

TL500: Red Hat Training: DevOps Culture and Practice Enablement

Course Code: TL500

Duration: 5 days

Instructor-led Training (ILT) | Virtual Instructor-led Training (VILT)

OVERVIEW

Experience the possibilities of DevOps through proven open culture and practices used by Red Hat to support customer innovation

Red Hat Training: DevOps Culture and Practice Enablement (TL500) is a five-day, immersive class offering students an opportunity to experience and implement cultural shifts that are utilized in many successful DevOps adoption journeys. Many agile training offerings focus on a particular framework, delivery mechanism, or technology. Instead, DevOps Culture & Practice combines the best tools from many leading frameworks to blend continuous discovery and continuous delivery with cultural and technical practices into a unique, highly-engaging experience simulating real-world scenarios and applications.

To achieve the learning objectives, participants should include multiple roles from an organization. Business product owners, architects, developers, and site reliability engineers will gain the experience of working outside of their traditional silos. The daily routine simulates a real-world delivery team, where cross-functional teams learn how collaboration breeds innovation. Armed with shared experiences and best practices, the team can apply what it has learned to help the organization's culture and mission succeed in the pursuit of new projects and improved processes.

This course includes a copy of the Red Hat Training: Open Practices for your DevOps

Journey course book, which can be used as a resource as students take the learnings from this course and apply them to other real world scenarios.

SKILLS COVERED

What are the learning outcomes you can expect?

To help you make the shift to DevOps, this course takes you through a five-day, immersive simulation. You will gain and apply experience with more than 30 practices, including:

- **Cultural tools** such as social contracts, real-time retrospectives, and team sentiments to align teams.
- **Powerful planning practices** such as impact mapping, event storming, and metrics-based process mapping.
- **Iterative and incremental delivery practices**, including scrum and kanban.
- **Technical programming practices**, including continuous integration, continuous delivery, infrastructure as code, test automation, and test-driven development.

You will be exposed to these and other open practices that are used in Open Innovation Labs and can be found in the [Open Practice Library](#).

You will also participate in hands-on labs that demonstrate how we use Red Hat OpenShift Container Platform and Red Hat Ansible in conjunction with Jenkins to automate the build and deployment of a sample to-do list application and its required infrastructure.

How will my organization and team be impacted?

Many companies are finding that their current organizational structure and software development practices are not ready to deliver the outcomes of digital transformation: shorter time to market, quick feedback loops, and user-driven products. These companies need to adopt and practice DevOps culture and methods to be successful in their digital transformation.

This course introduces you to real-world DevOps culture principles and modern software development practices. You will develop a modern software application using Red Hat OpenShift Cluster Platform, Red Hat Ansible Automation, and other industry-standard DevOps software, tools, and techniques. You'll be prepared to use DevOps principles and open source solutions to start and lead the digital transformation journey at your organization.

Red Hat has created this course in a way intended to benefit our customers, but each company and infrastructure is unique, and actual results or benefits may vary.

How will I be impacted?

As a result of attending this course, you will have experienced DevOps culture, been exposed to numerous DevOps practices, and implemented a small application using what you've learned. You should be able to demonstrate these skills:

- Understand and implement multiple open practices
- Deploy a small multi-tiered application to an OpenShift cluster
- Work as an effective member of an agile team
- Discover, prioritize, and document desired software features and functionality

- Develop software using pair and mob programming styles

WHO SHOULD ATTEND?

- This experience demonstrates how individuals across different roles must learn to share, collaborate, and work toward a common goal to achieve positive outcomes and drive innovation. Business product owners, developers, site reliability engineers, and engineers will especially benefit from experiencing this scenario, which includes some technical aspects that rely on working with software systems. You will know how to deliver continuous value to customers through an enhanced understanding of changing customer needs and the ability to more rapidly get new ideas to market. Our instructors will share experiences and best practices learned from engaging directly with customers during Red Hat services engagements.

PREREQUISITES

- Knowledge of agile practices is helpful
- Experience using agile practices and methodologies such as scrum is beneficial

MODULES

Module 1: What is DevOps?

- Brainstorm and explore what principles, practices, and cultural elements make up a DevOps model for software design and development.

Module 2: Collaborative practices to establish culture and shared understanding

- Learn and experience practices that

facilitate great conversation and alignment across stakeholder groups such as priority sliders, pair programming, mob programming, conducting retrospectives, visualizing work, assessing team sentiment, and performing agile estimation.

Module 3: Understanding the Why and Who of software delivery

- Use the impact mapping discovery practice to connect deliverables to measurable impact. Learn how to use human-centered design, design thinking, and Lean UX to develop empathy with users and stakeholders.

Module 4: Domain-driven design and storytelling

- Learn and practice the powerful Event Storming tool to visualize and map event-driven systems to produce emergent architectures for iterative and incremental delivery.

Module 5: Prioritization and pivoting

- Experience the collection of ideas, aligning them to target outcomes, and using economic prioritization practices and value slicing to build product backlogs that can deliver incremental value.

Module 6: Agile practices

- Cover agile delivery practices, including Kanban, Scrum, sprint planning, daily standup, showcase, retrospective, and backlog refinement.

Module 7: Design of experiments

- Set up, execute, and measure the results of experiments by utilizing platform's advanced deployment features, including A/B Testing, Blue/Green Deployments, Feature Toggles, Dark Launches, and Canary Deployments.

Module 8: Value stream and process mapping

- Delve into the practices of value stream mapping and metric-based process mapping to establish non-functional improvements that you can make to product delivery and execution of value streams.

Module 9: Continuous integration, deployment, and delivery

- Explore the foundational practices of continuous integration, continuous deployment, and continuous delivery.

Module 10: Non-functional requirements

- Learn how to elaborate non-functional areas that are unlikely to be captured by using practices primarily focused on the functional aspects of a solution.

Module 11: Testing

- Develop an understanding of test-driven development and business-driven development foundational practices, often called automated testing.

Module 12: Everything as code and GitOps

- Explore continuous integration/continuous delivery pipelines using Jenkins and Tekton and sing a GitOps approach to codify

everything for repeatability. Experience how to extend pipelines to cover non-functional testing, monitoring, and observability.

END OF PAGE