Software Engineering In Computer Science

Introduction

Computer science is the study of computers. It deals with software; a set of instructions that govern a computer on how to perform its tasks and hardware; the tangible physical components of computer elements of a computer system. Through the study of computer science, we learn about computer design, programming, information processing, algorithmic solutions of problems, and the algorithmic process itself. Computer science consists of various areas like networking, software engineering and system security.

Computer science has various areas of specialization as mentioned before. On the area of software engineering and the skills, one gets from majoring in software engineering areas am writing to discuss on.

Difference Between Software Engineering And Computer Science

Software engineering is a 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 deals with the application of theories, methods and tools and try to discover solutions to problems. Software engineering does not only deal with the technical processes but also activities like software project management.

Computer science is concerned with the theories and methods which underlie computers and software systems whereas software engineering is concerned with the practical problems of producing software.(Ian Sommerville, 2001, p. 7)

Categories Of Computer Software

There are seven different categories of computer software as discussed below. We begin with the system software which is the collection of programs written which aid processes of other programs. We also have application software that deals with programs that solve specific business needs. Scientific software deals with algorithms. Embedded software that implements and controls features and functions for the end-user and the system itself.

Product-line software provides specific capabilities for use by different users. Web applications are hypertext files that present information using text and limited graphics. Finally, artificial intelligence software deals with use of non-numerical algorithms to solve complex problems. (Pressman, n.d., pp. 7 & 8)

Processes

The software process is the road map that helps us to create a timely, high-quality result when building a product or a system. The processes provide stability, control, and organization to an activity that can, if left uncontrolled, become quite chaotic. The quality, timeliness and long-term

viability of a product you build are the best indicator of the efficacy of the process that you use. (Pressman, 2009, p. 20)

There are different software process models as discussed. Begin with the waterfall model, it takes the fundamental process activities of specification, development, validation and evolution and represents them as separate process phases. We also have incremental development which interleaves the activities of specification, development and validation. Lastly is the integration and configuration which relies on the availability of reusable components or systems.(Ian Sommerville, 2016, p. 44)

Principle That Guide Practice

Principles that guide practice establish a foundation from which software engineering is conducted. The practice provides you with the details you will need to drive along with the road map. One ensures that has an appropriate method for the work, use automated tools when they are appropriate for the task.(Pressman, n.d., p. 96)

The principles discussed below are fundamental to the practice of software engineering. They are stated in the order; divide and conquer, understand the use of abstraction, strive for consistency, focus on the transfer of information, build software that exhibits effective modularity, look for patterns. When possible, represent the problem and its solution from a number of different perspectives, remember that someone will maintain the software. (Pressman, n.d., p. 100)

Sociotechnical Systems

Sociotechnical systems are enterprise systems that are intended to help deliver a business goal. It might be to increase sales, reduce material used in manufacturing, collect taxes and maintain safe airspace. A sociotechnical system includes two or more technical but crucially the people who understand the purpose of the system. Technical computer-based systems include the hardware and software components but not the procedures and processes. (I. Sommerville, 2011, p. 267)

The sociotechnical system stack consists of the society, organization, business processes, application system, communications and data management, operating system and equipment. Software failures are contained within the enclosing layers of the system stack and do not seriously affect the operation or other layers in the system(I. Sommerville, 2016, p. 293)

Software Project Management

Project management has become vital in every organization. Effective use of project management techniques has resulted into the production of high-quality and reliable products. It also leads to managing resources efficiently, making customers more satisfied. Cause to rise in profit margins. By completing the project within time and budget leads to high productivity.(Sabharwal, 2008, p. 25,26)

Testing Web Applications

WebApp testing is a collection of related activities with a single goal, to uncover errors in WebApp content, function, usability and security. When users encounter an error in the content, they will go for the content and function they need which may fail. Users should learn to eliminate errors the cause failure. (Pressman, n.d., p. 529)

The dimensions of quality as a consequence of good design. The qualities include content is evaluated in both a syntactic and semantic level, the function is tested to uncover errors that indicate lack of conformance to customer requirements. The structure is assessed to ensure that web app is properly delivered, usability, navigability. Performance to ensure that the system is responsive to user interaction, compatibility, interoperability and security. (Pressman, n.d., p. 530)

Architectural Design

It deals with understanding how a system should be organised and designing the overall structures of the system. It can be designed in two levels, architecture in the small and in the large. Architecture in the small is concerned with the way individual programs are decomposed into components. Architecture in large is concerned with the architecture of complex enterprise systems are distributed over different computers owed by different companies.

There are various categories of architectural styles. To begin with, the data centered architectures where data store resides at the centre of this architecture and can be accessed frequently by other components. Data flow architectures is applied when input data are to be transformed through a series of computational or manipulated components into output data. (Pressman, n.d., p. 251)

The call and return architecture enables one to achieve a program structure that is relatively easy to modify and scale. In object-oriented architectures, the components of a system encapsulate data and the operations that must be applied to manipulate the data. Layered architectures accomplish operations that progressively become closer to the machine instruction set.(Pressman, n.d., p. 253)

User Interface Design

This creates an effective communication medium between a human and a computer. User interface begins with the creation of different models of system function. A software engineer establishes a user model, creates a design model, the end user develops a mental image that is often called the user’s mental model or system perception, and the implementers of the system create an implementation model. (Pressman, n.d., p. 321)

The interface design steps are as follows, first, using information developed during interface analysis, define interface objects and actions. Next you define the events that will cause the state of the user interface to change. Depicting each interface state as it will actually look to the end-user follows. Finally indicate how the user interprets the state of the system from information provided through the interface(Pressman, n.d., p. 328).

Reliability Engineering

Reliability is the probability of failure-free operation over a specified time, in a given environment, for a specific purpose. Availability is the probability that a system, at a point of time, will be operational and able to deliver the requested service. System reliability depends on where and how the system is used. Software specializations are often incomplete or incorrect, and it is left to software engineers to interpret how the system should behave.

Reliability requirements are in two types the functional requirements and non-functional requirements. Fault tolerance is a runtime approach to dependability in which systems include mechanisms to continue in operation, even after a software or hardware fault has occurred and the system state is erroneous. To provide fault tolerance, the system architecture has to be designed to include redundant and diverse hardware and software.

Safety Engineering

Safety-critical systems are divided into two classes primary and secondary. Primary safety-critical software is the controller of the system, malfunctioning can cause a hardware malfunction which may in turn result into a human injury or environmental damage. Secondary safety-critical software can indirectly result in an injury, malfunctioning may result in a design fault of the object being designed.(Ian Sommerville, 2016, p. 341)

The software processes used to develop safety-critical software are based on the processes used in software reliability engineering. Requirements and verification errors and omissions may mean that reliability is not enough. A document body of evidence that provides a convincing and valid argument that a system is adequately safe for a given environment.(Ian Sommerville, 2016, p. 361)

Security Engineering

Security is a system attribute that reflects the ability of the system to protect itself from malicious internal or external attacks. These external attacks areas are result of access of computers and mobile devices by outsiders as these devices are networked. There are four types of security threats as discussed below. The interception threats allow an attacker to gain access to an asset, interruption threats allow an attacker to make part of the system unavailable. Modification threats allow an attacker to tamper with a system asset, fabrication threats allow an attacker to insert false information into a system.(Pressman, n.d., p. 378)

Companies access the risks and losses that my arise from certain attack to the system asset. This makes security risk management rather a business than being a technical issue. To make risk management effective a company should have a documented security policy that sets out as follows. The assets that must be protected, the level of protection that is required for different types of assets. The responsibilities of individual users, managers, and the organization, existing security procedures and technologies that should be maintained.(Pressman, 2009, p. 341)

Applications Of Software Engineering

There four areas where software engineering is important. It is a digital learning environment for schools, teachers argue that using interactive software systems to support education can lead

to both improved learner motivation and a deeper level of knowledge and understanding in students. A digital learning environment is a framework in which a set of general-purpose and specially designed tools for learning may be embedded using the system(Ian Sommerville, 2016, p. 32)

A patient information system for mental health care, maintains information about patients suffering from mental health problems and the treatments they have received. It is used to generate management information that allows health service managers to assess performance against local and government targets. It also provides medical staff with timely information to support the treatment of patients.(Ian Sommerville, 2016, p. 35)

A wilderness weather station, to help monitor climate change and to improve the accuracy of weather forecasts in remote areas. It monitors the instruments, power and communication hardware and report faults to the management system. Allows for dynamic reconfiguration where parts of the software are replaced with new versions and where backup instruments are switched into the system in the event if system failure.(Ian Sommerville, 2016, p. 37)

An insulin pump control system, is a medical system that simulates the operation of the pancreas. It ensures that the system shall be available to deliver insulin when required. Also the system shall perform reliability and deliver the correct amount of insulin to counteract the current level of blood sugar.(Ian Sommerville, 2016, p. 32)

Summary

Computer science is a vast topic of study that deals with theories and methods of solving problems. When it comes to specializing into software engineering we get to apply the theories and methods into use and solve the problems we encounter. Computer software is divided into different categories namely system, application, scientific, embedded, product-line, web application and artificial intelligence software.

Processes increase the quality of production in industries and also increase production rate as time is managed. The process models are waterfall, incremental development, integration and configuration models. Practise gives details on the type of method to carry out. Sociotechnical enterprises are intended to help in business goals like increasing sales, reduce material used in manufacturing, collect tax and maintaining efficient airspace.

By completing projects in allocated time and budget causes high productivity. Web application testing is done to uncover errors in web application content, function, usability and security. Architecture design can be in two levels architecture in small for individual programs and architecture in large that deals with complex enterprise systems. User interface creates an effective communication medium between a human and a computer. Reliability creates a failure-free operation.

Safety-critical systems are grouped into two primaries which causes direct injury to hardware and user, and secondary that causes indirect injury. A document body of evidence that provides a convincing and valid argument that a system is adequately safe for a given application in a given environment. System attack can be either internally or externally. Types of security threats are interception, interruption, modification and fabrication. Software engineering can be applied

in learning institutions, wilderness weather stations, in insulin pump control systems and patient information systems for mental health care.

Conclusion

Software engineers can be managers of organizations as they have the skills to succeed such organizations. Users are able to use websites and avoid errors that may cause failure. Through software processes their production of quality product and also good time management. Through software engineering, one is able to put the methods and tools acquired in computer science into us.

References

1. Pressman, R. S. (2009). Software engineering: A practitioners approach (7th ed). London: McGraw-Hill.
2. Pressman, R. S. (n.d.). Software engineering: A practitioner’s approach.
3. Sabharwal, S. (2008). Software engineering : Principles and techniques. new Delhi: New Age International Publishers.
4. Sommerville, I. (2011). Software Engineering (9th ed). London: Pearson.
5. Sommerville, Ian. (2001). Software engineering (6th ed). Harlow, England ; New York: Addison-Wesley.
6. Sommerville, Ian. (2016). Software engineering (10th ed). Boston: Pearson.