

Adrian Menezes
9624
TE Comps B

SE Assignment -1

Q1)

As technology changes, user requirements & environment on which software is working also changes. So every org is ranked based on the software engg principles used. Implementing & maintaining large size of software programmes require a specific method modularize the task so that size of software can't harm software quality. Software engg provides methodology for implementing complex software systems with high quality. Without any std method or management, it is difficult to address defect in product & correct them as early as possible. SE provides this functionality. Extending previous software to add new functionality requires more cost in terms of time & effort to develop, as compared to process developing new software to provide functionality. SE provides a way in which software system can be able to scale as needed in future.

Q2)

Waterfall Model: sequential & linear approach

- Each ~~phase~~ phase must be completed before moving to next one
- Clear & structured, suitable for project with well defined requirements, minimal changes & stable scope
- limited flexibility for changes, difficult to adapt to evolving requirements, potential for late-stage error discovery.

* V-Model: Parallel development & testing approach. Each dev phase is followed by a corresponding testing phase

- Strong emphasis on validation & verification, clear documentation

reduces risk by identifying issues early.

- limited adaptability to changing requirements, potential for miscommunication betn development & testing phases.

* Incremental Model :- Similar to iterative models but the software is built in increments each delivering specific functionality

- Early delivery of fn modules reduced time to market allows for better integratn testing
- Requires careful planning to define increments, possible integratn challenges

* Iterative Model :- Similar to agile but with more structured & defined phases. Each iteratn may include a subset of software functionality

- Allows for iteratn refined features & early feedback suitable for progress with evolving requirements
- Requires clear planning & co-ordinatn between iteratn, potential for ^{creep} scope

- Q3]. The CMM model applicatn in software develop has sometimes been problematic. Applying models ~~that~~ aren't integrated within & across an org could be costly in training appraisal & improved activities
- The capability maturity model integratn (CMMI) project was formed to sort out problem of using multiple models for software develop process, thus the CMMI models has superseded the CMM model, though the CMM model continues to be a general theoretical process capability model used in public domain.
 - CMMI framework consists of a collectn of computer programs based on knowledge engg, ~~SE~~ SE, integrated product & process development & provider sourcing.
 - CMMI framework has 3 grps as
 - 1) CMMI for develop (CMMI-DEV)
 - 2) CMMI for service (CMMI-SVC)
 - 3) CMMI for acquisatn (CMMI-ACC)

Q4] Prescriptive Process Model

- 1) Developed to bring order & structure to software develop process
- 2) Accommodate changing requirements
- 3) More popular
- 4) Waterfall model & incremental model are a few examples of prescriptive process model

Evolutionary process Model

- 1) Stages consists of growing increments ^{evolutn} of an operational software product with
- 2) Improvement required in product
- 3) Less popular
- 4) Spiral & prototyping model & RAD model.

- Q5) • Incremental Model: When a project can be divided into smaller f^n increments, allowing certain modules to be developed & delivered independently while ensuring integratⁿ & testing along the way
- RAD model: When there is a need to quickly produce a working prototype to gather user feedback & make refinement before proceeding with full development.
- Waterfall model: When req. are stable & changes are minimal, making it possible to plan & execute project in linear seq. phases
- Agile method (SCRUM): When flexibility & adaptability are crucial & project can be divided into smaller increments with freq. iteratⁿ allowing for continuous feedback & changes

- Q6) Waterfall model: • First approach used software develop process. It's also called as classical life cycle model or linear sequential model. In waterfall model or linear seq. model. In waterfall model any phase of develop process begins only if prev phase is completed. Agile software develop describes an approach to software develop unclear which req. and solⁿ evolve through collaborative effort of self-organising & cross f^n teams & their customers. It advocates adaptive planning, develop, early delivery, & improvements encourages rapid & flexible responses to change. The term agile was popularized in this context by the manifesto for agile software development.

- Q7) Waterfall: Development speed: linear & sequential methodology where each phase must be completed before moving onto the next. This can lead to larger development cycles.

Metrics:- Time taken for each phase

Adaptability to change:- Less adaptable due to rigid structure

Metrics: No. of change req., impact analysis time & delays caused by change request.

Customer satisfactn: Less customer involvements till end, which affects satisfactn

Metrics: Customer feedback at end of prog post-deployment support requirement

2) Agile (Scrum & KANBAN):- Development speed: Incremental develop, quicker delivery

Metrics: No. of user stories completed per sprint, cycle.

Adaptability to change: Highly flexible, due to regular iteratn

Metric: No. of changes incorporated per sprint. Time taken to respond to change

Customer satisfactn: Continuous customer feedback & collab, increased satisfactn

Metrics: Regular customer feedback scores, frequency of customer involvement.

Q3] Features	Waterfall model	Incremental model	Prototyping model	Spiral model
Requirement Specs	well understood	not well understood	not well understood	well understood
Understanding req	well understood	not well understood	not well understood	not well understood
Availability of reusable components	NO	YES	YES	YES
Risk analysis	only at beginning	no risk analysis	no risk analysis	YES
User involvement	only at beginning	intermediate	high	high
Implementation time	long	less	less	depends on project
Flexibility expertise required	rigid high	less high	high medium	flexible high
Cost ctrl	yes	no	no	yes
Resource ctrl	yes	yes	no	yes