

SEPM

Q1. What is software. Explain its components and Features in Detail?

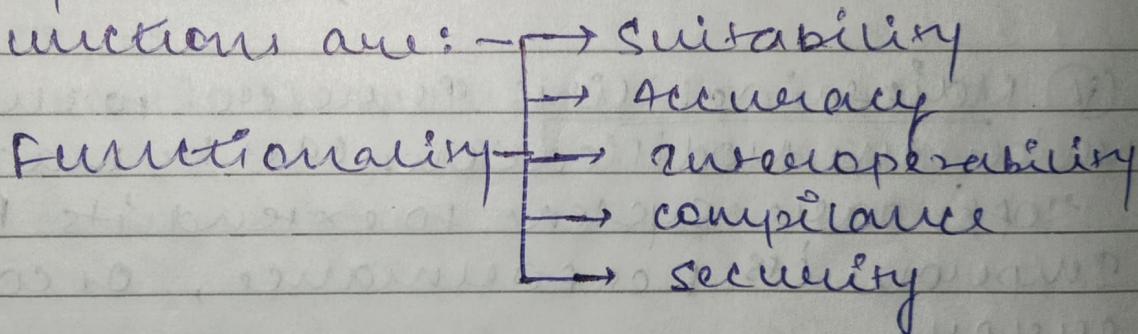
Ans. Software is defined as a collection of computer programs, procedures, rules, and data.

Software characteristics are classified into 6 major components:-

Features of software :-

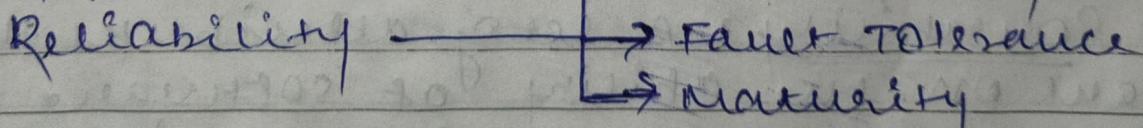
(1) Functionality : It refers to the degree of performance of the software against its intended purpose.

Required functions are: -



(ii) Reliability: A set of attributes that bears on the capability of software to maintain its level of performance under the given condition for a stated period of time.

Required functions are: -



(iii) Efficiency : It refers to the ability of the software to use system resources in the most effective & efficient manner. The software should make effective use of

(2)

Storage space & executing command as per desired timing requirements.

Efficiency → in time
→ in Resources

(IV) Usability: It Refers to the extent to which the software can be used with ease. the amount of effort or time required to learn how to use the software.

Usability → Understandability
→ Learnability
→ Operability

(V) Maintainability: It refers to the ease with which the modifications can be made in a software system to extend its functionality, improve its performance, or correct errors.

Maintainability → Testability
→ Stability
→ Changeability
→ Operability

(VI) Portability: A set of attributes that focus on the ability of software to be transferred from one environment to another, without or minimum changes.

Portability → Adaptability
→ Usability
→ Replaceability

There are 3 components of software:-

Software components

(1) Programs —

- source code & object code
- means of software
- one part of entire software / subset of software

➤ Program is a list of instructions that tell a computer what to do.

(2) Documentation —

Source info. about the product contained in design documents, detailed code components, etc.

- contains various manuals (User Or Operational Manual)
- cited at the end of each please.

(3) Operating procedures —

Step by step instructions compiled by an organization to help workers carry out complex Routine Operations.

- Instructions for the initial setup of use of software system.
- consist of some troubleshooting instructions.
- It can be given in manuals.

Q2. Describe software process. Explain software process framework.

Ans. A software process (software methodology) is a set of related activities that leads to the production

(4)

of the software.

These activities may involve the development of the software from scratch, or, modifying an existing system.

Framework is a standard way to build & deploy applications.

Software Process Framework is a foundation of complete software engineering process.

- SPF includes all set of unique activities.
- It also includes no. of framework activities that are applicable to all software projects.

Process Framework

Framework Activities

Task Sets

Task

ACTIONS

Milestones, Work products

Task Sets

QA Points

Unique activities

SPF

Software Process Framework (SPF)
A Generic process framework encompasses
5 activities which are given below
one by one:-

- (1) Communication - In this, Heavy communication with customers & other stakeholders, Requirements gathering is done.
- (2) Planning - In this, we discuss the technical related tasks, work schedule, risks, required resources etc.
- (3) Modeling - This is about building representations of things in the 'real world'. In modeling activity, a product's model is created in order to better understanding & requirements.
- (4) Construction - In the software eng., construction is the application of set of procedures that are needed to assemble the product.
we generate the code & test the product in order to ^{make} better understanding of requirements product.
- (5) Deployment - In this, a complete or non-complete products or software are presented to the customers to evaluate & give feedback. On the basis

(6)

of their feedback we modify the products for supply better product.

Universal activities include -

- ① Risk Management
- ② Software quality assurance (SQA)
- ③ Software configuration management (SCM)
- ④ Measurement
- ⑤ Formal technical reviews (FTR)
- ⑥ Work Product preparation & production.

Software process includes:

- ① Tasks - focus on a small, specific objective.
- ② Activities - group of related tasks & actions for a major objective
- ③ Action - set of tasks that produce a major work product.

Q3. Explain Process assessment & improvement. Explain its model.

Ans. Software processes are assessed to ensure their ability to control the cost, time & quality of software. Assessment is done to improve the software process followed by an organization.

Software process improvement (SPI) cycle includes:

- Process Measurement

- Process analysis
- Process change

Capability maturity model (CMM) & CMMI

capability maturity model integration

CMM was developed by SEI & evolved into CMMI later. It is an approach based on which an organization's process maturity is determined.

CMM's 5 Maturity levels | CMMI's levels

- | | |
|---|--|
| <ol style="list-style-type: none"> ① Initial level ② Repeatable level ③ Defined level ④ Managed level ⑤ Optimizing level | <ol style="list-style-type: none"> ① Initial ② Managed ③ Defined ④ Quantitatively managed ⑤ Optimized |
|---|--|

Common standard CMMI Appraisal Method for Process Improvement (CCAMMP)

It is a method used by software engineering institute for providing quality rankings with respect to capability maturity model integration (CMMI).

CMM Based Appraisal for Internal Process Improvement (CBAPP)

SEI (CMMI) based assessment Methodology provides diagnostics, enables & encourages an organization to understand its maturity : & gives the organization

(8)

an insight into its software development capability by assessing the strength & weakness of the current process.

Q4. What is Prescriptive Process Models?
Explain different types of Model in detail.

Ans . defines as the model prescribes a set of activities, actions, task, quality assurance & change the mechanism for every project.

There are 3 Types of Prescriptive process Models. They are:-

① The waterfall Model
linear & sequential model approach.

Requirement analysis

System Design

Implementation

Testing

Deployment

Maintenance

- Work Based on fixed Dates Requirements & Outcomes.
- Consistent communication is not Required.

(9)

- self contained
- Team members work independently
- Often Required to provide status reports.
- Chronological Order.

Advantage -

- Small Projects
- Basic Model
- Well known technology, fixed cost, Predefined Deadline
- Simple & Easy
- Documented
- Small Project
- Clear Goals & Diff

Disadvantage :-

- Customer dissatisfaction
- Risk factor
- No Feedback
- No Experiment
- No Parallelism
- High Risk
- 60% efficient
- Maintenance
- Not change friendly.

(2) Incremental Process Model

also known as 'Successive Version Model'.

In this, a series of releases, called 'increments', are built & delivered to the customer. First, a simple working

(10)

System (core product), that addresses Basic Requirements, is delivered.

- Customer feedback is recorded after each incremental delivery.

Advantages:

- Flexible to change requirements.
- changes can be done throughout the development stages.
- Easy to test bcoz of small iterations
- The initial cost is lower.

Disadvantages:

- Requires good planning & design.
- The total cost is higher.
- It affects system growth.

(3) RAD Model

- Rapid Application Development (RAD) is another form of Incremental Model.
- High speed adaptation of the linear Sequential Model in which fully functional system in a very short time (2-3M)
- Only applicable when Requirements are well understood.
- It's main features of RAD Model are that it focuses on the reuse of Templates, tools, Processes & code.

RAD Model has following phases -

(1) Business Modelling

- (2) Data Modeling
- (3) Process Modeling
- (4) Application Generation
- (5) Testing & Tuning.

Advantages:

- Flexible & adaptable to changes.
- Due to code generators & code reuse, there is a reduction of manual coding.
- Due to prototyping in nature, there is possibility of user rejects.

Disadvantages:

- can't be used for small projects.
- when technical risk is high, it is not suitable.
- not all application is compatible with R.A.D.

Q5. what do you mean by unified process:
explain its phases?

Ans. A Unified Process (UP) [20] is a software development process that uses the UML language to represent models of the software system to be developed. It is iterative, architecture centric, user driven & risk confronting.

One development cycle is divided into 4 consecutive phases.

- Inception phase

(12)

- elaboration phase
- construction phase
- transition phase
 - ... production phase
- each phase is concluded with a well-defined milestone;
- each phase has a specific purpose.

D Inception -

1. communication & planning are main
2. identifies scope of the project using use-case model.
3. customer requirements are identified & then it becomes easy to make a plan of the project.
4. Project plan, project goal, risks, use-case model, Project description, are made.
 - A Vision Document
 - An Initial use-case model
 - an initial project glossary
 - an initial business case
 - an initial risk assessment
 - a project plan, showing phases & iterations.
 - A Business Model.
 - Project milestones; the lifecycle objectives

② Elaboration -

- Planning & modeling are main.
- Executable arch. baseline.
- Detailed evaluation, development plan is carried out & minimizes the risks.

③ Construction -

- The software product integrated on the adequate platforms.
- The user manual.
- A description of the current release.

④ Transition

- "Beta Testing" to validate the new system against user expectations.
- Parallel Operation with a legacy system that it's replacing.
- Conversion of Operational Databases.
- Training of user & maintainers.
- Roll out the product to the Marketing, distribution & sales teams.

Q6. what do you mean by spiral model? explain its phases?

Ans. • Spiral Model is a system development lifecycle Model.

- It allows refinement throughout each stage of the Model.
- It is a evolutionary software process model which is a combination of an iterative nature of prototyping & systematic

(14)

aspects of traditional waterfall model.
Defined by Barry Boehm in his 1998 article.

concept development project

new Product development projects

product enhancement projects

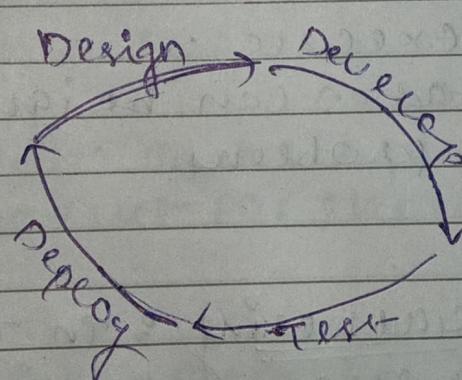
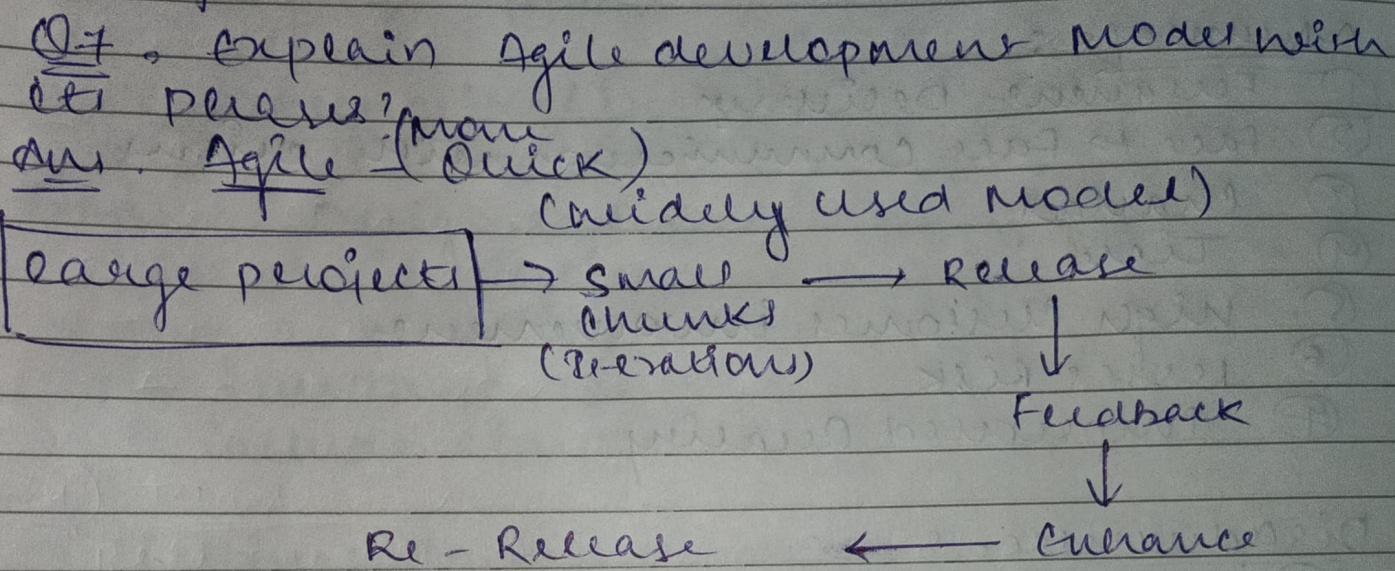
product maintenance projects

Each phase

Phases of Spiral Model

- ① Customer communication tasks required establishing effective communication between developer & customers.
- ② Planning - tasks required defining Resources, timelines, and other projects related info. estimation, scheduling, risk analysis.
- ③ Modeling analysis & design tasks required to asses both technical & management risks.
- ④ Engineering tasks required building one or more representations of the application.
- ⑤ construction & release tasks required to construct, test, install & provide user support.
- ⑥ customer evaluation - tasks required to obtain customer feedback based on an evaluation of the software representation

Created during one engineering stage &
Implemented during the installation
stage.



- ① Planning
- ② Req. Analysis
- ③ Design
- ④ Coding
- ⑤ Unit Test & Test
- ⑥ Acceptance Test

with Agile changes all well known customer
Feedback frequently with lower costs.

- Iterative approach to Project Management
helps teams to deliver value to their
customers faster & with fewer headaches.
- Agile Methodology is a way to manage
a Project by breaking it up into

(16)

Several phases.

① involves collaboration with stakeholders

② continues improvement at every stage.

Advantages:

- ① Frequent Delivery
- ② Face to face communication with client
- ③ Changes
- ④ Time
- ⑤ High customer involvement
- ⑥ Lesser risk
- ⑦ Improved Quality

Disadvantages:

- ① Dedicated Teams
- ② Hard to execute.
- ③ Documentation can be ignored.
- ④ Maintenance problems

Phase:-

- ① Requirement Gathering: In this phase, you must define the requirements, you should explain business opportunities & plan the time & effort needed to build the project. Based on this information, you can evaluate technical & economic feasibility.
- ② Design the Requirements: When you have identified the project, work with stakeholders to define Requirements, you can use

User flow diagram or the high level UML diagram to show the work of new Features & show how it will apply to your existing system.

③ Construction/ Iteration: When the team defines the Requirements, the work begins. Designers & Developers start working on their Project, which aims to deploy a working product. The product will undergo various stages of improvement, so it includes simple, minimal functionality.

④ Testing: In this phase, the Quality Assurance team examines the product's performance & looks for the bug.

⑤ Deployment: In this phase, the Team issues a product for the user's work environment.

⑥ Feedback: After releasing the product, the last step is feedback. In this, the team receives feedback about the product & works through the feedback.