The DevOps Field Guide: A Digital Companion for Young Padawans

Course: Full Stack DevOps Engineering (FSDE)

Summary

Welcome to the front lines of DevOps. While your textbooks provide the *theory*, this guide provides the *tools*. This channel is overall good: https://www.youtube.com/c/devopstoolkit

Phase I: The Culture (Weeks 1-2)

Before we automate, we must understand why.

1. Visualizing "The Phoenix Project" (YouTube)

- **The Resource:** The Phoenix Project: A Must-Read for Anyone in IT
- **Why:** You need to understand the "Four Types of Work." If you don't visualize "Unplanned Work" (firefighting), your ShopSmart project will fail.
- **Key Concept:** Watch how "The Three Ways" (Flow, Feedback, Learning) map to a real pipeline.

2. The DevOps Landscape (Roadmap.sh)

- **The Resource:** An interactive "GPS" for your career. https://roadmap.sh/
- **Why:** Click on terms like "CI/CD" or "IaC" to see how they connect. It proves that Linux is the foundation for everything that follows.

Phase II: The Foundation (Weeks 3-5)

Mastering the Command Line and Version Control.

3. Gamifying Git (Oh My Git! & Learn Git Branching)

- **The Resource:** Interactive games that visualize the Git repository. https://ohmygit.org/ https://learngitbranching.js.org/
- **Why:** You will break your repo. These tools let you visualize the difference between `git merge` and `git rebase` safely.
- **Target Lab:** Week 5 (Pull Requests). Use these to practice before you push bad code to your team's main branch.

4. Linux Command Line Mastery (Cheat Sheets)

- **The Resource:** PDF Cheat Sheets. https://linux-commands.labex.io/
- **Why:** Your AWS server is a Linux machine. When you can't install Node.js because of a "Permission Denied" error, you need to know `chmod` and `chown`.
- **Crucial Commands:** `ps aux | grep node` (Find zombie processes), `kill -9` (Force stop).

Phase III: The Container Era (Weeks 7 & 10)

Making your code portable.

5. Docker Deep Dives (Bret Fisher)

- **The Resource:** The "Node.js in Docker".
  Node.js Rocks in Docker, 2023 Ed. (DockerCon 2023)
- **Why:** The #1 struggle for students is `node_modules`. If you mount your local folder to Docker, you will break the container. Bret explains the "Anonymous Volume" hack to fix this.
- **Target Lab:** Week 7 (Containerizing Frontend/Backend).

6. Kubernetes Sandboxes (Killercoda)

- **The Resource:** Browser-based K8s clusters. https://killercoda.com/
- **Why:** Running Minikube on a laptop can be slow and buggy. Killercoda gives you an instant cluster to practice `kubectl apply` and `kubectl get pods` without crashing your computer.
- **Target Lab:** Week 10 (Deploying to K8s).

Phase IV: Automation & Cloud (Weeks 6, 9, 11)

Connecting Code to the Cloud.

7. GitHub Actions Crash Course (TechWorld with Nana)

- **The Resource:** Visual breakdowns of CI/CD YAML files.
  GitHub Actions Tutorial - Basic Concepts and CI/CD Pipeline with Docker
- **Why:** YAML is strict. One wrong space breaks the pipeline. Nana visually explains "Jobs," "Steps," and "Runners."
- **Target Lab:** Week 6 & 11 (Automated Testing pipelines).

8. Infrastructure as Code (DevOps Directive)

- **The Resource:** Sid Palas's "Complete Terraform Course."
  Complete Terraform Course - From BEGINNER to PRO! (Learn Infrastructure as Code)
- **Why:** It explains *State Management*. If you lose your `terraform.tfstate` file, you lose your infrastructure. Sid teaches you how to lock state using DynamoDB to prevent team disasters.
- **Target Lab:** Week 9 (Provisioning AWS via Code).

Phase V: The AI Context (Week 12+)

Bridging DevOps and your AI Major.

9. MLOps: Made With ML

- **The Resource:** Open-source MLOps curriculum. https://madewithml.com/
- **Why:** You are CS & AI students. "Models" are just code that needs versioning. Use this to apply CI/CD principles to your Machine Learning models.

May the Source Code be with you.