Sanjay Ghodawat University, Kolhapur

2020-21

Established as State Private University under Govt. of Maharashtra. Act No XL, 2017

EXM/P/09/00

Year and Program: SY	**School of Technology**	**Department:** CSE
Course Code:CST211	**Course Title:** Software Engineering	**Semester : III**
Day : Monday **Date :** 8th Feb 2021	**End Semester Examination**	**Time : 2 Hrs.** **Max Marks : 50**

Question Paper

Instructions: 1) All Questions are compulsory.

		Marks	Bloom's Level	CO
Q.1	**Answer the following (any Two)-**			
a)	Describe Software myths that exists along with the Realities associated with those myths.	8	L2	CST211.1
b)	Explain in detail about software characteristics with examples	8	L2	CST211.1
c)	List out the various umbrella activities which support software development process and discuss about their necessity in maintaining the quality in both software process and product that is being developers for railway reservation	8	L1	CST211.2
Q.2	Answer the following (any Two)-			
a)	Explain in detail about verification and validation	8	L2	CST211.3
b)	Explain in detail about Agile Manifesto values and its principles?	8	L2	CST211.4
c)	Explain in detail about Scrum Methodology?	8	L2	CST211.4
Q.3	Answer the following (any Two)-			
a)	Describe the need and usage of epic, user stories in agile development.	9	L2	CST211.5
b)	List out the steps to create a Sprint release plan and Explain it.	9	L3	CST211.5
c)	Short notes on Black box testing and White box testing	9	L4	CST211.6

****

Year and Program: SY	**School of Technology**	**Department:** CSE
Course Code:CST211	**Course Title:** Software Engineering	**Semester : III**
Day : Monday **Date :** 8th Feb 2021	**End Semester Examination**	**Time : 2 Hrs.** **Max Marks : 50**

Model Answer

1.a The development of software requires dedication and understanding on the developers' part. Many software problems arise due to myths that are formed during the initial stages of software development. Unlike ancient folklore that often provides valuable lessons, software myths propagate false beliefs and confusion in the minds of management, users and developers.

Managers, who own software development responsibility, are often under strain and pressure to maintain a software budget, time constraints, improved quality, and many other considerations. Common management myths are listed in Table.

Table Management Myths(2)

The members of an organization can acquire all-the information, they require from a manual, which contains standards, procedures, and principles;	Standards are often incomplete, inadaptable, and outdated. Developers are often unaware of all the established standards. Developers rarely follow all the known standards because not all the standards tend to decrease the delivery time of software while maintaining its quality.
If the project is behind schedule, increasing the number of programmers can reduce the time gap.	Adding more manpower to the project, which is already behind schedule, further delays the project. New workers take longer to learn about the project as compared to those already working on the project.
If the project is outsourced to a third party, the management can relax and let the other firm develop software for them.	Outsourcing software to a third party does not help the organization, which is incompetent in managing and controlling the software project internally. The organization invariably suffers when it out sources the software project.

In most cases, users tend to believe myths about the software because software managers and developers do not try to correct the false beliefs. These myths lead to false expectations and ultimately develop dissatisfaction among the users. Common user myths are listed in Table.

Table User Myths(2 marks)

Brief requirement stated in the initial process is enough to start development; detailed requirements can be added at the later stages.	Starting development with incomplete and ambiguous requirements often lead to software failure. Instead, a complete and formal description of requirements is essential before starting development. Adding requirements at a later stage often requires repeating the entire development process.
Software is flexible; hence software requirement changes can be added during any phase of the development process.	Incorporating change requests earlier in the development process costs lesser than those that occurs at later stages. This is because incorporating changes later may require redesigning and extra resources.

In the early days of software development, programming was viewed as an art, but now software development has gradually become an engineering discipline. However, developers still believe in some myths-. Some of the common developer myths are listed in Table.

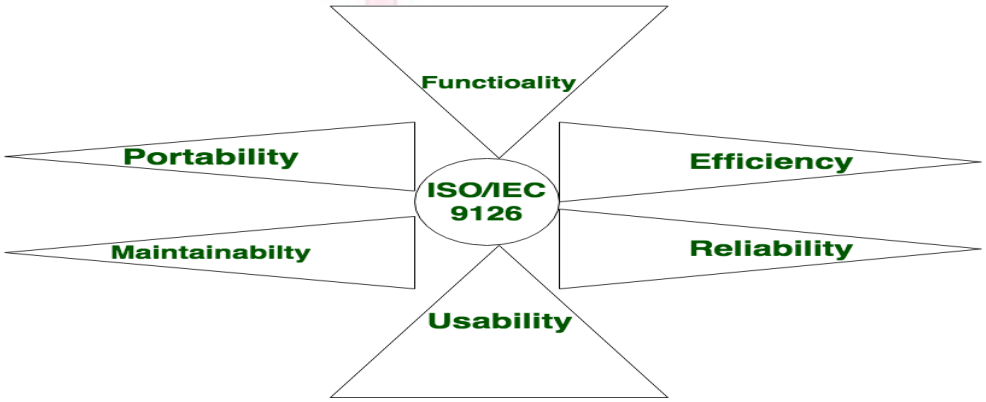
Table Developer Myths(2 marks)

Software development is considered complete when the code is delivered.	50% to 70% of all the efforts are expended after the software is delivered to the user.
The success of a software project depends on the quality of the product produced.	The quality of programs is not the only factor that makes the project successful instead the documentation and software configuration also playa crucial role.
Software engineering requires unnecessary documentation, which slows down the project.	Software engineering is about creating quality at every level of the software project. Proper documentation enhances quality which results in reducing the amount of rework.
The only product that is delivered after the completion of a project is the working program(s).	The deliverables of a successful project includes not only the working program but also the documentation to guide the users for using the software.
Software quality can be assessed only after the program is executed.	The quality of software can be measured during any phase of development process by applying some quality assurance mechanism. One such mechanism is formal technical review that can be effectively used during each phase of

	development to uncover certain errors.

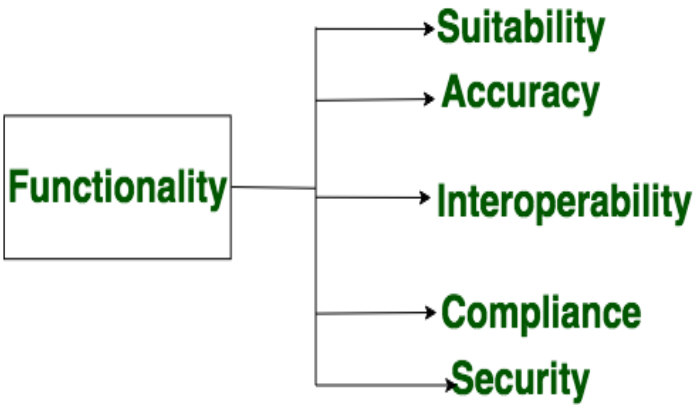
---

1. **b.** Software Characteristics are classified into six major components: (8 marks)

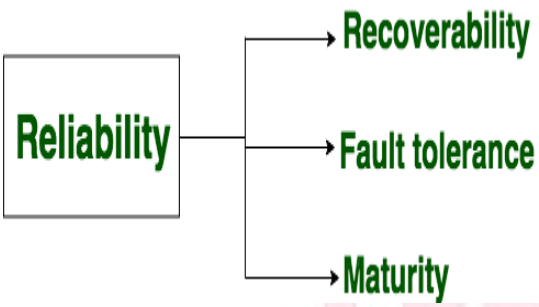


These components are described below:

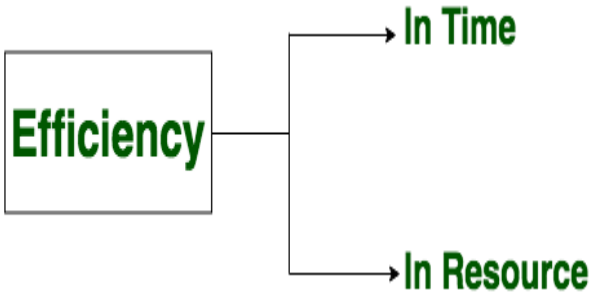
- **Functionality:**
  It refers to the degree of performance of the software against its intended purpose.
  Required functions are:



- **Reliability:**
  A set of attribute that bear on capability of software to maintain its level of performance under the given condition for a stated period of time.
  Required functions are:

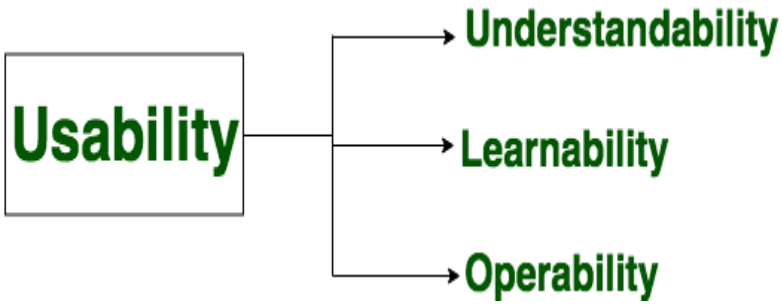


- **Efficiency:**
  It refers to the ability of the software to use system resources in the most effective and efficient manner.the software should make effective use of storage space and executive command as per desired timing requirement.
  Required functions are:

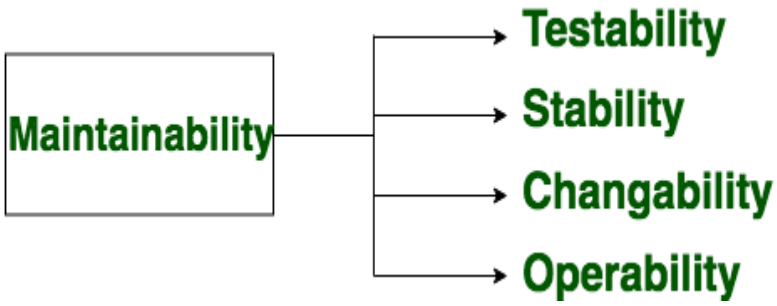


- **Usability:**
  It refers to the extent to which the software can be used with ease.the amount of effort or time required to learn how to use the software.

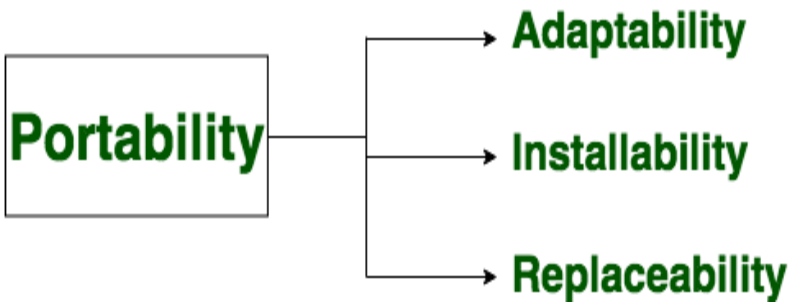
  Required functions are:



- **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.
  Required functions are:



- **Portability:**
  A set of attribute that bear on the ability of software to be transferred from one environment to another, without or minimum changes.
  Required functions are:



1.c. List the umbrella activities - 2 marks)
Quality in software process in railway system -3 marks)
Quality in software product in railway system -3 marks)

2.a. Verification – 4 Marks
Validation – 4 Marks
2.b. Agile 4 core values – 3 marks
Agile 12 principles – 5 marks
2.c. Scrum Definition – 2marks
Scrum diagram -3 marks
Explanation- 3 marks

3.a. Need of epic and user story in agile-4 Marks
Usage of user story, creating in format and explanation with example – 5 marks
b. Definition of sprint release plan – 2 marks
Steps to create release plan- 2 marks
Explanation – 5 marks
c. Comparison 9 points- 9 marks
