Fr. Conceicao Rodrigues College of Engineering,Mumbai

SOFTWARE ENGINEERING (CSC601)

Assignment -II

Date: 17-10-23

CO5: Identify risks, manage the change to assure quality in software projects.

Assignment 2

1. What is risk assessment in the context of software projects, and why is it essential?
2. Explain the concept of software configuration management and its role in ensuring project quality.
3. How do formal technical reviews (FTR) contribute to ensuring software quality and reliability?
4. Describe the process of conducting a formal walkthrough for a software project.
5. Why is it important to consider software reliability when analyzing potential risks in a project?

Rubrics :

Indicator	**Average**	**Good**	**Excellent**	**Marks**
Organization (2)	Readable with some mistakes and structured (1)	Readable with some mistakes and structured (1)	Very well written and structured (2)	
Level of content(4)	Minimal topics are covered with limited information (2)	Limited major topics with minor detailsare presented(3)	All major topics with minor details are covered (4)	
Depth and breadth of discussion(4)	Minimal points with missing information (1)	Relatively more points with information (2)	All points with in depth information(4)	
Total Marks(10)				

Name: Anuf Shaikh
Class: TE-COMPS 'B'
Roll no: 9639

SOFTWARE ENGINEERING

Assignment-II

1. What is risk assessment in the context of software projects, and why is it essential?

Risk assessment in software projects is the process of identifying, analyzing and priortizing potential risks or uncertainties that could affect the projects success. It involves evaluating the likelihood and impact of these risks and developing strategies to mitigate or manange reasons:

a) Project success: Idendentifying and addressing risks early can prevent project failiure or delays, ensuring that the project is completed on time and within budget.

b) Cost control effective risk assessment helps allocate resources effecieny reducing the likelihood of unexpected expenses of overuse.

c) Quality assurance: It ensures the quality of the software by identifying and mitigating risks related to defects, security vulnerabilities, or performance issues.

d) stakeholder confidence: Transparent risk assessment and

management build must among project stakeholders, such as clients, by demonstrating a proactive approach to project challenges.

e) Schedule adherence: Mitigating risks can help in adhering to project timelines, which is essential for meeting deadlines and market demands.

f) Resource allocation

g) Legal and regulartory compliance.

2. Explain the concept of software configuration management and its role in ensuring project quality.

Software configuration management (SCM) is a net of process and practices that help control and manage changes to software throughout its development lifecycle. Its printig role is to ensure project quality by maintaining the integrity, consistency and traceability of software components and their related documentation.

Hores, how SCM accomplishes this:

a) Version Control: SCM tools such as Git, enables developers to track changes to the source code.

) Configuration identification: SCM defines and manages software components, including source code, documentation, and other artifacts.

) Change control: SCM establishes a formal process for requesting reviewing and opposing changes to the software.

) Configuration auditing: Regular audits ensure that the software configuration aligns with the defined standards and requirements.

e) Baseline management: SCM creates baselines, which are stable and well-tested configurations of the software.

3. How do formal technical reviews (FTR) contribute to ensuring software quality and reliability?

Formal Technical Reviews (FTRs) play a crucial role in ensuring software quality and reliability by facilitating thorough examination and verification of software artifacts. Here's how FTR's contribute:

1) Defect identification: FTR's involve a systematic examination of software documentation, code and design.
2) Knowledge sharing: FTRs encourage knowledge sharing among team members.
3) Consistency and standards: FTRs ensure that the software conforms to established coding and design standards.
4) Improved documentation: Through FTRs, documentation quality is enhanced for understanding the software overall.
5) Risk mitigation: By identifying and addressing issue early, FTRs reduce the risk of major problems emerging later in the software development process, which can be costly and time-consuming to fix.

4. Describe the process of conducting a formal walkthrough for a software project.

The process for conducting a formal software project walkthrough:

i) Preparation
- Schedule and define the purpose
- Select artifacts and distribute them.

ii) Participant Preparation
- Participants review artifacts and make notes.

iii) Conduct the walkthrough:
- Gather the team and appoint a mentor.
- Present, discuss, and document issues

iv) Issue Tracking and Resolution:
- Assign issue for resolution.
- Follow up on issue status.

v) Documentation:
- Create a summary report and share it.

vi) Closure:
- Close the process when issues are resolved. This streamlined process ensured a collaborative review, issue resolution, and improved software quality.

5. Why is it important to consider software reliability when analyzing potential risks in a project?

i) User satisfaction: Unreliable software can lead to a poor user experience, causing furtration and dissatisfaction.

ii) Cost implication: Unreliable software can lead to higher support and maintainance costs.

iii) Project Delays: Development teams divert their attension from planned tasks to address defects and stability problems.

iv) Reputation damage: Reliability problems can damage an organization's reputation.

v) Legal and compliance risks: Depending on the domain, vulnerable software can result in legal and compliance risks.

vii) Security vulnerabilities: Unreliable software can be suspeptible to security vulnerabilities, putting sensitive data at risk.

viii) Operational disruption: Software thats' not reliable can disrupt business operations, causing dountime and affecting productivity.

~~viii~~ix) Maintainance Burden: ~~Unr~~ Unreliable software often requires constant maintence, diverting resources from new development initiatives and innovation.

x) Stakeholder confidence: Stakeholders, including investors and projects sponsors, may lose confidence in the projects success if reliability risks are not adequately addressed.