

# Introduction to Software Engineering

CS-C3150, Software Engineering

Casper Lassenius

# Software is Everywhere!

- The world is dependent on software
  - Businesses
  - Governments
  - Infrastructure
  - Services
  - Devices and Appliances
  - Machines and Transportation
  - Artificial Intelligence / Machine Learning



# Software is Expensive!

- When developing a high-tech product, the cost of software development often outweighs that of hardware
- Large software projects can cost several hundreds of millions
- Example: The Apotti system for patient data in the Helsinki region had cost over 800 M€ by the end of 2022
- Software engineering aims at developing software cost-efficiently



# Software Failures Can Be Catastrophic!

- Human lives can be lost
  - E.g., Therac-25, Tesla
- Careers can be lost
  - E.g., Volkswagen CEO Herbert Diess
- Money can be lost
  - E.g., Helsinki city payroll system, Knight Capital Group losing \$440M in 30 minutes
- Software engineering is about creating working and reliable software, given real-world constraints



**Software systems  
must be  
professionally  
developed, operated, and maintained**

# Professional Software Engineering

- Skilled / chartered software engineers developing software for someone else as a profession
- Typically involves teams of people, not single individuals doing everything
  - From one team of a few people to thousands of people working simultaneously on the same piece of software
- Typically involves larger systems than the one developed by a single individual
  - There can be hundreds of millions of lines of code in a system
  - There can be many different technologies in a system

# Software Engineering

- Engineering discipline
  - Using theories and methods to solve practical problems under real-world constraints (e.g., financial, organizational, schedule, ...)
- Concerns all aspects of software production
  - From understanding what system to build to maintenance and retirement
  - Not only technical implementation (“coding”), but also process, organizational, management, method and tools to support software development and operation

# **Software Engineering**

**“The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software;**

**that is, the application of engineering to software”**

**IEEE Std. 610.12–1990 Glossary of Software Engineering Terminology &  
ISO/IEC/IEEE Systems and Software Engineering Vocabulary**

# SWEBOK

## Software Engineering Body of Knowledge

- Promotes a consistent worldwide view of software engineering
- Defines 18 knowledge areas, specifying the scope of software engineering
- Characterizes the contents of software engineering within each knowledge area
- Here: V4, in “beta”, official version is V3 from 2014

#	Knowledge Area
1	Software requirements
2	Software architecture
3	Software design
4	Software construction
5	Software testing
6	Software engineering operations
7	Software maintenance
8	Software configuration management
9	Software engineering management
10	Software engineering process
11	Software engineering models and methods
12	Software quality
13	Software security
14	Software engineering professional practice
15	Software engineering economics
16	Computing foundations
17	Mathematical foundations
18	Engineering foundations

# Why Should You Care?

- There is a large need for software engineering experts in industry
  - Software engineer, Scrum master, Product owner, tester, architect, team lead, software engineering manager, ...
- Many positions, both management and expert (e.g. architect), are well paid
- Gives a good basis for software entrepreneurship
- Work on something that matters!



**Thanks & Have Fun!**