



SOFTWARE ENGINEERING

cuu duong than cong . com

Chapter 1 - INTRODUCTION

cuu duong than cong . com

Topics covered

- Professional software development
 - What is meant by software engineering.
- Software engineering ethics
 - A brief introduction to ethical issues that affect software engineering.

cuu duong than cong . com

cuu duong than cong . com



Software engineering

- The economies of ALL developed nations are dependent on software (?)
- More and more systems are software controlled
- Software engineering is concerned with theories, methods and tools for professional software development.
- ...

cuu duong than cong . com



Software costs

- Software costs often dominate computer system costs
 - The costs of software on a PC are often greater than the hardware cost.
- Software costs more to maintain than it does to develop
 - For systems with a long life, maintenance costs may be several times development costs.
- Software engineering is concerned with cost-effective software development.

cuu duong than cong . com



Software products

- Generic products
 - Stand-alone systems that are marketed and sold to any customer who wishes to buy them.
 - Examples – PC software such as graphics programs, project management tools; CAD software; software for specific markets such as appointments systems for dentists.
- Customized products
 - Software that is commissioned by a specific customer to meet their own needs.
 - Examples – embedded control systems, air traffic control software, traffic monitoring systems.



Product specification

- Generic products
 - The specification of what the software should do is owned by the software developer and decisions on software change are made by the developer.
- Customized productsong than cong . com
 - The specification of what the software should do is owned by the customer for the software and they make decisions on software changes that are required.

cuu duong than cong . com



FAQs about software engineering

Question	Answer
What is software?	Computer programs and associated documentation. Software products may be developed for a particular customer or may be developed for a general market.
What are the attributes of good software?	Good software should deliver the required functionality and performance to the user and should be maintainable, dependable and usable.
What is software engineering?	Software engineering is an engineering discipline that is concerned with all aspects of software production.
What are the fundamental software engineering activities?	Software specification, software development, software validation and software evolution.
What is the difference between software engineering and computer science?	Computer science focuses on theory and fundamentals; software engineering is concerned with the practicalities of developing and delivering useful software.
What is the difference between software engineering and system engineering?	System engineering is concerned with all aspects of computer-based systems development including hardware, software and process engineering. Software engineering is part of this more general process.

FAQs about software engineering

Question	Answer
What are the key challenges facing software engineering?	Coping with increasing diversity, demands for reduced delivery times and developing trustworthy software.
What are the costs of software engineering?	Roughly 60% of software costs are development costs, 40% are testing costs. For custom software, evolution costs often exceed development costs.
What are the best software engineering techniques and methods?	While all software projects have to be professionally managed and developed, different techniques are appropriate for different types of system. For example, games should always be developed using a series of prototypes whereas safety critical control systems require a complete and analyzable specification to be developed. You can't, therefore, say that one method is better than another.
What differences has the web made to software engineering?	The web has led to the availability of software services and the possibility of developing highly distributed service-based systems. Web-based systems development has led to important advances in programming languages and software reuse.



Essential attributes of good software

Product characteristic	Description
Maintainability	Software should be written in such a way so that it can evolve to meet the changing needs of customers. This is a critical attribute because software change is an inevitable requirement of a changing business environment.
Dependability and security	Software dependability includes a range of characteristics including reliability, security and safety. Dependable software should not cause physical or economic damage in the event of system failure. Malicious users should not be able to access or damage the system.
Efficiency	Software should not make wasteful use of system resources such as memory and processor cycles. Efficiency therefore includes responsiveness, processing time, memory utilisation, etc.
Acceptability	Software must be acceptable to the type of users for which it is designed. This means that it must be understandable, usable and compatible with other systems that they use.

Software engineering

- Software engineering is an engineering discipline that is concerned with **all aspects of software production** from the early stages of **system specification** through to **maintaining the system** after it has gone into use.
- Engineering discipline
 - Using appropriate theories and methods to solve problems bearing in mind organizational and financial constraints.
- All aspects of software production
 - Not just technical process of development. Also project management and the development of tools, methods etc. to support software production.



Importance of software engineering

- More and more, individuals and society rely on advanced software systems. We need to be able to produce reliable and trustworthy systems economically and quickly.
- It is usually cheaper, in the long run, to use software engineering methods and techniques for software systems rather than just write the programs as if it was a personal programming project. For most types of system, the majority of costs are the costs of changing the software after it has gone into use.



Software process activities

- **Software specification**, where customers and engineers define the software that is to be produced and the constraints on its operation.
- **Software development**, where the software is designed and programmed.
- **Software validation**, where the software is checked to ensure that it is what the customer requires.
- **Software evolution**, where the software is modified to reflect changing customer and market requirements.



Software process activities – An alternative

- **defining** the software development process to be used
- **managing** the development project
- **describing** the intended software product
- **designing** the product
- **implementing** the product
- **testing** the parts of the product
- **integrating** the parts and testing them as a whole
- **maintaining** the product



General issues that affect most software

- **Heterogeneity**
 - Increasingly, systems are required to operate as distributed systems across networks that include different types of computer and mobile devices.
- **Business and social change**
 - Business and society are changing incredibly quickly as emerging economies develop and new technologies become available. They need to be able to change their existing software and to rapidly develop new software.
- **Security and trust**
 - As software is intertwined with all aspects of our lives, it is essential that we can trust that software.



Software engineering diversity

- There are many different types of software system and there is no universal set of software techniques that is applicable to all of these.
- The software engineering methods and tools used depend on the type of application being developed, the requirements of the customer and the background of the development team.

cuu duong than cong . com



Application types

- Stand-alone applications
 - These are application systems that run on a local computer, such as a PC. They include all necessary functionality and do not need to be connected to a network.
- Interactive transaction-based applications
 - Applications that execute on a remote computer and are accessed by users from their own PCs or terminals. These include web applications such as e-commerce applications.
- Embedded control systems
 - These are software control systems that control and manage hardware devices. Numerically, there are probably more embedded systems than any other type of system.



Application types

- Batch processing systems
 - These are business systems that are designed to process data in large batches. They process large numbers of individual inputs to create corresponding outputs.
 - Virus scanning
- Entertainment systems
 - These are systems that are primarily for personal use and which are intended to entertain the user.
- Systems for modeling and simulation
 - These are systems that are developed by scientists and engineers to model physical processes or situations, which include many, separate, interacting objects.



Application types

- Data collection systems
 - These are systems that collect data from their environment using a set of sensors and send that data to other systems for processing.
- Systems of systems
 - These are systems that are composed of a number of other software systems.

cuu duong than cong . com



Software engineering fundamentals

- Some fundamental principles apply to all types of software system, irrespective of the development techniques used:
 - Systems should be developed using a managed and understood **development process**. Of course, different processes are used for different types of software.
 - **Dependability** and **performance** are important for all types of system.
 - Understanding and managing the **software specification** and requirements (what the software should do) are important.
 - Where appropriate, you should **reuse** software that has already been developed rather than write new software.



Software engineering ethics

- Software engineering involves **wider responsibilities** than simply the application of technical skills.
- Software engineers must **behave** in an **honest** and **ethically responsible** way if they are to be respected as professionals.
- **Ethical behaviour** is more than simply upholding the law but involves following a set of principles that are morally correct.



ACM/IEEE Code of Ethics

- The professional societies in the US have cooperated to produce a code of ethical practice.
- Members of these organisations sign up to the code of practice when they join.
- The Code contains some Principles related to the behaviour of and decisions made by professional software engineers, including practitioners, educators, managers, supervisors and policy makers, as well as trainees and students of the profession.

IEEE code of ethics

1. **to accept responsibility** in making decisions consistent with the safety, health and welfare of the public, and to disclose promptly factors that might endanger the public or the environment;
2. **to avoid real or perceived conflicts of interest** whenever possible, and to disclose them to affected parties when they do exist;
3. **to be honest and realistic** in stating claims or estimates based on available data;
4. to reject bribery in all its forms;
5. **to improve the understanding of technology**, its appropriate application, and potential consequences;
6. **to maintain and improve our technical competence** and to undertake technological tasks for others only if qualified by training or experience, or after full disclosure of pertinent limitations;
7. to seek, accept, and offer **honest criticism** of technical work, to acknowledge and correct errors, and to credit properly the contributions of others;
8. **to treat fairly** all persons regardless of such factors as race, religion, gender, disability, age, or national origin;
9. **to avoid injuring others**, their property, reputation, or employment by false or malicious action;
10. **to assist colleagues and co-workers** in their professional development and to support them in following this code of ethics.



ACM Code of Ethics and Professional Conduct

- 1.1 Contribute to society and human well-being.
- 1.2 Avoid harm to others.
- 1.3 Be honest and trustworthy.
- 1.4 Be fair and take action not to discriminate.
- 1.5 Honor property rights including copyrights and patent.
- 1.6 Give proper credit for intellectual property.
- 1.7 Respect the privacy of others.
- 1.8 Honor confidentiality.



ACM SE Code of Ethics and Professional Practice

(short version)

- **PUBLIC** - Software engineers shall act consistently with the public interest.
- **CLIENT AND EMPLOYER** - Software engineers shall act in a manner that is in the best interests of their client and employer consistent with the public interest.
- **PRODUCT** - Software engineers shall ensure that their products and related modifications meet the highest professional standards possible.
- **JUDGMENT** - Software engineers shall maintain integrity and independence in their professional judgment.
- **MANAGEMENT** - Software engineering managers and leaders shall subscribe to and promote an ethical approach to the management of software development and maintenance.
- **PROFESSION** - Software engineers shall advance the integrity and reputation of the profession consistent with the public interest.
- **COLLEAGUES** - Software engineers shall be fair to and supportive of their colleagues.
- **SELF** - Software engineers shall participate in lifelong learning regarding the practice of their profession and shall promote an ethical approach to the practice of the profession.



Summary

- Software engineering is an engineering discipline that is concerned with all aspects of software production.
- Essential software product attributes are maintainability, dependability and security, efficiency and acceptability.
- The high-level activities of specification, development, validation and evolution are part of all software processes.
- The fundamental notions of software engineering are universally applicable to all types of system development.



Summary (cont.)

- There are many different types of system and each requires appropriate software engineering tools and techniques for their development.
- The fundamental ideas of software engineering are applicable to all types of software system.

cuu duong than cong . com



Summary (cont.)

- Software engineers have responsibilities to the engineering profession and society. They should not simply be concerned with technical issues.
- Professional societies publish codes of conduct which set out the standards of behaviour expected of their members.

cuu duong than cong . com



Decide Initial Team Issues

- Set the meeting agenda and time limits.
- Choose the team leader (leadership strategy?).
- Decide how the team will communicate.
- Identify the customer.
 - The party or parties who want this application.
- Get an understanding of the project in general terms.
 - Don't be embarrassed if project seems too vague to you.
 - Probe until you are comfortable.

Set Team Expectations

- Get everyone's commitment to taking required time
 - Define an expected average number of hours per week
 - If not forthcoming:
 - Industrial: alert management
 - Academic: inform instructor; implement written mutual evaluations
 - Gather dates of planned absences
- Choose team emphasis: accomplishment / learning
 - Accomplishment (capable product): get a good mix of leadership, technical, writing, customer relations
 - Learning: sacrifice accomplishment by allowing members to experience new activities.
 - Understand manager's / instructor's emphasis.





Specify How the Team Will Communicate

- General policy:
 - if in doubt, communicate. Redundancy is OK!
- Usual Meeting place and time
- Alternative meeting
- Standards:
 - The MS WORD is used for documenting
 - E-mail should be via any compatible Yahoo email (especially for attachments)
- Preferred mode of electronic communication:
 - Ex: Unless a communication is of very limited interest to the group, it should be posted to the group site, www.xxx.yyy with automatic notification to every member. The “subject” format should be Attn. <name(s)>: subject matter.
- Alternative mode of electronic communication:
 - For 1-1 communication of very limited group interest, members will use e-mail and/or telephone.
- Acknowledgement:
 - Team members should acknowledge all electronic communication specifically targeted to them, whether asked to acknowledge or not. Senders should follow up on all significant communication that is not acknowledged.

