Manuscript Title: Font Size 24 Pt

**Author Details1, Author Details2 (Maximum Authors 6)**

1Affiliation of First Author.

2Affiliation of Second Author.

***Abstract: -*** This electronic document is a “live” template. The various components of your paper [title, text, heads, etc.] are already defined on the style sheet, as illustrated by the portions given in this document. DO NOT USE SPECIAL CHARACTERS, SYMBOLS, OR MATH IN YOUR TITLE OR ABSTRACT. This electronic document is a “live” template. The various components of your paper [title, text, heads, etc.] are already defined on the style sheet, as illustrated by the portions given in this document. DO NOT USE SPECIAL CHARACTERS, SYMBOLS, OR MATH IN YOUR TITLE OR ABSTRACT. This electronic document is a “live” template. The various components of your paper [title, text, heads, etc.] are already defined on the style sheet, as illustrated by the portions given in this document. DO NOT USE SPECIAL CHARACTERS, SYMBOLS, OR MATH IN YOUR TITLE OR ABSTRACT.

***Keywords:*** *Risk Management, process management etc.*

1. **Introduction**

All manuscripts must be in English. These guidelines include complete descriptions of the fonts, spacing, and related information for producing your proceedings manuscripts. This template provides authors with most of the formatting specifications needed for preparing electronic versions of their papers. All standard paper components have been specified for three reasons: (1) ease of use when formatting individual papers, (2) automatic compliance to electronic requirements that facilitate the concurrent or later production of electronic products, and (3) conformity of style throughout a conference proceedings. Margins, column widths, line spacing, and type styles are built-in; examples of the type styles are provided throughout this document and are identified in italic type, within parentheses, following the example. PLEASE DO NOT RE-ADJUST THESE MARGINS. Some components, such as multi-leveled equations, graphics, and tables are not prescribed, although the various table text styles are provided. The formatter will need to create these components, incorporating the applicable criteria that follow.

1. The software detail ought to mirror the qualities of the item that the client needs. Nonetheless, the advancement association may likewise have necessities, for example, maintainability that are excluded in the specification.
2. Certain software quality traits, for example, maintainability, ease of use, dependability can't be actually indicated and estimated.
3. At the beginning periods of software process it is hard to characterize a total software specification. Hence, in spite of the fact that product may adjust to its specification, clients don't live up to their quality desires.

1. **PR FRAMEWORK FOR SPM**

The objective of 3PR system research was to build up a straightforward task project management framework customized for SPM. This exploration was directed as a major aspect of a greater research project. The objective of the greater research project was to build up a product management usefulness metric. Such a metric improvement concentrate required a legitimate SPM system.

A portion of the past models and structures are process oriented, and people management is to some degree disregarded. Some give an all-encompassing perspective without sufficient subtleties. Some are ostensibly finished. In particular, the majority of these investigations need approval. Subsequently, we built up the 3PR system and approved it. This structure was effectively utilized as an establishment for the improvement of a SPMefficiency metric.

The 3PR system comprises of four principle regions in SPM:

* + People management
  + Process management
  + Product management
  + Risk management

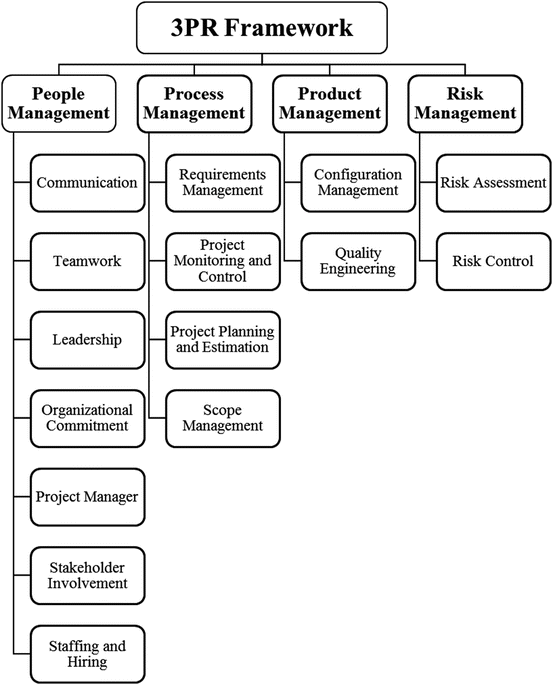


Fig: 3PR Framework for SPM

1. **SOFTWARE QUALITY CHALLENGE**

In the software industry, the developers will never state that the product is free of imperfections, dissimilar to other modern item makers generally do. This distinction is because of the accompanying reasons.

### Product Complexity

It is the quantity of operational modes the item allows. Regularly, a industrial item permits just not exactly a couple of thousand methods of activity with various blends of its machine settings. Be that as it may, software bundles permit a large number of operational conceivable outcomes. Thus, guaranteeing of all these operational conceivable outcomes accurately is a noteworthy test to the software industry.

### Product Visibility

Since the industrial items are obvious, the greater part of its deformities can be identified during the manufacturing procedure. Likewise the absence of a section in a industrial item can be effectively identified in the item. Be that as it may, the imperfections in programming items which are put away on diskettes or CDs are imperceptible.

### Product Development and Production Process

In an industrial product, defects can be distinguished during the accompanying stages −

* **Product development** − In this stage, the originators and Quality Assurance (QA) staff checks and tests the item model to distinguish its faults.
* **Product production planning** − During this stage, the generation procedure and tools are planned and arranged. This stage additionally gives chances to investigate the item to distinguish the imperfections that went unnoticed during the development stage.
* **Manufacturing** − In this stage, QA methodology are applied to recognize detect failures of items themselves. Defects in the item distinguished in the primary time of assembling can as a rule be amended by an adjustment in the item's plan or materials or in the production tools, in a way that wipes out such imperfections in items made in future.

Be that as it may, on account of software, the main stage where defects can be identified is the development stage. If there should be an occurrence of software, product production planning and manufacturing phases are not required as the manufacturing of software copies and the printing of software manuals are conducted automatically. The variables influencing the detection of imperfections in software items versus other industrial items are appeared in the accompanying table. Be that as it may, on account of software, the main stage where defects can be identified is the development stage.

If there should be an occurrence of software, product production planning and manufacturing phases are not required as the manufacturing of software copies and the printing of software manuals are conducted automatically. The variables influencing the detection of imperfections in software items versus other industrial items are appeared in the accompanying table.

Be that as it may, on account of software, the main stage where defects can be identified is the development stage. If there should be an occurrence of software, product production planning and manufacturing phases are not required as the manufacturing of software copies and the printing of software manuals are conducted automatically. The variables influencing the detection of imperfections in software items versus other industrial items are appeared in the accompanying table.

|  |  |  |
| --- | --- | --- |
| **Characteristic** | **Software Products** | **Other Industrial Products** |
| Complexity | More than thousands of functioningchoices | thousand functioning choices |
| visibility of product | Unseen Product Hard to discover defects by vision | Noticeable Product Effective discovery of defects by vision |
| Nature of development and production process | can defect defects in only one phase | can detect defects in all of the following phases   * Product development * Product production planning * Manufacturing |

These attributes of software, for example, complexity and imperceptibility make the advancement of SQA approach and its effective execution a profoundly expert test.

1. **ACTIVITIES OF SQM**

SQM is split into three main activities:

1. **Quality planning**

The choice of proper methods and standards from this structure and adjust for a particularsoftware project. Before software development starts, quality planning mustoccur. QP includes the production of objectives and targets for product, just as the formation of a vital arrangement that will push you to effectively meet the goals you spread out. QP is regularly viewed as the most significant part of SQM, as it builds up a solid diagram for the remainder of the procedure to pursue – prompting the most ideal finished result.

1. **Quality assurance**

The development of a structure of authoritative techniques and principles that lead to great quality software. The QA period of SQM includes the genuine structure of the software program. With great SQM set up, item execution will be checked en route to guarantee that all measures are being pursued. Reviews might be performed and information will be gathered all through the aggregate of the procedure.

This stage can include:

* encouraging documentation process benchmarks, for example, the production of well-characterized building records utilizing standard layouts
* mentoring how to lead standard procedures, for example, quality audits
* performing in-process test information recording strategies
* identifying models, assuming any, that ought to be utilized in software advancement forms

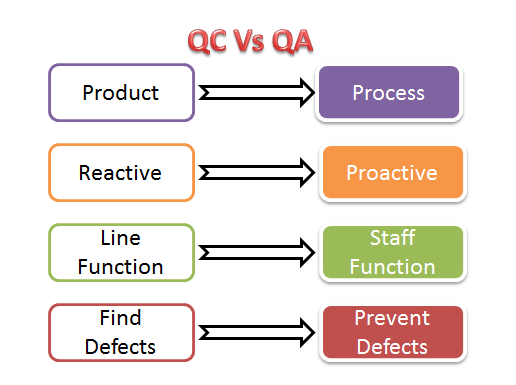


Fig 2: QC Vs QA

1. **Quality control**

Meaning of procedures guaranteeing that software development keeps the quality methodology and standards. The progression of SQM where testing at long last becomes possibly the most important factor, quality control is set up to find bugs, assess usefulness and then some. Contingent upon the consequences of the quality control stage, you may need to return to advancement to iron out crimps and make some little last alterations. Having a product quality administration plan set up can ensure that all industry measures are being pursued and that your end-client will get a well-grown, great item.

Activities include:

* release testing of software, including legitimate documentation of the testing procedure
* examination of software and related documentation for non-conformance with measures
* follow-up audit of software to guarantee any required changes itemized in past testing are tended to

Quality management gives a free beware of the product and software development process. It guarantees that project

expectations are predictable with organizational measures and objectives.



Fig 3. Importance of main software project management areas

1. **CONCLUSION**

Quality Assurance is to check whether the item created is fit for use. For that, Organization ought to have procedures and principles to be pursued which should be enhanced anperiodic premise. It focuses primarily on the nature of item/ service that we are giving to the clients during or after usage of software. A framework for SPM titled 3PR structure is introduced. The system comprises of four fundamental SPM zones: people management, process

Management, product management, and risk management.Fifteen project managements were recognized and arranged under these fundamental zones.

**REFERENCES**

1. Demir KA (2008) Measurement of software project management effectiveness. DoctoralDissertation, Naval Postgraduate School, Monterey, California, USA
2. Philips D (2000) The software project manager’s handbook, principles that work at work.IEEE Computer Society, Los Alamitos
3. El Emam K, Koru AG (2008) A replicated survey of IT software project failures. IEEE Softw25(5):84–90
4. Humphrey WS (1996) Using a defined and measured personal software process. IEEE Softw13(3):77–88
5. Humphrey WS (1997) Introduction to the personal software process. Addison-Wesley,Reading
6. Hughes B, Cotterell M (2002) Software project management, 3rd edn. McGraw-Hill International(UK) Ltd, Berkshire
7. Boehm BW(1991) Software risk management: principles and practices. IEEE Softw 8(1):32–41
8. Jones C (1994) Assessment and control of software risks. Prentice-Hall, Englewood Cliffs
9. Muller R (2003) Determinants for external communications of IT project managers. Int J ProjManag 21:345–354
10. Jones C (2004) Software project management practices: failure versus success. Crosstalk – JDefSoftwEng 17(10):5–9
11. Cicibas H, Unal O, Demir KA (2010) A comparison of project management software tools(PMST). In: Proceedings of the Software Engineering Research and Practice (SERP 2010), pp560–565, July 12–15, 2010, Las Vegas, Nevada, USA