

CS 303-Software Engineering Lecture 1

By
Sara Rehmat
MS (CS)

General Information

- **Grade Distribution:**

- Assignments: 20%
- Quizzes: 10%
- Midterms (30%)
- Final Exam (40%)

Text Books

- Software Engineering: A Practitioner's Approach, 6th Ed. Roger S. Pressman, McGraw-Hill, 2009
- Software Engineering, 9th Ed. Ian Sommerville. Pearson Education

Course Objectives

- Students will learn the major software engineering issues such as software requirements, design, process, and management.
- Students will also conduct team projects to practice how to apply major software engineering concepts and methods to software development problems.

Course Outline

- Will be shared in the next class

Introduction to Software Engineering

- What is software?
 - Software is: (1) **instructions** (computer programs) that when executed provide desired features, function, and performance; (2) **data structures** that enable the programs to adequately manipulate information and (3) **documentation** that describes the operation and use of the programs.
- What is Engineering?
 - Engineering is all about designing, building and testing things that solve real-world problems.

Introduction to Software Engineering

- What is Software Engineering?
 - **IEEE** defines software **engineering** as: (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 the above statement.
- *Software engineering* is concerned with theories, methods and tools for professional software development.
- 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.

Why Software Engineering?

- Software is everywhere.
- The economies of ALL developed nations are dependent on software.
- More and more systems are software controlled.
- 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 Engineering ensures that good software are developed.

**SO YOU ARE A SOFTWARE
ENGINEER**

**TELL ME MORE ABOUT HOW DOES
IT FEEL TO KNOW EVERYTHING**



Attributes of good software

- Maintainability:
 - Evolve to meet the changing needs of customers.
- Dependability:
 - includes reliability, security and safety.
- Efficiency:
 - responsiveness, processing time, memory utilisation, etc.
- Acceptability:
 - understandable, usable and compatible with other systems that the customers use.

General issues affecting 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

History of Software Engineering

- The notion of “Software Engineering ” first proposed in 1968 at a conference held to discuss “Software Crisis”.
- Projects running over-budget
- Projects running over-time
- Software was very inefficient
- Software was of low quality
- Software often did not meet requirements
- Projects were unmanageable and code difficult to maintain
- Software was never delivered

History of Software Engineering contd..

- The causes of the software crisis were linked to the overall complexity of hardware and the software development process.
- “The major cause of the software crisis is that the machines have become several orders of magnitude more powerful! To put it quite bluntly: as long as there were no machines, programming was no problem at all; when we had a few weak computers, programming became a mild problem, and now we have gigantic computers, programming has become an equally gigantic problem.”

Edsger Dijkstra, The Humble Programmer (EWD340),
Communications of the ACM

Software Engineering, Computer Science, System Engineering

- Computer science focuses on theory and fundamentals; software engineering is concerned with the practicalities of developing and delivering useful software.
- 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.

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

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 – contd..

- 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.
- 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 – contd..

- 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

References

- Software Engineering, 9th Ed. Ian Sommerville. Pearson Education
- Software Engineering: A Practitioner's Approach, 6th Ed. Roger S. Pressman, McGraw-Hill, 2009

Questions