**AZURE DEVOPS**

* Software as a service (SaaS) platform from Microsoft
* Platform to implement all your DevOps Process
* Azure DevOps portal: Https://dev.azure.com
* Organization:
* Organization is a mechanism for organizing and connecting Groups of related projects.
* Each Projects must be in an organization
* Project:
* A project is a place where actual complete product development is done by team.
* All the data related to software is stored under a project.
* 2 types of projects: Public and private
* Team:
* A Team is a group of people who are responsible for the software Development
* Azure Boards

Azure Boards helps in,

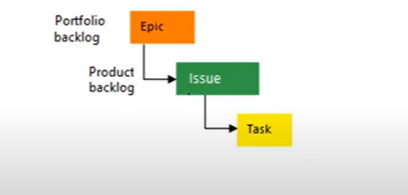
* Track the work with Kanban Board
* Work with Backlog
* Team Dashboard
* Reporting
* Azure Board provide Drag and drop feature to update Kanban Board
* It provides a Clear picture of work Done/Dong by team member

Work Item:

* It is a unit (small or large) of work has several characteristics and is a part of product development.

1. Work items with Basic process: 1) Epic, 2) Issue, 3) Task

* Choose Basic when your team wants the simplest model that uses Issues ,Tasks and Epics to track work



Workflow:

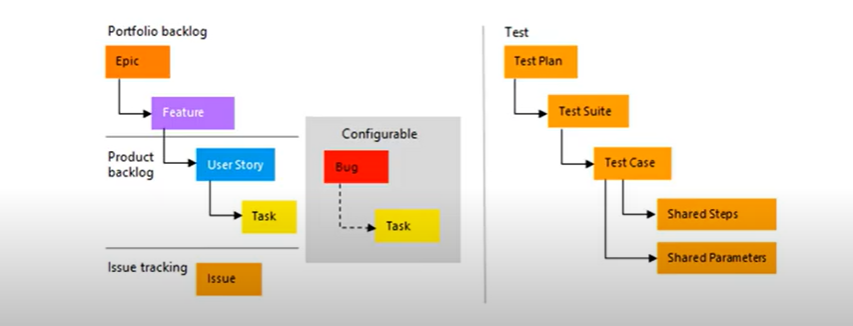
* Workflow is the process of updating work item progress
* TODO – InProgress – Done

Backlog:

* Backlog is a collection of work items which will be used for future development
* 2 types:

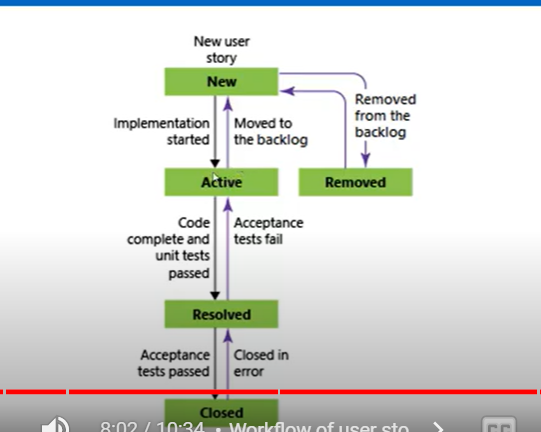
1. Product backlog: it is an ordered list of everything that is known to be needed in the product
2. Sprint backlog: Collection of work items which are in TODO states
3. Work items with Agile process: 1) Bug 2)Epic 3)Feature 4)Issue 5)Task 6)Testcase 7)User story .

* Choose Agile when your teams uses agile planning methodologies, Including scrum, and tracks development and test activities separately



1. Bug: Something which is missed or implemented wrong way
2. Epic: An epic represents a business initiative to be Accomplished
3. Feature: A Feature represents a shippable component of a software
4. Issue: Any other custom type
5. Task: Smallest unit of work
6. Test Case: Test case for a feature
7. User Story: Implementation of new work

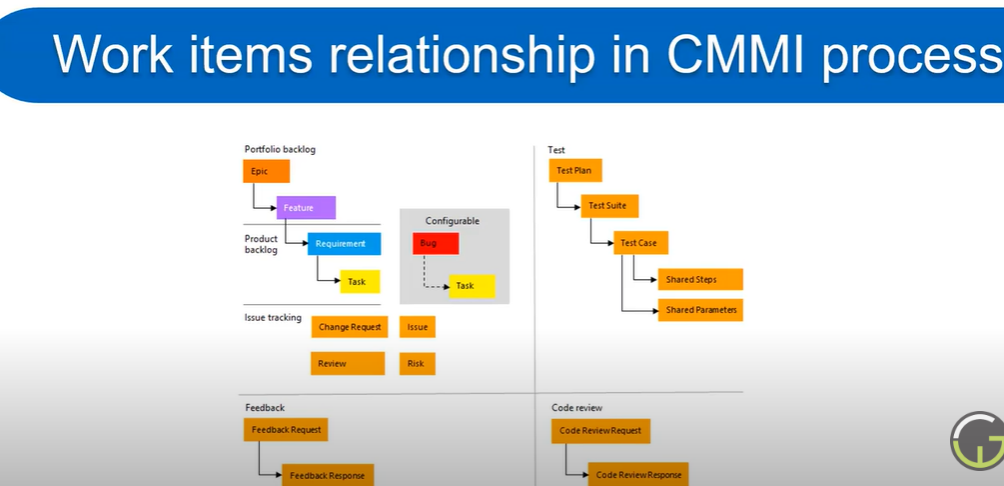
Workflow of user story:

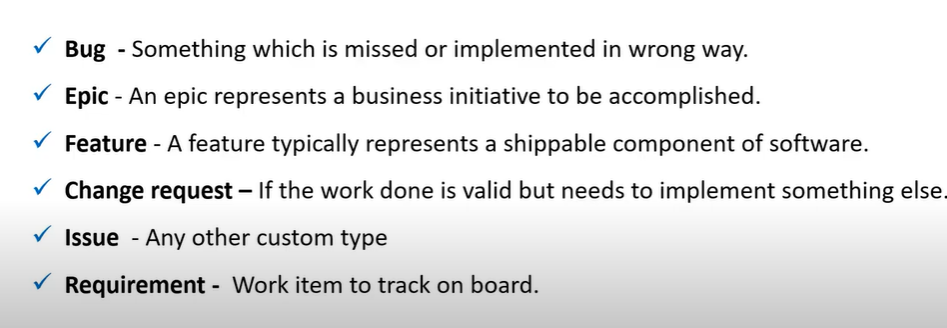


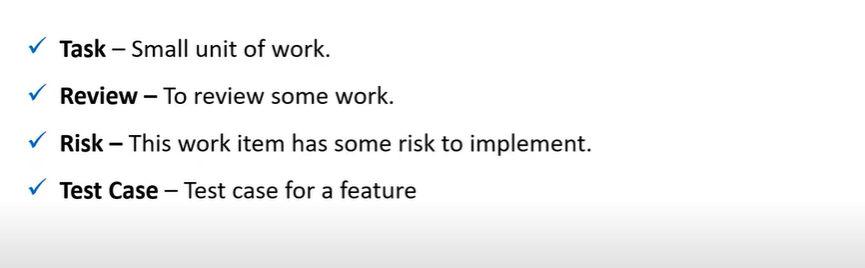
C) Work items with CMMI process:(Capability maturity model integration)

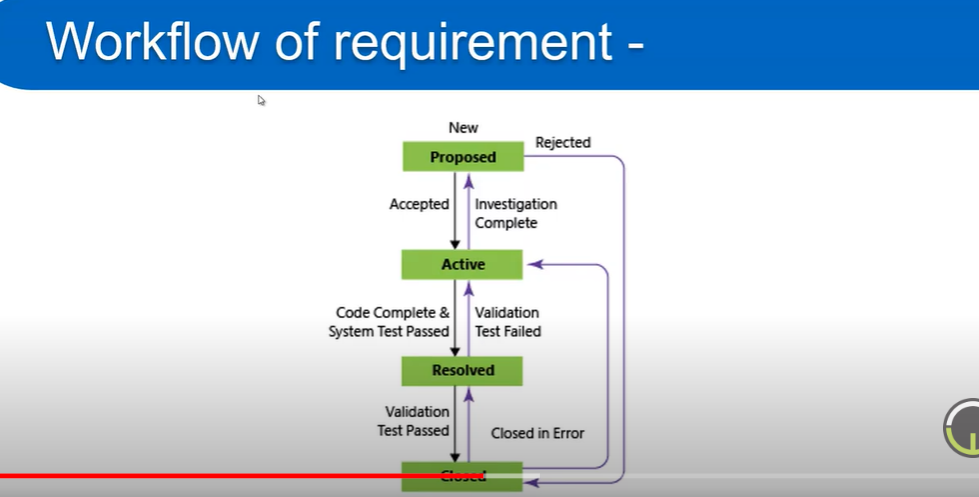
1)Bug 2) Change Request 3) Epic 4) Feature 5) Issue 6)Requirement 7)Review 8)Risk 9)Task 10)Testcase

* Choose CMMI process when your team follows more formal project methods that require a framework for process improvement and an auditable record of decisions



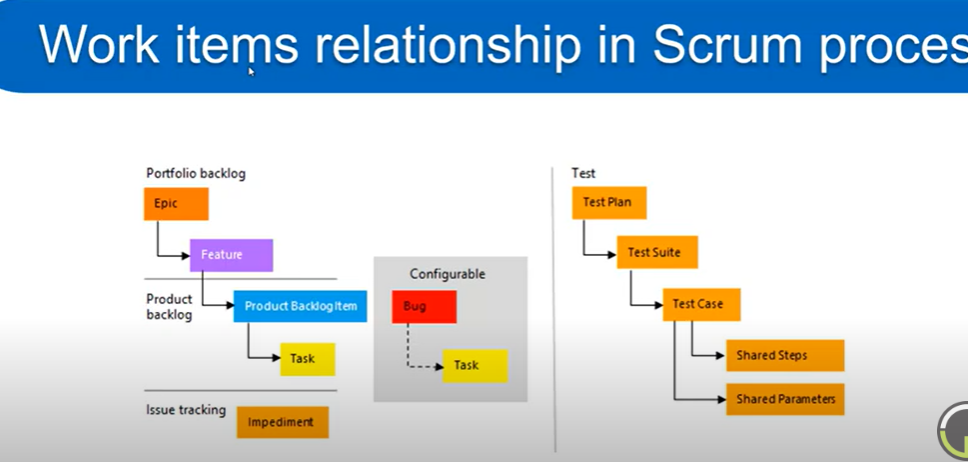


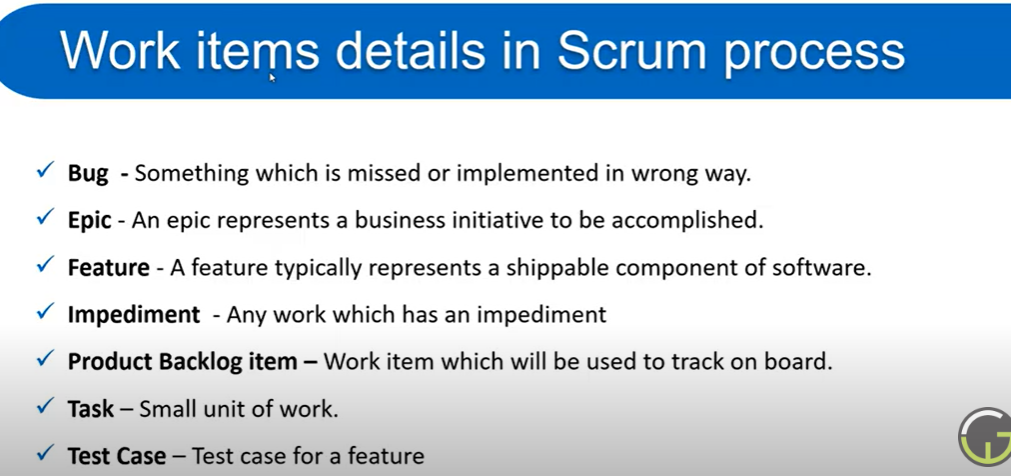


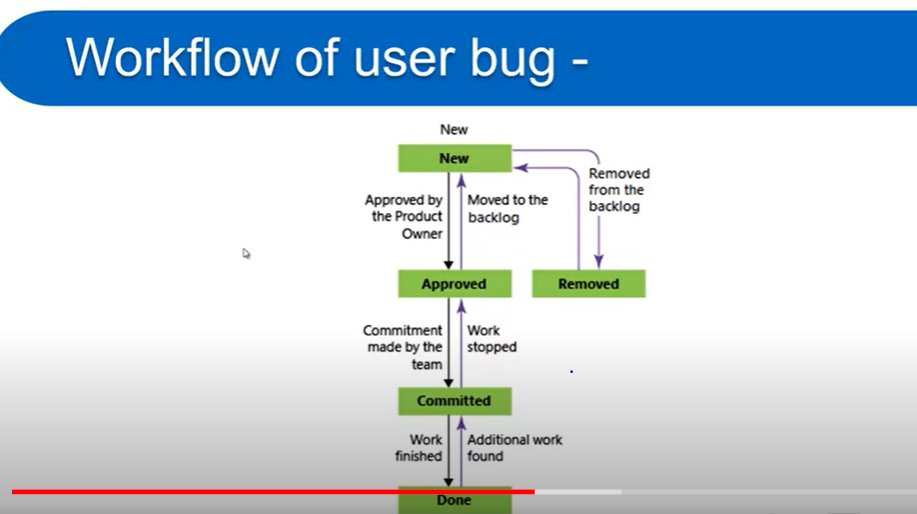


D. Work items with Scrum Process: 1)Bug 2)Epic 3)Feature 4)Impediment 5)Product backlog item 6)Task 7)Testcase

* This Process works great if you want to track product backlog items and bugs on the Kanban board, or Break PBIs and bugs down into tasks on the task board







**Query:** A Query is a combination of few logics which is applied on work items

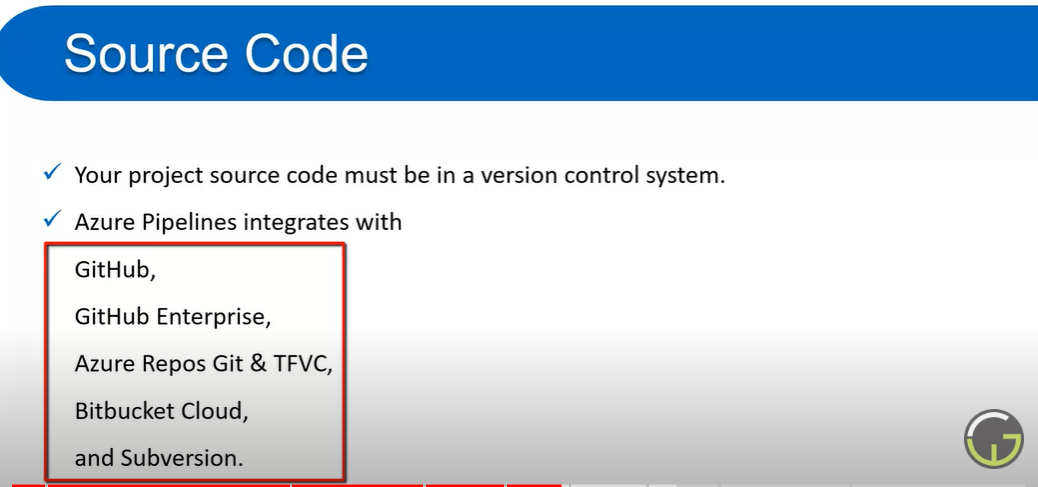
* Azure Repos:
* Repo is a container which is used to store and manage your code in a systematic way.
* Version control systems are types of software’s which are used to track and manage each and every change in code, Done by an individual or a team.
* In Azure Repo there are 2 type of version control

1. Git: Distributed
2. TFVC (Team foundation version control): Centralized

* Tortoise Git: Another tool for Cloning git repo etc.…
* Git add - -all : for adding all the files
* Git commit -m “message” : for commit the change
* Git pull : to update the local repo as remote repo
* Git Checkout “branch name” :To move to another branch
* Git Push –set -upstream origin “branch name” :When we create a branch in local and need to push it to server repo
* Branch Policies : The way to protect Branches
* Branch Lock: If we lock any branches, it means no changes can happen to the branch.
* Tags: Use tags to mark a point in history that is relevant to the repository. 2 types

1. Lightweight tags: It’s a point to specific commit
2. Annotated tags: its contain more information such as the trigger, message and date

* Azure Pipelines:
* Pipeline in DevOps is a set of process (automated or can be triggered manually) which is used to make your project code available to users
* Azure pipeline is a cloud service that you can use to automatically build and test your code project and make it available to others



* Simple CI Pipeline script example for a Dotnet Project:



* $(Build.ArtifactStagingDirectory) : Predefined Variable in Azure Devops
* Predefined variables URL:

https://docs.microsoft.com/en-us/azure/devops/pipelines/build/variables?view=azure-devops&tabs=yaml

* How To create a Agent : Goto project settings -> Under pipelines Agent pool ->Create new agent pool ->Open the created pool ->Add new agent ->Download and configure agent settings in our local machine .(Personal access token we get from user settings-Personal access token)
* Service Connection : With the help of service connection we will be able to connect to any third party tool like GitHub, azure docker etc… Project settings ->service connections
* Task Group: Grouping of all required tasks into one group is called task group. We can add new task to the task group based upon our requirements and we can apply it to different pipelines.
* Libraries : To Create a group of variables we use libraries And we can use these variables in our pipeline.
* Azure Market place : Used for installing third party tools and services.

* Key Concepts in Pipeline
* Agent:
* When you build or deployment runs ,the system begins one or more jobs .An agent is installable software that runs one job at a time
* To build your code or deploy your software using azure pipelines ,You need at least one agent
* In azure DevOps we can use two types of agent : 1)Microsoft hosted agent :Azure pipelines 2)Private : Default
* Approvals:
* Approvals are a set of validations which are required before a deployment can be performed
* Artifacts:
* An Artifacts is a collection of files or packages which are created by a build run. This Artifacts are then made available for the next task that is deployment.
* Environment:
* An environment is the place where we deploy our application
* An environment is a collection of resources. Eg:VM’s,Container,Web App
* A Release pipeline can deploy the code on one or more VM’s (Environment) after the build pipeline is completed.
* Job:
* A job represents an execution boundary of a set of steps , all of the steps run together on the same agent.
* Stage:
* It is used to mark separation of concerns, Each stage contains one or more job.
* Release pipeline (CD Pipeline)
* Deployment group: Before creating release pipeline we need to create Deployment group . Deployment group is the connection between azure devops and EC2 instance.
* In Azure DevOps Portal -> Pipeline –> Deployment Group ->Create Deployment group -> Set name and Description ->Select type of target (as per your requirement) -> Checkbox: Use a personal access token ->Copy the script to clipboard option ->open your instance -(Windows or Linux) -> open PowerShell as admin -> paste and copied content and enter.
* Now go to release pipeline and create new pipeline -> Open artifacts and select source of artifact which is created by build pipeline -> Setup project details and other Details .

Now to go to Stage ->Add a stage ->Choose a template ->Setup stage details like name etc.

Now we need to set tasks ->open jobs and tasks -> Setup details like stage name, Configuration type, actions, website name etc.. -> if you want to update port, update add bind settings ->After configure all the details save the details and create pipeline.

* Build Pipeline Tutorial https://www.youtube.com/watch?v=2CZVIxRF22c&list=PLaFzfwmPR7\_Ifxq-udm66fhReFeGOe2x\_&index=38
* Release pipeline tutorial <https://www.youtube.com/watch?v=ZSynootLhCk&list=PLaFzfwmPR7_Ifxq-udm66fhReFeGOe2x_&index=41>
* Multistage release pipeline tutorials https://www.youtube.com/watch?v=CWeBLNe0Pho&list=PLaFzfwmPR7\_Ifxq-udm66fhReFeGOe2x\_&index=42
* Azure Artifacts (Packages)
* Azure Artifacts enables developers to share and consume packages from Different feeds and public registries.
* Packages can be shared between same team, same organization even publically.
* Azure artifacts supports different packages like Nuget ,Npm ,Python etc…
* Artifacts feeds are organizational constructs that allow you to store ,manage , and group your packages and control who to share it with. You can store all the following package types in a single feed :npm,NuGet ,Maven,Python and universal packages
* Deployment Strategies:
* Recreate
* Remap
* Blue/Green
* Canary
* A/B testing

* Azure Test plans:
* A central location where your team can coordinate all your manual test activities ,Trcak progress and get critical insights.
* Inside test plan there is test suits and we can create multiple test suits according to our need and inside test suit we can create multiple test cases.
* Different types of suits : Static suit ,Requirement based suit ,Query based suit
* We can use shared parameters for different test plans and we can set parameter in the path : Test plans ->Parameters
* We can use test configuration for testing your application on different OS , Web browsers and versions.

Path : test plans -> configuration -> New Test Configuration

ARM

