Shruti Patil
9567

Page No.:
Date:

SE Assignment 1

1) What is significance of recognizing software requirements in software engineering process?

→ Requirements engineering is the process ~~that~~ of identifying, eliciting, analyzing, specifying, validating and managing the needs and exceptions of stakeholders for a software system.

It is an iterative proces that involves several steps:-

1) Requirements Elicitation:- This step involves ~~seve~~ interviews surveys, focus groups and techniques to gather information from stakeholder about the needs and expectation.

2) Requirements Analysis:- This step involve analyzing the gathered information to identify the high-lev goals and objectives, any constraints or limitations of the software system.

3) Requirement specification:- This step involve documenting the requirements in a clear, consistent and unambiguous manners. It involves prioritizing and grouping the requirements into managable chunks

4) Requirement validation:- This step involves checking that the requirements are complete, consistent and accurate it also involves checking that the requirement are testable and that they meet the need and expectations of stakeholder.

5) Requirement management:- This step involves managing t requirements throughout the software development life cycle including tracking and controlling changes and ensurin that the requirements are still valid and relevant.

Recognizing software requirements in software engineering provides a solid foundation for the development process which helps to reduce the risk of failure and ensure that the system is delivered on time, within budget and required quality standards.

2) Describe the main characteristics of different process models used in software development.

→ Software process models is an abstraction of the software development process. It is a representain of the order of activities of the process and the sequence in which they are performed.

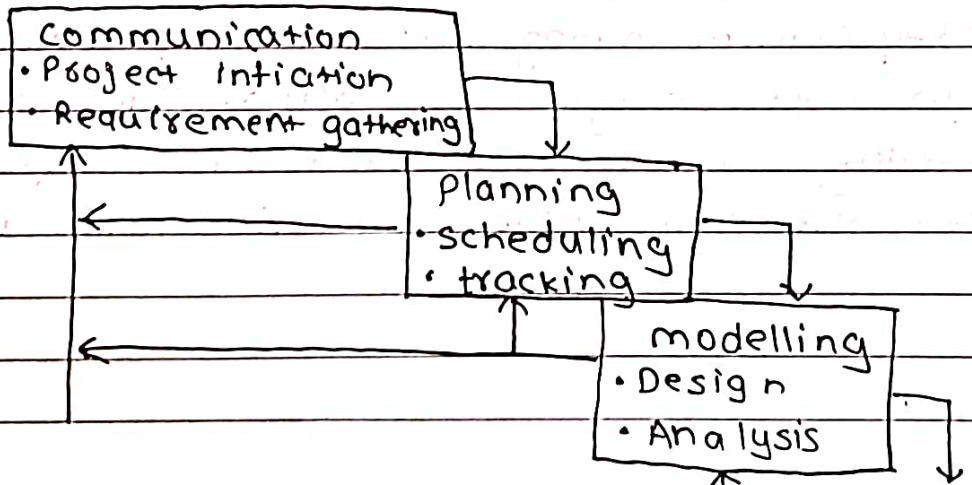
The 5 generic process framework activities

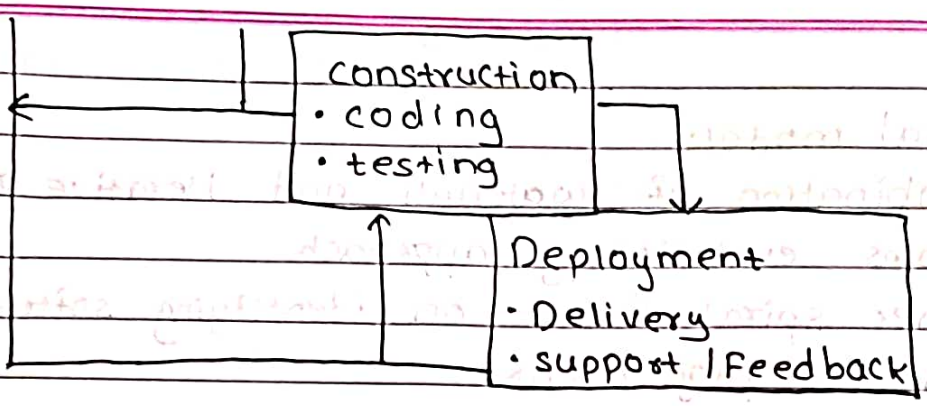
1) Communication
2) Planning
3) Modelling
4) Construction
5) Deployment.

a) Waterfall model :-

It is sequential, plan driven-process where you must plan and schedule all your activities before starting the project. It does support iteration, so changes can cause confusion.

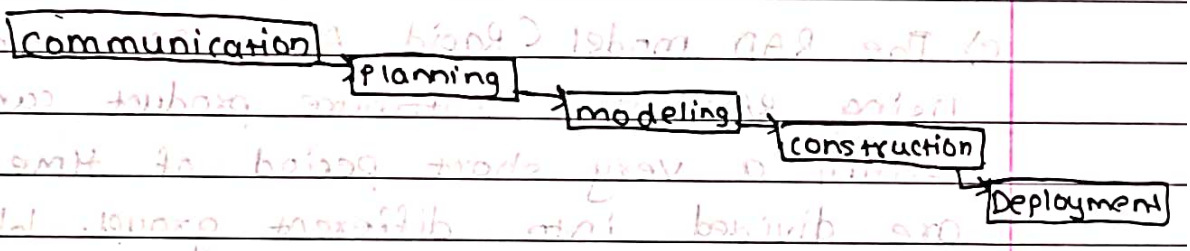
Water model with feedback :-



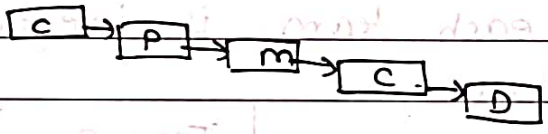


b) Incremental model :- Similar to iterative models but the software is built in increments, each delivering specific functionality. It is efficient as the developers only on focus on what is imp and bugs are fixed as they arise, but you need a clear and complete definition of the whole system before you start.

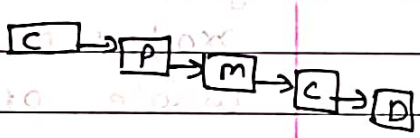
Increment # 1



Increment # 2



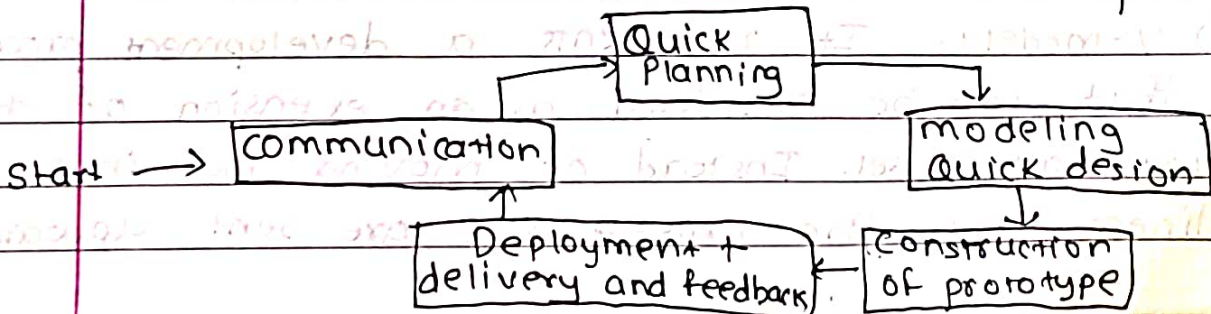
Increment # 3



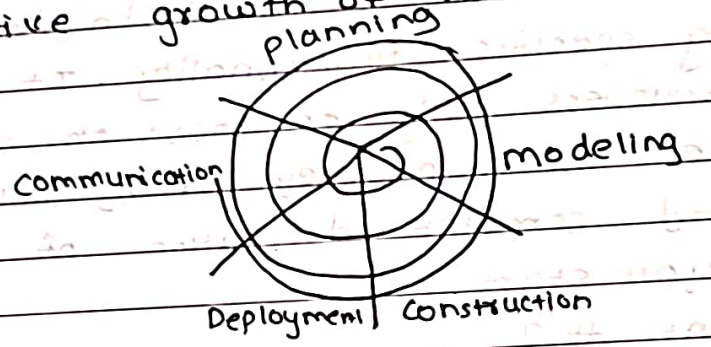
c) Prototyping model :-

Follows an evolutionary and iterative approach used when requirements are not well understood. Focuses on those aspects of software that are visible to the customer/user.

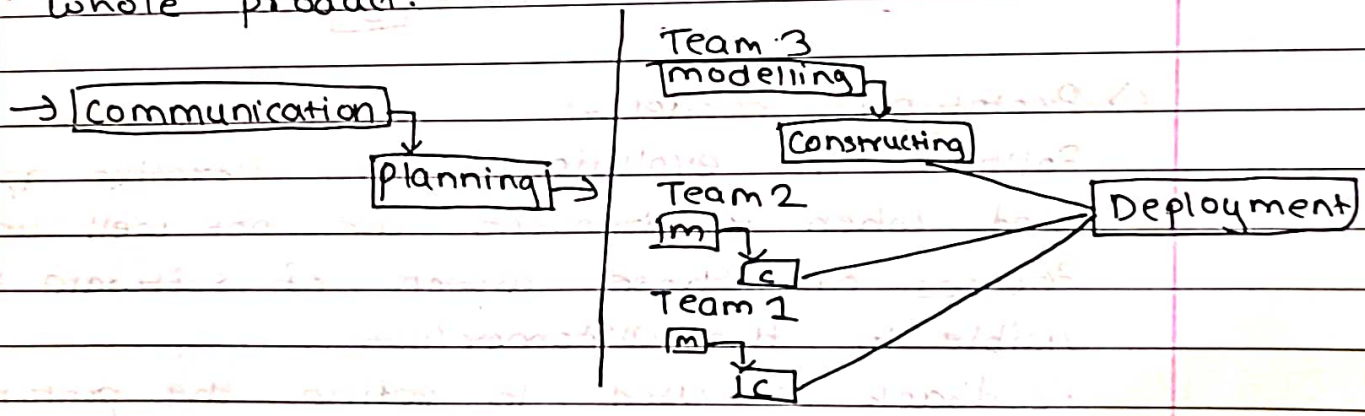
Feedback is used to refine the prototype



d) Spiral model:-
combination of waterfall and iterative prototyping
follows evolutionary approach.
Inner spiral focuses on identifying software requirements and project risks.
Outer spiral take on a classical waterfall approach after requirements have been defined, but permit iterative growth of the software.

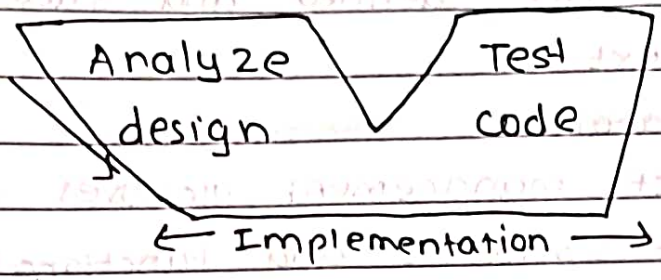


e) The RAD model (Rapid Action Development)
Using RAD model, software product can be developed within a very short period of time. Requirements are divided into different groups. When all groups are ready with their final product the product of each team is integrated to form a whole product.



f) V-model :- It represents a development process that can be considered as an extension of the waterfall model. Instead of moving down in a linear way, the process steps are bent upward

after coding phase.



g) Iterative model:- Each iteration may include a subset of the software's functionality

Q3) How does the capability Maturity Model (CMM) contribute to improving software development processes?

→ CMM was developed by the Software Engineering Institute (SEI) at carnegie mellon university in 1987. It defines the process characteristics that should exist if an organization wants to establish a software process that is complete.

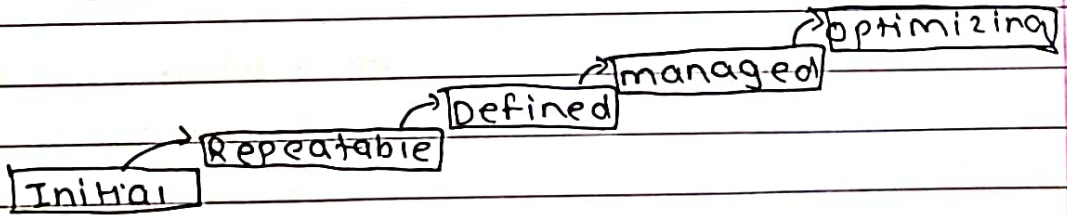
The SEI CMM presents 2 types of metal models:-

As a continuous model

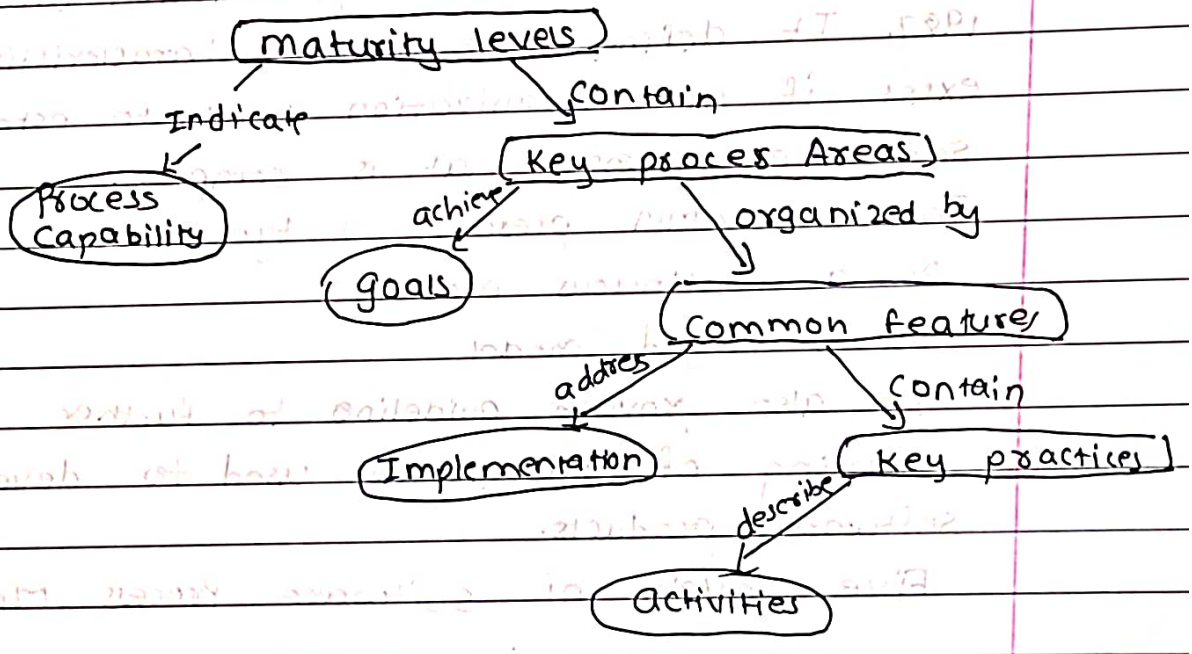
As a staged model.

It also provides guideline to further enhance the maturing of the process used to develop those software products.

Five levels of Software Process Maturity:-



Level 1 :- Initial
Few processes are defined and success depends o individual effort.
level 2 :- Repeatable
Basic project management processes are establish to track cost, shedule and functionality.
Level 3 :- Defined
All projects use an approved, tailored version of th organizations standard software proces for develop and maintaining software.
level 4 :- manages.
Detailed measure of the software processes and product quality are collected.
Level 5 :- Optimizing
continuous process improvement is enabled by quantitative feedback from the proces and from piloting innovative ideas and technologies



4) Explain the differences between perscriptive process models and evolutionary process models.

→

Prescriptive models	Evolutionary models
1) Developed to bring order and structure to the software development process	1) Evolutionary software process do not establish the max speed of the evolution. Due to this development process becomes slow.
2) Defines a distinct set of activities, actions, tasks that are required to engineer high quality software	2) Evolutionary process models lack flexibility, extensibility and high quality
3) more popular	3) less popular
4) Provides complete and full developed systems.	4) Time does not allow a full and complete system to be developed.
5) ex:- waterfall model, incremental model	5) eg:- prototyping, spiral and concurrent models
6) It can accomodate changing requirement	6) Improvement is required in the product

05) Provide examples of situations where using a specific process model would be more suitable.

→ a) Waterfall model:- It is suitable for projects with well-defined and stable requirement, where changes are unlikely. For ex:- when developing software for critical systems like medical devices or aerospace apps,

where changes can be expensive and risky.

b) V-model (validation or verification) :- For projs that require a strong focus on testing and val like safety systems or financial software, the v-m ensures a systematic approach to verifying an validating each development phase in parallel.

c) Incremental model :- When a project has a tight schedule but the full set of requirements can't b defined upfront the incremental model works wel It's suitable for projects involving e-commerce platforms where features can be developed and delivered in stages.

d) RAD model :- It is useful in scenarios where there's a need for rapid software development and quick delivery typically in busines-focused projects. for. eg :- Customer relationship management (CRM) system for a sales team that requires constant updates and improvements to stay competitive

e) The prototype model :- It is beneficial when the client's requirements are not well-defined or are subject to frequent changes. When creating a user interface for a new mobile app, using a prototype allows stakeholder to visualize the design early on and provide feedback for requirement before the final product is developed

f) Spiral model :- In situations where the project involves a high level of risk assesment and continuous refinement such as large scale government projects, this model provides a structured approach for iterative development while managing risks effectively.

g) Agile model :- Agile methologies like scrum are beneficial when requirements are constantly evolving and the development team needs to adapt quickly Its useful for projects involving web development, mobile apps where iterative development and customer feedback are crucial.

26) Compare and contrast the waterfall model and Agile methodologies in terms of project planning and progress tracking

→ The waterfall and Agile models differ significantly in terms of project planning and progress tracking

1) Waterfall model :-

a) Project Planning :- It follows sequential approach planning is done extensively at the beginning of the project, where all requirements are gathered and documented before development begins.

b) Progress tracking :- progress is tracked through predefined milestones each phase must be completed before moving to the next, making it challenging to accomodate changes later in the project.

2) Agile methodologies :-

a) Project Planning :- Agile approaches (like scrum or kanban) emphasize flexibility and adaptability. Planning is done in iterations, with focus on delivering smaller, functional increments of the projects.

b) Progress tracking :- Agile relies on frequent iterations and feedback loops. Teams track progress using tools like burndown charts, velocity measurements and daily stand up meetings. This allows for course correction and adjustment as needed.

Waterfall model follows a rigid, linear planning process with limited room for changes, while Agile methodologies promote iterative planning and ongo adjustments based on regular feedback, fostering a more adaptable approach for project management.

Q7) Apply process metrics to evaluate the efficiency and effectiveness of waterfall agile (both scrum and kanban) methodologies, considering factors such as development speed, adaptability to change and customer satisfaction.

→ Project metrics refer to quantitative measures used to access various aspects of a sofware development projects progres, quality, efficiency and performance. It provides valuable insights into the project's health.

1) Waterfall:-

a) development speed :- longer development cycles might lead to slower speed compared to agile methodologies.

b) Adaptibility to chang:- It is less adaptable to change due to its sequential nature, which can lead to challange when change is required.

c) Customer satisfaction:- since change can't be easily incorporated mid-proces, customer satisfaction might be impacted if initial requirements do not meet ther evolving needs

2) Agile (scrum):-

a) Development speed:- short iterations (sprints) can result in faster development speed and more frequent release.

b) Adaptibility to change:- Agile methodologie like scrum are designed to embrace change,

allowing teams to adjust requirement between sprints
c) Customer satisfaction:- frequent feedback loops and incremental improvements often lead to higher customer satisfaction.

3) Agile (Kanban):
a) development speed:- It is focused on continuous delivery which results in steady development speed as work items are pulled through the process
b) Adaptability to change :- It is flexible and allows for changes to be introduced at any point, making it adaptable to shifting requirements.
c) Customer satisfaction:- Continuous delivery and focus on flow can lead to better customer satisfaction by quickly addressing their needs

Q8)

These factors helps in evaluating and selecting a suitable model for a given project, considering its unique characteristics and goals. Each model has its strength and weakness making this comparison relevant for making informed desision in software development.