

Continuous Delivery

CD.Training

A Collection of On-Line, Self-Paced Study
Programmes

David Farley

2024

Introduction to CD.Training

CD.Training courses have been created by Dave Farley, incorporating his decades of experience and expertise in developing world-class software, consulting and delivering training for a wide range of organisations.

*Transform your software development process
Develop critical engineering skills at scale
More cost-effective and convenient than off-site training
Invest in learning with minimal disruption to work priorities*

These are not simple sit at a screen, tick-the-box courses!
These courses achieve real change in skill and practice, by combining information, explanation and implementation, with several hours of video lessons by Dave Farley, practical exercises, and suggestions for wider exploration and learning.

These courses can be studied by:

- [individuals](#) - flexibly, at their own pace, and in their own place: and,
- [in teams](#) - a cohort working together on the practical exercises, sharing the learning and exploring implications for working practices.

All Our Courses come with Lifetime Access & Money Back Guarantee!

CD.Training Courses

Our range of of on-line training programmes are organised into Training Pathways, where software developers, team-leaders and organisations build expertise from one course to the next, to support you in your goal to create high quality software for your users.

- [Continuous Delivery: Better Software Faster](#)
- [Get Going with CD Pipelines](#)
- [Anatomy of a Deployment Pipeline](#)
- [TDD & BDD: Design Through Testing](#)
- [Understanding Acceptance Testing](#)
- [Acceptance testing with BDD](#)
- [ATDD: Stories to Executable Specifications](#)

About Dave Farley

Inventor of the Deployment Pipeline and Creator of the CD YouTube channel, Dave is a pioneer and expert practitioner in CD, DevOps, TDD and software engineering. Author of the award-winning "Continuous Delivery" book, best-seller "Modern Software Engineering" and the handbook "Continuous Delivery Pipelines."

Dave has an illustrious career developing world class software, and is passionate about helping software developers around the world to improve the design, quality and reliability of their software, through his consultancy, YouTube channel, and training courses.

<https://courses.cd.training>

Testimonials

"Dave Farley's Continuous Delivery Ltd. courses have been instrumental in our software delivery transformation, bringing about visible improvements in speed, efficiency, and quality. I would recommend them to any organisation aiming to optimise their software delivery process"

- Chris Chomicki

"The absolute foundation of software engineering! If you want to learn how to build software, you should stop everything, forget anything you thought you knew about software engineering, and (re)start your training with this course."

- Panagiotis Badredin

"This amazing course is „customer focus in a box“! The ATDD & BDD course is just amazing! Never saw the topic being conveyed that clear and actionable."

- Christian Gesell

"Transformational! This course has deeply changed not only the way I see testing but the way I see software development as a whole, starting from requirements gathering to delivering and maintaining production software. I highly recommend it!"

- Francesc Castells

"Until taking this course I thought I knew what CD was all about. I thought our pipelines were encapsulating all that CD had to offer. I was wrong. In this course you'll learn what CD is really about, both the technical disciplines as well as the business and cultural implications. If you truly want to delivery value into the hands of your customers then this course is for you!"

- Jacob Hodges

"Game Changer. This course showed me how to increase the quality of our work and speed up development while having fun, all while spending more time actually writing code. Yes, all of the above at the same time!"

- Boyan Zlatanov

"Great Course for any organization. This is a great course that explains the importance of Continuous Delivery and gives instructions on how to apply it in your business. This is a course any manager should take."

- Markus Klause

"Everything you need to understand what TDD really is and how it works. The course is well structured. It helps go with your own pace. It covers everything you need in order to start with TDD, and to do it right. I can't be more satisfied."

- Paolo Carollo

"Very enlightening and a very powerful tool because it gives me the notion of everything I need to think about every time I have to implement a continuous delivery approach, or guide someone implementing it, on any kind of technology landscape."

- João Rodrigues

"The course made me feel energised and inspired to tackle large projects with more energy and control. It also motivated me to be optimistic about what is achievable."

- junior dev

Continuous Delivery - Better Software Faster

Purpose

Get the benefits of Continuous Delivery for your software, your team and your business. Understand the foundational principles, learn the CD mindset, and evaluate your practice against the essential techniques, so you can improve and transform your software development process to get fewer bugs, quicker releases, less stress and higher quality code, and have more FUN doing it!

Objectives

- Understand the elements of Continuous Delivery from first principles, to “get the CD Mindset”.
- Practise the essential techniques to implement a Continuous Delivery approach and build Better Software Faster.
- Understand how an efficient Deployment Pipeline will organise your software development efforts, to move quickly and reliably.
- Learn about examples from organisations who have transformed their businesses, so you can adopt practical models to implement Continuous Delivery.
- Assess your current Continuous Delivery capability, and get a targeted report from Dave Farley to guide you to make progress through continuous improvement.

Content

Continuous Delivery is a holistic approach practised by many of the most successful businesses on the planet! This course encompasses technical, organisational and cultural performance and will teach you the Continuous Delivery techniques you need to deliver quality software into the hands of your customers more quickly.

Explore why CD is the way to develop better software faster, and is better for business. And then go through the essential elements that you need to implement CD for your software, your team and your organisation:

- [Achieve repeatability and reliability](#)
- [Applying scientific principles](#)
- [The Deployment Pipeline](#)
- [Reducing the Cycle Time](#)
- [Automated Testing](#)
- [Automation - Release to Production](#)
- [Controlling the Variables](#)
- [Working in Small Steps](#)
- [Making Evidence-based Decisions](#)
- [Working in Small Autonomous Teams](#)
- [Agility in Practice](#)

Conclude with an on-line self-assessment tool, where you can rate how well you are currently working with Continuous Delivery, and get a tailored report from Dave Farley offering targeted advice to help you make progress.

Anatomy of a Deployment Pipeline

Purpose

To improve the speed and efficiency of your software development process, by exploring, in-depth, each stage of the Deployment Pipeline and learning best practices and techniques to turn new ideas into quality features for users quickly and reliably.

Objectives

- Create a Deployment Pipeline that organises your software development efforts to move quickly and reliably, from “idea” to valuable software in the hands of users.
- Improve your “Lead Time”, eliminate waste and deliver fast, while adopting a lean approach to software development by virtue of your automated, efficient Deployment Pipeline.
- Use your Deployment Pipeline as a “machine” to apply scientifically rational, engineering principles: Use it as a “falsification” mechanism, where you can fix, and learn from, failed tests quickly.
- Reduce your software production time by at least 50% by optimising your Deployment Pipeline as the only route to production and “the last word” on releasability.
- Have a Deployment Pipeline that optimises for learning, amplifies skills and empowers your development team.

Content

The Continuous Delivery Deployment Pipeline is a machine for organising the whole software development process. It includes all steps to go from new code to a deployable release, including all tests, version control, fast feedback and automation.

This in-depth course starts with a quick recap of Continuous Delivery principles and will take you through each stage of the Deployment Pipeline:

[How to Build Your Next Deployment Pipeline](#)

[The Commit Cycle](#)

Development Environment, Version Control, Commit Stage, Continuous Integration Strategies, Test Driven Development, and the Artifact Repository

[The Acceptance Cycle](#)

Acceptance Stage, Manual Testing, Performance Testing, Testing Non-Functional Requirements, and Data Migration

[Release Into Production](#)

[The Whole Deployment Pipeline](#)

Infrastructure as Code, Regulation & Compliance, Measuring Success.

We conclude with a Real World Example of a Deployment Pipeline currently in operation for a FinTech project.

Get Dave's book - "Continuous Delivery Pipelines", free when you buy this course.
Get Going with CD Pipelines

Get Going with CD Pipelines

Purpose

Learn how to organise your software development, from writing new code to releasing new software, in one efficient Continuous Delivery Pipeline. A short course to get started with sound principles and essential techniques for your Deployment Pipeline.

Objectives

- Get the knowledge and confidence to build, and use, a Continuous Delivery Pipeline to organise your software development efforts, to move quickly and reliably, from “*idea to valuable software in the hands of users*”.
- Adopt a continuous improvement approach to includes all steps from new code to a deployable release, including: all tests, fast feedback, automation, and the ability to deploy into production.
- Develop ideas about how to optimise your Continuous Delivery Pipeline, to reduce your software production time by at least 50%
- Create a Continuous Delivery Deployment Pipeline that optimises for learning, amplifies your skills and empowers your development team.

Course Content

The Continuous Delivery Pipeline is a ‘machine’ for organising the whole software development process. The Deployment Pipeline helps us to create Better Software Faster by working efficiently and applying the proven, effective techniques of Continuous Delivery.

The course starts with a recap of the seven essential techniques needed to implement, and get the benefits of, Continuous Delivery, and concludes with a case study of the impressive LMAX Pipeline.

The lessons comprise:

[The Benefits of the Continuous Delivery Pipeline](#)

[What is Included in a CD Pipeline](#)

[How to Build your first CD Pipeline.](#)

[The LMAX Case Study](#)

[A practical exercise to apply the lessons to building and enhancing a new Continuous Delivery Pipeline.](#)

TDD & BDD - Design Through Testing

Purpose

Learn how TDD and BDD combine to strengthen the design of your code, improve your programming skills and significantly reduce bugs in production. A course for software technicians, with exercises to practise the techniques, and demos by Dave Farley.

Objectives

- To understand the benefits of TDD and BDD, and the design principles that underpin the effective use of TDD as a practice to create high-quality software
- Develop the skills and techniques to write long-lasting, resilient tests, and significantly reduce defect-rates in your code
- Adopt TDD as an approach to software Design rather than focus on “after-the-fact” unit-testing
- Adopt a Behaviour Driven Development (BDD) approach to the use of TDD to drive the creation of higher-quality software.
- Create better tests and learn how to “listen to the code” to modify designs for the better.

Content

Test Driven Development (TDD) is really more about Test Driven Design than it is about testing, and is one of those very rare software engineering practices that can make a real difference to the quality of your code.

The course starts with an introduction to the benefits of Continuous Delivery and TDD, the principles of “How Test First Improves Design”, and “Red, Green, Refactor”, and progresses through a series of practical exercises to practise the techniques, followed by advice and demos by Dave Farley to show how he would approach these problems.

The lessons include:

- How to Write Great Tests*
- Use of Dependency Injection and Mocking*
- TDD Patterns and Anti-Patterns*
- BDD and the Importance of Naming*
- Design Problems and Solutions*
- Refactoring*
- How to ‘listen to the code’*
- Acceptance Test Driven Development*
- How Tests Improve Design*

Acceptance Testing Training

A Collection of Three Courses

CD Ltd has a collection of 3 courses, in which Dave Farley demonstrates, through a mix of theory and worked examples from real-world projects, the practices and techniques of a BDD-approach to Acceptance Testing. Learn to analyse problems in a way that helps to determine which product features to develop, learn to write better stories and specifications, reduce reliance on manual testing and produce better outcome-focused software for your users.

Each course has been designed to meet the needs of different roles in the Software development process. Pick the course that is right for you!

Understanding Acceptance Testing

For entrepreneurs and leaders, domain experts and specialists. Learn to create better requirements that communicate business goals and ideas for new features, with your technical colleagues, leading to resilient better-designed software.

Acceptance Testing with BDD

For Product Owners, QA/Testers, BAs and Technical Leads. Learn techniques to analyse problems, find Stories and Examples, so you can write Specifications for new software that are understandable, stable and resilient to change.

ATDD & BDD: From Stories to Executable Specifications

Our most technical course on Acceptance Testing, for Developers, Programmers, Engineers and Technologists who turn ideas for new products into software. Learn to translate Stories and Examples into Executable Specifications, and practise techniques for building durable tests and infrastructure for Automated Acceptance Testing.

SHOULD I START WITH THE FIRST COURSE AND PROGRESS TO STUDY ALL THREE?

No. These courses are not a progression from 'basic' to 'advanced'. They each contain the same core content, with additional modules tailored to suit the different roles involved in software development. You only need the one course that's the best fit for you.

Or, you can buy a combination of courses, to suit the different members of your team, whatever their role in writing requirements, coding and testing.

Course Content

These three courses explore "Acceptance Testing", "Behaviour Driven Development (BDD)" and "Specification by Example" in the context of Dave Farley's "Four-Layer Model".

What tools and techniques are effective? How we can avoid the common pitfalls which can lead to tests that are difficult, fragile, slow and prone to intermittent failure.

Through a mix of theory, practical exercises and worked examples, we demonstrate the practices and techniques of high-level functional testing using "Executable Specifications" as the model for the behaviour of a system, and use this as a guiding principle for the development process.

These courses start with the questions - What is Acceptance Testing? and How do we implement it across our organisation? and conclude by looking at how we create effective, efficient test cases in complex systems.

Understanding Acceptance testing includes:

- The team roles and responsibilities of Acceptance Testing
- Defining the Behaviour of the System
- Event Storming - Understanding Your Problem
- Capturing & Translating Requirements
- Specification by Example
- How Test First Improves Design,
- Properties of Good Acceptance Tests
- What to Test, and
- Techniques to write requirements, or "specifications" that describe WHAT we want the system to do, and not HOW it will do it.

In the **Acceptance Testing with BDD** course, we cover all these topics and develop these ideas and techniques further - exploring the link between an effective, story-based, approach to requirements and practising the skills to map from Story to Acceptance Test.

We share real world examples of Executable Specifications and set some problems for you to practise writing tests using these techniques:

- Story-Mapping
- Domain Specific Language
- Story-Mapping - Organising Requirements
- Test First and TDD
- Three Practical Exercises in Story-Mapping and Finding Stories & Examples
- Video Demonstrations of Writing Stories as a Team

In **ATDD & BDD: From Stories to Executable Specifications**, we progress further, using Dave Farley's "Four-Layer Approach" to organising and developing testing infrastructure to make tests easy to write, easy to understand, and flexible in the face of system change. We cover more technical and tricky topics, such as:

- Acceptance Tests & BDD
- The Language of Exec Specs
- Building a DSL for Testing
- Test Isolation
- Protocol Drivers and Stubs
- Testing Asynchronous Systems
- Testing Time
- Dealing with Intermittent Tests
- Managing Test Isolation, Intermittency and Asynchrony

The course concludes by looking at the scope of Acceptance Testing and Test strategy - how to build effective, efficient test cases in complex systems.