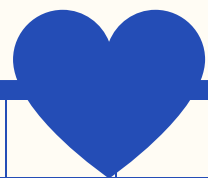


# SDLC

**Software  
Development Life  
Cycle**



@HIRALAL SINGH

# SDLC



What is SDLC?

SDLC stands for Software Development Life Cycle. It's a step-by-step process used by software teams to plan, build, test, and maintain software.



# SDLC

## **Phases of SDLC (Easy Explanation):**

# SOLC

## 1. Plan (What to build?)

- Understand what the customer wants.
- Gather requirements from stakeholders.
- Do cost, time, and resource estimation.
- Create a plan and a requirement document.

# SDLC

## 2. Design (How to build it?)

- Decide the system architecture and tools.
- Choose the best technologies.
- Plan the database, UI, and integrations.

# SDLC

## 3. Implement (Build it)

- Developers write the actual code.
- Break tasks into smaller daily goals.
- Follow coding standards and version control.

# SDLC

## 4. Test (Does it work?)

- Test the code to find bugs.
- Use both manual and automated testing.
- Ensure the software meets user expectations.

# SDLC

## 5. Deploy (Release to users)

- Move the tested build to production.
- Set up configurations and install the software.
- Make it live for real users.



# SDLC

## 6. Maintain (Keep it running)

- Fix bugs and handle user issues.
- Monitor performance and security.
- Keep improving the software with updates.

# SDLC

## SDLC Questions for DevOps Interviews

@HIRALAL SINGH

# INTERVIEW TIME



What is SDLC and why is it important in DevOps?

SDLC is the Software Development Life Cycle. It outlines all the steps from planning to maintaining software. In DevOps, SDLC ensures smooth collaboration between development and operations, automating and optimizing each phase for faster and more reliable delivery.



# INTERVIEW TIME



What role does DevOps play in the SDLC?

DevOps integrates development and operations throughout the SDLC to improve automation, testing, deployment, and monitoring



# INTERVIEW TIME



At which SDLC phases  
is DevOps most  
involved?

DevOps is actively involved in:

- **Implementation:** CI tools like Jenkins automate builds.
- **Testing:** Automated testing ensures code quality.
- **Deployment:** CD pipelines automate release processes.
- **Maintenance:** Monitoring tools track performance and errors.



# INTERVIEW TIME



How does CI/CD fit into the SDLC?

CI/CD automates the Implement, Test, and Deploy phases:

- CI (Continuous Integration): Automates code integration and testing.
- CD (Continuous Deployment): Automates release to production after testing.



# INTERVIEW TIME



Why is having separate build and production environments important?

It ensures code can be tested safely before going live. This protects users from bugs or incomplete features and allows teams to fix issues early.



# INTERVIEW TIME



How does DevOps help in the Maintenance phase of SDLC?

DevOps uses monitoring and alerting tools like Prometheus, Grafana, and ELK to detect issues, ensure uptime, improve security, and apply patches automatically when needed.



# INTERVIEW TIME



How does DevOps help in the Maintenance phase of SDLC?

DevOps uses monitoring and alerting tools like Prometheus, Grafana, and ELK to detect issues, ensure uptime, improve security, and apply patches automatically when needed.

# INTERVIEW TIME



Can you explain how you've used automation in the SDLC process?

Yes, I've automated the build and test process using Jenkins, and used Ansible for deployment scripts. Monitoring is done using Prometheus + Grafana to automatically alert on performance drops