO

- Teams
- Slack
- Azure Service Bus
- Datadog
- Grafana

# Service Hooks

## Supported events

- Work item update
- Code pushed
- Build complete
- Release deployed

# Project Configuration

Configure settings for the project

- Create and configure iterations
- Create and manage areas of the project

# Team Configuration

Configure settings for individual teams

- Team settings such as backlog, workdays and bug handling
- Manage iterations
- Manage areas
- Create templates for work items

# Service Connections

Configure connection to third party tools

- Azure
- DockerHub
- GitHub
- Jira
- NuGet
- Npm

# Project Menu Items

# Overview

## Summary

- Provides project description
- Readme file or Wiki

# Overview

## Dashboard

- Create dashboards
- Pipeline status, work item count, burndown chart, etc.

# Overview

## Wiki

- Document where the code resides
- Link work items and code commits
- Uses Git in the background
- Markdown

# Boards

Manage work items with different boards

- Boards
- Backlog
- Sprints
- Queries
- Delivery Plans

# Boards

Plan team capacity per sprint  
Burndown chart for sprints

# Boards

## Create custom queries

- Work items where status change > 180 days
- All closed work items in a timeframe
- Work items approved by team member XYZ

# Boards

## Delivery plans

- Visualize milestones and delivery plan over all teams
- Update iterations
- Used by Product Owner / Release Manager

# Azure DevOps Features and Stories


[Settings](#)

Azure DevOps team all up



		Publish				End Q2 21		today					
> Teams		< November 2020		October 2020		November 2020		December 2020		January		February >	
> Azure DevOps Features	2020 - 9/30	+ 3 Feature	10_2020 10/1 - 10/30	+ 5 Feature	11_2020 11/1 - 11/30	+ 15 Feature	12_2020 12/1 - 12/31	+ 31 Feature	01_2021 1/1 - 1/29	+ 9 Feature	02_2021 2/1 - 2/26	+ 3 Feature	03_2021
> Azure Pipelines Stories	2020 - 9/30	+ 28 User Story	10_2020 10/1 - 10/30	+ 18 User Story	11_2020 11/1 - 11/30	+ 17 User Story	12_2020 12/1 - 12/31	+ 22 User Story	01_2021 1/1 - 1/29	+ 30 User Story	02_2021 2/1 - 2/26	+ 1 User Story	03_2021
> Azure Release Stories	2020 - 9/30	+ 1 User Story	10_2020 10/1 - 10/30	+ 1 User Story	11_2020 11/1 - 11/30	+ 1 User Story	12_2020 12/1 - 12/31	+ 1 User Story	01_2021 1/1 - 1/29	+ 1 User Story	02_2021 2/1 - 2/26	+ 1 User Story	03_2021
> Azure Artifacts Stories	2020 - 9/30	+ 8 User Story	10_2020 10/1 - 10/30	+ 3 User Story	11_2020 11/1 - 11/30	+ 9 User Story	12_2020 12/1 - 12/31	+ 7 User Story	01_2021 1/1 - 1/29	+ 5 User Story	02_2021 2/1 - 2/26	+ 3 User Story	03_2021

# Repos

Overview of everything Git related

- Files
- Commits
- Branches
- Tags
- Pull Requests

# Repos

## Advanced Security (GitHub Advanced Security for Azure DevOps)

- Comprehensive security suite
- Secret scanning
- Dependency scanning
- Static code scanning
- 49\$ per month per active committer → too expensive

# Pipelines

## Pipelines

- YAML pipelines
- Used for build and deployments tasks
- File is stored in Git repo

# Pipelines

## Environments

- Can be created in a pipeline or portal
- Overview of when and what was deployed to a specific environment
- Configure approvals and checks before deployment is executed

# Pipelines

## Outdated features

- Releases
- Task groups
- Deployment groups

# Pipelines

## Library

- Create variable groups which can be shared across multiple pipelines
- Securely store sensitive files such as certificates
- However, using Azure Key Vault is the recommended way

# Test Plans

Needs additional license

Used for manual testing and exploratory testing

- Test cases can be linked to work items
- Specify test configuration, e.g. operating system, browser, etc.
- Tester can add screenshots or screen recordings

# Artifacts

Create private feeds for packages

- NuGet
- Npm
- Maven
- Pip

# Git

# Git

Distributed version control

Commit changes with a message

Creates a history of all commits

# Git

Commits are made to a branch

Branches are individual work areas

Use .gitignore file to exclude files or extensions

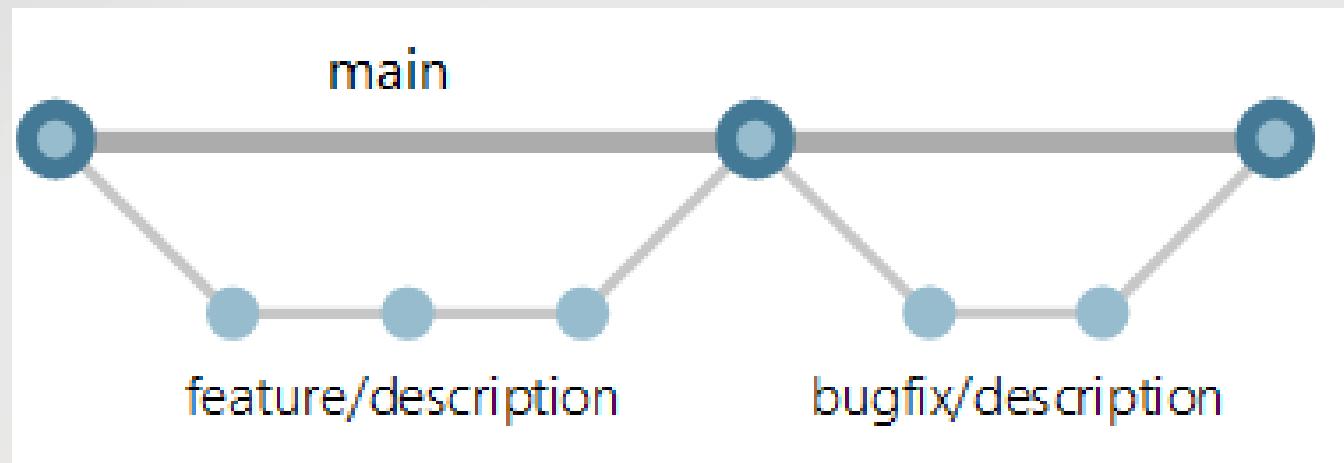
# Git Branches

Main branch should only contain tested / deployable code

Developers work with “feature” branches

Once a feature is completed, feature branch will be merged into main branch

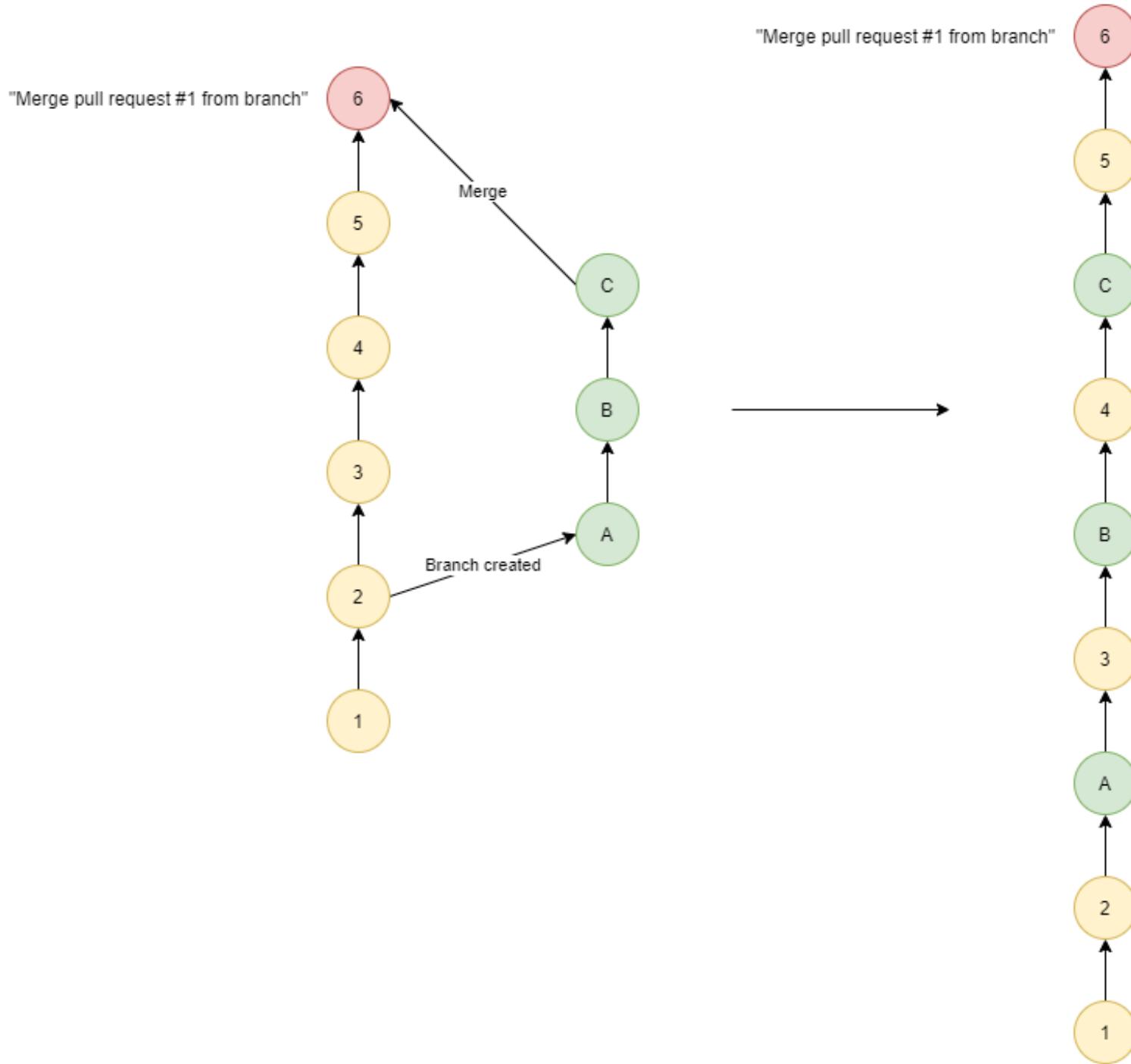
# Git Branches



# Git Branches

Various merge strategies are available

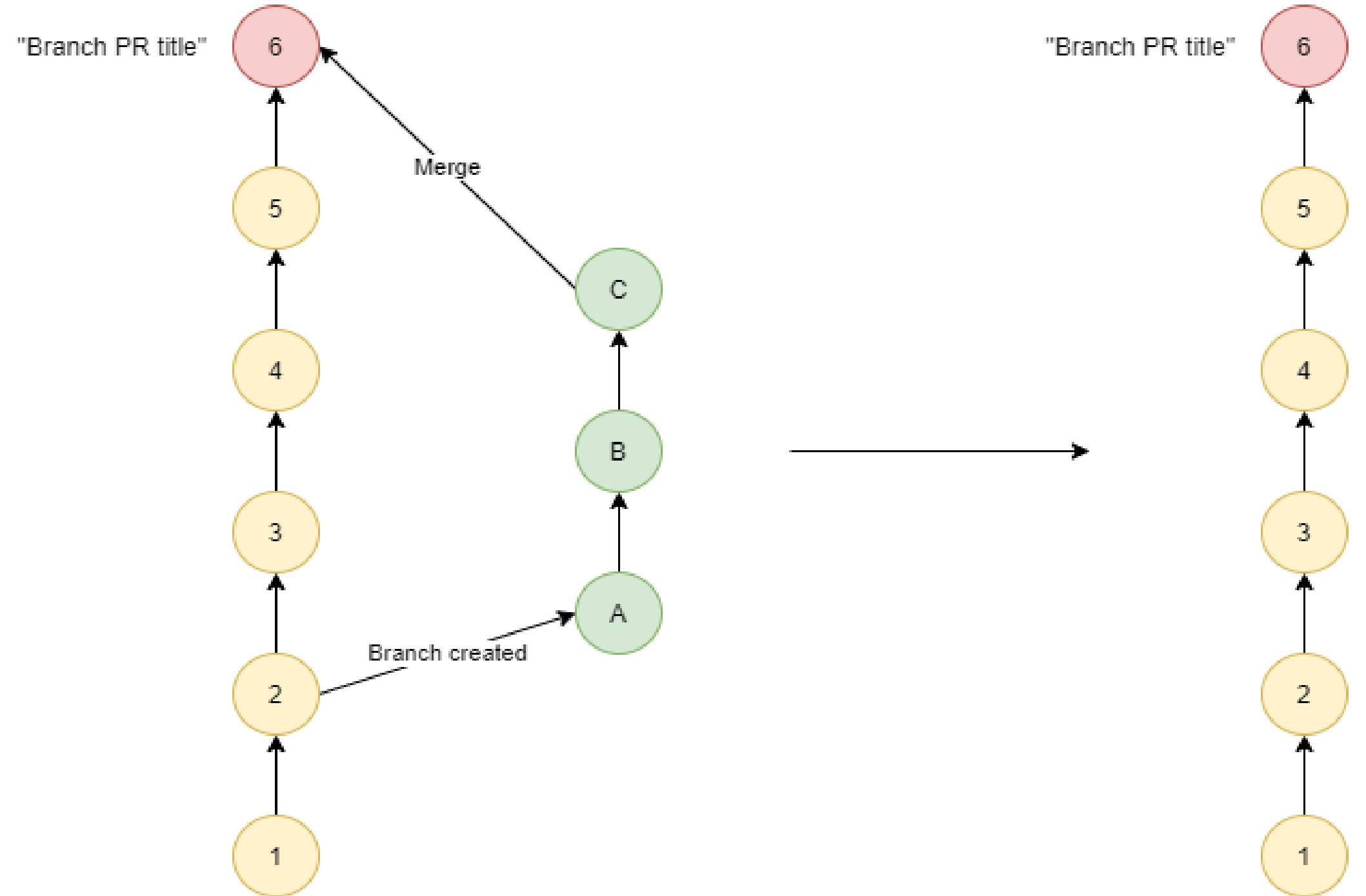
- Basic Merge



# Git Branches

Various merge strategies are available

- Basic Merge
- Squash Merge



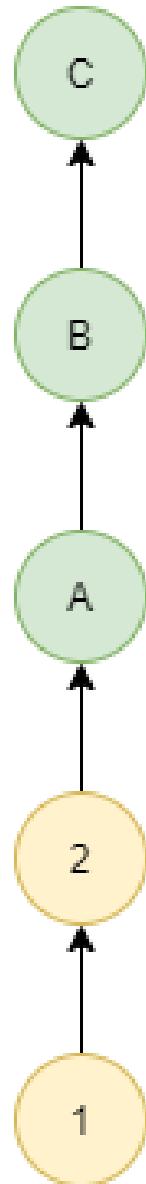
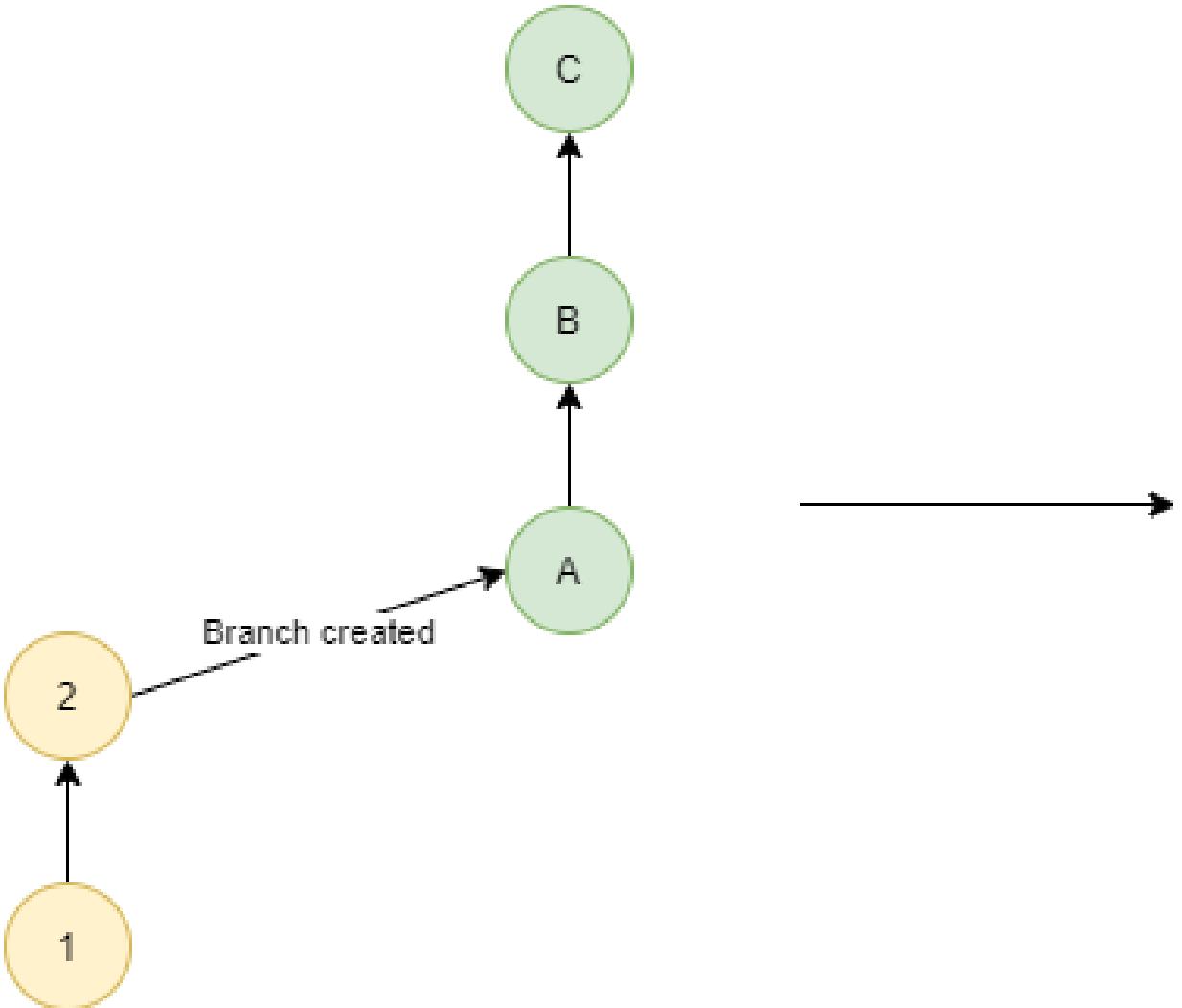
# Git Branches

Various merge strategies are available

- Basic Merge
- Squash Merge
- Fast-Forward Merge

Base Branch

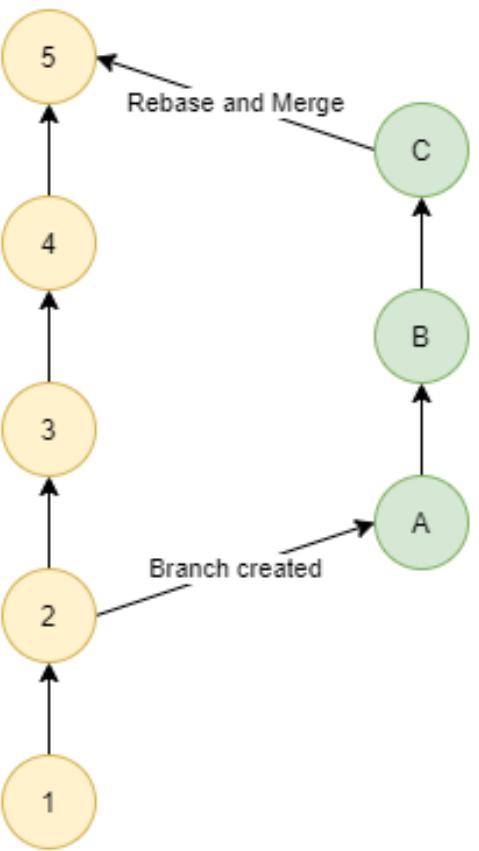
Branch Being Merged



# Git Branches

Various merge strategies are available

- Basic Merge
- Squash Merge
- Fast-Forward Merge
- Rebase



# Merge Conflicts

Git tries to determine the correct version if multiple developers change the same file

If Git can't figure it out → merge conflict and developers have to manually resolve it

# Merge Conflicts

Pull destination branch and merge it locally to resolve the conflict

Use short-lived branches and merge often

# Demo Project

P Presentation +

Overview

Boards

Repos

**Files**

Commits

Pushes

Branches

Tags

Pull requests

Advanced Security

Pipelines

Test Plans

Artifacts

Presentation is empty. Add some code!

#### Clone to your computer

HTTPS

SSH

<https://ProgrammingWithWolfgang@dev.azure.com/ProgrammingW>



OR



Clone in VS Code



Generate Git Credentials

Having problems authenticating in Git? Be sure to get the latest version [Git for Windows](#) or our plugins for [IntelliJ](#), [Eclipse](#), [Android Studio](#) or [Windows command line](#).

#### Push an existing repository from command line

HTTPS

SSH

```
git remote add origin  
https://ProgrammingWithWolfgang@dev.azure.com/ProgrammingWithWolfgang/Pre
```



#### Import a repository

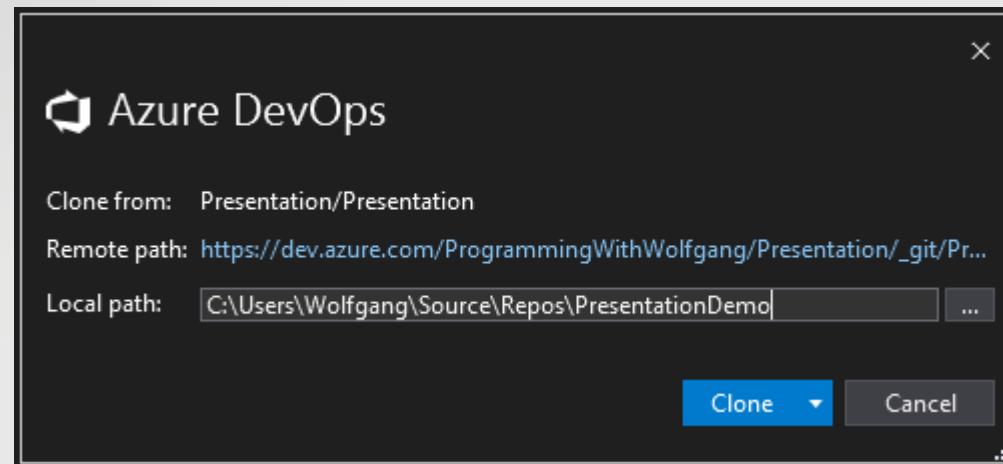
Import

#### Initialize **master** branch with a README or gitignore

Add a README

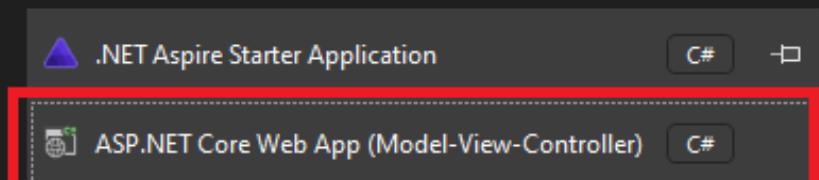
Add a .gitignore: None

Initialize



# Create a new project

## Recent project templates



Search for templates (Alt+S)

All languages All platforms All project types

**.NET Aspire Application** New  
A project template for creating an empty .NET Aspire app.  
C# .NET Aspire API Cloud Common Service Web

**.NET Aspire App Host** New  
A project template for creating a .NET Aspire app host (orchestrator) project.  
C# .NET Aspire Cloud Common

**.NET Aspire Service Defaults** New  
A project template for creating a .NET Aspire service defaults project.  
C# .NET Aspire API Cloud Common Service Web

**.NET Aspire Test Project (xUnit)** New  
A project that contains xUnit.net integration tests of a .NET Aspire AppHost project.  
C# .NET Aspire API Cloud Common Service Test

**Console App**  
A project for creating a command-line application that can run on .NET on Windows, Linux and macOS  
C# F# VB.NET macOS Windows Linux Console

**Next**

# Configure your new project

## ASP.NET Core Web App (Model-View-Controller)

C#

Linux

macOS

Windows

Cloud

Service

Web

Project name

PresentationDemo

Location

C:\Users\Wolfgang\source\repos\PresentationDemo\



Solution

Create new solution

Solution name ⓘ

PresentationDemo

 Place solution and project in the same directory

Project will be created in "C:\Users\Wolfgang\source\repos\PresentationDemo\PresentationDemo\\PresentationDemo\"

[Back](#)[Next](#)

# Additional information

## ASP.NET Core Web App (Model-View-Controller)

C#

Linux

macOS

Windows

Cloud

Service

Web

### Framework ⓘ

.NET 8.0 (Long Term Support)

### Authentication type ⓘ

None

Configure for HTTPS ⓘ

Enable container support ⓘ

### Container OS ⓘ

Linux

### Container build type ⓘ

Dockerfile

Do not use top-level statements ⓘ

Enlist in .NET Aspire orchestration ⓘ

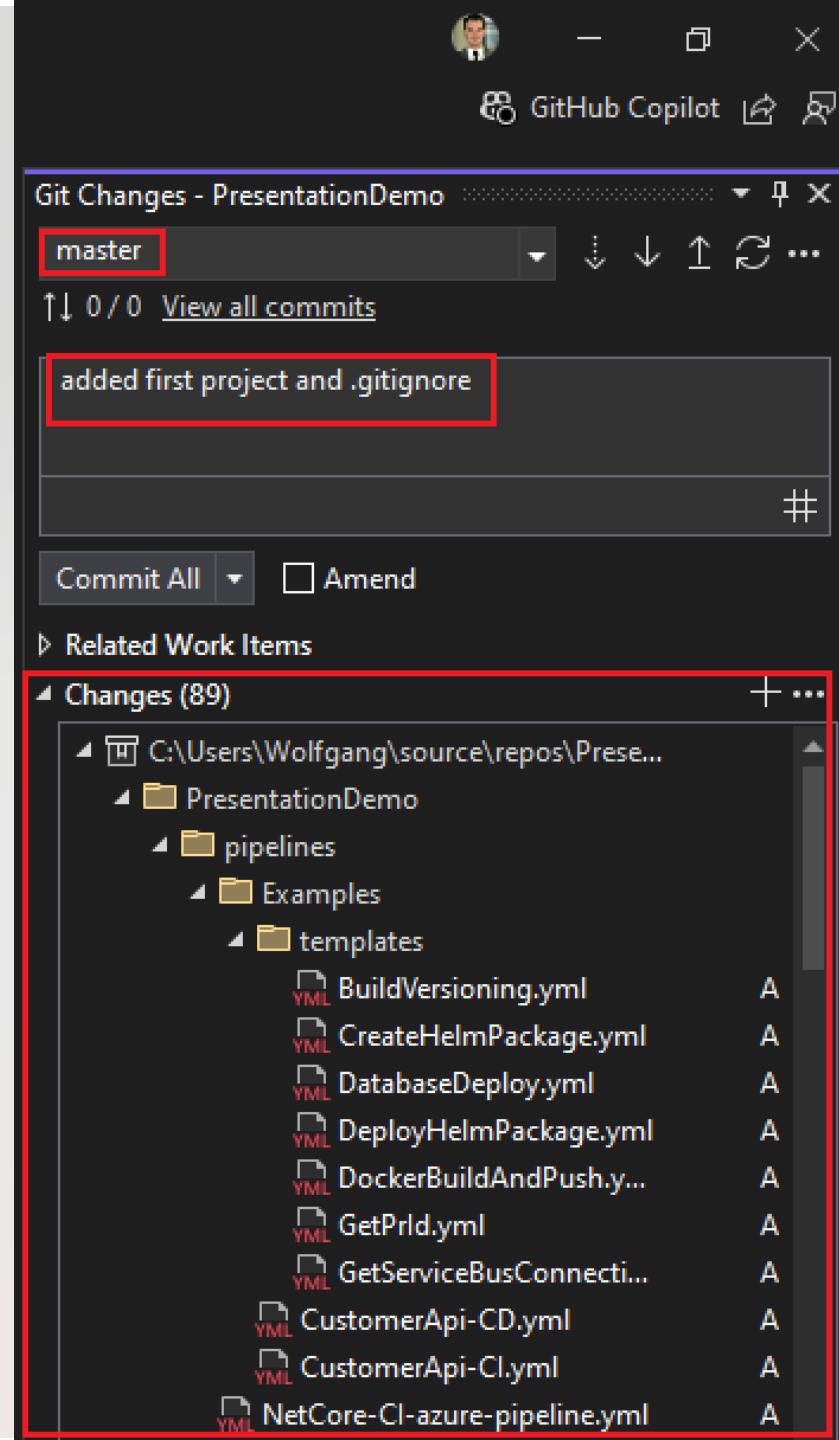
Back

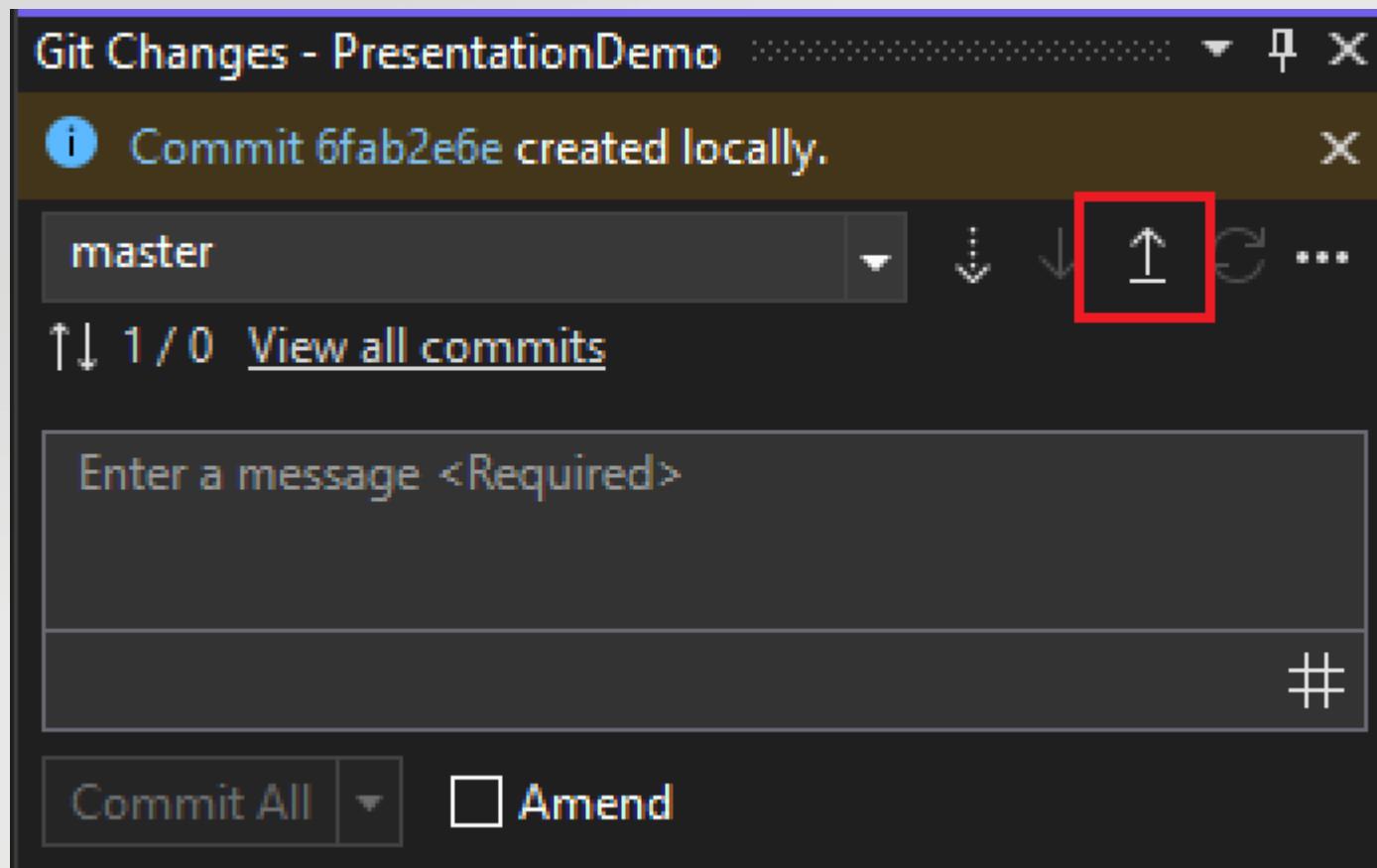
Create

The screenshot shows the Visual Studio IDE interface with the following details:

- Menu Bar:** File, Edit, View, Git, Project, Build, Debug, Team, Test, Analyze, Tools, Extensions, Window, Help.
- Search Bar:** Search -> PresentationDemo
- Toolbars:** Standard toolbar with icons for Undo, Redo, Cut, Copy, Paste, Find, Replace, etc.
- Project Explorer:** Shows the project 'PresentationDemo' with its files.
- Status Bar:** Debug, Any CPU, PresentationDemo.AppHost, https://
- Code Editor:** The current file is '.gitignore'. The code content is as follows:

```
1 ## Ignore Visual Studio temporary files, build results, and
2 ## files generated by popular Visual Studio add-ons.
3 ##
4 ## Get latest from https://github.com/github/gitignore/blob/master/VisualStudio.gitignore
5
6 # User-specific files
7 *.rsuser
8 *.suo
9 *.user
10 *.userosscache
11 *.sln.docstates
12
13 # User-specific files (MonoDevelop/Xamarin Studio)
14 *.userprefs
15
16 # Build results
17 [Dd]ebug/
18 [Dd]ebugPublic/
19 [Rr]elease/
20 [Rr]eleases/
21 x64/
22 x86/
23 [Aa][Rr][Mm]/
24 [Aa][Rr][Mm]64/
25 bld/
26 [Bb]in/
27 [Oo]bj/
28 [Ll]og/
29
30 # Visual Studio 2015/2017 cache/options directory
31 .vs/
```





P Presentation +

Overview

Boards

**Repos**

Files

Commits

Pushes

Branches

Tags

Pull requests

Advanced Security

Pipelines

Test Plans

Artifacts

Project settings

◆ Presentation

> 📁 PresentationDemo

📄 .gitignore

⌚ master

⌚ / Type to find a file or folder...

**Files**

Contents History

**Set up build** **Clone**

Name ↑	Last change	Commits
📁 PresentationDemo	5m ago	<a href="#">6fab2e6e</a> added first project and .gitignore Wolfgang...
📄 .gitignore	5m ago	<a href="#">6fab2e6e</a> added first project and .gitignore Wolfgang...



- P **Presentation** +
-  Overview
-  Boards
-  Repos
-  **Pipelines**
-  **Pipelines**
-  Environments
-  Releases
-  Library
-  Task groups
-  Deployment groups
-  Test Plans
-  Artifacts



## Create your first Pipeline

Automate your build and release processes using our wizard, and go from code to cloud-hosted within minutes.

[Create Pipeline](#)



Azure DevOps

ProgrammingWithWolfgang / Presentation / Pipelines



Presentation



Overview



Boards



Repos



Pipelines



Pipelines



Environments



Releases



Library



Task groups



Deployment groups



Test Plans

Connect

Select

Configure

Review

New pipeline

## Where is your code?

Azure Repos Git YAML

Free private Git repositories, pull requests, and code search

Bitbucket Cloud YAML

Hosted by Atlassian

GitHub YAML

Home to the world's largest community of developers

GitHub Enterprise Server YAML

The self-hosted version of GitHub Enterprise



Other Git

Any generic Git repository



Subversion

Centralized version control by Apache

Use the [classic editor](#) to create a pipeline without YAML.



P **Presentation**

+

✓ Connect

**Select**

Configure

Review

Overview

Boards

Repos

Pipelines

Pipelines

Environments

Releases

Library

Task groups

Deployment groups

New pipeline

## Select a repository

Filter by keywords

Presentation ▾



Presentation



## Xamarin.iOS

Build a Xamarin.iOS project.



## Starter pipeline

Start with a minimal pipeline that you can customize to build and deploy your code.



## Existing Azure Pipelines YAML file

Select an Azure Pipelines YAML file in any branch of the repository.



## .NET Core Function App to Windows on Azure

Build a .NET Core function app and deploy it to Azure as a Windows function App.



## Android

Build your Android project with Gradle.



## Ant

Build your Java projects and run tests with Apache Ant.



## ASP.NET Core

Build and test ASP.NET Core projects targeting .NET Core.



## C/C++ with GCC

Build your C/C++ project with GCC using make.



## Deploy to Azure Kubernetes Service

Build and push image to Azure Container Registry; Deploy to Azure Kubernetes Service



## Docker

Build a Docker image



## Docker

Build and push an image to Azure Container Registry

Azure DevOps ProgrammingWithWolfgang / Presentation / Pipelines

Search

P Presentation + ✓ Connect ✓ Select ✓ Configure Review

New pipeline

## Review your pipeline YAML

Variables Save and run

Presentation / azure-pipelines.yml \* ↕ Show assistant

```
1 # ASP.NET Core
2 # Build and test ASP.NET Core projects targeting .NET Core.
3 # Add steps that run tests, create a NuGet package, deploy, and more:
4 # https://docs.microsoft.com/azure/devops/pipelines/languages/dotnet-core
5
6 trigger:
7 - master
8
9 pool:
10 - vmImage: ubuntu-latest
11
12 variables:
13 - buildConfiguration: 'Release'
14
15 steps:
16 - script: dotnet build --configuration $(buildConfiguration)
17 - displayName: 'dotnet build $(buildConfiguration)'
```

74

Azure DevOps ProgrammingWithWolfgang / Presentation / Pipelines

P Presentation +

✓ Connect ✓ Select Configure Review

New pipeline

## Configure your pipeline

ASP.NET Build and test ASP.NET projects.

.NET Core ASP.NET Core (.NET Framework) Build and test ASP.NET Core projects targeting the full .NET Framework.

.NET .NET Desktop Build and run tests for .NET Desktop or Windows classic desktop solutions.

Universal Windows Platform Build a Universal Windows Platform project using Visual Studio.

Xamarin.Android Build a Xamarin.Android project.

Xamarin.iOS Build a Xamarin.iOS project.

Starter pipeline Start with a minimal pipeline that you can customize to build and deploy your code.

Existing Azure Pipelines YAML file Select an Azure Pipelines YAML file in any branch of the repository.

Show more

Select an existing YAML file

Select an Azure Pipelines YAML file in any branch of the repository.

Branch master

Path /PresentationDemo/pipelines/NetCore-CI-azure-pipeline.yml

Presentation

Cancel Continue

✓ Connect ✓ Select ✓ Configure Review

New pipeline

## Review your pipeline YAML

Variables Run ▾

Run

❖ Presentation / PresentationDemo/pipelines/NetCore-CI-azure-pipeline.yml

Show assistant

```
1  name : .NetCore-PresentationDemo-CI
2  trigger: none
3
4  pool:
5    vmImage: 'ubuntu-latest'
6
7  variables:
8    buildConfiguration: 'Release'
9
10 steps:
11   - task: DotNetCoreCLI@2
12     inputs:
13       command: 'restore'
14       projects: '**/PresentationDemo*.csproj'
15       displayName: 'Restore Nuget Packages'
16
17   - task: DotNetCoreCLI@2
18     inputs:
19       command: 'build'
20       projects: '**/PresentationDemo*.csproj'
21       arguments: '--no-restore'
22       displayName: 'Build projects'
23
24   - task: DotNetCoreCLI@2
25     inputs:
26       command: 'test'
27       projects: '**/*Test.csproj'
28       arguments: '--no-restore--no-build'
29       displayName: 'Run Tests'
30
31   - task: DotNetCoreCLI@2
32     inputs:
33       command: 'publish'
34       publishWebProjects: false
35       projects: '**/PresentationDemo.csproj'
36       arguments: '--configuration $(buildConfiguration) --no-restore'
37       modifyOutputPath: false
38       displayName: 'Publish PresentationDemo'
39       condition: and(succeeded(), ne(variables['Build.Reason'], 'PullRequest'))
```

#NetCore-PresentationDemo-CI • added first project and .gitignore

⌚ Presentation

Cancel :

Summary Code Coverage

Manually run by Wolfgang Ofner [View 7 changes](#)

Repository and version

⌚ Presentation  
master ac557633

Time started and elapsed

⌚ Just now

Related

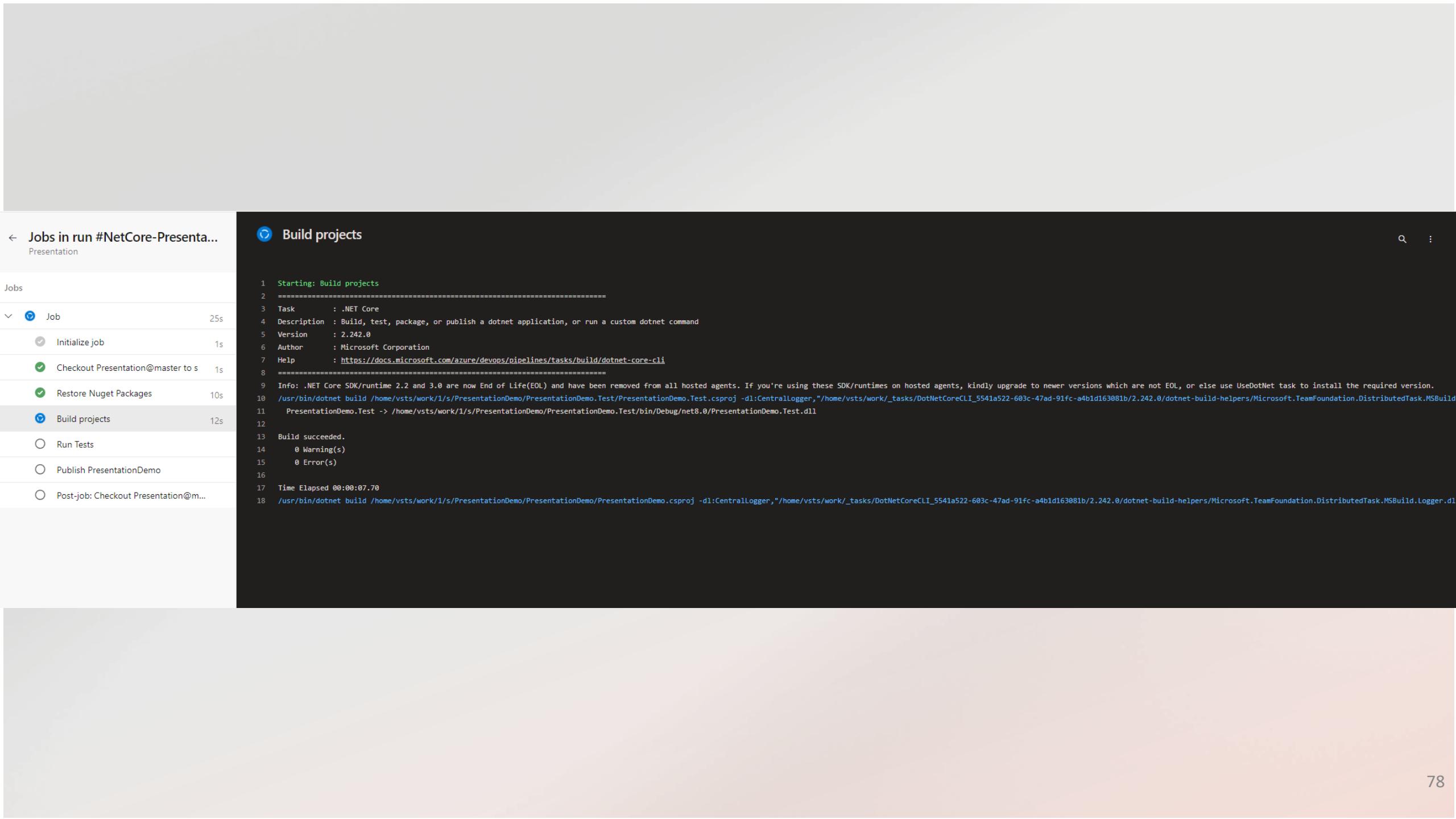
⌚ 0 work items  
⌚ 0 artifacts

Tests and coverage

⌚ Get started

Jobs

Name	Status	Duration
⌚ Job	Queued	



## ← Jobs in run #NetCore-Presenta...

Presentation

Jobs

Job

37s

Initialize job

1s

Checkout Presentation@master to s

1s

Restore Nuget Packages

10s

Build projects

14s

Run Tests

5s

Publish PresentationDemo

3s

Post-job: Checkout Presentation...

<1s

Finalize Job

<1s

Report build status

<1s

## Job

1 Pool: [Azure Pipelines](#)

2 Image: ubuntu-latest

3 Agent: Hosted Agent

4 Started: Just now

5 Duration: 37s

6

7 ► Job preparation parameters

42 ✅ [100% tests](#) passed

43 Job live console data:

44 Finishing: Job

## ✓ #NetCore-PresentationDemo-CI • added first project and .gitignore

Presentation

ⓘ This run is being retained as one of 3 recent runs by master (Branch).

Summary Tests Code Coverage

### Summary

1 Run(s) Completed ( 1 Passed, 0 Failed )

1

Total tests

+1



1 ● Passed  
0 ● Failed  
0 ● Others

100%

Pass percentage

904ms

Run duration ⓘ

0

Tests not reported

↑ 100%

↑ +904ms

Bug ▾ Link

Filter by test or run name



Hooray! There are no test failures.

Change the test outcome filter to view tests relevant to you.

# Pull Request Policies

# Pull Request Policies

The main branch should always be in a deployable state

Create pull request policies to protect the main branch

# Pull Request Policies

- Enforce reviewers
- Validate that the code builds before merging
- Run unit tests
- Enforce linking of work items (no work should be done without a work item)
- Deploy the application to a test environment
- Enforce merge strategies

## Project Settings

MicroserviceDemo

- General
  - Overview
  - Teams
  - Permissions
  - Notifications
  - Service hooks
  - Dashboards
- Boards
  - Project configuration
  - Team configuration
  - GitHub connections
- Pipelines
  - Agent pools
  - Parallel jobs
  - Settings
  - Test management
  - Release retention
  - Service connections
  - XAML build services
- Repos
  - Repositories**
  - Artifacts
    - Storage
  - Test
    - Retention

## All Repositories

Filter by keywords

- MicroserviceDemo

## MicroserviceDemo

Settings Policies Security Approvals and checks

### Repository Policies

- Off Commit author email validation  
Block pushes with a commit author email that does not match the following patterns.
- Off File path validation  
Block pushes from introducing file paths that match the following patterns.
- Off Case enforcement  
Avoid case-sensitivity conflicts by blocking pushes that change name casing on files, folders, branches, and tags. [Learn more](#)
- Off Reserved names  
Block pushes that introduce files, folders, or branch names that include platform reserved names or incompatible characters. [Learn more](#)
- Off Maximum path length  
Block pushes that introduce paths that exceed the specified length. [Learn more](#)
- Off Maximum file size  
Block pushes that contain new or updated files larger than this limit.

### Branch Policies

Protect important branch namespaces in this repository with pre-merge checks and policies

- master  Default  Compare

← MicroserviceDemo +

Filter by keywords

master Default Compare

master

Settings Policies Security Approvals and checks

### Branch Policies

Note: If any required policy is enabled, this branch cannot be deleted and changes must be made via pull request.

On **Require a minimum number of reviewers**  
Require approval from a specified number of reviewers on pull requests.

Minimum number of reviewers

Allow requestors to approve their own changes  
 Prohibit the most recent pusher from approving their own changes  
 Allow completion even if some reviewers vote to wait or reject  
 When new changes are pushed:  
 Require at least one approval on every iteration. [Learn more](#)  
 Require at least one approval on the last iteration  
 Reset all approval votes (does not reset votes to reject or wait)  
 Reset all code reviewer votes

On **Check for linked work items**  
Encourage traceability by checking for linked work items on pull requests.

Required  
Block pull requests from being completed unless they have at least one linked work item.

Optional  
Warn if there are no linked work items, but allow pull requests to be completed.

On **Check for comment resolution**  
Check to see that all comments have been resolved on pull requests.

Required  
Block pull requests from being completed while any comments are active.

Optional  
Warn if any comments are active, but allow pull requests to be completed.



On

### Limit merge types

Control branch history by limiting the available types of merge when pull requests are completed.

#### Allowed merge types:

##### Basic merge (no fast-forward)

Preserves history exactly as it happened during development

##### Rebase and fast-forward

Creates a linear history by replaying the source branch commits onto the target without a merge commit

##### Squash merge

Creates a linear history by condensing the source branch commits into a single new commit on the target branch

##### Rebase with merge commit

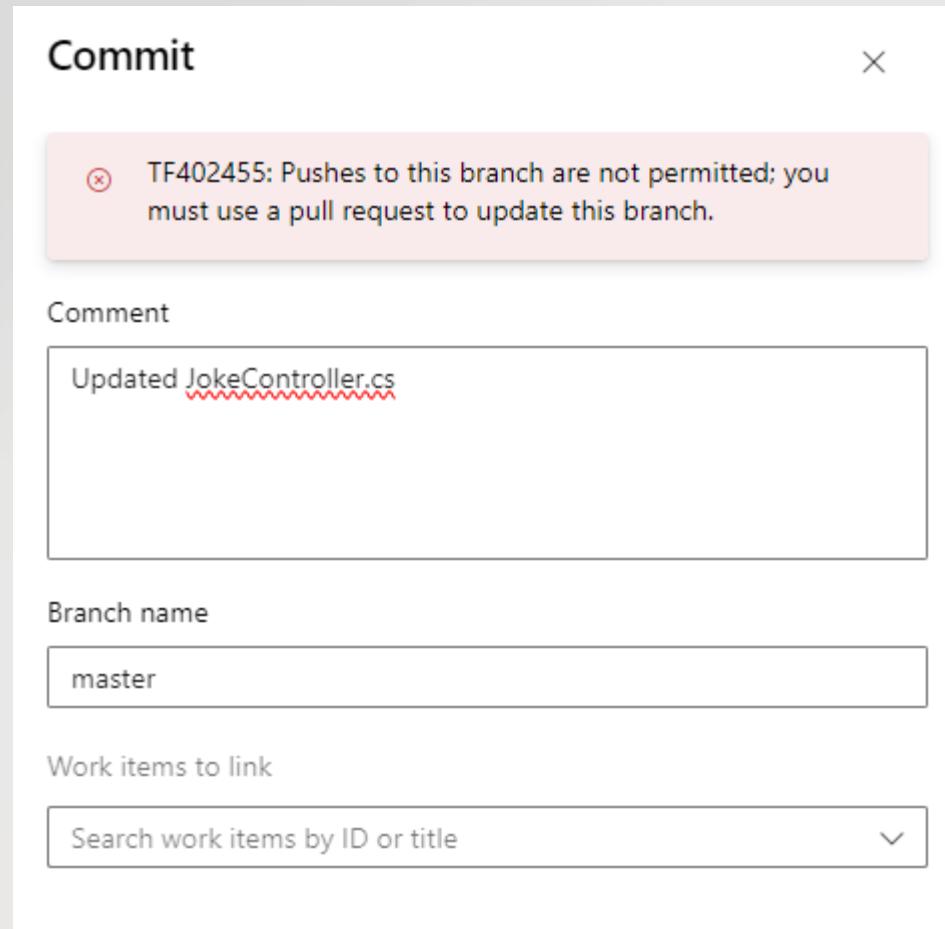
Creates a semi-linear history by replaying the source branch commits onto the target and then creating a merge commit

### Build Validation 1

Validate code by pre-merging and building pull request changes.

Enabled	Name ↑	Path filter	Trigger
<input type="checkbox"/> On	CustomerApi-Cl Required	*/CustomerApi/*	Automatic Expires after 12 hours

# Enforce Pull Requests



Wiki

# Wiki

Create documentation using markdown

Have your documentation in the same place as  
your code and work items

Seamlessly link work items and code commits

# Wiki

Some items are only available when the repository is checked out

- .attachments folder with all uploaded files
- .order file with order of files

# 3<sup>rd</sup> Party Repository Access

# 3<sup>rd</sup> Party Repository Access

3<sup>rd</sup> parties might need access to your repository

- GitOps to monitor branches and deploy changes
- Access a private NuGet feed
- Manage Agent Pools

# 3<sup>rd</sup> Party Repository Access

Allow access with Personal Access Token

- Token has an expire date
- Define permissions as needed
- Token value is only displayed after creation

## Create a new personal access token

X

Name

Organization

Expiration (UTC)

Scopes  
Authorize the scope of access associated with this token

Scopes  Full access  Custom defined

---

**Work Items**  
Work items, queries, backlogs, plans, and metadata

Read  Read & write  Read, write, & manage

---

**Code**  
Source code, repositories, pull requests, and notifications

Read  Read & write  Read, write, & manage  Full  Status

---

**Build**  
Artifacts, definitions, requests, queue a build, and update build properties

Read  Read & execute

---

**Release**  
Read, update, and delete releases, release pipelines, and stages

Read  Read, write, & execute  Read, write, execute, & manage

---

**Test Management**  
Read, create, and update test plans, cases, and results

Read  Read & write

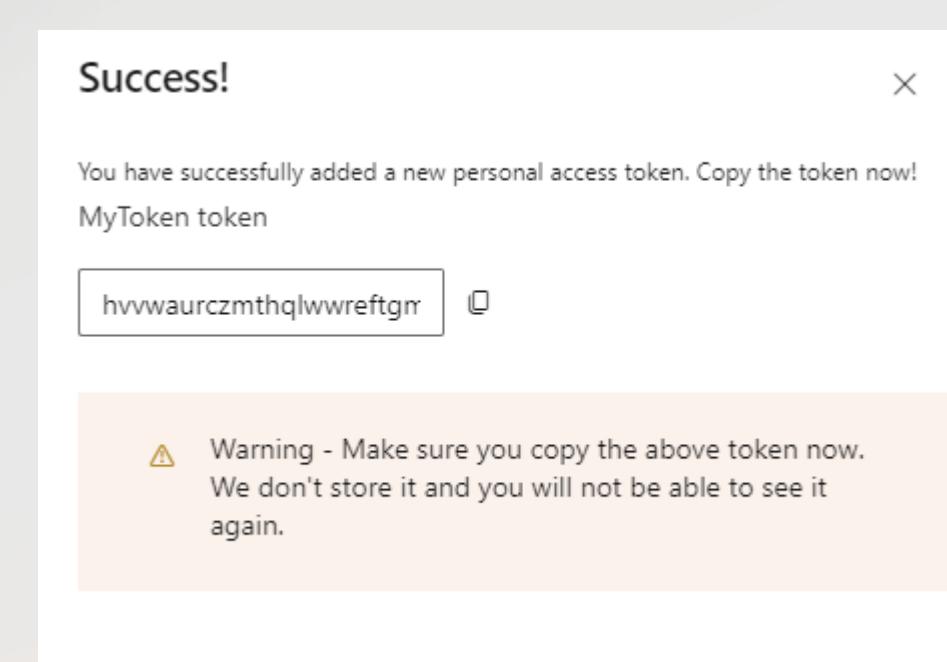
---

**Packaging**  
Create, read, update, and delete feeds and packages

Read  Read & write  Read, write, & manage

---

[Show all scopes \(29 more\)](#)



# 3<sup>rd</sup> Party Repository Access

Use public SSH keys

- 3<sup>rd</sup> party has the same access as the user who added the SSH key
- Use PATs whenever possible

# SSH Public Keys

Azure DevOps

User settings  
Wolfgang Ofner

Account  
Profile  
Time and Locale  
Permissions

Preferences  
Notifications  
Theme  
Usage

Security  
Personal access tokens  
SSH public keys  
Authorizations

**SSH Public Keys**

Connect to your Git repos through SSH Public Keys when you can't use the recommended Git Credential Managers or Personal Access Tokens to securely connect to a Git provider.

Compare the server fingerprint when you connect via Git to ensure you've connected to Azure DevOps. The fingerprint should match one of the following:

Server MD5 Fingerprint 97:70:33:82:fd:29:3a:73:39:af:6a:07:ad:f8:80:49 (RSA)  
Server SHA256 Fingerprint ohD8VZEXGWo6Ez8GSEJQ9WpafgLfsOfOtGGQCQo6Og (RSA)



**There are no SSH public keys added to your account on this organization.**

Visit the [documentation](#) to learn how to generate your first SSH Key and select New Key to add it to your account.

Add New SSH Key

Name\*  
MyKey

Public Key Data\*  
ssh-rsa  
AAAAAB3NzaC1yc2EAAAABIwAAAQEAkIOUpkDHrfHY17SbrmTlpNLTGK9Tjom/BWDSU  
GPI+nafzIHDTYw7hdI4yZ5ew18JH4JW9jbhUFrvQzM7xIELEVf4h9IFX5QVkbPppSwg0cdaa3  
Pbv7kOdJ/MTyBIWXFCR+HAo3FXRitBqxiX1nKhXpHAZsMcIq8V6RjsNAQwdsdMFvSIVK/7XA  
t3FaoJoAsncM1Q9x5+3V0Ww68/eIFmb1zuUFljQJKprrx88XypNDvjYNby6vw/Pb0rwert/En

One Project  
vs  
Multiple Projects

# Project Management

One project per product with several teams  
Microsoft recommendations

- Use single project due to lowered administrative effort
- Use multiple project if you want custom processes or different administrative policies
- Private vs. public projects

# GitHub

## RECOMMENDED

### Enterprise

Security, compliance,  
and flexible deployment

Starting at

\$ 21 USD per user/month

Start a free trial

Contact Sales

← Everything included in Team, plus...

- Access to GitHub Codespaces
  - Repository rules
  - Multiple reviewers in pull requests
  - Draft pull requests
  - Code owners
  - Required reviewers
  - Pages and Wikis
  - Environment deployment branches and secrets
  - 3,000 CI/CD minutes/month  
Free for public repositories
  - 2GB of Packages storage  
Free for public repositories
  - Web-based support
- 50,000 CI/CD minutes/month  
Free for public repositories
- 50GB of Packages storage  
Free for public repositories

#### EXCLUSIVE ADD-ON

- Premium support

### Team

Advanced collaboration for  
individuals and organizations

\$ 4 USD per user/month

Continue with Team ▾

- Unlimited public/private repositories
- Dependabot security and version updates
- 2,000 CI/CD minutes/month  
Free for public repositories
- 500MB of Packages storage  
Free for public repositories
- Issues & Projects
- Community support

#### FEATURED ADD-ONS

- GitHub Copilot Access
- GitHub Codespaces Access

### Free

The basics for individuals  
and organizations

\$ 0 USD per user/month

Create a free organization

#### FEATURED ADD-ONS

- GitHub Secret Protection
- GitHub Code Security

Feature	Organization	Enterprise
Purpose	A collaborative space for teams and projects.	A container for managing multiple organizations.
Billing	Billing is managed at the organization level.	Billing is centralized for all included organizations.
Management	Administrators manage repository access, team permissions, and organization-level settings.	Administrators can centrally enforce policies, manage user accounts, and audit activity across all organizations.
Security	Offers security features like required two-factor authentication (2FA) and team-based access controls.	Provides more advanced features such as IP allow lists, audit log streaming, and Enterprise Managed Users (EMUs).
Scale	Best suited for small to medium-sized businesses or large open-source projects.	Designed for large corporations with complex structures that need to manage multiple departments or business units.

# Azure DevOps vs GitHub

Feature	GitHub Enterprise	Azure DevOps
Primary Focus	Git-based code hosting, collaboration, and open-source ecosystem. A "developer-first" platform.	A comprehensive suite of tools for the entire software development lifecycle (SDLC).
Version Control	<b>Git</b> only. The de-facto standard for modern Git-based workflows.	<b>Git</b> and <b>Team Foundation Version Control (TFVC)</b> . TFVC support caters to legacy enterprise projects.
CI/CD	<b>GitHub Actions</b> : Flexible, event-driven, and has a vast marketplace of pre-built actions.	<b>Azure Pipelines</b> : Highly mature, robust, and offers advanced release management features like approval gates and multi-stage pipelines. Supports both YAML and a classic GUI editor.
Project Management	<b>GitHub Issues and Projects</b> : Simple, built-in tools for tracking work and creating Kanban-style boards.	<b>Azure Boards</b> : A powerful and highly customizable tool for agile, Scrum, and Kanban methodologies with advanced reporting and portfolio management.
Security	<b>GitHub Advanced Security</b> : Provides code scanning (CodeQL), secret scanning, and dependency management (Dependabot).	<b>Azure Pipelines</b> and <b>Microsoft Entra ID</b> : Strong identity management and role-based access controls. Security features are often integrated across the broader Azure ecosystem.
Third-Party Integration	Excellent for open-source and modern cloud-native tools. Integrates with thousands of GitHub Actions and external services.	Deeply integrated with the Microsoft ecosystem (Visual Studio, Microsoft Entra ID, Azure services) but also supports integrations with many third-party tools.

# Azure DevOps and GitHub

Wolfgang Ofner  
Senior Cloud Architect

