

Quality & Quality Assurance

Introduction

Quality is an important factor when it comes to any product or service. With the high market competition, quality has become the market differentiator for almost all products and services.

Therefore, all manufacturers and service providers out there constantly look for enhancing their product or the service quality.

In order to maintain or enhance the quality of the offerings, manufacturers use two techniques, **quality control and quality assurance**. These two practices make sure that the end product or the service meets the quality requirements and standards defined for the product or the service.

There are many methods followed by organizations to achieve and maintain required level of quality. Some organizations believe in the concepts of **Total Quality Management (TQM)** and some others believe in **internal and external standards**.

The standards usually define the processes and procedures for organizational activities and assist to maintain the quality in every aspect of organizational functioning.

When it comes to standards for quality, there are many. ISO (International Standards Organization) is one of the prominent bodies for defining quality standards for different industries.

Therefore, many organizations try to adhere to the quality requirements of ISO. In addition to that, there are many other standards that are specific to various industries.

As an example, SEI-CMMI ([Software Engineering Institute-Capability Maturity Model Integration](#)) is one such standard followed in the field of software development.

Since standards have become a symbol for products and service quality, the customers are now keen on buying their product or the service from a certified manufacturer or a service provider.

Therefore, complying with standards such as ISO has become a necessity when it comes to attracting the good number of customers.

Quality Control

Many people get confused between quality control (QC) and quality assurance (QA). Let's take a look at quality control function in high-level.

As we have already discussed, organizations can define their own internal quality standards, processes and procedures; the organization will develop these over time and then relevant stakeholders will be required to adhere by them.

[*The process of making sure that the stakeholders are adhered to the defined standards and procedures is called quality control. In quality control, a verification process takes place.*](#)

Certain activities and products are verified against a defined set of rules or standards.

Every organization that practices QC needs to have a Quality Manual. The quality manual outlines the quality focus and the objectives in the organization.

[*The quality manual gives the quality guidance to different departments and functions. Therefore, everyone in the organization needs to be aware of his or her responsibilities mentioned in the quality manual like SOP- Standard Operation Plan.*](#)

Quality Assurance

Quality Assurance is a broad practice used for assuring the quality of products or services. There are many differences between quality control and quality assurance.

In quality assurance, a constant effort is made to enhance the quality practices in the organization.

Therefore, continuous improvements are expected in quality functions in the company. For this, there is a dedicated quality assurance team commissioned like IQAC (Internal Quality Assurance Cell of MMDU)

Sometimes, in larger organizations, a ‘Process’ team is also allocated for enhancing the processes and procedures in addition to the quality assurance team (*like some of the faculty members are also deputed as Academic –Auditors w.r.t. IQAC-MMDU*).

Quality assurance team of the organization has many responsibilities. *First and foremost responsibility is to define a process for achieving and improving quality.*

Some organizations come up with their own process and others adopt a standard processes such as ISO or CMMi (**Capability Maturity Model Integration**). Processes such as CMMi allow the organizations to define their own internal processes and adhere by them.

Quality assurance function of an organization uses a number of tools for enhancing the quality practices. These tools vary from simple techniques to sophisticated software systems.

The quality assurance professionals also should go through formal industrial trainings and get them certified. This is especially applicable for quality assurance functions in software development houses.

Since quality is a relative term, there is plenty of opportunity to enhance the quality of products and services.

The quality assurance teams of organizations constantly work to enhance the existing quality of products and services by optimizing the existing production processes and introducing new processes.

Software Engineering-Formal Technical Reviews FTR

Software review is an important part of **Software Development Life Cycle (SDLC)** that assists software engineers in validating the quality, functionality, and other vital features and components of the software. A *formal technical review is a software quality assurance activity* performed by software engineers (and others).

The objectives of the FTR are:-

- (1) To **uncover errors** in function, logic, or implementation for any representation of the software;
- (2) To verify that the software under review meets its **requirements**;
- (3) To ensure that the software has been represented according to **predefined standards**;
- (4) To achieve software that is developed in a **uniform manner**; and
- (5) To make projects more **manageable**.

In addition, the FTR serves as a training ground, enabling junior engineers to observe different approaches to software analysis, design, and implementation. The FTR also serves to promote backup and continuity because a number of people become familiar with parts of the software that they may not have otherwise seen.

The FTR is actually a class of reviews that includes walkthroughs, inspections, round-robin reviews and other small group technical assessments of software. Each FTR is conducted as a meeting and will be successful only if it is **properly planned, controlled, and attended**. In the sections that follow, guidelines similar to those for a walk through are presented as a representative formal technical review.

The Review Meeting

Regardless of the FTR format that is chosen, every review meeting should abide by the following constraints:

- Between three and five people (typically) should be involved in the review.
- Advance preparation should occur but should require no more than two hours of work for each person.
- The duration of the review meeting should be less than two hours.

Given these constraints, *it should be obvious that an FTR focuses on a specific (and small) part of the overall software. For example, rather than attempting to review an entire design, walkthroughs are conducted for each component or small group of components.* By narrowing focus, the FTR has a higher likelihood of uncovering errors.

The focus of the FTR is on a work product (e.g., a portion of a requirements specification, a detailed component design, a source code listing for a component). *The individual who has developed the work product—the producer—informs the project leader that the work product is complete and that a review is required. The project leader contacts a review leader, who evaluates the product for readiness, generates copies of product materials, and distributes them to two or three reviewers for advance preparation (like a research paper is to be submitted to 3-4 reviewers before its final publication).* Each reviewer is expected to spend between one and two hours reviewing the product, making notes, and otherwise becoming familiar with the work. Concurrently, the review leader also reviews the product and establishes an agenda for the review meeting.

The review meeting is attended by the review leader, all reviewers, and the producer. One of the reviewers takes on the role of the recorder; that is, the individual who records (in writing) all important issues raised during the review. The FTR begins with an introduction of the agenda and a brief introduction by the producer. The producer then proceeds to walk through the work product, explaining the material, while reviewers raise issues based on their advance preparation. When valid problems or errors are discovered, the recorder notes each.

At the end of the review, all attendees of the FTR must decide whether to:-

- (1) **Accept** the product without further modification,
- (2) **Reject** the product due to severe errors (once corrected, another review must be performed), or
- (3) **Accept the product provisionally** (minor errors have been encountered and must be corrected, but no additional review will be required).

The decision made, all FTR attendees complete a sign-off, indicating their participation in the review and their concurrence with the review team's findings.

Review Reporting and Record Keeping

During the FTR, a reviewer (the recorder) actively records all issues that have been raised. These are summarized at the end of the review meeting and a review issues list is produced. In addition, a formal technical review summary report is completed.

A review summary report answers three questions:

- 1. What was reviewed?**
- 2. Who reviewed it?**
- 3. What were the findings and conclusions?**

The review summary report is a single page form (with possible attachments). It becomes part of the project historical record and may be distributed to the project leader and other interested parties.

The review issues list serves two purposes:

- (1) To identify problem areas within the product and**
- (2) To serve as an action item checklist that guides the producer as corrections are made. An issues list is normally attached to the summary report.**

It is important to establish a follow-up procedure to ensure that items on the issues list have been properly corrected.

Review Guidelines

Guidelines for the conduct of formal technical reviews must be established in advance, distributed to all reviewers, agreed upon, and then followed. A review that is uncontrolled can often be worse than no review at all. The following represents a minimum set of guidelines for formal technical reviews:

1. Review the product, not the producer. An FTR involves people and egos. Conducted properly, the FTR should leave all participants with a warm feeling of accomplishment. Conducted improperly, the FTR can take on the aura of an inquisition. Errors should be pointed out gently; the tone of the meeting should be loose and constructive; the intent should not be to embarrass or belittle. The review leader should conduct the review meeting to ensure that the proper tone and attitude are maintained and should immediately halt a review that has gotten out of control.

2. Set an agenda and maintain it. One of the key maladies of meetings of all types is drift. An FTR must be kept on track and on schedule. The review leader is chartered with the responsibility for maintaining the meeting schedule and should not be afraid to nudge people when drift sets in.

3. Limit debate and rebuttal. When an issue is raised by a reviewer, there may not be universal agreement on its impact. Rather than spending time debating the question, the issue should be recorded for further discussion off-line (if there is any conflict with some point, there should not be any end endless discussion on it, let it be discussed it separately)

- 4. Enunciate problem areas, but don't attempt to solve every problem noted. A review is not a problem-solving session.** The solution of a problem can often be accomplished by the producer alone or with the help of only one other individual. Problem solving should be postponed until after the review meeting.
- 5. Take written notes.** It is sometimes a good idea for the recorder to make notes on a wall board, so that wording and priorities can be assessed by other reviewers as information is recorded.
- 6. Limit the number of participants and insist upon advance preparation.** Two heads are better than one, but 14 are not necessarily better than 4. Keep the number of people involved to the necessary minimum. However, all review team members must prepare in advance. Written comments should be solicited by the review leader (providing an indication that the reviewer has reviewed the material).
- 7. Develop a checklist for each product that is likely to be reviewed.** A checklist helps the review leader to structure the FTR meeting and helps each reviewer to focus on important issues. Checklists should be developed for analysis, design, code, and even test documents.
- 8. Allocate resources and schedule time for FTRs.** For reviews to be effective, they should be scheduled as a task during the software engineering process. In addition, time should be scheduled for the inevitable modifications that will occur as the result of an FTR.
- 9. Conduct meaningful training for all reviewers.** To be effective all review participants should receive some formal training. The training should stress both process-related issues and the human psychological side of reviews.
- 10. Review your early reviews.** Debriefing can be beneficial in uncovering problems with the review process itself. The very first product to be reviewed should be the review guidelines themselves.

Types of Software Reviews:

There are mainly three types of software reviews, all of which are conducted by different members of the team who evaluate various aspects of the software. Hence, the types of software review are:

1. Software Peer Review:

Peer review is the process of evaluating the technical content and quality of the product and it is usually conducted by the author of the work product, along with some other developers. According to **Capacity Maturity Model** the main purpose of peer review is to provide “a disciplined engineering practice for **detecting or correcting defects** in the software artifacts, preventing their leakage into the field operations”. In short, peer review is performed in order to determine or resolve the defects in the software, whose quality is also checked by other members of the team.

Types of Peer Review:

- **Code Review:** To fix mistakes and to remove vulnerabilities from the software product, systematic examination of the computer source code is conducted, which further improves the quality & security of the product.
- **Pair Programming:** This is a type of code review, where two programmers work on a single workstation and develop a code together.
- **Informal:** As suggested by its name, this is an informal type of review, which is extremely popular and is widely used by people all over the world. Informal review does not require any documentation, entry criteria, or a large group of people. It is a time saving process that is not documented.
- **Walkthrough:** Here, a designer or developer lead a team of software developers to go through a software product, where they ask question and make necessary comments about various defects & errors. This process differs from software inspection and technical review in various aspects.
- **Technical Review:** During the process of technical review a team of qualified personnel review the software and examine its suitability to define its intended use as well as to identify various discrepancies.
- **Inspection:** This is a formal type of peer review, wherein experienced & qualified individuals examine the software product for bugs and defects using a defined process. Inspection helps the author improve the quality of the software.

2. Software Management Review:

These reviews take place in the later stages by the management representatives. *The objective of this type of review is to evaluate the work status.* Also, on the basis of such reviews decisions regarding downstream activities are taken.

3. Software Audit Reviews:

Software Audit Review or software review is a type of external review, wherein one or more auditors, who are not a part of the development team conduct an independent examination of the software product and its processes to assess their compliance with stated specifications, standards, and other important criterion's. This is done by managerial level people.

Formal Review Vs Informal Review:

Formal and informal review are two very important types of reviews that are used most commonly by software engineers to identify defects as well as to discuss ways to tackle these issues or discrepancies.

Therefore, to understand these important types of software review, following is a comparison of the two:

Formal Review:

A type of peer review, **formal review** follows a formal process and has a specific formal agenda. It has a well-structured and regulated process, which is **usually implemented at the end of each life cycle**. During this process, a formal review panel or board considers the necessary steps for the next life cycle.

Features of Formal Review:

- This evaluates conformance to specification and various standards.
- Conducted by a group of 3 or more individuals.
- The review team petitions the management of technical leadership to act on the suggested recommendations.
- Here, the leader verifies that the action documents are verified and incorporated into external processes.

Informal Review:

Unlike Formal Reviews, Informal reviews are applied **multiple times** during the early stages of software development process. The major difference between the **formal and informal reviews** is that the former follows a formal agenda, whereas the latter is conducted as per the need of the team and follows an informal agenda. Though *time saving, this process is not documented and does not require any entry criteria or large group of members.*

Features of Informal Review:

- Conducted by a group of 2-7 members, which includes the designer and any other interested party.
- Here the team identifies errors & issues as well as examine alternatives.
- It is a forum for learning.
- All the changes are made by the software designer.
- These changes are verified by other project controls.
- The role of informal review is to keep the author informed and to improve the quality of the product.

Process of Software Review:

The process of software review is a simple one and is common for all its types. It is usually implemented by following a set of activities, which are laid down by IEEE Standard 1028. All these steps are extremely important and need to be followed rigorously, as skipping even a single step can lead to a complication with the development process, which can further affect the quality of the end product.

1. Entry Evaluation:

A standard check-list is used by entry criteria in order to ensure an ideal condition for a successful review.

2. Management Preparation:

During this stage of the process, a responsible management ensures that the software review has all the required resources, which includes things like staff, time, materials, and tools.

3. Review Planning:

To undergo a software review, an objective is identified. Based on the objective, a recognized team of resources is formed.

4. Preparation:

The reviewers are held responsible for preparing group examination to do the reviewing task.

5. Examination and Exit Evaluation:

In the end, the result made by each reviewer is combined all together. Before the review is finalized, verification of all activities is done that are considered necessary for an efficacious software review.

SPM Next Online Lecture on 10th April, 2020 at 12:00 Noon

Topics: - Project Reporting, Post-Implementation Review

BCA-604 SPM

Lecture by Prof. (Dr.) Munishwar Rai

Date: - 10-04-2020 12:00 Noon Onwards

Just wait for a moment.....



Project Reporting:

How to Report Successfully?

Reporting on the status of a project is *a way to monitor and manage the project*, as well as getting the information out from report as per the need of the stakeholders. You'll be creating lots of different types of progress reports throughout the life cycle of the project. But the project status report can do a lot of things other reports can't. It's a vital communication tool among different stakeholders of the project, and it can provide *a documented history of the project*, which makes planning for your next project easier.

Some other objectives of status reports include:

- Improve communications across organization among the different stakeholders.
- Simplify the communication process
- Keep stakeholders informed about status and quality of the project
- Deliver important messages based on the report analysis to the intended target user for the refinement.
- Improve organizational support for your project/your team, to remove the errors/faults highlighted in the report.

The best practices of progress reporting on a project

- **Communicate:** The report should not communicate everything by itself, but use the project's progress status report to deliver the right data to the right party at the right time.
- **Consistency:** It means use the same format, distribution cycle and method in all the reports. Don't mix things up, that only disrupts the effectiveness of the communication aspect of the report.
- **Establish Metrics:** When planning for the project, figure out how you're going to measure its progress (metrics), and then stick to this metrics as you report on the project throughout its life cycle.
- **Simplify:** You want the report to be effective, so avoid it with unnecessary details. Stay to the point and just report on what needs reporting on.
- **Verify:** Your audience doesn't want opinions or unsubstantiated facts. Make sure that you're giving only as per the requirements of the stakeholders.
- **Standards:** Like consistency, keeping standards of process and a template for reporting makes sure your report is clear like we have to follow the same format while filling a feedback form.
- **Use Tools:** There are project management tools that incorporate these best practices, streamlining the process.

Here are some of the things to need to include when you generate Project Management Report.

If you're reporting to stakeholders, they are not interested in details. Keep your presentation light and to the point. Their time is precious/limited, respect it. If they want to know more about a specific aspect of the project, only then you can go in for the deeper dive.

1. General Project Info

To start with, you're going to need to just put down the basics. *What is the project name? Who is the project manager? What are the number of resources?* All this information is essential, if obvious, to track the paperwork. Don't assume your stakeholder is familiar with all this information. It's also especially useful for when you're doing historic research for future projects.

2. General Status Info

The General Status information can be in the form of a stamp to the report with the data that will distinguish it from the others reports that will be associated with the project paperwork. So, here you want to include on *what date the report was generated, who is the author* and so on.

3. Milestone Review

The milestones are major phases of your project. They're a good way to break up the larger project into more manageable parts. *They're not as small as tasks, but made up of those tasks which together make up a phase of the project,* you'll note where you are in terms of meeting those milestones against where you planned to be at this point in the project's life cycle. Hence we can check the progress of the project w.r.t. milestones.

4. Project Summary

Include a short summary of the *forecasted completion date and costs of the project.* Be sure to include the tasks that are facing issues, *how those problems might impact the deadline and costs, what you're planning to do to resolve these issues* and what the results will be once you have fixed the problem.

5. Issues and Risks

List the issues that have arisen over the development of the project to date. *Note what they are, how you're resolving them and what impact they'll have on the overall project.* What are you doing to get the project back on track?

6. Project Metrics

You should have established the metrics for status reporting during the project's planning phase. It's impossible to know that your project is succeeding without measuring its effectiveness (*as we are conducting different tests/exams for the students for their success in their life*) .These metrics are a way to show you're on track and evaluate what, if anything, needs attention.

Reporting tools help you monitor and track your projects online, so all your project data is up-to-the-minute, not days or weeks old. Most PM tools let you generate some kind of project reports.

Post-Implementation Review in Project Management

A post-implementation review is a process to evaluate whether the objectives of the project were met. You can also use it to see how effective the project was managed. *This helps to avoid making similar mistakes with future projects and learning how to run the project better.*

Post-implementation review is the last step in your project cycle and usually involves an independent party, which can act more objectively in making their determinations about how the project was run. This provides the stakeholders of the project the confidence to know that the objectives of the project were met successfully.

What Is Post-Implementation Review?

The project might be over, but the process continues. That means that if you delivered a product or a service, the project might be completed, but you still need to check on the viability of the product or service. You might have achieved all the goals you set out for the project, but *what about the business needs that product or service was responding to?*

Think of it as an ongoing step in your project closure process. It's a post-project review or post-implementation review, which is part of your project management responsibilities. *It's also a great way to identify project successes, deliverables, and achievements and learn lessons from those parts of the project that didn't work out as planned activity.*

What Is the Post-Implementation Review Process?

To get the most out of your project, you want to employ a post-implementation review process. While this can start at any time after the initial project has been complete, starting it sooner than later makes sure that the project details are still fresh in the team's mind.

How to Conduct a Post-Implementation Review?

After the project's deliverables have gone through at least one successful business cycle, you can get started on the review. Here are some of the best practices for conducting the review include the following.

- 1. Trust:-** To get the information you need, you want *honesty* from your participants. Therefore, tell them you want openness, without fear. *The more critical and truthful their observations about the project are, the more successful the review.*
- 2. Objectivity:-** You don't want interpersonal issues ,observations with bad feelings or to settle old scores. Seek objectivity, or as close to an impartial critique as can be expected, i.e. biasfree review.
- 3. Documentation:-** Like all project management, you want to create proper documentation that illustrates how you went from Point A to Point B. By documenting the practices and procedures that created the successes in the project, you'll be able to *follow them again in future projects as a reference document.*
- 4. Hindsight:-** As you develop a narrative as to what worked and what didn't, what surprises arose during the project and how you deal with them, understand that this *hindsight vision can also help as you look forward towards new projects.*
- 5. Improvement:-** *The point of this review process is not to blame individuals or teams for mistakes, but to learn from experience and then apply that knowledge to future projects.* Stay focused on what's next, rather than looking back as a means of applying guilt.

Post-Implementation Review Methods

There are many ways to gather the information you want to determine what worked and what didn't in your project. Here are some examples.

- **Gap Analysis:-** This method of assessing *how a plan differed from the actual application* is always a powerful tool to see what benchmarks you met, and which you didn't. You can start with your project charter and see how closely you adhered to your objectives. Look at your deliverables. Are they at a quality level you expected? When there are gaps discovered, figure out how they can be closed.

- **Project Goals:-** Its mean, did you achieve the goals of your project? Are your deliverables functioning as planned? What was the error rate of the project? Can the deliverables adjust to changes in the market? How well-trained and supported are end-users? What controls and systems are in place and are they working? Are problems being addressed? Did you planned goal align with your result?
- **Stakeholders:-** How satisfied are your stakeholders? Were users' needs met? What effect did the project have on them? If there is dissatisfaction, why is that and what can you do to resolve it?
- **Cost:-** How much did the project end up costing? What are the costs involved in operating the project's result? Are the costs aligned to the benefits of the project? If this isn't the case, how can you improve the cost next time?
- **Benefits:-** Did the project achieve the benefits projected, and if not why and how can that be improved? What opportunities are there to further the results? Are there other changes you could apply to help maximize the project's results?
- **Lessons:-** Did the project's deliverable, schedule and budget all meet expectations, and if not why? What were some of the issues that arose during the running of the project and how could they be avoided for the next project? What went well, and what can you learn from that experience?
- **Report:-** Document what you learned from the review, whether there is actions needed to get the beneficial results you want and list the lessons you've learned, noting how the project can impact future projects, so you can build on success and avoid problems.

Conclusion:-

There are many ways to close a project, but too often the post-implementation review is neglected. When you're dealing with a lot of people and asking them for criticism of the project, there's the potential to step on someone's toes and create hurt feelings that can creates some unpleasant political issues within your team or organization. Therefore, be clear that what you're interested in is not a personal attack, but a systemic overview of process and how everyone together can work towards improving it. That's why it often helps to hire an independent party to collect the post-project data.

Don't forget to review all the project documentation. It'll help you better assess what worked and what didn't, and provide you with an overview of the project and where there might have been unforeseen holes that you can then fill in with upcoming projects.

When you're done with the review be completely transparent. Share your findings in a report and make sure everyone has access to these documents. Your goal is to create better projects, and that information isn't proprietary. Everyone has a need to know it.

SPM Next Online Lecture on 14th April, 2020 at 11:00 Noon

Topics: - Project organization and responsibilities