ONE PAGE SUMMARY

IT & Software › Full Stack Development › CI/CD for Full Stack

Introduction to CI/CD for Full Stack

Streamline Development with CI/CD for Full Stack Applications

DEFINITION

CI/CD stands for Continuous Integration and Continuous Deployment, essential for automating software development. It helps developers integrate code changes and deploy them efficiently and reliably.

KEY CONCEPTS

Continuous Integration

A practice where developers frequently integrate their code changes into a shared repository, ensuring early detection of issues.

Continuous Deployment

An extension of CI that automatically deploys code changes to production after passing tests, reducing manual intervention.

Version Control

A system that records changes to files over time, allowing developers to track and manage code revisions.

Automated Testing

Using scripts to automatically test code for bugs and ensure functionality before deployment.

Pipeline

A set of automated processes that code changes go through, from integration to deployment.

EXAMPLES

→ Using GitHub Actions for CI/CD workflows.

→ Deploying a web app to Heroku after passing tests.

→ Setting up Jenkins to automate build and test processes.

MEMORY TIPS

★ Remember 'CI' as 'Code Integrates' and 'CD' as 'Code Deploys'.

★ Think of a pipeline as a conveyor belt for code.

★ Visualize a test suite as a safety net for your code.

COMMON MISTAKES

✗ Neglecting to write automated tests before deployment.

✗ Failing to regularly merge code changes into the main branch.

✗ Overlooking environment configurations leading to deployment failures.

QUICK RECAP

CI/CD is crucial for modern software development, enabling faster and more reliable releases. It involves automating integration and deployment processes to enhance collaboration and efficiency.