

- **Some of devops-tools:**

1. Version Control Tool: Git (GitLab, GitHub, Bitbucket)

Git is perhaps the best and most widely used version control tool in a development era characterized by dynamism and collaboration. Version control provides developers with a means by which they can keep track of all the changes and updates in their codes such that in the event of a mishap, it is quite easy to return to and use the previous versions of the code and Git happens to be the best for many reasons.

2. Build Tool: Maven

Maven is one of the important DevOps tools for building projects. Unlike the ANT build system, Apache Maven is more than just an automation build framework. It is also designed to manage reporting, documentation, distribution, releases, and dependencies processes. Written in Java language, Maven can build and manage projects written in Java or C#, Ruby, Scala, and other languages using project object model (POM) plugins.

3. Continuous Integration Tool: Jenkins

Jenkins is an integration DevOps tool. For continuous integration (CI), Jenkins stands out as it is designed for both internal and plugin extensions. Jenkins is an open-source Java-based automation CI server that is supported by multiple operating systems including Windows, macOS, and other Unix OSs. Jenkins can also be deployed on cloud-based platforms.

4. Configuration Management Tool: Chef

Configuration management (CM) refers to the maintenance and control of the components of large complex systems in a known, consistent, and determined state throughout the DevOps life cycle. Components of an IT system may include servers, networks, storage, and applications.

5. Configuration Management Tool: Puppet

Puppet is also open-source and uses declarative programming for system configuration, deployments, and server management DevOps tools. It is organized into reusable modules for the speedy setup of pre-configured servers and is compatible with most platforms. Like Chef, it also uses IAC, adopts a master-slave architecture, and features an intuitive user interface for ease of real-time reporting, node management, and several other tasks.

6. Configuration Management Tool: Ansible

Ansible is an open-source CM DevOps tool that is also used for deployment, automation, and orchestration. While Ansible leverages infrastructure as a code architecture, it uses SSH connection for its push nodes thus agentless. Of the three, Ansible is considered easy to learn and use as its Playbooks are written in YAML with minimal commands and are readable by humans.

7. Container Platforms: Docker

Container platforms are application solutions that allow developers to build, test, and ship applications in resource-independent environments. Each

container comprises a complete runtime environment including the specific application, its libraries, source code, configurations, and all its dependencies. Container platforms offer orchestration, automation, security, governance, and other capabilities.

DevOps heavily relies on containerization and microservices for efficient application development and deployment with Docker and Kubernetes as the most widely used container technologies.

8. Container Platforms: Kubernetes

Kubernetes, on the other hand, is an automation orchestration platform that enables developers to run containerized applications across Kubernetes clusters referring to a group of nodes. Developers harness Kubernetes to automate such processes as container configuration, scaling, networking, security, and more to achieve speed and efficiency in production.

9. Communication and Collaboration: Slack

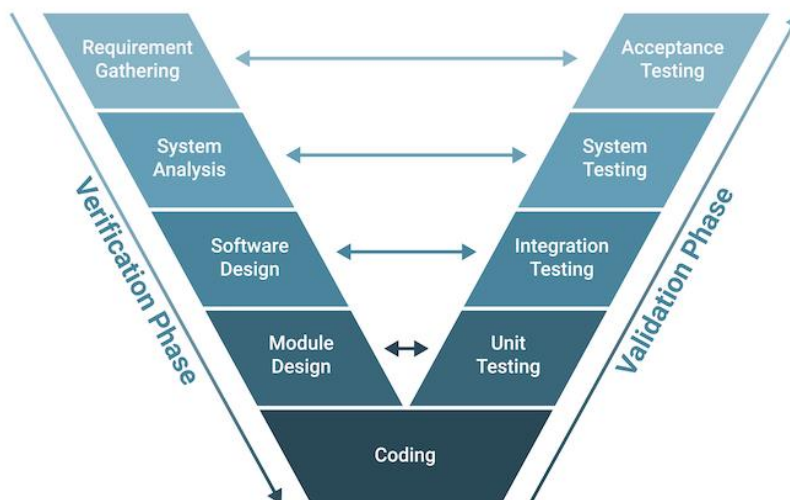
Workplace communication and collaboration technologies are as numerous and as diverse as can be imagined. And when it comes to deciding which tools best suit specific business requirements, several factors go into consideration such as integration and automation capabilities, security, user experience, as well as whether to develop, buy or rent.

AGILE	DEVOPS
Allow collaboration with customer feed back	Bringing development and operation togrther

Rapid change and faster release	Continues test and delivery
Need small team	Need large team
Work by " SPRINTS "	PUSH to put deadline for tasks

V MODEL:

In waterfall model is we move to the next stage there is no chance to get back if something is found wrong in later stage .V model provides means of testing of software at each stage in reverse manner



Design Patterns	Architectural Patterns
Design patterns are reusable solutions to specific design problems that developers encounter during the software design phase. They describe proven approaches to solving recurring design challenges and provide guidelines for structuring code and classes. Design patterns help improve code maintainability, flexibility, and extensibility.	Architectural patterns, on the other hand, are higher-level patterns that provide guidance for organizing the overall structure and behavior of an entire software system. They define the fundamental principles and components of a system and help address system-level concerns such as scalability, maintainability, and performance.

Data ops	ML ops
Data Ops focuses on the operational aspects of managing and delivering high-quality data for various data-driven processes and analytics. It aims to streamline data workflows, improve data quality, and ensure efficient data integration, transformation, and delivery.	ML Ops is a set of practices and methodologies that aim to operationalize and manage the entire machine learning lifecycle, from development to deployment and maintenance. It combines concepts from software engineering, DevOps, and data engineering to address the unique challenges of deploying and managing machine learning models in production environments.