CI/CD

Aleix Llusà Serra

Master's degree in Machine Learning and Cybersecurity for Internet Connected Systems
Universitat Politècnica de Catalunya
https://epsem.upc.edu

26 d'octubre de 2026



This work is licensed under a Creative Commons "Attribution-ShareAlike 4.0 International" license.

1 Introduction

2 Version control systems

3 Agile development

4 Continuous integration

5 Continuous deployment

6 References

What is Software engineering?

Software engineering vs computer science?

$\longrightarrow$ Assignment 1

What is Software engineering?

Definition

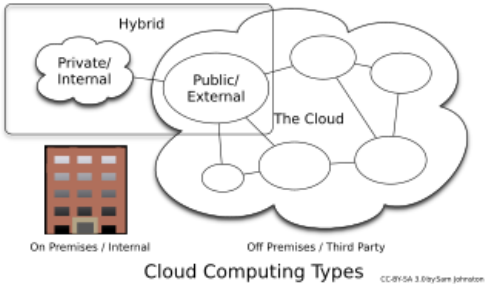
Software engineering is programming integrated over time (Winters, Manshreck i Wright 2020)

Programming tasks (development) and software engineering tasks (development, modification, maintenance). The addition of time adds an important new dimension to programming. A software engineering task is a team effort. (Winters, Manshreck i Wright 2020)

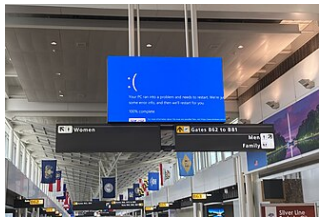
What should avoid Software engineers?



©Freepik



Cloud Computing Types



©Wikimedia Commons

No firefighting. No disturbing when at the café. Sleeping at night.



Jean Béraud



Vincent van Gogh

Which are the critical factors?

Definition

An organization's velocity is a critical factor in its ability to compete with other players, maintain product and service quality, or adapt to new regulation. (Winters, Manshreck i Wright 2020)

Opportunities: Free software

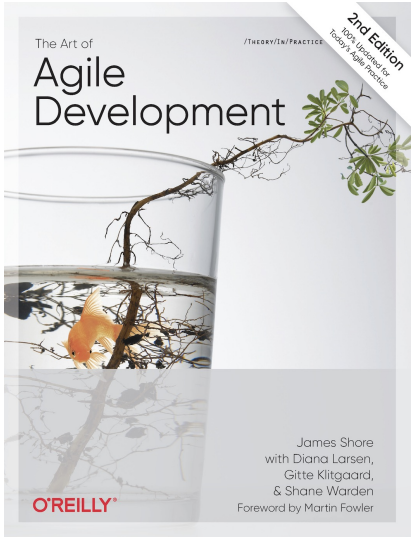
Free Software Foundation (FSF):
"Free software" means software that respects users' freedom and community. Roughly, it means that the users have the freedom to run, copy, distribute, study, change and improve the software. Thus, free software is a matter of liberty, not price.
`https://www.gnu.org/philosophy/free-sw.html`

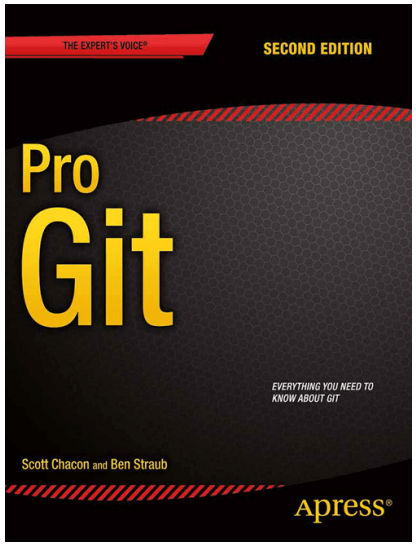
Introduction

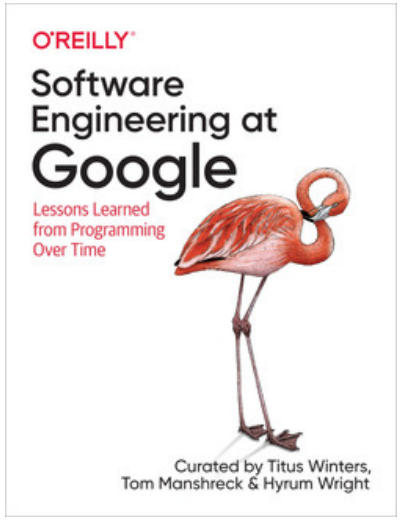
What is this course about?

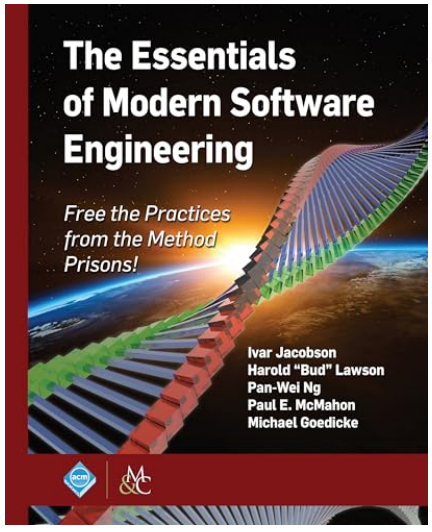
- Version control systems
- Agile development and DevOps
- Continuous integration
- Continuous deployment

Some books









(Shore i Warden 2008)

https://www.jamesshore.com/v2/books/aoad2

https://discovery.upc.edu/permalink/34CSUC_UPC/11q3oqt/alma991005248477306711

(Chacon i Straub 2014) https://git-scm.com/book/en/v2

(Winters, Manshreck i Wright 2020) https://discovery.upc.edu/permalink/34CSUC_UPC/18e7aks/alma991005250023406711

(Jacobson et al. 2019)

Some tools

- Git
- Gitlab CI/CD
  `https://docs.gitlab.com/ee/topics/build_your_application.html`
- Laboratory: `https://gitlab.com/meritupc/2026`

Forge

Manage all the project's lifecycle:

- revision control system
- code review and history
- documentation
- bug tracking system
- release management
- workflow discussion
- continuous integration
- project and task management

Examples: Github, Gitlab, Openproject...

References

- Chacon, Scott i Ben Straub (2014). ProGit. 2a ed. Apress. ISBN: 1484200772. URL: https://git-scm.com/book/en/v2.
- Jacobson, Ivar et al. (2019). The Essentials of Modern Software Engineering: Free the Practices from the Method Prisons! ACM. ISBN: 9781947487277. DOI: 10.1145/3277669.
- Shore, James i Shane Warden (2008). The Art of Agile Development. O'Reilly. ISBN: 9780596527679. URL: https://discovery.upc.edu/permalink/34CSUC_UPC/8e3cvp/alma991005248477306711.
- Winters, Titus, Tom Manshreck i Hyrum Wright (2020). Software Engineering at Google. O'Reilly. ISBN: 9781492082798. URL: https://discovery.upc.edu/permalink/34CSUC_UPC/8e3cvp/alma991005250023406711.