Chapter 4 Requirements Engineering

Use cases: identify the actors in an interaction and which describe the interaction itself; A set of use cases should describe all possible interactions with the system; UML sequence diagrams may be used to add detail to use-cases by showing the sequence of event processing in the system.

RE processes: Elicitation, Analysis, Validation, Management. Its an iterative activity in which these processes are interleaved. Requirement elicitation: Start, User requirements elicitation, System req. elicitation; Requirements specification: Business requirements specification, User requirements specification, System requirements specification and modeling; Requirements validation: Feasibility study, Prototyping, Reviews.

Ways of writing a system requirements specification: Natural language, Structured natural language, Design description languages, Graphical notations, Mathematical specifications.

Metrics for nonfunctional requirements: Speed, Size, Ease of use, Reliability, Robustness, Portability

Non-functional requirements (Product requirements, Organizational requirements, External requirements) define system properties and constraints e.g. reliability, response time and storage requirements, and may be more critical than functional requirements.

Requirements completeness and consistency: They should include descriptions of all facilities required. There should be no conflicts or contradictions in the descriptions of the system facilities.

System stakeholder types: End users, System managers, System owners, External Stakeholders

Agile methods: use incremental requirements engineering and may express requirements as user stories which is practical for business systems but not for critical systems.

Functional requirements: Statements of services the system should provide, how the system should react to particular inputs and how the system should behave in particular situations.

Requirements Abstraction: A contract defined its needs in a sufficiently abstract way that a solution is not pre-defined. The requirements must be written so that several contractors can bid for the contract, offering different ways of meeting the client organization’s needs. Once a contract has been awarded, the contractor must write a system definition for the client in more detail so that the client understands and can validate what the software will do. Both of these documents may be called the requirements document for the system.

System requirements: A structured document setting out detailed descriptions of the system’s functions, services and operational constraints. Defines what should be implemented so may be part of a contract between client and contractor.

The process of establishing the services that a customer requires from a system and the constraints under which it operates and is developed.

The system requirements are the descriptions of the system services and constraints that are generated during the requirements engineering process may range from a high-level abstract statement of a service or of a system constraint to a detailed mathematical functional specification.

Ethical principles: PUBLIC, CLIENT AND EMPLOYER, PRODUCT, JUDGMENT, MANAGEMENT, PROFESSION, COLLEAGUES, SELF.

Web software engineering: software reuse is the dominant approach for constructing web-based systems. Incremental and agile development.

Ethical behavior is more than simply upholding the law but involves following a set of principles that are morally correct.

Competence: engineers should not misrepresent their level of competence. They should not knowingly accept work which is out with their competence.

Engineering discipline: Using appropriate theories and methods to solve problems bearing in mind organizational and financial constraints.

Essential attributes of good software: Maintainability, Dependability (reliability, safety and security), Efficiency (responsiveness, processing time, memory utilization), Acceptability

The development of interactive web-based systems and mobile apps which require a blend of software and graphical design skills. 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.

Good software should deliver the required functionality and performance to the user and should be maintainable, dependable and usable.

Software engineering is an engineering discipline that is concerned with all aspects of software production.

Difference between software engineering and computer science: 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.

The fundamental software engineering activities: software specification, software development, software validation and software evolution.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Generic products – Stand-alone system: CAD, appointments systems for dentists.

Customized products – Commissioned by a specific customer to meet their own needs: traffic monitoring systems.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

UML - A visual language for specifying, constructing and documenting the artifacts of systems.