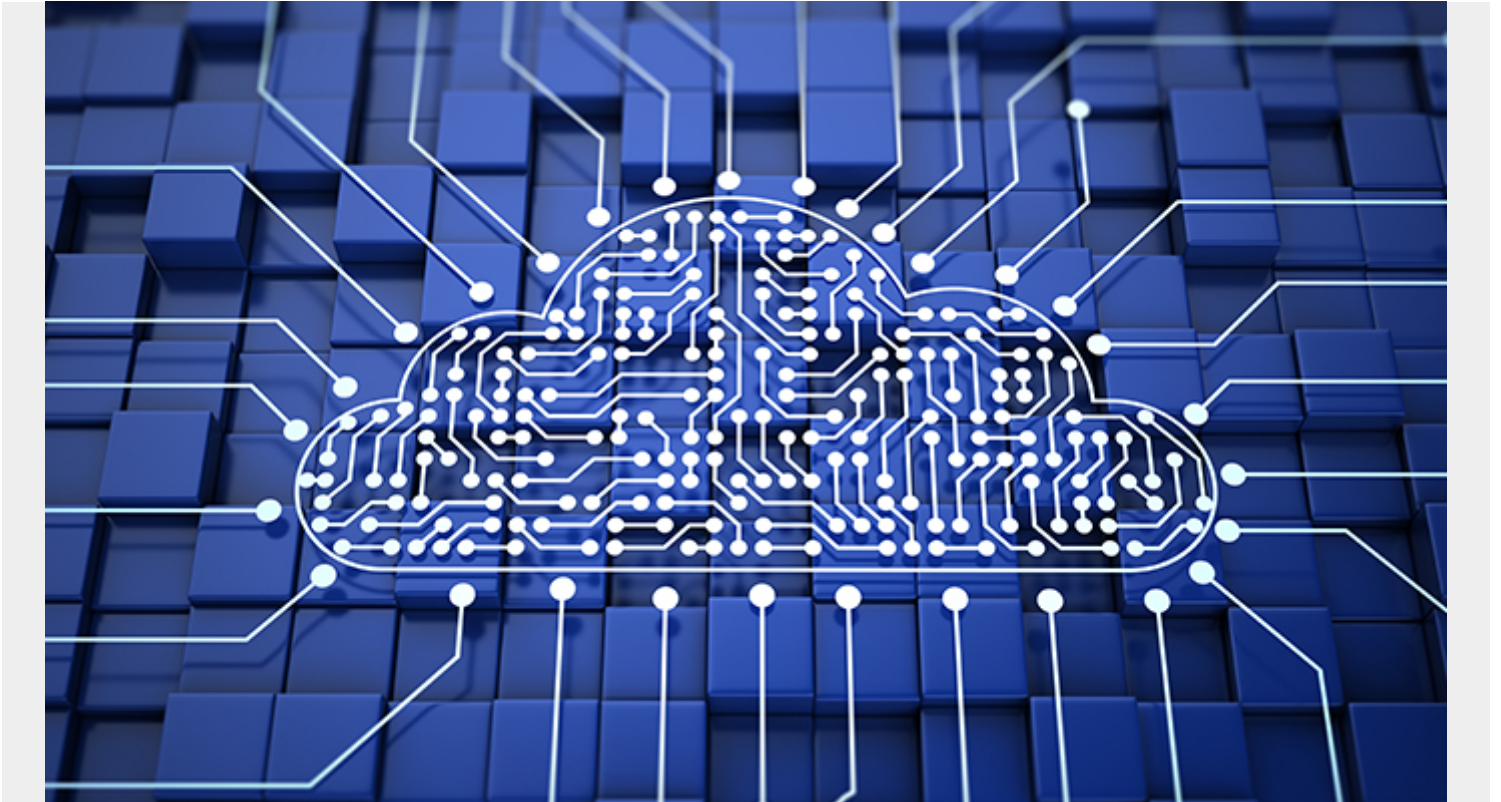


# AZURE DEVOPS: A BEGINNER'S GUIDE



Azure [DevOps](#) (formerly Visual Studio Team Services) is a hosted service providing development and collaboration tools. With a Free tier to get started and no need to run your own agents you can quickly get up and running with the many tools available.

- Git repositories for source control
- Build and Release pipelines for [CI/CD](#) automation
- Agile tools covering Kanban/scrum project methodologies
- Many pre-built deployment tasks/steps to cover the most common use cases and the ability to extend this with your own tasks.
- Hosted build/release agents with ability to additionally run your own
- Custom dashboards to report on build/release and agile metrics.
- Built in wiki

Azure DevOps has a lot of inbuilt functionality that allows teams to get up and running with managing their project and automating their workflows to increase productivity with a very short initial learning curve.

Some key terminology that is used within Azure DevOps that will be referenced in this article:

- **Organisation** - This is your Azure DevOps account name. Typically you will have one Azure DevOps account and manage multiple projects within.
- **Project** - You can split your organisation into multiple projects. This allows you to manage access control and split your code and pipelines from one and other.

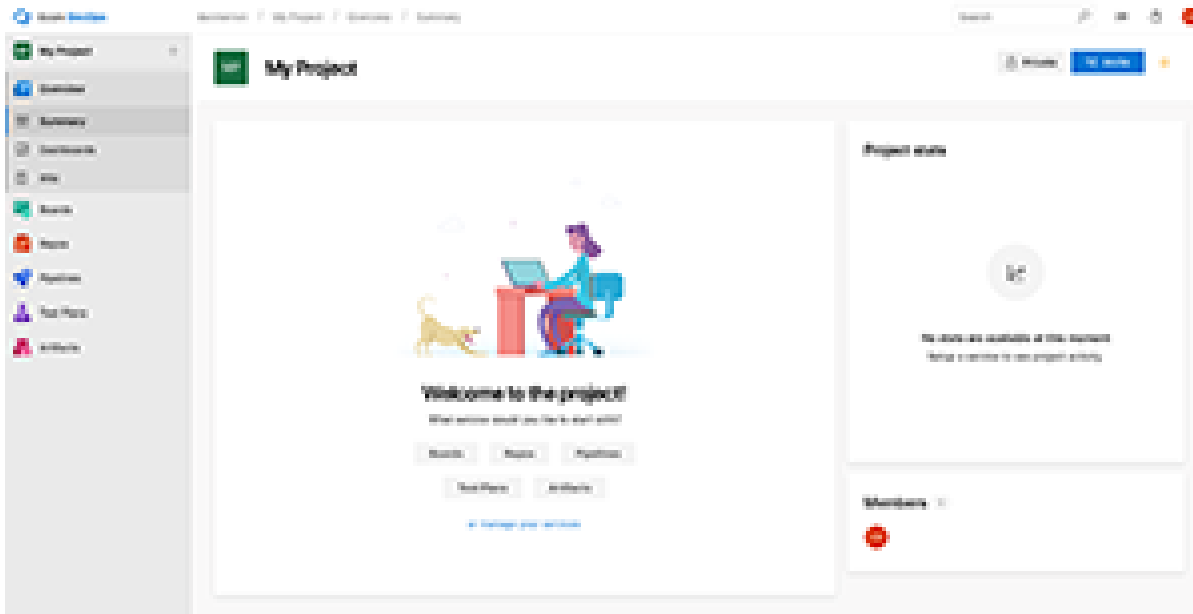
- **Repo** - Git or Team Foundation Services Source Control

## Getting Started

Signing up to use Azure DevOps is really simple and requires just a Microsoft Account and a few details about your organisation. You can sign up by visiting the Azure DevOps Services web page at <https://azure.microsoft.com/en-gb/services/devops/> and clicking 'Start for free'.

Once Registered you will have a dedicated organisation URL which will follow the notation: <https://<your org name>.visualstudio.com>

## Projects



Once logged into your Azure DevOps dashboard the first thing you will see is a prompt to create a project. Creating a project is as simple as providing a name/description & setting the publicity of the project. Additionally, you can choose between using Git and Team Foundation Source Control and your teams agile method.

## Boards

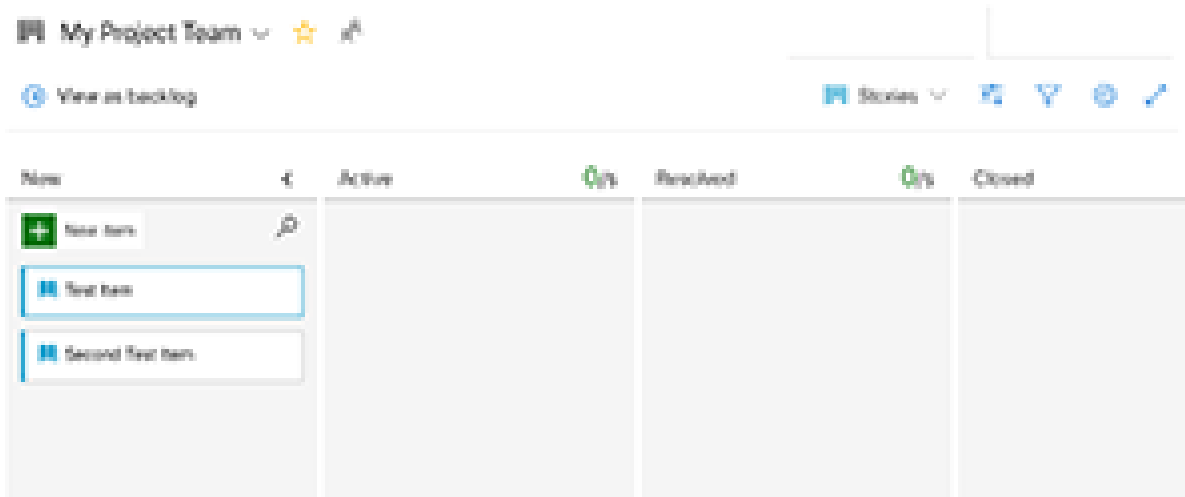
Boards within Azure DevOps mimic the functionality of Atlassian's Issue & Project Tracking software, Jira. Within Boards you can create kanban boards, story items, feature items, backlog items, sprints and query items to query your entire project items.

Essentially, Boards is your ultimate, built in agile project planning and management tool and has all of the functionality you would expect to find in a stand alone piece of software with the ability to link work items and tasks with your pipelines. If you've never used Agile methodologies before, reading the following blog posts will help you understand how Boards can help you manage your teams workloads and priorities.

- <https://www.dev.blogs.bmc.com/sprint-zero/>
- <https://www.dev.blogs.bmc.com/scaled-agile-framework-safe-explained/>
- <https://www.dev.blogs.bmc.com/agile-certifications/>

- <https://www.dev.blogs.bmc.com/accelerating-agile-delivery-with-itsm-integration/>
- <https://www.dev.blogs.bmc.com/devops-vs-agile-whats-the-difference-and-how-are-they-related/>

Weirdly, Boards has a boards subsection within itself. The main Boards Heading is the whole project planning/tracking piece within Azure DevOps. The second boards subheading is the KanBan boards created within your project. You can create multiple kanban boards for different teams and projects and manage tasks within them.



## Work Items

Work Items is a section within boards to manage, document, assign and prioritise bugs, epics, features, issues, tasks, test cases and user stories for later inclusion on kanban boards and/or sprints.

## Repos

The Repo's functionality differs between choosing Git or Team Foundation Source Control as your Git version when creating a project. The most common choice is Git and therefore this article will focus on that. Within repo's you can create or import repositories and manage them using your standard git GUI or git CLI. All of the functionality you would expect from Git is there with the added benefit of linking commits and pull requests to Work Items and CI.

Additionally Azure DevOps boasts an inbuilt IDE for quickly editing code within your project files, found under the Files subheading.

## Work Items

Work Items interface showing a list of items with columns for ID, Title, Assigned To, and State.

ID	Title	Assigned To	State
4	My first bug	Unassigned	New
3	My first bug	Unassigned	New
2	Second Test Item	Unassigned	New
1	Test Item	Unassigned	New

## Pipelines

Pipelines within Azure DevOps is the CI/CD tool that allows the building, testing and deployment of code using DevOps practices.

## Build Pipelines

Build Pipelines provide you with the functionality required to build and compile your application code using a host of [pre defined build tasks](#). The ability to link your Build Pipelines with either GitHub or Azure DevOps Git Repositories means you use CI to automatically trigger your builds and report their status on every commit, pull request or merge. In previous versions of Azure DevOps (VSTS) build pipelines could be managed using the web GUI, however in the current version the only option to create and manage build pipelines is by using YAML pipelines.

## Release Pipelines

Release pipelines are very similar to build pipelines however are for deploying your applications to your servers. Release pipelines can still be created using the GUI (and exported to YAML if required) however YAML release pipelines are soon to be implemented in Q3 2018.

<https://docs.microsoft.com/en-us/azure/devops/release-notes/>. Creating release pipelines is a very intuitive, user friendly drag and drop process and within a few minutes you can be deploying your applications to your servers with very little, if any, coding.

A list of built in Build, Test and Release tasks can be found in the official Azure DevOps Documentation which can be used within your pipelines

<https://docs.microsoft.com/en-us/azure/devops/pipelines/tasks/?view=azure-devops&viewFallbackFrom=vsts>. Additionally, there is a Task Marketplace where you can find and install many additional tasks that are not pre bundled. <https://marketplace.visualstudio.com/azuredevops>

## Pricing

Given there is a lot of open source tools that can be combined to deliver the functionality that Azure DevOps does, it's important to know that you can get started with Azure DevOps for free if you're working on an open source project or if your team is smaller than 5. Aside from that, Azure DevOps

starts at £22.35 +VAT per month for up to 10 Users. Concurrency using the Free Hosted Build Agents is where the free tier limitation begins - but you can solve that by running your own self hosted build agents.

To summarise, Azure DevOps is an all in one Agile focussed project tracking and planning tool coupled with Developer and DevOps tools for writing, building and deploying code to your server estate. It has a nice GUI and excellent integrations (e.g. Slack and email notifications). Getting started is very straight forwards - so much so that you can be managing tasks and building/deploying applications relatively quickly and pain free.

Microsoft seem to be putting a lot of focus into Azure DevOps at the moment and features are being improved/added frequently so there will be lots of nice things to come.