Azure DevOps

Index:

INTRODUCTION

Azure devops is a cloud computing devops tool provided by **microsoft** and is a collaborative tool among developers, operational staff and management staff for better quality and faster deployment.

- Continuous Integration and Continuous Delivery are important basic processes used by Azure Devops. They automatically and continuously update and deploy the product.

Azure Services Vs Azure Server

There are two ways through which we can use these services of azure devops. A developer can use these service on both cloud or on-premises.

1. Azure Services: is used on cloud.
2. Azure Servers: are used to access these tools on-premise.

Azure Services

Azure service is a **cloud service** tool to run developer operations across domains.

Some of the features of azure services are

1.Plan and track work

2.Build and Deploy the work

3.Ensure quality of work by getting feedback

4.Integrate and extend service to the work

Azure Server

1) Azure Servers is a tool hosted **on-premises.**

2) Azure Servers is built on **SQL server**.

3) On premises is generally preferred when users need their data to stay within their **network.**

4) On premises is also preferred when users want to access services which cannot be provided by Azure services like Integrating SQL server reporting.

Fundamental features like authentication, data protection, security are also different between Azure service and Azure server.

- For Authentication through devop services, you connect over the public internet You either authenticate with Microsoft account credentials or with **Azure AD credentials**, depending on your organization setup.
- It is **recommended by microsoft** that you configure your organizations to use Azure AD rather than Microsoft accounts.
- With Azure DevOps Server, you connect to an intranet server. You authenticate with Windows Authentication and your Active Directory (AD) domain credentials. This process is transparent and you never see any kind of sign-in experience.

Steps to Start Azure Services

1) For accessing azure services, register an azure account here: Create Azure Account.

2) Azure devops services can be launched through azure portal or azure demo generator.

3) To launch services through portal go to Azure Portal and in search bar search for "my azure devops organization". Then follow these steps

Create an organization->create a project->configure project name, description and visibility(Private and public) of the project.

4) To launch services through demo generator go to Azure Demo Generator and login though an azure account and accept all the actions.

5) To Add/Invite team members to your organization

Click on invite -> Use email ID or GitHub username->choose the project you intend to add users on.

Azure Board

Azure Boards helps teams to manage and discuss tasks through dashboards like

- KanBan board
- Backlogs
- Team dashboard
- Integrated Reporting

Azure Board provides certain set of tools for this like.

1) Work Items

- It tracks features and requirements being developed, code defects, bugs and other things.
- Work Items are similar to **ISSUES** in Git hub.
- Each work item is assigned with an **identifier** unique to project.
- Work items provided are based on project templates choosen while creating project like Basic, Agile, Scrum, CMMI.
- Every work item has a **type** like

  Epic: Work items that track significant features or requirements.

  Tasks: Work items that track user stories,bugs and other small items of work.

  Issues: Work items that track even smaller works like time in days or hours.

2) Board

Boards and Backlogs are used for collaborative interaction. Azure services mainly uses **kanban board.**

- You can add and update the status of work using the Kanban board, assign work to team members, tag with labels to support queries, and filter data.
- You can also add and update the status of work from New, Active, Resolved, and Closed using the Kanban board for an agile process, and add tasks as child items to user stories.

3) Sprint

It forecasts work based on effort estimate and determine how much work can be done in a sprint.

4) Queries

Queries are filtered list of work items based on criteria in queries. Like find out groups of work items that are something in common.

Azure Repos

Azure repos is a set of version control tools which provides unlimited private repositories to manage project code. It also tracks changes done to code.

- As you edit code VCS takes snapshots of the file and saves it for latter references.

There are two types of repos.

- Pull Request is a feature used for reviewing and merging of code into single process.

1) GIT

It is a **distributed version control** system and most widely used. In Git, the local copy of code includes the complete version control repository. In Azure Repos, Git is standard Git, and the commonly used clients and tools are vscode, command line, xcode, etc.

2) Team Foundation Version Control (TFCV)

TFCV is a **centralized version control.** Team Foundation Version Control TFVC is a centralized version control system. In TFVC, team members have only one version of each file on their dev machines. Historical data is maintained only on the server. Branches are path-based and created on the server. The commonly used IDE's or clients are visual studio, eclipse, xcode, etc.

Azure Pipelines

1) It combines continuous integration and continuous deployment to build, test and deploy project code to target users.

2) we can use pipelines through YAML file as code and classic editor on web portal.

3) In both methods, you need to configure azure pipelines to use Git Repo and push the code the code to version control repository. This action activates a default triggers the pipeline.

YAML File Vs Classic Editor

There are two platforms where we can use pipelines through YAML file as code and classic editor on web portal.

- In both methods, you need to configure azure pipelines to use Git Repo and push the code the code to version control repository. This action activates a default triggers the pipeline.

YAML file:

In here, the pipeline is versioned with code and the main structure is based on code. This build structure can be modified by altering **azure-pipelines.yml file.**

Classic Editor:

It is used to create a pipeline without using yaml file. Here, a build and Release pipelines are created to build, test the code and consume, deploy the artifacts to the target.

Azure Test Plans

As the name suggests, Azure test plan provides tools and features for different types of testing like

1) Planned Manual Testing:

Testing done by organizing test suites and test plans by designated testers.

2) User Acceptance testing:

Testing done by designated **user acceptance testers** to verify value delivered to meet customer requirements by reusing artifacts by created by developers.

3) Exploratory Testing:

Developing teams like developers, testers and product owners explore the software without using test plans or test suites.

4) **Stakeholders Feedback:** Testing done by user outside the development team such as marketing and sales team.

Azure Test Plans include three types of test management artifacts:

Test plans: Used to group together test suites and individual test cases. This includes static test suites, requirement-based suites, and query-based suites.

Test Suites: Used to group test cases into separate testing scenarios within a single test plan. This makes it easier to view the completed scenarios.

Test Cases: Used to validate individual parts of a code or app deployment to ensure that it performs accurately, is error-free, and meets business and customer requirements.

- **Testing from Kanban Board**: To test from a Kanban board: Open the Kanban board-> Select the Project, Team, and Boards->Open the menu for the work item->click + Add test. Add the inline tests. Click Run test. Update the status of the test (Pass test or Fail test).

Azure Artifacts

Azure Artifacts enables to create package feeds like Maven, NPM, NuGet from public and privates sources, with teams of any size.

1) With a single click, you can add fully integrated packages to CI/CD pipeline.
2) It also has a feature to ensure **immutability** and performance for the components you create or consume.
4) **Immutability: O**nce a particular package is published, the version number is permanently reserved. No other package can be uploaded with same number.
5) Immutability can only be override by creating a new feed.