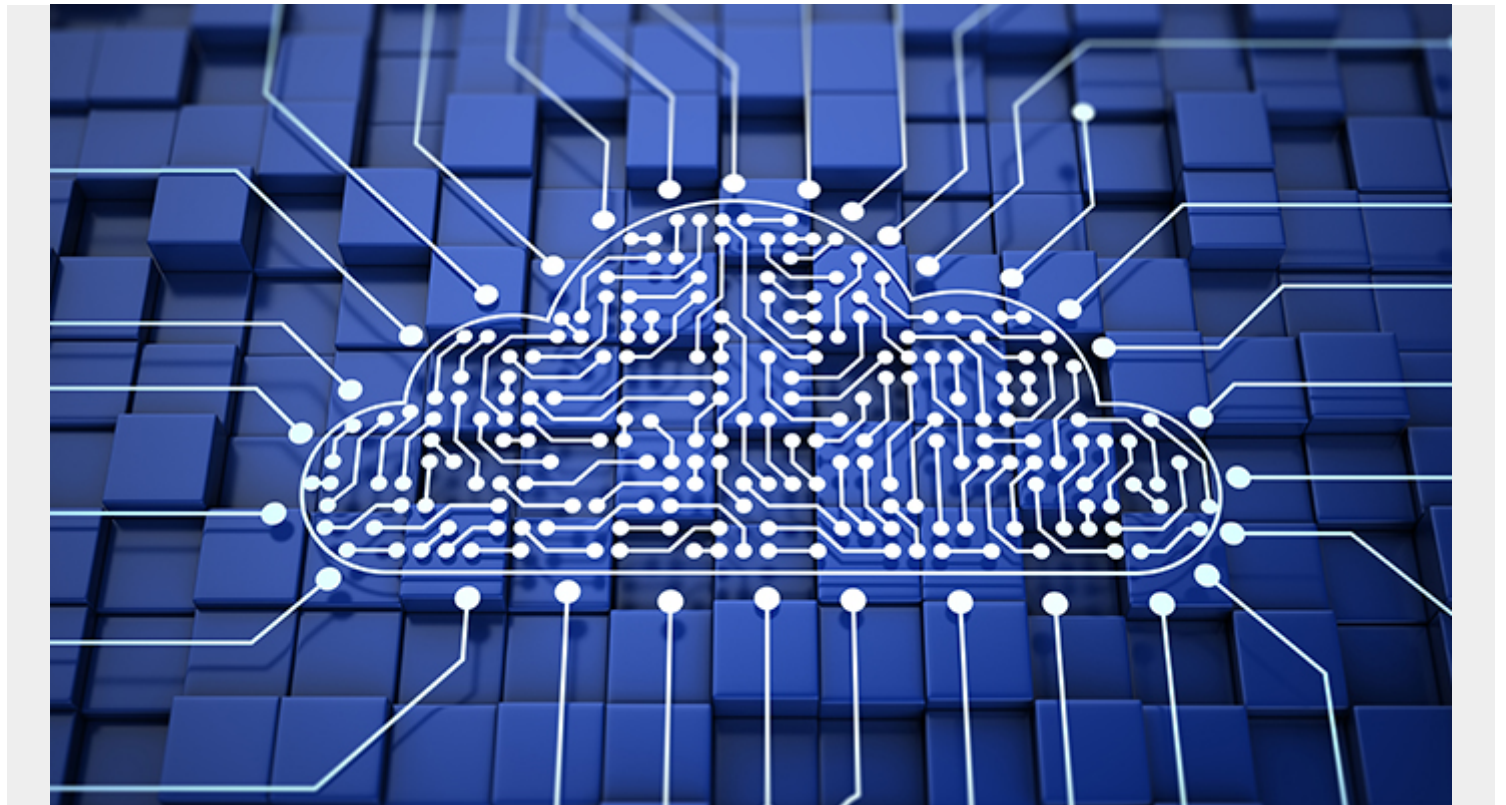


WHAT IS AZURE DEVOPS? A BEGINNER'S GUIDE

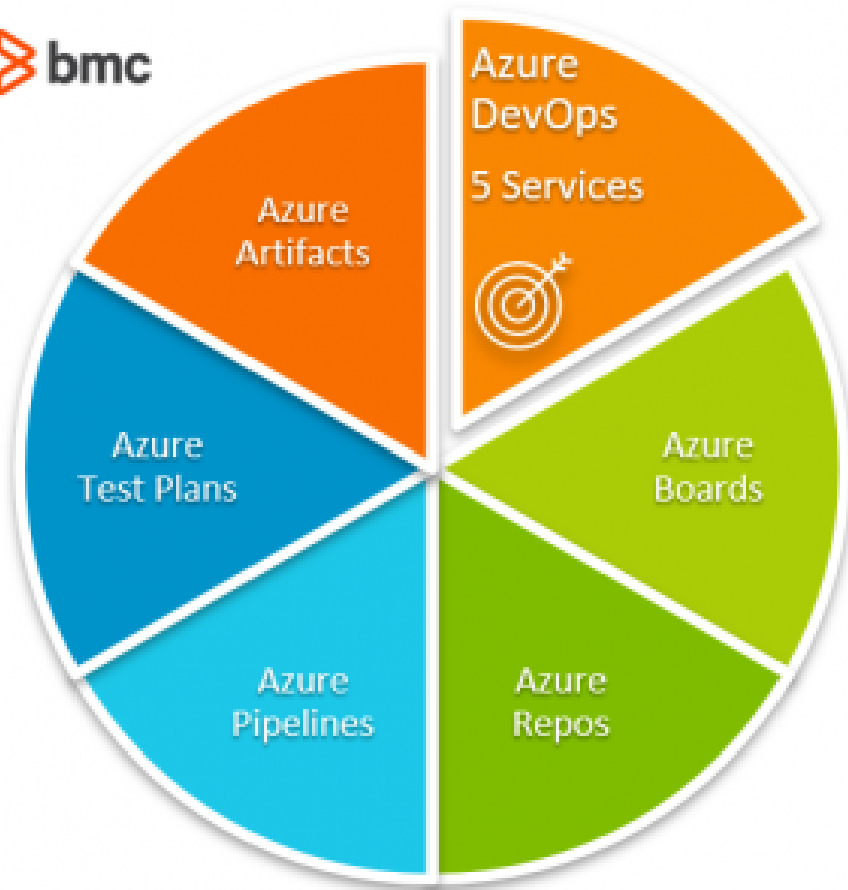


DevOps has paved the way for faster and more agile software development processes by unifying teams, processes, and technologies to create an ever-evolving [software development lifecycle](#) (SDLC). This has led to more robust and efficient SDLCs, now capable of handling any user request, market demand, or technological issue.

A range of tools is available in the market to facilitate DevOps, such as CI/CD tools, [version control systems](#), artifact repositories, [IaC tools](#), and monitoring tools. With the increased demand for cloud-based technologies, DevOps tools have also transitioned to cloud offerings. These cloud offerings can be used by teams spread across the world with nearly unlimited scalability and efficiency.

In this article, we will explore such a cloud-based DevOps service offered by Microsoft called Azure DevOps.

(Explore our [DevOps Guide](#), a series of articles & tutorials.)



What is Azure DevOps used for?

Azure DevOps is a service offered by Microsoft based on the Azure [cloud computing platform](#) that provides a complete set of tools to [manage software development projects](#). It consists of:

- Five key services
- An extensive marketplace that contains extensions to further extend the Azure DevOps platform and integrate with third-party services

Azure DevOps core services

Core Azure DevOps services include:

1. Azure Boards
2. Azure Pipeline
3. Azure Repos
4. Azure Test Plans
5. Azure Artifacts

Azure DevOps comes in two variants:

- The cloud-based Azure DevOps service
- The Azure DevOps Server

The Azure DevOps Server, previously known as the Team Foundation Server (TFS), is a DevOps server solution that is targeted for on-premise deployments. It consists of all the tools available in the cloud-based Azure DevOps service to power any DevOps pipeline.

This also server offers a free variant called Azure DevOps Server Express, aimed at [individual developers](#) and small teams of up to five team members. It can be installed in any environment.

Azure guarantees [99.9% availability](#) for all the paid DevOps services including paid user-based extensions. Moreover, it provides 99.9% availability to execute load testing and build and deploy operations in paid Azure Test Plans (Load Testing Service) and Azure Pipelines.

Azure DevOps pricing

The cost will be one of the primary concerns when considering any DevOps solution.

The cloud-based Azure DevOps services come as both free and paid options. Additionally, the service offerings are provided in two varieties as individual services and complete service bundles.



Azure DevOps: Comparing Services & Pricing

Service	Service Type	License Type	Free	Paid
Azure Pipelines	Individual	Consumption-based	1 Free Microsoft Hosted CI/CD (1800 minutes) 1 Free self-hosted CI/CD (unlimited minutes)	\$40 per Microsoft hosted CI/CD (unlimited minutes) \$15 per self-hosted CI/CD (unlimited minutes)
Azure Artifacts	Individual	Consumption-based	2 GB free storage	\$2 per extra GB (Price per GB decreases with consumption)
Azure DevOps Basic Plan Pipelines Boards Repos Artifacts	Service Bundle	User-based	Up to 5 free users (subject to limitations)	\$6 per additional user
Azure DevOps Basic + Test Plan Pipelines Boards Repos Artifacts Test Plans	Service Bundle	User-based		\$52 per user (unlimited usage)

In addition to the above, there are special pricing options for open-source projects and Visual Studio subscribers to get free access to the Azure DevOps services depending on the subscription level.

(Visit the [Azure DevOps pricing page](#) for details & up-to-date pricing.)

Azure DevOps registration

Registering for Azure DevOps is a simple and straightforward process that requires only a Microsoft account. Simply visit this page and click on “Start for free.”

When registering, you will need to provide some additional information such as organization name, project name, version control type (repo), etc.

- **Organization** refers to the Azure DevOps account name. The organization can contain multiple projects.
- **Projects** allow users to separate projects, control access, and split the code, tests, and pipelines to keep them within the assigned projects. A project can be either public or private, with Git or Team Foundation server as the version controlling system. Additionally, projects can be configured with a work item process like [Agile or Scrum](#) that will be used in Azure Boards to manage the project.

Once the registration is complete, you will gain a dedicated organization URL in the following notation:

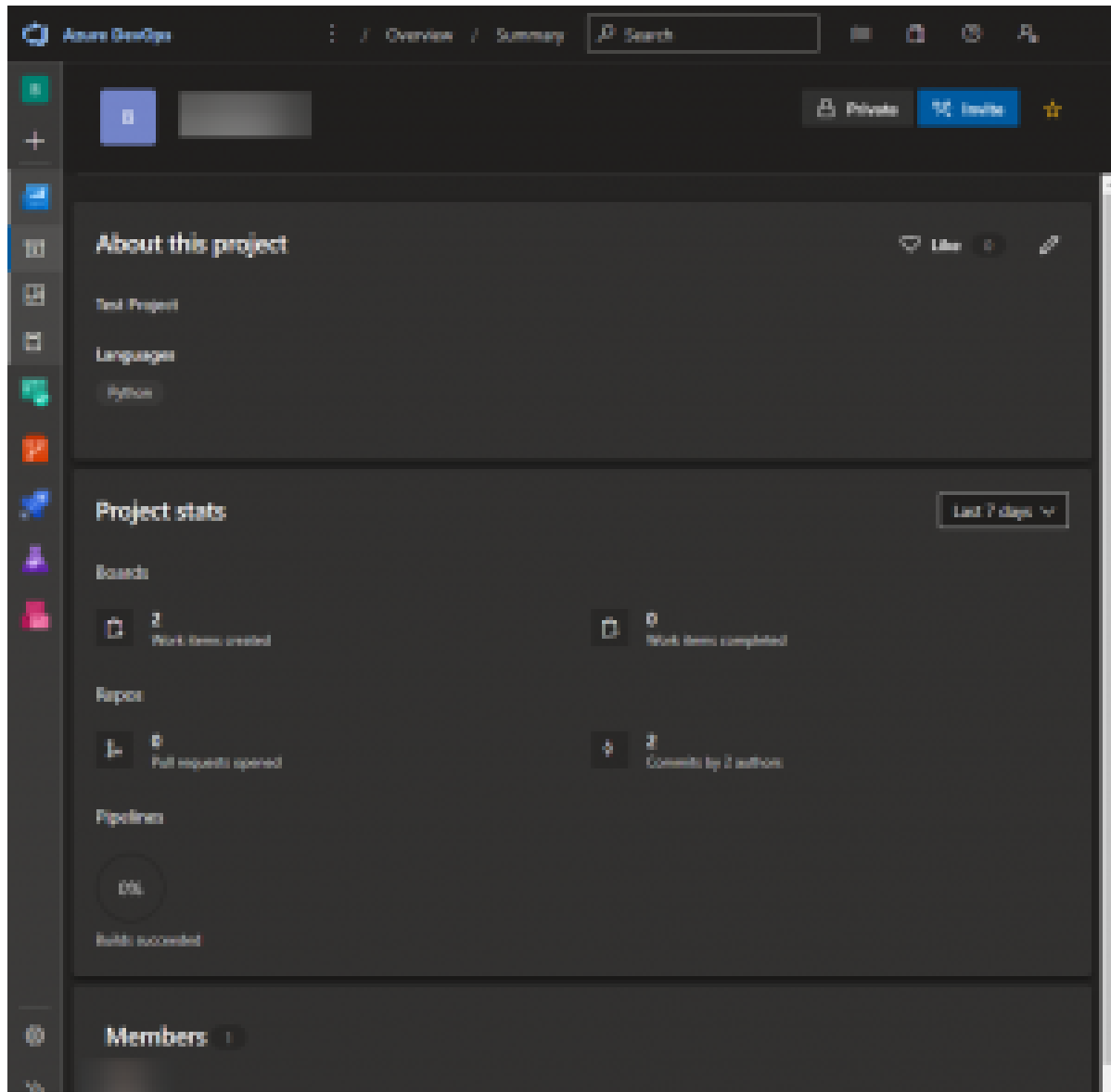
`https://<organization name>.visualstudio.com`

Users can manage all their projects and use the DevOps services by visiting this URL.

Azure DevOps Services

Azure DevOps consists of five services—which we’ll explore in this section. All these services can be grouped under individual projects so that users can have proper isolation between different projects using different technologies and catering to different needs.

Project summary view:



Azure Boards

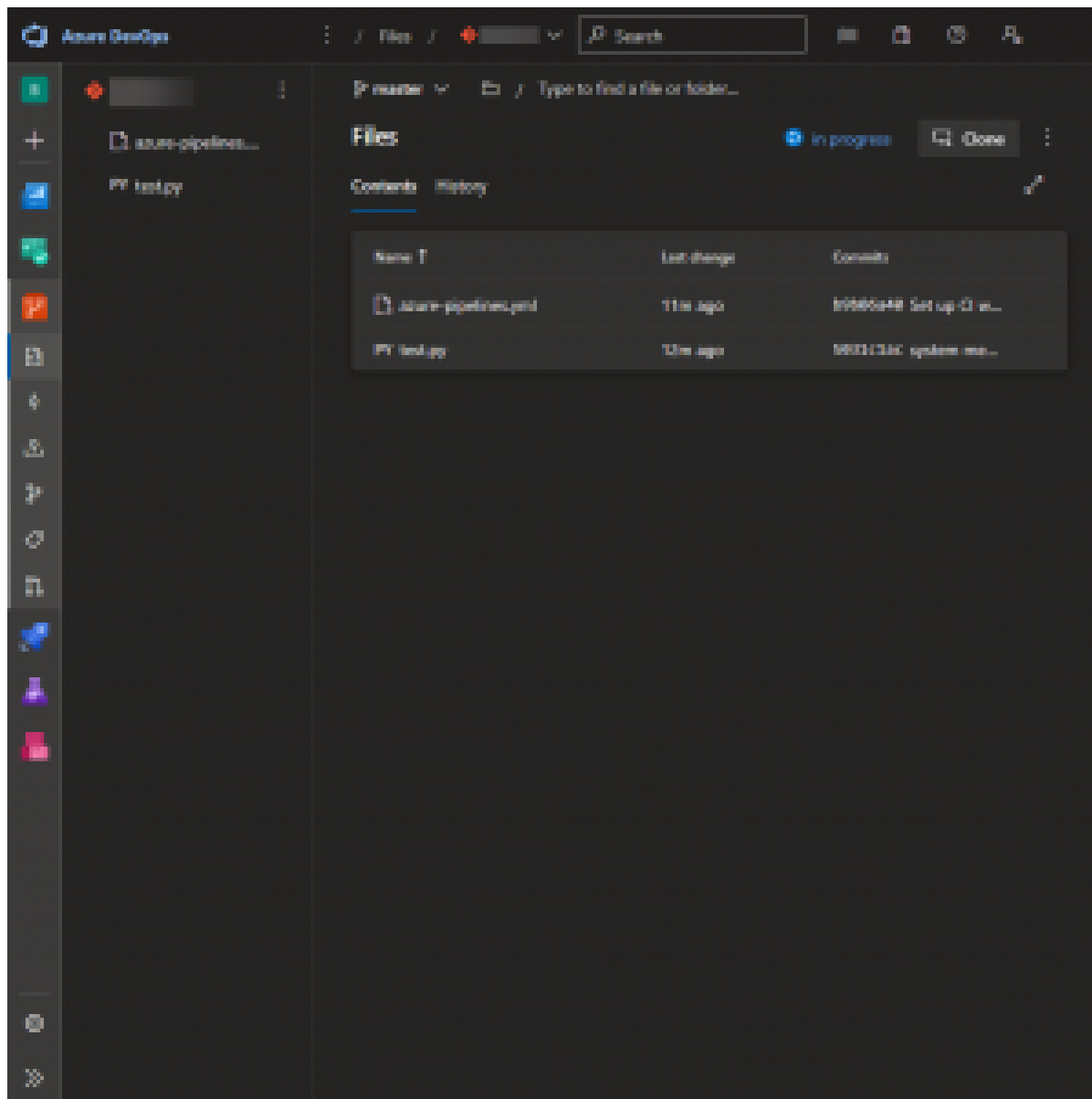
The Boards service in Azure DevOps is the management hub of the project.

Boards can be used to plan, track, and collaborate between team members. With Azure, the Boards team can create Work items, Kanban boards, backlogs, dashboards, and custom reports to track all aspects of the project.

You can also customize boards to suit the exact workflow requirements and gain meaningful insights through built-in reporting and monitoring tools. Additionally, Azure Boards comes with first-party integrations with services like Microsoft Teams and Slack, which enables efficient [ChatOps](#).

Azure Repos

The Azure Repos are code repositories that enable users to manage their codebases. These are private and cloud-based repositories that support both Git and TFVC version control systems.



Azure Repos can

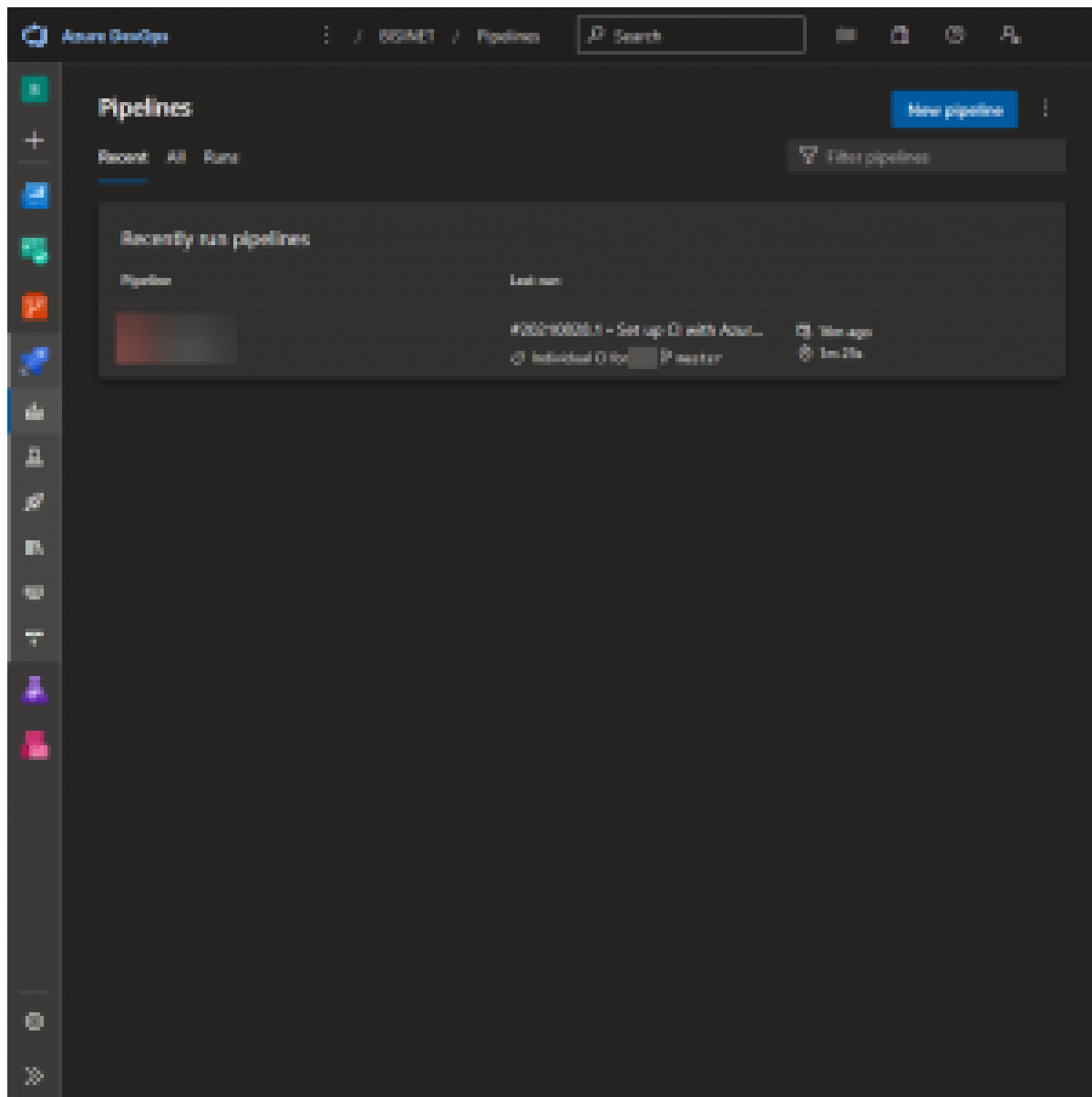
support projects of any scale, from individual hobby projects to enterprise developments. They also consist of the following features:

- Support for any Git client (IDE, Text Editor, CLI)
- Semantic code search
- Collaboration tools to interact with other team members
- Direct integration with CI/CD tools
- Branch Policies to enforce [code quality standards](#)

Platform-agnostic services like Azure allows repo users to use any IDE or tool they are familiar with to interact with the Azure Repos in any operating system.

Azure Pipelines

Pipelines are the [CI/CD](#) tool that facilitates automated building, [testing](#), and deployment. Azure Pipelines supports any programming language or platform which enables users to create pipelines that support Windows, Linux, and macOS using cloud-hosted agents.



These pipelines are easily extensible through the extensions available in the marketplace. Besides, they support advanced workflows that can be used to facilitate:

- Multi-phase builds
- Test integrations
- Custom reporting functions

On top of that, Azure Pipelines provide [native container support](#), enabling them to push containers to container registries from the pipeline directly. The pipelines offer flexibility to deploy to multiple environments from [Kubernetes clusters](#) to serverless functions and even deploy to [other cloud providers](#) such as AWS or GCP.

Azure Test Plans

Test Plans is the Azure DevOps service that allows users to integrate a cloud-based testing platform to manage all the [testing requirements](#) such as:

- Planned manual testing
- User acceptance testing (UAT)
- Exploratory testing

- Gathering feedback from stakeholders

Azure Test Plans allow users to create test plans and execute test cases within a pipeline. This can be combined with Azure Boards to create a test that can be executed from the Kanban boards and plan and author tests collaboratively.

Test Plans support creating UAT plans for user acceptance testing and assign users from the DevOps platforms. It also supports the Test and Feedback browser extension to easily enable exploratory testing for interested parties without utilizing third-party tools. Furthermore, Test Plans enable users to test on any platform while having end-to-end traceability and powerful data gathering tools to diagnose any remedy identified issues.

It is the only service in Azure DevOps with no free tier due to its rich toolset that is only accessible for commercial users.

Azure Artifacts

This is the artifact library service by Azure DevOps that can be used to create, store, and share packages (development artifacts). Azure Artifacts enable users to integrate fully featured package management functionality to CI/CD pipelines.

Moreover, Azure Artifacts enable users to manage all package types like npm, Maven, etc., and keep them organized in a central library scoped only to the specific project.

Azure Cloud Services

Azure DevOps is one of the leading cloud-based DevOps services that offer a robust and feature-rich toolset to create and manage a complete DevOps process. It enables users to:

1. Cater to any DevOps need regardless of the [programming language](#), technology, or the targeted platform.
2. Deploy anywhere from containers to third-party clouds.

Azure DevOps facilitates all these with unparalleled scalability and availability without the hassle of maintaining specific software to carry out separate DevOps tasks.

Azure DevOps vs. GitHub

Should you use Azure DevOps instead of GitHub? The [differences between GitHub and Azure DevOps](#) mean each offers something distinct to your situation. The choice depends on your situation and the capabilities and benefits each one brings. In considering Azure DevOps and GitHub, both support Git and collaborative software development in both public and private modes.

Azure DevOps is an enterprise-level software development management tool with an integrated build server and comprehensive tools that support project creation, software development and testing, and ongoing management and maintenance. It also offers advanced security and compliance features, along with governance capabilities.

It contrasts with the lightweight, small-team-friendly, open-source option that is GitHub. The open-source nature of GitHub means it has broad community support, built-in social features, is developer-friendly, and has a large and active user base. The community around Azure DevOps is

smaller, mostly Microsoft-focused enterprise users.

Azure DevOps vs. Jira

The Jira software development tool, available as SaaS or on-premises, is another option for those evaluating the Azure DevOps development platform. [Jira](#) shares some features with Azure DevOps, such as extensibility, Scrum and Kanban boards, customizable workflows, roadmaps for project management with dashboards and reporting, version control automation and orchestration, and repository management.

Jira is different from Azure DevOps in that its core strength is supporting Agile project management, cross-team collaboration, and tracking issues across multiple development platforms. It has different versions for software, business, and IT teams, with an available mobile app. It offers advanced search for finding code issues and best-practices playbooks.

Azure DevOps is a complete solution with built-in CI/CD, along with Git and TFVC support, code repositories, and testing tools. It includes Agile tools and deep DevOps integration, including integrated Azure Pipelines. With Jira, you need third-party CI/CD and testing tools, as well as those for version control code repositories.

In choosing between Jira and Azure DevOps, you will need to look at functionality, integration, performance, and available support.

Related reading

- [BMC DevOps Blog](#)
- [AWS vs Azure vs GCP: Comparing The Big 3 Cloud Platforms](#)
- [GitHub, GitLab, Bitbucket & Azure DevOps: What's The Difference?](#)
- [Azure Compliance: 3 Keys for Getting Started](#)
- [Azure Certifications: An Introduction](#)
- [SRE vs DevOps: What's The Difference?](#)