# **Azure Dev.-Ops+ Jenkins**

**A purple and pink infinity symbol with white text

Description automatically generated**

**Azure deVops**

**What is cloud computing 🡪**

* Cloud computing is the delivery of computing services-such as storage, Servers, database, networking, Software and more over the internet
* instead of physical hardware or data centers, organization can access and use resources on demand and pay accordingly
* cloud provides virtualized infrastructure, platform and applications

**Types Of cloud computing🡪**

There are Three types of Cloud

1. **IaaS(Infrastructure As a service)**

* Provide virtualized Computing resources like virtual machine, Storage, Networks

**Cloud Provider Responsibility**:

* **Physical Infrastructure**: Security of the data center, servers, storage, and network.
* **Networking**: Ensuring network availability and security at the infrastructure level.

**Customer Responsibility**:

* **Operating System**: Installing, updating, and securing the OS.
* **Applications**: Deploying and managing applications.
* **Data**: Data security, encryption, backups, and compliance.
* **Firewalls and Network Configurations**: Configuring firewalls, security groups, and virtual networks within the cloud environment.

1. **PaaS (Platform as a service)**

* Provide a platform to develop ,test, and deploy applications without Worring about underlying infrastructure

**Cloud Provider Responsibility**:

* **Physical Infrastructure**: Data center, servers, networking, and storage.
* **Operating System**: OS updates and patches.
* **Platform Runtime**: Runtime environment for applications (e.g., Java, .NET).
* **Middleware**: Frameworks and tools for application development (e.g., databases, web servers

**Customer Responsibility**:

* + **Applications**: Development, deployment, and management of their own applications.
  + **Data**: Securing, managing, and backing up their data.

**-----------------------------------------------------------------------------------------**

1. **SaaS(software As service)**

* Provide access to fully managed applications via the internet

** Cloud Provider Responsibility:**

* Everything: Infrastructure, networking, servers, storage, OS, applications, and updates.
* Application Security: Securing the application, ensuring availability, updates, and compliance.

** Customer Responsibility:**

* User Data: Managing, securing, and backing up their own data.
* User Access and Identity: Configuring user accounts, roles, and permissions**.**

**---------------------------------------------------------------------**

**What is OnPrem Application🡪**

* The software is hosted locally on companies own server like Cisco, rather than on cloud
* The company has full control over its data ,software, And servers
* The company IT team must take care Of al Aspects

Comparison With Cloud🡪

on-prem applications require more maintenance and upfront costs but offer more control. No cloud provider is involved.

**Customer Responsibility:**

* **Physical Security**: Protecting the physical servers and data center.
* **Networking**: Managing firewalls, routers, and connections.
* **Servers**: Installing, configuring, and maintaining servers and storage.
* A screenshot of a table

  Description automatically generated**Operating System (OS)**: Installing and updating the OS.
* **Applications**: Deploying, managing, and securing applications.
* **Data**: Backups, recovery, security, and compliance

**Shared Responsibility Model with a Traditional Workflow Example (Developer, Support, Ops, Testing)**:

**1. On-Premises (On-Prem)🡪**In the on-premises model, everything is managed by the company’s internal IT teams. The company owns the physical infrastructure (servers, networks) and manages the entire software stack, from hardware to applications.

Workflow:

* Developers: Write code for the banking application, including front-end, back-end, and database components.
* Operations (Ops): Manage the servers, networking, and storage. They install and configure the operating systems, middleware, and runtime environments.
* Testing: Test the application in staging environments hosted on internal servers.
* Support: Provides post-deployment support and deals with infrastructure or application issues.

Challenges:

* High costs for hardware and infrastructure maintenance.
* Slow scalability and difficulty in handling peak loads.

**2. IaaS (Infrastructure as a Service)🡪** With IaaS, the company moves the physical infrastructure (servers, storage, networking) to a cloud provider like AWS or Azure, but they are still responsible for the operating system, middleware, application, and data.

Workflow:

* Developers: Still write the code, but now they deploy the application to virtual machines in the cloud.
* Operations (Ops): No longer manage physical servers. Instead, they configure virtual machines (VMs) and handle the OS, networking (firewalls, security groups), and storage.
* Testing: Test the application in cloud-hosted environments, using cloud-based staging servers.
* Support: Support continues to monitor the virtual machines and application performance, dealing with issues related to the OS and the application.

----------------------------------------------------------------------------------------------------------------------------------

**3. PaaS (Platform as a Service)🡪** In PaaS, the cloud provider handles the infrastructure, OS, runtime, and middleware, while the company is only responsible for the application code and data.

Workflow:

* A white text with black text

  Description automatically generatedDevelopers: Focus entirely on writing and deploying the code. They use cloud services like Azure App Service to host the application.
* Operations (Ops): Minimal involvement since the platform manages the infrastructure and middleware. They focus on configuring platform settings and scaling.
* Testing: Testing teams use automated testing services within the PaaS environment.

Support: Support focuses more on monitoring the application itself, as infrastructure management is done by the cloud provider.

**---------------------------------------------------------------------**

**A waterfall with many colorful icons

Description automatically generated with medium confidenceWaterfall Model**

**Waterfall Model (Easy Explanation)**

**Explanation:**The Waterfall model follows a step-by-step process, where each phase must be finished before moving to the next one. These phases include gathering requirements, designing, coding, testing, and deployment. Once you complete one phase, you cannot go back and change it easily.

Real-time example:  
Think of building a house. First, you design the entire house (requirements), then lay the foundation (development), build the walls (testing), and finally add finishing touches like paint (deployment). Once you’ve built the foundation, you can’t easily go back and change the layout.

**A diagram of a system

Description automatically generated**

**Agile Model**

**Explanation:**Agile is a flexible way of developing software. Instead of completing the whole project in one go, it breaks the work into small pieces called "sprints." Each sprint lasts 1-4 weeks, and after each one, the team checks the progress and makes changes if needed, based on feedback from the customer.

Real-time example:  
Imagine you're building an online store. In the first sprint, you’d create a simple version with just user login and product listing. After testing it and getting feedback, in the next sprint, you’d add features like payment options or search filters. This process repeats, allowing you to improve the site with every sprint.

------------------------------------------------------------------------------------

**DevOps (Development + Operations)**

**Explanation**:  
DevOps combines software development (Dev) and IT operations (Ops) to work together smoothly. The goal is to make software development, testing, and deployment faster by automating tasks. It focuses on improving collaboration between teams and uses tools like Continuous Integration (CI) and Continuous Delivery (CD) to make the process efficient.

Real-time example:  
Think of a car factory where engineers (developers) and assembly workers (operations) work together to build cars faster. By automating repetitive tasks (like machines assembling parts), they can build and release new car models faster without losing quality.

**CI/CD (Continuous Integration/Continuous Deployment)**

**Explanation**:  
CI/CD is a DevOps practice. In Continuous Integration (CI), developers frequently merge their code into a shared system, and automated tests check for errors. Continuous Deployment (CD) automatically sends changes to production once these tests pass, so new features can be released quickly.

**Real-time example**:  
In an online banking app, developers might add new features like a "money transfer" option. With CI/CD, the code for this feature is tested automatically. Once the tests pass, it’s automatically made available to users, speeding up the release process.

**Additional Important Questions:**

1. **What are the benefits of the Agile Model?**  
   Agile offers flexibility, regular customer feedback, quicker releases, and lowers the risk of project failure by delivering in smaller chunks.
2. **How does DevOps improve software development?**  
   DevOps automates repetitive tasks, makes teams work better together, and speeds up software delivery through practices like CI/CD.
3. **What’s the difference between Continuous Integration and Continuous Deployment?**  
   Continuous Integration (CI) ensures that developers frequently merge and test code. Continuous Deployment (CD) pushes changes to production automatically once tests pass.
4. **What is the Waterfall Model best suited for?**  
   It's best for projects with clear, unchanging requirements, like building hardware systems where changing plans later would be too costly.
5. **Why is CI/CD important in modern software development?**  
   CI/CD helps teams find and fix bugs early, speeds up releases, and ensures high-quality software by automating testing and deployment.

A diagram of a company's company

Description automatically generated

