

BSE101- Software Requirements Engineering

- CA - 40%
 - 2 TESTS
- EXAM - 60%
- www.LeChaamwe.weebly.com
 - Lecture Notes
 - Undergraduate

Introduction to Software Engineering

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

Software costs

- Software costs often dominate 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 costs

- Software engineering is concerned with cost-effective software development

What is Software?

- Software encompasses:
 - (1) **instructions** (computer programs) that when executed provide desired features, function, and performance;
 - (2) **data structures** that enable the programs to adequately store and manipulate information and
 - (3) **documentation** that describes the operation and use of the programs.

What is software engineering?

- Software engineering is an engineering discipline which is concerned with all aspects of software production
- Software engineers should adopt a systematic and organised approach to their work and use appropriate tools and techniques depending on the problem to be solved, the development constraints and the resources available



Software Engineering Definition

The IEEE definition:

*Software Engineering: (1) The application of a **systematic, disciplined, quantifiable approach** to the **development, operation, and maintenance** of software; that is, the application of engineering to software. (2) The study of approaches as in (1).*



8

Software Applications

1. **System software:** such as compilers, editors, file management utilities
2. **Application software:** stand-alone programs for specific needs.
3. **Engineering/scientific software:** Characterized by "number crunching" algorithms. such as automotive stress analysis, molecular biology, orbital dynamics etc.



9

Software Applications

4. **Embedded software** resides within a product or system. (key pad control of a microwave oven, digital function of dashboard display in a car)
5. **Product-line software** focus on a limited marketplace to address mass consumer market. (word processing, graphics, database management)



Software Applications

6. **WebApps** (Web applications) network centric software. As web 2.0 emerges, more sophisticated computing environments is supported integrated with remote database and business applications.
7. **AI** software uses non-numerical algorithm to solve complex problem. Robotics, expert system, pattern recognition game playing



What is the difference between software engineering and computer science?

- Computer science is concerned with theory and fundamentals;
- software engineering is concerned with the practicalities of developing and delivering useful software
- Computer science theories are currently insufficient to act as a complete underpinning for software engineering



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 process



What is the difference between software engineering and system engineering?

- System engineers are involved in system specification, architectural design, integration and deployment



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 editing, graphics programs, project management tools; CAD software; software for specific markets such as appointments systems for dentists.



15

Software products

- **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.



What are the attributes of good software?

- The software should deliver the required functionality and performance to the user and should be maintainable, dependable and usable
- Maintainability
 - Software must evolve to meet changing needs



What are the attributes of good software?

- Dependability
 - Software must be trustworthy
- Efficiency
 - Software should not make wasteful use of system resources
- Usability
 - Software must be usable by the users for which it was designed



What are the key challenges facing software engineering?

- Coping with legacy systems, coping with increasing diversity and coping with demands for reduced delivery times
- Legacy systems
 - Old, valuable systems must be maintained and updated



What are the key challenges facing software engineering?

- Heterogeneity
 - Systems are distributed and include a mix of hardware and software
- Delivery
 - There is increasing pressure for faster delivery of software



Professional and ethical responsibility

- 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.



Issues of professional responsibility

- *Confidentiality*
 - Engineers should normally respect the confidentiality of their employers or clients irrespective of whether or not a formal confidentiality agreement has been signed.
- *Competence*
 - Engineers should not misrepresent their level of competence. They should not knowingly accept work which is outwith their competence.



Issues of professional responsibility

- *Intellectual property rights*
 - Engineers should be aware of local laws governing the use of intellectual property such as patents, copyright, etc. They should be careful to ensure that the intellectual property of employers and clients is protected.



Issues of professional responsibility

- *Computer misuse*
 - Software engineers should not use their technical skills to misuse other people's computers.
 - Computer misuse ranges from relatively trivial (game playing on an employer's machine, say) to extremely serious (dissemination of viruses).



Ethics

- Ethics is the branch of philosophy that studies what's right and wrong.
- Ethical rules are rules to follow in our interactions with other people and in our actions that affect other people.



What is Ethics?

- They apply to all of us and are intended to achieve good results for people in general
- Different ethical theories attempt to achieve the same goal:
 - to enhance human dignity, peace, happiness, and well-being.



What is Ethics?

- "Do not lie" is a command of an ethical act, if it complies with ethical rules.
- For example, it is an ethical act not to lie, yet it is unethical to tell the murderer the truth of his victim's whereabouts.



What is Ethics?

- Its principle of universality: We should follow rules of behavior that can universally apply to everyone.
- In other words, "Do unto others as you would have them do unto you" (Luke 6:31).



Legality vs. Ethics

- Illegal acts break the law while unethical acts may not be illegal
- Ethics
 - Branch of philosophy that deals with what is considered right or wrong
 - Right and wrong not always clear
 - Consider
 - Company sells profiles of customers with information collected through cookies
 - Company allows personal use of Web but secretly monitors activity
 - Company knowingly sells tax software with bugs



Code of ethics

- A declaration of principles and beliefs that govern how employees of a corporation are to behave
- Inspirational and disciplinary
- All-encompassing and stable over time



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.



ACM/IEEE Code of Ethics

- The Code contains eight 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.



Code of ethics - preamble

- **Preamble**
 - The short version of the code summarizes aspirations at a high level of the abstraction; the clauses that are included in the full version give examples and details of how these aspirations change the way we act as software engineering professionals.



Preamble

- Without the aspirations, the details can become legalistic and tedious; without the details, the aspirations can become high sounding but empty; together, the aspirations and the details form a cohesive code.



Preamble

- Software engineers shall commit themselves to making the analysis, specification, design, development, testing and maintenance of software a beneficial and respected profession.
- In accordance with their commitment to the health, safety and welfare of the public, software engineers shall adhere to the following Eight Principles:




Code of ethics - principles

- 1. PUBLIC
 - Software engineers shall act consistently with the public interest.
- 2. 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.




Code of ethics - principles

- 3. PRODUCT
 - Software engineers shall ensure that their products and related modifications meet the highest professional standards possible.
- 4. JUDGMENT
 - Software engineers shall maintain integrity and independence in their professional judgment.




Code of ethics - principles

- 5. MANAGEMENT
 - Software engineering managers and leaders shall subscribe to and promote an ethical approach to the management of software development and maintenance.
- 6. PROFESSION
 - Software engineers shall advance the integrity and reputation of the profession consistent with the public interest.




Code of ethics - principles

- 7. COLLEAGUES
 - Software engineers shall be fair to and supportive of their colleagues.
- 8. 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.



Ethical dilemmas

- Disagreement in principle with the policies of senior management
- Your employer acts in an unethical way and releases a safety-critical system without finishing the testing of the system
- Participation in the development of military weapons systems or nuclear systems



Questions

