

# Génie Logiciel

## Introduction to Software Engineering

Sylvain Lobry

16/09/2022

## Intro to Software Engineering

# What is a Software Engineering?

- *IEEE Standard Glossary of Software Engineering Terminology*: “The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software”
- Arrêté ministériel de 1983: “l'ensemble des activités de conception et de mise en œuvre des produits et des procédures tendant à rationaliser la production du logiciel et son suivi”

## Intro to Software Engineering

# What is a Software Engineering?

- Historically, started in the October 1968 NATO conference on Software Engineering (Garmisch, Germany), chaired by Friedrich L. Bauer, Louis Bolliet and H. J. Helms
- Domain created by a group of scientists as a reaction to 2 issues:
  - Most softwares are not reliable
  - Hard to finish in time a software that meets requirements.
- Prerequisite: programming

## Intro to Software Engineering

# Software Engineering Body of Knowledge (SWEBOK)

- Published by IEEE computer society (initiated in 1998, last version: 2013)
- International standard (ISO/IEC TR 19759:2005)
- Guide defining the body of knowledge of a Software Engineer
- 15 areas that every software engineer should know
- Was proposed as a formal requirements for the profession of software engineer
- Guide to SWEBOK downloadable for free on [IEEE's website](#)

## Intro to Software Engineering

# Software Engineering Body of Knowledge (SWEBOK)

- Software requirements
- Software design
- Software construction
- Software testing
- Software maintenance
- Software configuration management
- Software engineering management
- Software engineering process
- Software engineering models and methods
- Software quality
- Software engineering professional practice
- Software engineering economics
- Computing foundations
- Mathematical foundations
- Engineering foundations

## Intro to Software Engineering

# Software Engineering Body of Knowledge (SWEBOK)

- Software requirements
- Software design
- Software construction
- Software testing
- Software maintenance
- Software configuration management
- Software engineering management
- Software engineering process
- Software engineering models and methods
- Software quality
- Software engineering professional practice
- Software engineering economics
- Computing foundations
- Mathematical foundations
- Engineering foundations

## Intro to Software Engineering

# Software requirements

- Different types of requirements
- Analysis of the requirements
- Quantification, evaluation & validation of requirements

## Intro to Software Engineering

# Software design

- Architecture of the software
- User interface



## Intro to Software Engineering

# Software construction

- THE CODE
- Choice of a language
- Planning for reusability, changes
- Unit testing
- Profiling

## Intro to Software Engineering

# Software testing

- Different levels of testing
- Different types of tests
- Measurements of tests

## Intro to Software Engineering

# Software maintenance

- Cost of maintenance
- Different types of maintenance (technical, management)
- Techniques for maintenance

## Intro to Software Engineering

# Software configuration management

- Planning of the software configuration
- Control
- Auditing

## Intro to Software Engineering

# Software engineering management

- Planning
- Reviewing
- Closure

## Intro to Software Engineering

# Software engineering process

- Definition of a process
- Software life cycles
- Measurement

## Intro to Software Engineering

# Software engineering models and methods

- Structure to software engineering
- Models as abstractions of software components
- Methods as an organization for a systematic approach

## Intro to Software Engineering

# Software quality

- Definition of a quality
- Management of quality
- Cost of quality



## Intro to Software Engineering

# Software engineering professional practice

- Legal issues
- Documentation
- Working in groups
- Communication

## Intro to Software Engineering

# Software engineering economics

- Basics of economy and finance
- Life cycles
- Economic analysis

## Intro to Software Engineering

# The foundations

- computing
- mathematical
- engineering

## Intro to Software Engineering

# “No silver bullet”

- Many tools, ideas, practices have been developed for software engineering
- *No Silver Bullet – Essence and Accident in Software Engineering* by Fred Brooks, 1987:
- Decomposes complexity of a software project between:
  - Accidental complexity, introduced because of development choices
  - Essential complexity, inherent to the problem
- Argues that accidental complexity has substantially decreased and that essential complexity harder to reduce
- No single technology can reduce by an order of magnitude essential complexity

# Génie Logiciel

## Software quality

Sylvain Lobry

16/09/2022

Resources: [www.sylvainlobry.com/GenieLogiciel](http://www.sylvainlobry.com/GenieLogiciel)

## Software Quality

# Definition of software quality

- ISO 9000:2015 standard
  - **“degree to which a set of inherent characteristics of an object fulfils requirements”**
  - Requirements can be implicit or explicit
- Side note: ISO = International Organization for Standardization

## Software Quality

# Examples of desirable properties

- **Fiability:** the software is valid (obtains desired outcome) and robust (able to cope with errors during the execution or erroneous input)
- **Safety:** when software is safety-critical, additional requirements
- **Performances:** expression of requirements in execution or processing speed and in terms of resources (e.g. memory)
- **Compatibility:** the software can be used with other softwares (either defined or meeting a specific format)
- **Portability:** the software can be used in various environments (e.g. OS)
- **Ergonomics:** the software can be easily (or naturally) used by the end-user
- Many more in few slides

## Software Quality

# I want it all!

- Quality comes at a cost. Non qualitative software as well -> Notion of “Cost of Software Quality”
- Often “level of quality in a software product can be inferred from the cost of activities related to dealing with the consequences of poor quality”
- 4 types of cost of quality:
  - Prevention (training, infrastructure, tools)
  - Appraisal (review, testing)
  - Internal failure (before delivery of the product)
  - External failure (after delivery of the product)



## Software Quality

# Control lists

- List of (technical) elements to be verified during the actual inspection
- For instance, one list per programming language
- Errors found should belong to the control list and identified
- Needs to be concise