

|  |  |  |
| --- | --- | --- |
| **Batch:B1** | **Roll No.:16010421119** | **Experiment No.:0**1 |

**Aim:** To study DevOps tools and case study of a company.



**Resources needed:** Internet



**Pre Requisite:** Knowledge of DevOps concepts.



**Theory:**

**What is DevOps?**

DevOps is a culture which promotes collaboration between Development and Operations Team to deploy code to production faster in an automated & repeatable way. The word 'DevOps' is a combination of two words 'Development' and 'Operations.'

DevOps helps to increases an organization's speed to deliver applications and services. It allows organizations to serve their customers better and compete more strongly in the market.

In simple words, DevOps can be defined as an alignment of development and IT operations with better communication and collaboration.

**Why is DevOps Needed?**

● Before DevOps, the development and operation team worked in complete isolation.

● Testing and Deployment were isolated activities done after design-build. Hence they consumed more time than actual build cycles.

● Without using DevOps, team members are spending a large amount of their time in testing, deploying, and designing instead of building the project.

● Manual code deployment leads to human errors in production

● Coding & operation teams have their separate timelines and are not in synch causing further delays.

There is a demand to increase the rate of software delivery by business stakeholders.

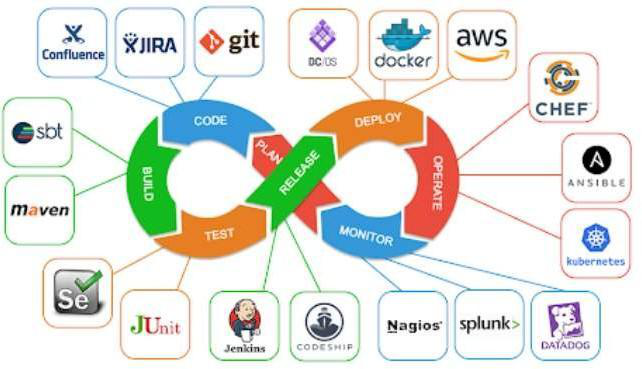
**CI/CD pipelines?**

A **CI**/**CD pipeline** helps you automate steps in your software delivery process, such as initiating code builds, running automated tests, and deploying to a staging or production environment.

**DevOps Automation Tools?**

It is vital to automate all the software development processes and configure them to achieve speed and agility. This process is known as DevOps automation.





**Figure 1: DevOps Processes and popular tools**



**Procedure: (group of 3-4 students, Zigsaw pedagogy)**

1. Explore on internet about DevOps and its tools.

2. Explore and study on internet any one case study/ success story of DevOps.

**Results: (Document based on concept understanding and presentation)**

1. Write in detail any two tools of each phases of DevOps process.

2. Write following questions based on your selected case study/ success story of DevOps: a. Write the existing problem faced by the company.

b. Which activities/problems are selected for DevOps solutions by the company? c. How company has benefited?

3. Presentation of document in the lab for all students.

**Questions:**

1. Do DevOps have planning tools?

a. No tools are available, it is done manually   
 b. Many tools are available   
**Ans: b. Many tools are available**   
2. Configuration management tool?

a. Ansible   
 b. Nagios   
**Ans: a. ansible**   
3. Continuous integration tool?

a. Git   
 b. Jenkins   
**Ans: b.Jenkins**





**Outcomes:**

**ConnectWise Update Rollout Process**

ConnectWise uses a structured process to rollout software updates, including monthly patches

for applications provided by ConnectWise. This structured process ensures smooth, error-free

deployments, improving efficiency and reliability.

**Tools in Each Phase of the DevOps Process for ConnectWise**

***1. Planning Phase***

**Jira**:

● **Functionality**: Jira is used for planning and tracking agile methodology projects. It enables teams

to plan sprints, assign tasks, track progress, and manage releases.

● **Features**:

o **Issue and Project Tracking**: Create and manage tasks, bugs, and user stories.

o **Agile Boards**: Visualize work with customizable Kanban and Scrum boards.

o **Reporting and Analytics**: Generate detailed reports on team performance, sprint progress,

and more.

**Confluence**:

● **Functionality**: Confluence is used for documentation and knowledge sharing. It allows teams to

create, share, and collaborate on documents and projects.

● **Features**:

o **Documentation**: Create and organize project documentation, meeting notes, and

requirements.

o **Collaboration**: Real-time editing and commenting to foster team collaboration.

o **Integration**: Seamless integration with Jira for linking documentation to project tasks.

***2. Development Phase***

**GitLab**:

● **Functionality**: GitLab is a web-based DevOps lifecycle tool that provides a Git repository

manager, CI/CD pipeline features, and monitoring tools.

● **Features**:

o **Version Control**: Manages code changes through Git, allowing for branching, merging,

and rollback.

o **CI/CD Pipelines**: Automates build, test, and deployment processes.

o **Code Review**: Facilitates code reviews and collaboration through merge requests.

**Jenkins**:

● **Functionality**: Jenkins is an open-source automation server used to automate the building, testing,

and deployment of applications.

● **Features**:

o **Automation**: Automates the CI/CD pipeline, including build, test, and deployment stages.

o **Plugins**: Extensive plugin ecosystem for integrating various tools and processes.

o **Scalability**: Supports distributed builds across multiple nodes to handle large-scale

projects.



***3. Deployment Phase***

**Docker**:

● **Functionality**: Docker is a platform that uses containerization to create, deploy, and run

applications in isolated environments.

● **Features**:

o **Containerization**: Encapsulates applications and their dependencies in containers,

ensuring consistency across environments.

o **Portability**: Containers can run consistently on any environment that supports Docker.

o **Resource Efficiency**: Efficiently uses system resources, enabling high-density application

deployments.

**Ansible**:

● **Functionality**: Ansible is an open-source automation tool used for configuration management,

application deployment, and task automation.

● **Features**:

o **Configuration Management**: Automates the setup and configuration of servers and

environments.

o **Provisioning**: Provisions infrastructure and deploys applications.

o **Idempotency**: Ensures that automation tasks can be repeated without causing adverse

effects.

***4. Monitoring Phase***

**Prometheus**:

|  |  |  |
| --- | --- | --- |
| ● ● | **Functionality**: Prometheus is an open-source systems monitoring and alerting toolkit. **Features**: | |
| o | **Metrics Collection**: Collects metrics from configured targets at specified intervals. |
| o | **Alerting**: Configures alerts based on thresholds and conditions. |
| o | **Data Visualization**: Integrates with Grafana for creating detailed visual dashboards. |

**ELK Stack (Elasticsearch, Logstash, Kibana)**:

● **Functionality**: The ELK Stack is a set of tools used for searching, analyzing, and visualizing log

data in real-time.

● **Features**:

o **Log Management**: Collects and processes log data from various sources.

o **Search and Analysis**: Provides powerful search capabilities and data analysis.

o **Visualization**: Kibana offers customizable dashboards for visualizing data trends and

patterns.

**DevOps Case Study: ConnectWise**

***a. Existing Problems Faced by ConnectWise***

● **Manual Deployment Errors**: Frequent human errors during manual deployments of

microservices, leading to system downtime and inconsistencies.

● **Slow Update Rollout**: Lengthy and inefficient update rollout processes, causing delays in

delivering patches and new features.

● **Poor Documentation**: Lack of proper documentation and tracking, leading to difficulties in

understanding the reasons behind code changes and maintaining consistency.



**b. Activities/Problems Selected for DevOps Solutions**

● **Automated Deployment**: Implementing tools like Jenkins and Docker to automate the deployment process, reducing human errors and ensuring consistency.

● **CI/CD Implementation**: Establishing a CI/CD pipeline using GitLab and Jenkins to automate build, test, and deployment processes, speeding up the update rollout.

● **Improved Documentation**: Utilizing Confluence and Jira for better documentation and tracking of code changes, ensuring transparency and future reference.

***c.* Benefits to the Company**

● **Increased Efficiency**: Automation of build and deployment processes significantly reduced manual efforts and sped up the update rollout.

● **Reduced Errors**: Automation minimized human errors during deployment, leading to more stable and reliable updates.

● **Enhanced Scalability**: Streamlined processes allowed for consistent and scalable deployments, accommodating growth and increased demand.

● **Better Traceability and Accountability**: Improved documentation and logging provided better traceability and accountability for code changes.

● **Higher Customer Satisfaction**: Regular and reliable updates enhanced the user experience, leading to higher customer satisfaction and retention.

By adopting DevOps practices and tools, ConnectWise was able to transform their update rollout process, resulting in greater efficiency, reliability, and overall business success.

**Conclusion: (Conclusion to be based on the Results and outcomes achieved)**

In conclusion, ConnectWise's adoption of DevOps tools and practices significantly enhanced their software update rollout process. By leveraging Jira, Confluence, GitLab, Jenkins, Docker, and Ansible, they achieved automation, reduced errors, improved documentation, and streamlined deployments. These improvements led to increased efficiency, scalability, and customer   
satisfaction, demonstrating the transformative impact of DevOps on their business operations and overall success.



**Grade: AA / AB / BB / BC / CC / CD /DD**

**Signature of faculty in-charge with date**



**References:**   
**Books/ Journals/ Websites:**

1. <https://www.amido.com/blog/a-case-study-of-dev-ops-at-netflix>

2. <https://insights.sei.cmu.edu/blog/devops-case-study-netflix-and-the-chaos-monkey/>

3. C**ASE STUDIES - HCL Technologies:** https://[www.hcltech.com](http://www.hcltech.com/) › infographic\_v2.2. 4.[**DevOps on AWS Case Study - Blue Sentry IT:**](https://bluesentryit.com/wp-content/uploads/English__DevOps_Case_Study_Commercial.pdf) https://bluesentryit.com/wp- [content/uploads/English](https://bluesentryit.com/wp-content/uploads/English__DevOps_Case_Study_Commercial.pdf) [DevOps\_Case\_Study\_C](https://bluesentryit.com/wp-content/uploads/English__DevOps_Case_Study_Commercial.pdf)ommercial.pdf   
5.[**CASE STUDY - Vodafone | DevOpsGroup:**](https://www.devopsgroup.com/customers/case-studies/full/vodafone/)



https:/[/www.devopsgroup.com/](http://www.devopsgroup.com/customers/case-studies/full/vodafone/)c[ustomers/case-studies/full/vodafone/](http://www.devopsgroup.com/customers/case-studies/full/vodafone/)

6. Effective DevOps: Building a Culture of Collaboration, Affinity, and Tooling at Scale, Jennifer Davis, Ryn Daniels, O'Reilly Media June 2016.

7. Practical DevOps: Implement DevOps in your organization by effectively building, deploying, testing, and monitoring code,Joakim Verona, Packt Publishing, 2nd Edition,2016