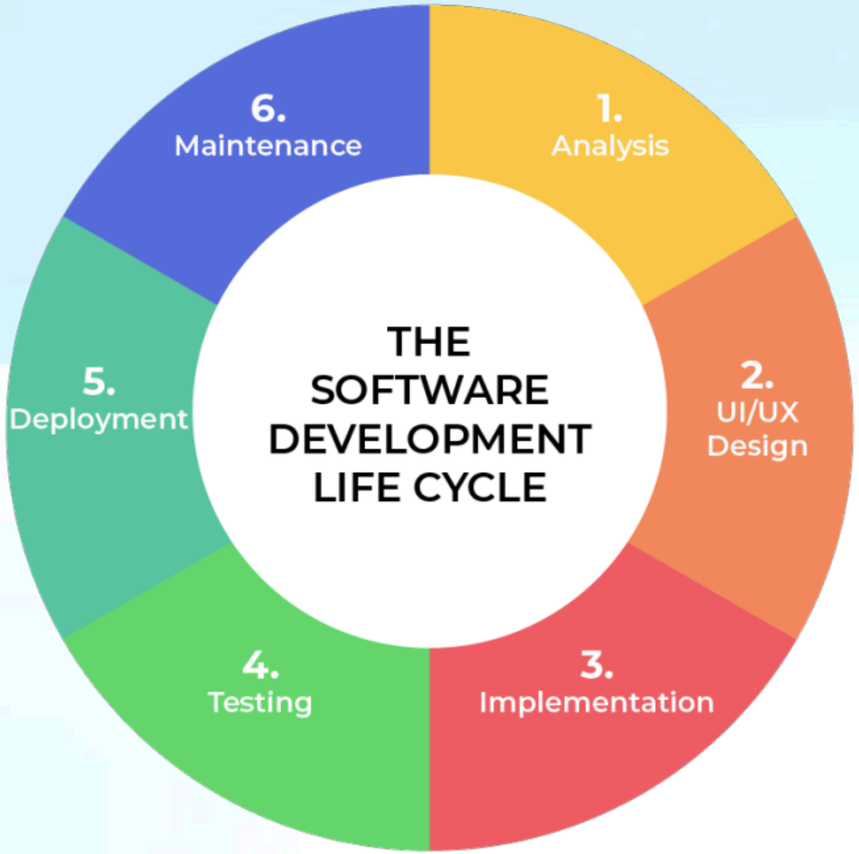
Road To DevOps In 2025

Akhilesh Mishra

SOFTWARE DEVELOPMENT LIFE CYCLE (SDLC)



- The SDLC refers to the overall process of planning, creating, testing, deploying, and maintaining software applications.
- It typically involves several phases, such as requirements gathering, design, development, testing, deployment, and maintenance.

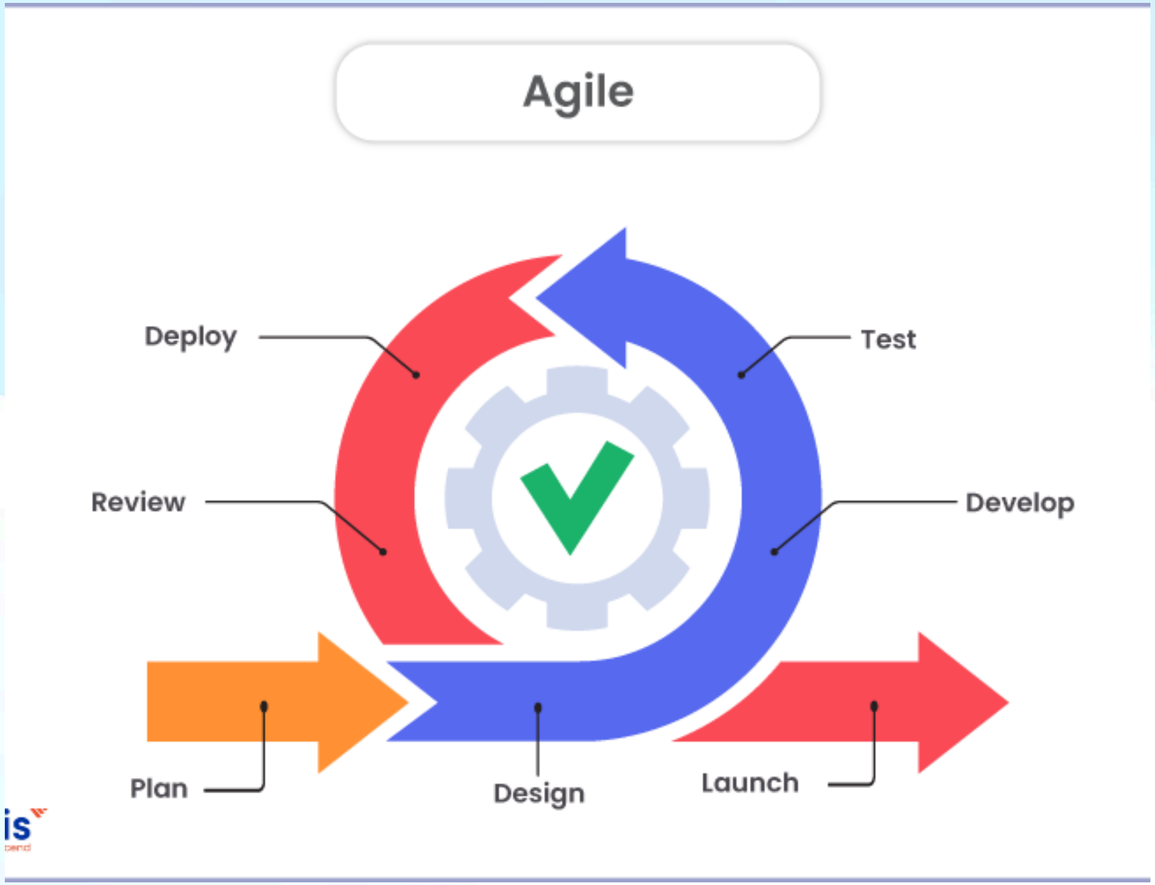
WHAT IS WATERFALL MODEL OF SDLC?

Waterfall —Old ways of SDLC

- Linear and sequential approach to software development
- Distinct phases: requirements gathering, design, development, testing, and deployment
- Emphasis on comprehensive planning and documentation upfront
- Limited flexibility for changes once the project has begun
- Longer development cycles and delayed feedback from end-users

WHAT IS AGILE MODEL OF SDLC?

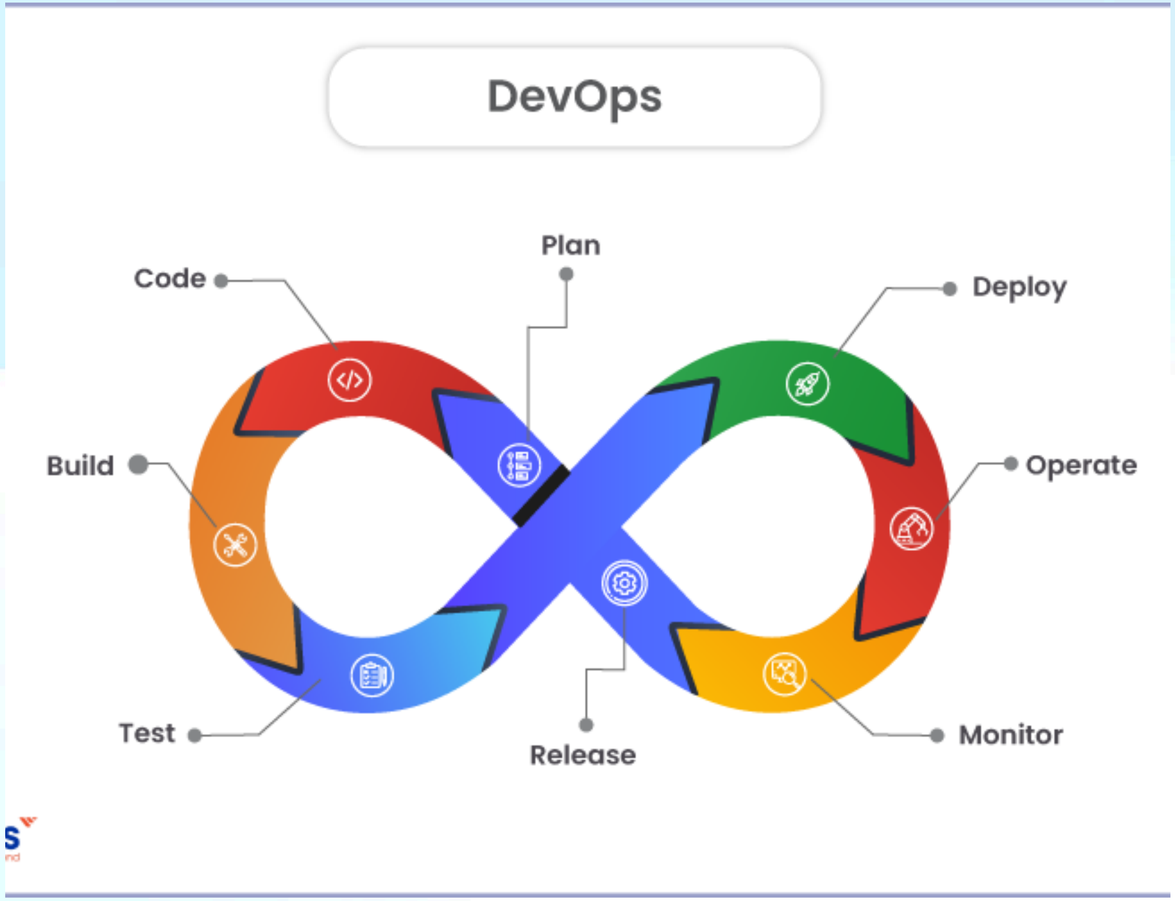
Agile — Improvement over Waterfall



- Iterative and incremental approach to software development
- Emphasis on working software over comprehensive documentation
- Encourages collaboration and communication between cross-functional teams
- Accommodates changes and incorporates user feedback throughout the development process
- Delivers working software in shorter iterations, enabling faster time-to-market and user validation

WHAT IS DEVOPS MODEL OF SDLC?

DevOps — Improvement over Agile



- Extends Agile principles to include operations and infrastructure
- Focuses on continuous integration, continuous delivery, and continuous deployment (CI/CD)
- Automates build, test, and deployment processes to enable frequent and reliable releases
- Encourages collaboration and shared responsibility between development and operations teams
- Enables faster feedback loops, improved quality, and increased agility in delivering software updates and features

THE DEVOPS LANDSCAPE AND ITS GROWING IMPORTANCE

Why DevOps adoption is growing?

- DevOps bridges the gap between development and operations teams, fostering collaboration and efficiency.
- The DevOps approach enables faster time-to-market, improved quality, and enhanced customer satisfaction.
- By automating processes and leveraging continuous integration and delivery (CI/CD), DevOps accelerates software release cycles.
- DevOps encourages a culture of shared responsibility, breaking down silos and promoting cross-functional teamwork
- DevOps enables businesses to respond quickly to changing market demands and customer needs.
- It promotes a focus on continuous improvement, allowing teams to learn, adapt, and optimize their processes.
- As the importance of software continues to grow, DevOps will play an increasingly critical role in driving innovation and business success.

KEY SKILLS AND TECHNOLOGIES YOU NEED TO MASTER

Must Have Skills:1

Linux:

- File/Directory management —Create, delete, copy, move, rename, find, file parsing, permissions, etc
- Package manager —Yum, apt, apt-get, def, etc
- Process and resource management —Resource usage(Memory, disk, cpu, IO) and Process management (PS)
- Basic Networking
- Input/output redirection, awk, cut, tail, head, pipe, alias, etc

Bash:

- Basic bash with input, multiline commands, etc
- Variables and Conditional statements
- Loops and arrays
- Output parsing, data manipulation
- Exit codes and scripts arguments

KEY SKILLS AND TECHNOLOGIES YOU NEED TO MASTER

Must Have Skills:2

Docker:

- Basic docker build and run commands
- Writing Dockerfile, pushing/pulling Docker images to and from public/private repositories
- Docker networking and volumes
- Best practises for writing Docker file
- Docker-compose to run multiple containers with dependencies

Networking:

- IP Addressing: Knowledge of IP addressing schemes, including IPv4 and IPv6, and understanding how to subnet and allocate IP addresses efficiently with CIDR blocks
- Network Security: Familiarity with firewalls, VPNs, and other security measures to protect network resources and prevent unauthorized access.
- Network Troubleshooting: Ability to diagnose and resolve network issues using tools like ping, traceroute, and packet analyzers (e.g., Wireshark) for effective problem-solving.
- Load Balancing: Knowledge of load balancing techniques and technologies (such as HAProxy or Nginx) to distribute network traffic efficiently and ensure high availability.
- DNS, TCP/UDP, HTTP/HTTPS, VPN, SSL

KEY SKILLS AND TECHNOLOGIES YOU NEED TO MASTER

Must Have Skills:3

Git and GitHub

- Familiarity with GitHub UI
- Basic Git commands such as pull, push, merge, rebase
- Resolving git conflicts
- Branching strategies used in companies

CI tool (GitHub Action or Jenkins):

- Basic syntax and events that triggers to CI pipeline
- Reusable workflows
- Using marketplace actions (for Github Actions)
- Security credentials management
- Building complex pipelines

KEY SKILLS AND TECHNOLOGIES YOU NEED TO MASTER

Must Have Skills:4

Cloud Platform(At least 1):

- Cloud Networking: Understanding of cloud networking concepts, such as Virtual Private Cloud (VPC), subnets, and security groups, for deploying and managing applications in cloud environments.
- Cloud storage such as persistent volumes, object storage and file share
- Identity and access management in detail.
- Compute services (Such as EC2 in AWS)
- Container services (Such as ECS and EKS in AWS)
- Serverless Functions(such as Lambda in AWS, and cloud Function in GCP)

Terraform(IAC tool):

- Basic syntax, provider, state management
- Different variable types, locals, resource and data source blocks, output block
- Writing/using private and public modules
- Loops, conditional statement, dynamic block
- Most used builtin functions

KEY SKILLS AND TECHNOLOGIES YOU NEED TO MASTER

Must Have Skills:5

Kubernetes

- Architecture, fundamental concepts
- Deploying micro services in Kubernetes using services, ingress and load balancers
- Namespace, RBAC, service accounts, context switching
- Scaling nodes and pods (HPA, VPA)
- Troubleshooting deployment failures and misconfiguration
- HELM, operators,
- Probes, volumes, daemon set, stateful sets

Python:

- Basic syntax, data structures, functions, recursion, basic OOP
- Conditional statements, loops, running system command, parsing the output
- Making API calls, manipulating json data, reading from files and writing to files
- Writing basic automation scripts using the Cloud provider SDK

KEY SKILLS AND TECHNOLOGIES YOU NEED TO MASTER

Good to have skills(You can get into DevOps without them)

Ansible(Configuration management tool)

- Used to bake images and configure the server

ArgoCD(GitOps tool)

- Used to deploy the application in Kubernetes.

Prometheus (Logging)

Grafana (Dashboarding and monitoring tool)

Elasticsearch(Log storage tool)

Database fundamentals

SQL

Development skills (Will be very useful in coming years)

Cloud certifications

A STEP-BY-STEP ROADMAP FOR YOUR DEVOPS JOURNEY

Phase 1

Learn Fundaments

- Use blogs and Youtube videos to learn fundamentals
- Start with Linux and networking, followed by Docker
- Then Start with Cloud fundamental and explore the Cloud resources using consoles
- Try basic Terraform, write simple resources
- Learn Kubernetes fundamentals
- Explore Git and GitHub Actions
- Then explore the concepts around CICD tool. I recommend GitHub Actions as it is easy to get started and simple to learn
- Learn basic Python and bash

A STEP-BY-STEP ROADMAP FOR YOUR DEVOPS JOURNEY

Phase 2

Get hands on

- Try to build basic projects using 2-3 tools at same time
- Build a 3-tier architecture with Cloud console
- Deploy 2 3-tier infra using Terraform
- Deploy a Dockerized application on Could VM
- Deploy a multi container application with Docker-compose
- Write basic bash and python scripts to do something meaningful
- Write a GitHub Action pipeline to build and deploy the application in cloud VM
- Deploy a simple application in Minikube(kubernetes)
- Use the different Kubernetes components and kubectl CLI in minikube
- Use a simple HELM chart to deploy prometheus and grafana in minikube
- Write a GitHub Action workflow to lint and test the code and also include devsecops tools such as checkoff for terraform and sonar cube

A STEP-BY-STEP ROADMAP FOR YOUR DEVOPS JOURNEY

Phase 3

Build complex projects

- Here you will build complex real world projects that simulate real scenarios.
- Deploy a complete Kubernetes cluster on Cloud using Terraform
- Deploy a microservices based application in Kubernetes cluster following best practices
- Use Github actions pipeline to build all the services and deploy them with ArgoCD
- Upgrading the Cluster
- Build complex Terraform modules and use the to deploy a scalable infra across multiple environments
- Build reusable GitHub Actions workflows and use them at multiple places
- Follow the devsecops best practises in CICD pipelines and use GitHub Advance security features
- Build complex automation that solves a real problem
- Explore the cost optimisation and build automation to automate the cost monitoring and control
- Explore databases logs and sql
- Build a complicated branching strategy to deploy application across multiple environments(dev, test, prod)

STRATEGIES TO ACE YOUR DEVOPS INTERVIEWS

REAL-WORLD SUCCESS STORIES AND LESSONS LEARNED

QUESTIONS?

VALUABLE RESOURCES

Youtube resources

- https://livingdevops.com/devops/best-devops-resources-to-learn-devops/

GitHub resources

- https://github.com/akhileshmishrabiz/Devops-zero-to-hero
- https://github.com/akhileshmishrabiz/python-for-devops
- https://github.com/akhileshmishrabiz/web-app-on-aws-ecs
- https://github.com/akhileshmishrabiz/flaskapp-awsec2

Simple Devops projects

- https://livingdevops.com/python/build-your-first-api-with-complete-crud-operations/
- https://awstip.com/terraform-to-deploy-aws-lambda-function-with-s3-trigger-4c8e231d5f0c?sk=d0ac19b8c5c0316a6aed535d9f461090
- https://awstip.com/securely-connect-ec2-instances-to-s3-buckets-via-private-network-with-vpc-gateway-endpoint-78e03356a9e7?sk=575902d3c8ed33bd18390165e9bf13ab
- https://towardsaws.com/learning-terraform-on-aws-let-me-show-how-its-done-31e38618ac6c?sk=0a3c6252f2917a175f8813ab6b25432e
- https://towardsaws.com/continuously-build-deploy-python-web-app-on-aws-with-github-action-a9de1421898c?sk=9be65762df1f58070460ee26bd291a65
- https://levelup.gitconnected.com/run-multi-container-web-app-with-docker-compose-8f4ad698e8e9?sk=5aead1507d204769ac49e1ac1ca66c47
- https://livingdevops.com/devops/everything-you-need-to-know-about-docker/
- https://livingdevops.com/terraform/how-to-use-terraform-functions/
- https://livingdevops.com/devops/migrating-rds-postgres-database-with-python/
- https://livingdevops.com/devops/python-lambda-layers-with-terraform-github-actions/