Name - Piyusha Rajendra Supe.
Roll no - 23C0315.
BE. Computer B.

Software Testing and Quality Assurance. (03/10/2025)

Assignment - STQA.

UNIT - 05 -

Q1. What is software ? "Software does not wear out" : State whether this statement is true or false Justify your answer.

→ • Software is a collection of programs, procedures and related documentation that performs specific tasks on a computer system.

• "Software does not wear out" is a true statement.

• Justified as follows -

• No physical degradation - Hardware components (like disks, processors and cables) physically deteriorate over time due to wear and tear. Software, being intangible, does not degrade physically with use - a program will execute the same way even after millions of runs.

- Failure Mode differences. -
Hardware failures are often due to physical aging. Software "failures" occur due to design defects, bugs, or changing requirements rather than physical deterioration.

- Software aging (conceptual) -
While software doesn't "wear out" like hardware, it can become obsolete or unstable due to:
  - Accumulated updates or patches introducing complexity.
  - Changing environments (new operating systems, hardware or user needs).
  - Security vulnerabilities over time.

This is called software aging - but its not wear and tear, its more about evolving complexity and incompatibility.

Q2. What are the different umbrella activities in software process framework? Explain them in brief.

→ - Software process framework defines a set of common activities that occur across all software projects, called umbrella activities.

- They support and overlay the main development process (like requirements, design, coding, testing)

- Major umbrella activities are -

(1) - Software project management - planning, scheduling, monitoring and controlling the project.

(2) Software quality assurance - defining and ensuring adherance to quality standards and procedures.

(3). Documentation - preparing and maintaining technical and user documentation throughout the project.

(4). Measurement and metrics - collecting data about processes, products and resources to improve decision making.

(5). Risk management - Identifying, analyzing and mitigating project risks.

(6) Reviews and audits - conducting formal and informal reviews to detect defects early.

(7) Reusability management - promoting and tracking reuse of software components.

Q3. What is software engineering? Software engineering is different than hardware engineering? State whether this statement is true or false.

→ - Software engineering is the application of systematic, disciplined and quantifiable approaches to the development, operation and maintenance of software.

- It focuses on methods, tools, processes and standards to produce high quality software within cost, time constraints.

- The statement "software engineering is different from hardware" is true.

- Software engineering deals with designing and manufacturing physical computer components.

- Software can be modified, updated and replicated easily; hardware changes require physical redesign or replacement.

- Software does not wear out physically hardware does.
- Development life cycle, cost structures and failure modes are fundamentally different between software and hardware.

UNIT - 06 -

Q4. Identify the importance of regression testing and explain it.

→ - Regression testing is the process of re-executing test cases after changes (bug fixes, enhancements, updates) to ensure existing functionality still works correctly.

- It is important because changes in one part of software can unintentionally affect other parts.

- Helps detect new defects introduced after modifications

- Ensures stability and reliability of the software after updates.
- Maintains user confidence and software quality during maintenance cycles.
- Saves time and cost in the long run by catching issues early.
- Can be automated for frequent and efficient execution in agile and continuous integration environments.

95. During which phases are system testing and acceptance testing conducted in software testing and what are reasons for each?

→ System testing –

- Conducted after integration testing is completed and before deployment.
- performed during the system / implementation phase of software development life cycle.
- Reason – To verify that the complete, integrated system meets the specified functional and non-functional requirements as a whole.

Acceptance testing –

- Conducted after system testing and before the software is released to production.

- Performed during the deployment/ installation or handover phase of the life cycle.
- Reason - To ensure the software meets customer needs or user needs and requirements and to obtain formal approval of client or stakeholders before going live.

Q6. What is impact of defect in different phases of software development?

→ - The later the defect is found in the SDLC, the higher its impact and cost to fix.

Here are what impacts it causes in each stage -

1) Requirement phase - Cause wrong or incomplete functionality to be designed and built. Very expensive to fix.
2) Design phase - Lead to incorrect architecture or module interaction.
3) Coding phase - Easier and cheaper to fix at this stage (unit testing or code review).
4) Testing phase defects - More costly because the software is already built and integrated.
5) Post release - production defects - Has the highest impact - customer dissatisfaction, service disruption, security issues, loss of reputation and high patch costs.