SE Assignment - 1

Q1)

As the technology changes the user requirements & environment on which software is working also changes. so every organization is based on software engineering principles used by organization.
Implementing & managing large size of software. Prommaging requires a specific method modulize the tasks so that size of software can't harm the software quality.

software engineering provides methoddology for impleming complex software systems with high quality.
without any standard method or management it is difficult to address defects in the product & correct them.
extending the previous software to add new functionali requires more cost in terms of time to develop & efforts taken by people as compare to the process of developing new software to provide that functionality.

Software engineering provides a way in which software system can be able to scale as needed in future.

(Q2) waterfall model -

It is a sequential & linear approach. Each Phase must be completed before moving to the next one.
clear & structured, suitable for projects with well defined requirements, minimal changes & stable scope.
limited flexibility for changes, difficult to adapt to evoluling requirements, potential for late-stage a error discovery.

V-model - It is a parallel development & testing approach. Each development phase is followed by corresponding testing phase. Strong emphasis on validation & verification, clear documentation, reduces risk by identifying issues early.
Limited adaptibility to changing requirements, potential for miscommunication between development & testing phases

Incremental model - similar to interactive models, but software is built in increments, each delievering specific functionality.
Early delivery of functional modules, reduced time to market, allows for better integration testing.
requires careful planning to define increments, possible integration challanges

Iterative model - similar to agile, but with more structured & define phases. Each iteration may include a subset of the softwares functianality. Allows for Iteration, refined features & early feedback, suitable for projects with evolving requirements.

(Q3) The CMM models application in software development has sometimes been problematic. Applying multiple models that are not integrated within & across an organization could be costly in training appraisals & improvement activities.
The capability maturity model integration (CMMI) project was formed to sort out the problem of using multiple models for software development processes, thus the CMMI model has superseded the CMI model, though the CMM model continues to be general theoritical process capability model used in the public domain.

CMMI framework consists of a collection of computer programs based on knowledge, engineering, software engineering, integrated product & process development & provider sourcing.

- CMMI framework has three groups as:
1) CMMI for development (CMMI - DEV)
2) CMMI for service (CMMI - SVC)
3) CMMI for aquisation (CMMI - ACQ)

Q4)

perospective process model	Evolutionary process model.
1) Developed to bring order & structure to the software development process.	1) stages consists growing increements of an operational software product with evolution.
2) It can accommodate changing requirement.	2) Improvement is required in the product.
3) It is more popular.	3) It is less popular.
4) eg. Water fall model & incremental models	4) eg. spiral model RAD model.

5)

Incremental model-

when a project can be divided into smaller functional increments, allowing certain modules to be developed & delivered independently while ensuring integration & testing along the way

RAD model - When there is need to quickly produce a working prototype to gather user feedback & make requirements before proceeding with full development.

Waterfall model - when requirements are stable & changes are minimal, making it possible to plan & execute the project in a linear sequence of phase

Agile model - when flexibility & adaptibility are crucial & the project can be divided into smaller increments with frequent interactions, allowing for continuous feedback.

Q6)

Waterfall model is first approach used in software development process.
It is also called as classical life cycle model or linear sequential model.
In waterfall model any phase of development process begins only if previous phase is completed.

Agile software development describes an approach to software development under which requirements & solutions. evolve through the the collabarative effort of self-organising & cross functional teams & their customers.

Q7)

1) waterfall

Development speed:

waterfall is a linear & sequential methodology where each phase must be completed before moving on next.

matrics: Time taken for each phase

adaptability to change -

waterfall is less adaptable to changes in requirements due to rigid structure.

matrics: number of change requests, impact analysis time & delays caused by change requirements

customer satisfaction -

waterfall may have limited customer involvement until the end, which could be affect satisfaction.

matrics - customer feedback at the end of project Post deployment.

2) Agile.

Development speed:

Agile methodologies, emphasise incremental development, allowing for quicker delivery of working features

matrics: number of user stories completed per sprint or cycle.

Adaptability to change:
Agile methodologies are highly adaptable to changing requirements due to regular interations & flexibility.
metrics: number of changes incorporated for sprint / cycle time to respond to change requests.

Customer satisfaction.
It involve continuous customer feedback & collaboration leading to improved satisfaction
metrics: regular customer feedback scores, frequency of customer involvement.

8)

Features	waterfall model	incremental model	Prototyping model	spiral model
requirement specification	well understood	not well understood	not well understood	well understood
understanding requirements	well understood	not well understood	not well understood	well understood
Availability of reusable components	NO	yes	yes	yes
Risk analysis	only at beginning	NO	NO	yes
User involvement	only at beginning	inter-mediate	high	high
implementation time	long	less	less	depends on Project
flexibility	Rigid	less	high	flexible
expertise required	high	high	medium	high
const control	yes	NO	NO	yes
Resource Control	yes	yes	NO	yes