Beirut Arab University

Faculty of Sciences

Department of Mathematics and Computer Science

CMPS303 : Software Engineering

Fall 2011

Summarizing of 20 topics

by

Salah Shami, Ziad Abdullah, Ali El-Zaart

and

CMPS303 Students

1. Waterfall model
Def: The waterfall model describes a development method that is linear and sequential.
Advantage: Users get a feel for the actual system, developers get to build something immediately, and specifications can be developed incrementally
Disadvantages: Developers may make implementation compromises in order to get a prototype working quickly, the process in not visible (few documents that reflect every version of the system), systems poorly structured
Application: Appropriate when requirements are well-understood, changes will be fairly limited during the design process, large interactive systems, mostly used for large systems engineering projects.
Relationship with..: all Software Development Models.

2. Evolutionary Models
Def: The Evolutionary EVO development model divides the development cycle into smaller, incremental waterfall models in which users are able to get access to the product at the end of each cycle
Advantage: Benefit not only business results but marketing and internal operations as well, reduction in risk for software projects and in costs by providing a structured, disciplined avenue for experimentation. EVO allows the marketing department access to early deliveries, facilitating development of documentation and demonstrations. Better fit the product to user needs and market requirements. Increase management visibility of project progress.
Disadvantages: Lack of process visibility, systems are often poorly structured;
Application: For small or medium-size interactive systems, For parts of large systems (e.g. the user interface), For short-lifetime systems
Application: useful when staffing is unavailable for a complete implementation
Relationship with..: water fall, prototyping models.

3. The incremental model/ Iterative and Incremental development
Def: The incremental model combines elements of the linear sequential model (applied repetitively) with the iterative philosophy of prototyping
Advantage: Customer value can be delivered with each increment so system functionality is available earlier, early increments act as a prototype to help elicit requirements for later increments, lower risk of overall project failure. The highest priority system services tend to receive the most testing.
Disadvantages: Increments should be relatively small (20,000 lines of code), Can be difficult to map the customer's requirements onto increments of the right size, Hard to identify common functions
Application: Incremental development is particularly useful when staffing is unavailable for a complete implementation by the business deadline that has been established for the project, increments can be planned to manage technical risks.
Relationship with..: water fall, prototyping models.

4. The spiral model
Def: It provides the potential for rapid development of incremental versions of the software. Using the spiral model, software is developed in a series of incremental releases. During early iterations, the incremental release might be a paper model or prototype. During later iterations, increasingly more complete versions of the engineered system are produced.
Advantage: Risks are explicitly assessed and resolved throughout the process.Software engineers can start working on the project earlier rather than wading through a lengthy early design process.
Disadvantages: Requires highly skilled people in risk analysis and planning, Requires more time, and is more expensive, Estimates of budget and time are harder to judge at the beginning of the project since the requirements evolve through the process
Application: The spiral model is a realistic approach to the development of large-scale systems and software
Relationship with..: spiral model is an evolutionary software process model that couples the iterative nature of prototyping with the controlled and systematic aspects of the linear sequential model..

5.Requirements
Def: The descriptions of the system services and constraints that are generated during the requirements engineering process.
Advantage: they are the basis of all software applications as they focus on what the application must d.
Disadvantages: Problems arise when requirements are not precisely stated (Ambiguous), Ambiguous requirements may be interpreted in different ways by developers and users.
Application: they are elicited from the customer outlining what they wish the system to do, and recorded in a language the customer understands. A requirements document then becomes the contract between the customer and the software developers on the product that will be delivered
Relationship with..:. The first phase in the software development life cycle

6. System Sequence Diagrams
Def: A system sequence diagram is a picture of one particular scenario of a use case. It showsthe events that external actors generate, the order in which the events occur, and how the events interact.
Advantage: they are the basis of all software applications as they focus on what the application must d.
Disadvantages: Problems arise when requirements are not precisely stated (Ambiguous), Ambiguous requirements may be interpreted in different ways by developers and users.
Application: they are elicited from the customer outlining what they wish the system to do, and recorded in a language the customer understands. A requirements document then becomes the contract between the customer and the software developers on the product that will be delivered
Relationship with..:. the first phase in the software development life cycle

7. Testing
Definition: testing is a process in SE that intends to show that a program does what it is intended to do and to discover program defects before it is put into use
Advantage: demonstrate that the software meets its requirements, discover if the behavior of software is incorrect, undesirable or does not conform to its specification
Disadvantage: the tester may be biased by previous experience, and the test value may not cover all possible value, interface errors are discovered late
Application: validation testing, and defect testing
Relationship with: verification and validation of code implemented

8. Maintenance
Definition: modifying a program after it has been put into use, modifying existing components and adding new components to the system
Advantage: needed for better quality software, reusing the application that will reduce cost
Disadvantage: If any error is present in the test script, sometimes it may lead to deadly consequences, it is costly in case of playback methods
Application: for long-term enhancement for a normal business activities
Relationship with: Evolutionary Development Model

9. Deployment
Definition: makes the system operational through installation of system and also focuses on training of user, configuring user access privileges
Advantage: manage activities such as release, install and activate, deactivate, update, security, uninstall
Disadvantage: Overall performance can be slower as one application server machine has to do lots of work
Application: pre-production environments such as deployment coordinators, production environments such as database administrator
Relationship with: Maintenance

10. Verification
Definition: it is a process conforming the software to its specification
Advantage: it shows defects that have been detected and corrected
Disadvantage: costs for critical systems
Application: formal verification, model checking, automated program analysis
Relationship with: Testing

11. Validation
Definition: is a process that demonstrate the developer and the system customer meet their requirements and work as intended successfully
Advantage: records of static analyses of the source code, involved with safety-related system development
Disadvantage: involves additional validation process and analysis than for non-critical systems
Application: reflect operational profile, security validation
Relationship with: Testing

12. Class Diagram/ Domain Diagram
Definition: is an illustration of the relationships and source code dependencies among classes in UML
Advantage: helps the analyst to understand the functionality of the system and models, generate a complete or partial system implementation from the system model
Disadvantage: focusing on a static view of the system
Application: developing object-oriented system model
Relationship with: system requirements specification, Use Cases description

13. Critical Path Method
Definition: is an algorithm for scheduling a set of project activities
Advantage: makes dependencies visible, organize large and complex projects, enables the calculation of the float (slack) of each activity
Disadvantage: for large and complex projects, there’ll be thousands of activities of dependency relationships
Application: Project management applications
Relationship with: Schedule feasibility Study

14. Use Case
Definition: show the interactions between a system and its environment, typically defining interactions between a role and a system
Advantage: capture requirements, easy to understand, help manage the complexity of large projects, provide good basis for the verification and validation of functional requirements
Disadvantage: lack of formality in the definitions, each use case captures a major functional abstraction that can cause numerous problems with functional decomposition
Application: end-to-end timing requirements for real-time applications
Relationship with: scope of the system, and sequence Diagram

15. Agile Development
Definition: agile methods universally rely on an iterative approach to software specification, development and delivery
Advantage: They are intended to deliver working software quickly to customer, who can then propose new and changed requirement to be included in later iterations of the systems
Disadvantage: the principles (customer involvement, incremental delivery, embrace change) are difficult to implement
Application: support business application development where the system requirements usually changed rapidly during the development process
Relationship with: iterative development and incremental development

16. Project management
Def: Project management involves the planning, monitoring, and control of the people, process, and events that occur as software evolves from a preliminary concept to an operational implementation..
Advantage: Project management software is designed to help business teams cost effectively complete a project on time. Keeping a project on track throughout its life cycle can help save money by eliminating the chance for a missed deadline.
Disadvantages: he disadvantages of Project Management can be grouped into 3 main categories: **overhead(***Cost overhead*: Project Management costs money, *Communication overhead*: Project Management introduces another layer of communication between management and team members, *Time Overhead*:), **obsession(***Methodology obsession Process obsession Stakeholder obsession*°,, and **non-creativity(***Technical non-creativity*: Project Management imposes deadlines on resources, *Managerial non-creativity*).
Application: used for large systems engineering projects.
Relationship with..: all Software Development Models.

17. Architecture
Def: The design process for identifying the sub-systems making up a system and the framework for sub-system control and communication.
Advantage: Stakeholder communication: may be used as a focus of discussion by system stakeholders. System analysis has a profound effect on whether the system can meet the critical requirements (performance, etc). Large-scale reuse: may be reusable across a range of systems.
Disadvantage: Using large-grain components improves performance but reduces maintainability.
Introducing redundant data improves availability but makes security more difficult. Localizing safety-related features usually means more communication so degraded performance.
Application: Large systems rarely conform to a single architectural model.
Relationship with..: System organization, Decomposition style, Control style

18. Analysis
Def: It uses a combination of text and diagrammatic forms to depict requirements for data, function, and behavior in a way that is relatively easy to understand, and more important, straightforward to review for correctness, completeness, and consistency.
Advantage: To validate software requirements, you need to examine them from a number of different points of view. Analysis modeling represents requirements in three “dimensions” thereby increasing the probability that errors will be found, that inconsistency will surface, and that omissions will be uncovered.
Disadvantage: Systems must be defined up front.
Relationship with..: feasibility study

19. Design
Def: is a process of problem solving and planning for a software solution. After the purpose and specifications of software are determined, software developers will design or employ designers to develop a plan for a solution.
Advantage: a software design may be as simple as a flow chart or text describing a planned sequence of events, output of the analysis of a software problem
Disadvantage: corrections must often wait for the maintenance phase.
Application: required in all
Relationship with..: Analysis

20. Implementation
Def: is the stage in the software engineering process at which an executable software system is developed, is the process of realizing the design as a program
Advantage: Reusable in the coding, configuration management
Disadvantage: critical and peripheral modules not distinguished
Application: required in all
Relationship with..: Design