Chhand Chavghule
9594, TE COMPS - B

SE - Assignment - 2

1.) Risk assessment in the context of software projects is the process of identifying, analyzing and prioritizing potential risks and uncertainties that could affect the successful completion of a software development project. These risks can ~~change~~ range from technical issues and ~~so~~ resources constraints to changes in project requirements, market conditions and external factors. The primary goal of risk assessment is to proactively manage and mitigate these risks to ensure the projects objectives are met. Following are key reasons as to why risks assessment is essential in software projects.

1.) Early problem identification :- spot problems before they escalate.

2.) Efficient resource allocation :- allocate resources effectively.

3.) Cost control :- identifying and managing risks can help control project costs.

4.) schedule management :- maintaining project timelines

5.) Quality assurance :- address quality risks to ensure the final product meets expectations.

6.) Reputation management :- protect organization's image and avoid legal issues by managing risks

7.) stakeholder communication :- keep clients, ~~and~~ management and team informed about potential challenges to set realistic expressions.

8.) Increasing project success rate - projects that manage risks effectively have a better chance of success.

Q.2) Software configuration management (SCM) is a set of practices and processes used to systematically control, organize, and track changes in software projects. Its primary role is to ensure the integrity, stability of a software system throughout its development lifecycle. Here's how SCM contributes to project quality.

1.) version control : SCM tracks and manages different versions of software ensuring the right version is used, reducing errors.

2.) change management : organizes changes, ensuring Through testing and documentation to prevent defects.

3.) Traceability : SCM links changes to specific requirements, enhancing understanding and meeting project requirements.

4.) configuration management :- It controls all software components, preventing configuration & release errors in each release.

5.) parallel development :- SCM allows multiple developers to work concurrently without conflicts, maintaining code quality.

6.) Automated build and development : Integration with SCM ensures consistent, error-free software building and development.

7.) Backup and Recovery :- SCM provides backup and recovery mechanisms to protect against data loss.

8.) Auditing and Compliance : Tracks changes for auditing and regulatory compliance, crucial in regulated ~~fdus~~ industries to ensure quality and compliance standards.

Q.3) formal technical reviews (FTR) are systematic, well structured processes for reviewing and evaluating various aspects of software development, such as requirements, design, code and documentation. FTRs play a crucial role in ensuring software quality and reliability Through the following mechanisms :

i.) Error detection and prevention
ii.) Knowledge sharing
iii.) Compliance
iv.) Requirement validation
v.) Risk Mitigation
vi.) Consistency
vii.) Quality improvement
viii.) Enhanced process.

Q-4.) A formal walkThrough in the context of a software project is a structured and systematic process for reviewing and evaluating software artifacts such as code, design documents or requirements. The primary goal is to identify issues, ensure quality and improve the overall project. The following is the step-by-step process for conducting a formal walkThrough.

i.) Preparation :- Preparing the artifact and assembling a review team.
ii.) Scheduling :- Scheduling a meeting and setting an agenda.
iii.) Conducting the walkthrough :- Conducting a structured review where team members discuss and document issues.
iv.) Resolution :- Resolving issues and assigning responsibilities.
v.) Documentation :- Documenting the review.
vi.) Follow up :- After the review, follow up on the assigned actions.
vii.) closure :- Closing the review process once all issues are addressed.
viii.) Feedback and Continuous improvement :- Gathering feedback to improve future reviews.

Q.5.) Considering software reliability is crucial when analyzing potential risks in a project for several reasons.

a.) User Expectations : Users expect software to be reliable ensure software meets user expectations.

b.) business Impact : Software failures have significant financial implications.

c.) Reputation : Safeguard the organization's image

d.) maintainence costs : Reducing long term support expenses.

e.) Safety critical applications : Avoid catastrophic consequences.

f.) Regulatory compliance : Ensure adherence to industry regulations.

g.) Data integrity : Protect data from corruption or loss.

h.) market competition : staly competitive with reliable software.

i.) Caustomer satisfaction: Enhance user experience and loyalty.

j.) Project success : critical for successful project outcomes.