**Board:**

Azure Boards is a standalone service within the Azure DevOps suite that helps teams plan, track, and discuss work across the entire software development process. It provides a flexible, customizable platform for managing work items, such as user stories, bugs, tasks, and issues, so you can track your work item's progress throughout the development lifecycle.

Azure Boards supports agile methodologies, including [Scrum and Kanban](https://learn.microsoft.com/en-us/azure/devops/boards/get-started/what-is-azure-boards?view=azure-devops#implement-agile-scrum-and-kanban-processes). It provides a range of features and integrations to help teams [collaborate](https://learn.microsoft.com/en-us/azure/devops/cross-service/cross-service-overview?view=azure-devops) and stay organized with [dashboards, reports](https://learn.microsoft.com/en-us/azure/devops/boards/get-started/what-is-azure-boards?view=azure-devops#configure-dashboards-and-power-bi-reports), and notifications.

**Work items:**

The four main states that are defined for the User Story (Agile process) describe a user story's progression. The workflow states are New, Active, Resolved, and Closed. (The Removed state supports removing a work item from appearing on the backlog; to learn more, see Move, change, or delete work items.)

EPIC:

Issues and Tasks are used to track work, while Epics are used to group work under larger scenarios.

Feature:

Features can take multiple sprints to complete, but should be sized to ensure a consistent flow of value to the customer. Features are broken down into stories. Tasks define the work required to complete a work item

User Story :

A user story is a small, self-contained unit of development work designed to accomplish a specific goal within a product. A user story is usually written from the user's perspective and follows the format: “As [a user persona], I want [to perform this action] so that [I can accomplish this goal].”

The workflow states are New, Active, Resolved, and Closed.

**Azure repositories:**

Azure repository is a set of version control tools that you can use to manage your code.

Version control systems are software that help you track changes you make in your code over time.

**CI (Build pipeline ) / CD (Release pipeline) :**

Azure Pipelines automatically builds and test code projects. It supports all major languages and project types and combines continuous integration, continuous delivery, and continuous testing to build, test, and deliver your code to any destination.

CI: CI stands for continuous integration, a fundamental DevOps best practice where developers frequently merge code changes into a central repository where automated builds and tests run.

Here code deploy from Repository to artifactory.

CD: CD can either mean continuous delivery (test environments) or continuous deployment(production).

Here we code deploy from arifactory to cloud

**Resource Group:**

Resource Groups belong to exactly one Subscription. A Subscription can have many resource groups, but a resource group may belong to only one subscription. The Resources themselves (the Azure cloud services) can be grouped together in Resource Groups.

For each project we have one resource group.

**Storage Account:**

An Azure storage account contains all of your Azure Storage data objects, including blobs, file shares, queues, tables, and disks. The storage account provides a unique namespace for your Azure Storage data that's accessible from anywhere in the world over

**Blobs (Binary Large Objects):** This storage type is typically used for unstructured data, such as documents, images, logs, audio, and video, including all associated metadata and organized into containers (think of this as a pseudo folder structure for the flat object namespace).

**File Shares:** Azure Files is Microsoft’s SMB implementation in the cloud. Think of it like Network Attached Storage (NAS). These shares are highly available from anywhere in the world and can be mounted directly on Windows, macOS, and Linux clients via SMB. This is traditional file storage in a folder hierarchy like you’d expect when using the SMB protocol.

**Queues:** Azure Queues are used for asynchronous messaging between application components, which is especially useful when decoupling those components (ex. microservices) while retaining communication between them. Another benefit is that these messages are easily accessible via HTTP and HTTPS.

**Tables:** Azure Tables are structured, schema-less NoSQL data stores in the cloud. Tables are commonly used to store flexible datasets such as user data for a web application and other kinds of metadata

**Disks:** Azure Disks are used as persistent block-level storage attached to VMs in Azure. If you run any VMs in Azure, then you’re probably already using Azure Disks. Outside of VMs you probably won’t find yourself using Azure Disks, but it’s technically part of the Azure Storage family so it’s worth mentioning.

**Storage Tiers:** The three data tiers that Azure offers for storage products are designed to help you save money, especially if all data doesn’t need to be readily available for on-demand retrieval:

* **Hot Tier:** optimizes the storage for frequently accessed data.
* **Cool Tier**: optimizes storage for infrequently accessed data, which must be stored for at least 30-days.
* **Archive Tier :**  optimizes the storage for rarely accessed data, which must be stored for at least 180-days. The retrieval latency time is also flexible at this tier, on the order of hours.

## Azure Functions:

## Azure functions service is a lightweight, serverless compute service with its own use. You can't replace a large website with Azure functions.

Here are some of the use cases of Azure functions,

* Scheduled Tasks
* Reminders and Notifications
* Lightweight Web API

The languages supported to developer azure functions are C#, JavaScript, F#, Java, Powershell, Python, and TypeScript.

**Azure SQL Database:**

Azure SQL Database is a fully managed platform as a service (PaaS) database engine that handles most of the database management functions such as upgrading, patching, backups, and monitoring without user involvement.

**Key vault:** Azure Key Vault is a cloud service that provides a secure store for secrets. You can securely store keys, passwords, certificates, and other secrets. Azure key vaults may be created and managed through the Azure portal. In this quickstart, you create a key vault, then use it to store a secret.

**Azure VM:**

A [virtual machine](https://www.serverwatch.com/guides/virtual-machines/) (VM) is simply a digital version of a physical computer. As a virtualized instance of a computer, a virtual machine can execute almost all tasks a physical computer can, including running operating systems and applications.

**Azure Virtual Networks, Subnets:**

If you let Azure create a virtual network when you create a VM, the name is a combination of the resource group name that contains the virtual network and -vnet. The address space is 10.0.0.0/24, the required subnet name is **default**, and the subnet address range is 10.0.0.0/24.

**DHCP and DNS:**

Domain Name System (DNS) is an Internet service that translates domain names (e.g., its.umich.edu) into IP addresses. Dynamic Host Configuration Protocol (DHCP) is a protocol for automatically assigning IP addresses and other configurations to devices when they connect to a network.

**Network Security Groups (NSG):**

You can use an Azure network security group to filter network traffic between Azure resources in an Azure virtual network. A network security group contains [security rules](https://learn.microsoft.com/en-us/azure/virtual-network/network-security-groups-overview#security-rules) that allow or deny inbound network traffic to, or outbound network traffic from, several types of Azure resources. For each rule, you can specify source and destination, port, and protocol.

**Front door:**

Azure Front Door is a modern cloud content delivery network (CDN) service that delivers high performance, scalability, and secure user experiences for your content and applications. Cloud-native and DevOps tools to automate and streamline deployment. Fully customizable rules engine for advanced routing capabilities.

HTTP ----- 80

HTTPS ----- 443

RDP ----- 3389

**Deployment Group:**

A deployment group is a logical set of deployment target machines that have agents installed on each one. Deployment groups represent the physical environments; for example, "Dev", "Test", or "Production" environment. In effect, a deployment group is just another grouping of agents, much like an agent pool.

* We have create deployment group in pipeline
* After creating vm release agent we will get shell script
* We need to run this script in vm

**Load balancer :**

Main use of the load balancer work is re-route the traffic to VMs

**Backend pool:**

* In load balancer we have concept backend pool
* All VMs we have add in backend pool
* To add backend pool all VMs are should be in same availability set.

**Frantend IP:**

* We have created frontend ip for access applications and need to remove all vm’s IP.
* In Load Balancer all VMs access through frontend ip for reroute traffic.

**Health Check:**

**Azure AD:**

Azure Active Directory (Azure AD), part of Microsoft Entra, is an enterprise identity service that provides single sign-on, multifactor authentication, and conditional access to guard against 99.9 percent of cybersecurity attacks.

AD is great at managing traditional on-prem infrastructure and applications. Azure AD is great at managing user access to cloud applications. You can use both together, or if you want to have a purely cloud based environment you can just use Azure AD.