Software defect Prevention

Kartheek
Department of Software Engineering
9304163778
BTH, Karlskrona
chilla.kartheek87@gmail.com

Prithvi Raj Sirigudi
Department of Software engineering
939401024238
BTH, Karlskrona
Prsi15@student.bth.se

Abstract

The paper mainly focuses on the defect prevention for the product. In this article the authors present a join view on the techniques/process/models for defect prevention. The article includes interpretation of what is defect prevention? Advantages and disadvantages of TPM's in defect prevention. Cost and benefits of defect prevention and the necessary recommendations for using the TPM's in defect prevention.

I. Introduction

1. Interpretation of definition of defect prevention

In general defect increases the cost and rework. Many organization's face defect problems' that impact the company growth and important operations [6]. The main challenges in any software product line development lies in decreasing the number of defects. Process failures are highly regard as reason behind defect occurrence rather compare to human failures. Identification and prevent the occurrence of process failure will prevent large extent of defect occurrence [6]. Defect Prevention is the strategy which is applied in the software development life cycle that prevents the root cause of occurrence of defects and eliminate the recurring. Defect prevention is highly regarded for total quality management. Defect prevention is considered as level 5 key process area in the Capability maturity model, which help to identify the defects encountered in past and use them as checkpoints for the future to prevent the recurring of similar defect1. IT organizations have the established process that have the defect prevention methodologies that help to improve the quality and productivity and reduce the development costs [1]. These established process are carried out as their responsibilities.

2. Techniques commonly used for defect Prevention

There exists several type of defect prevention techniques available to identify the defects… Among them some most commonly used techniques and their advantages and disadvantages are listed below. The defect prevention techniques that are chosen are Joint Application Development (JAD), Failure Models and Effects Analysis (FMEA), Fault Tree analysis (FTA).

a) Joint Application development (JAD):

IT is generally used in the initial phase of the development. It is a technique to join developers, testers, designers to form a collaborative team [2]. It helps to share common view on the requirements. This is and efficient technique during the requirement phase. It involves structured integration for collaboration. It is proved to produce high quality requirements specification document during requirement engineering phase [2]. JAD reduces the cost and time duration during the data analysis, data gathering and requirement phase. Effort of the JAD process is depend on the experience level of the participants. It is useful in developing of new system or enhancing the existing system.

b) Failure Models and Effect Analysis (FMEA)

This technique is used in the earlier stage of design. It helps to identify the failure modes and prioritize them based on the frequent occurrence, likelihood and impact. This help to extract which defect can cause serious impact and the preventive measure to be taken to avoid them [3]. IT is the proactive tool which involve many step to follow and which takes lot of time and effort to run. In this case the failure mode with highest potential is found out and action plan is defined to reduce it [3]. Even though a lot of effort is put into it provide improved quality and reliability of the software product.

c) Fault Tree Analysis (FTA)

It is highly based on the origin of the defect in the software system. Boolean logic is used for combination of events that might result in the fault occurrence in the software product [4]. For an effective FTA the skill and knowledge about its execution should be known to the participants. THE FTA start with one process specific failure and attempt to find the root cause and its corresponding relationship [4]. IT generally considers one fault at a time thus takes a lot of effort and time. It gives the information about the operating environment and human error if so any.

3. Advantages and disadvantages.

Technique	advantage	disadvantage
JAD [2]	• Knowledge and understanding from different experts are taken into consideration to alleviate the mistakes. • Active technical (developers, designer) and non technical (end users) collaborative participation reduces ambiguity.	• Wrong problem can be put forth due to improper background work. • Poor selection criteria for JAD evaluation. • Unequal amount of participation, only certain people ideas are considered.
FMEA [3]	• Identify and remove failure modes in earlier stage of development. • Focus on prevention. • It reduces the recurring of the same type of failure.	• Inexperience of the participants. • Focus on only major failure modes. • As it is inductive and top down multiple failure modes cannot be identified.
FTA [4]	• Identify the possible combination of specific failure. • Graphical representation of causes. • IT has a specific advantage to identify complex and multiple failures.	• Only one fault is covered at a time. To Cover all faults it takes lot of FTA's. • IT is time and resource depended.

4. Solution to Defect Prevention and Avoidance in the final product.

Now a day the organizations are concentrating on the defect prevention rather than the defect identification and then removal of the defect during later development stages. Testing is considered main method to identify the defects. Various testing schemes like unit testing, integration testing, regression testing, system testing are conducted to identify the defect after the production stage. Cost benefit analysis that help to compare the cost for using the technique and the quality of the product obtained. Many organizations don't adopt to the prevention methods but follow other detection methods to avoid defect. Approximate defect free product can be obtained by combining both defect prevention compliment and defect prevention method.

5. Cost and benefit analysis for the defect prevention techniques.

Cost Benefit analysis of the techniques

Cost invested in any product is estimated based on the effort put into that process. In this cost based analysis, the benefits are analyze based on the return of investments. Effort involves and the techniques effectiveness

JAD: JAD involve a lot of human resources. It involves developer, designer and testers involve in defect prevention [2]. The investment rate increase is dependent on the type of project that is being involved. Effort put into by the participants is dependent on the experience level, as they find it easier to identify unlike non experienced people [2]. JAD is highly useful in projects where requirements are loosely defined.

FMEA: It is defect prevention tools [3]. That involves identify and reduce defects in the earlier stage of development. This enables to reduce the development cost in the further phases. Considerable time frame is required to identify the defects within the development phase so these are some time constraints to be looked into but it is effective [3].

FTA: In this tool the defect prevention and the rot cause and the operating environment in which it is conducted and the involvement of any possible human error are also identified [4]. It involves one defect at a time. It takes a lot of time and effort as it identifies one defect at a time. It involves high development costa and less return of investment's [4].

6. Recommendations and motivation about the use of the techniques.

An emerging company XYZ is involved in developing software and maintenance services. This company was started four years ago and gained a good will from their customers [5]. A new software product especially focused on quality need to be delivered to the customer in mean time. This software product is not quite complex not to easy but moderate level [5]. The resources needed to accomplish this project time in estimated time are available within company [5]. An iterative incremental development life cycle model is implemented for execution of the project. The recommendations are outline below for outputs of project.

Requirements: JAD is one of the best defect prevention technique to be implemented in requirements phase as it involves all team members like analysts, developers, testers, etc. in requirements phase to sort out all the essential needs. Moreover, JAD provides stakeholders a clear idea in finalizing the requirements of a specified project [5]. So JAD is an effective defect prevention technique in order to meet all the requirements as the customer wanted.

Design: Since the development of the project is done incrementally prototyping will be the best for the developers to provide a clear visually understandable functionality of the product through design [5]. Moreover, prototyping provides software designer and implementer in having a valuable feedback from the users in the early stages of the project [5]. There are several measures but prototyping is the best for this type of described software industry.

Implementation: As described the project is mainly focused on quality of the product. Hence Test Driven Development will be the efficient defect prevention method. Implementing test driven development sometimes increases the work load of the developers in initial stages but the project will be accomplished in the best way as per the customer needs.

Testing: Testing is important phase which actually reflects the customer's satisfaction. Different types of testing like unit testing, system testing and integration testing should be involved for the best results prior to the delivery of the product. In this phase all the possible defects are detected and then precise solutions are figured out and fixed. Finally, the desired software product is delivered to the customer with satisfaction.

REFERENCES

[1] N. S. Eickelmann, "Empirical studies to identify defect prevention opportunities using process simulation technologies," in Software Engineering Workshop, 2001. Proceedings. 26th Annual NASA Goddard, 2001, pp. 22–25.

[2] E. J. Davidson, "Joint application design (JAD) in practice," J. Syst. Softw., vol. 45, no. 3, pp. 215–23, Mar. 1999.

[3] D. H. Stamatis, Failure Mode and Effect Analysis: FMEA from Theory to Execution. ASQ Quality Press, 2003.

[4] M. Towhidnejad, D. R. Wallace, and A. M. Gallo, "Fault tree analysis for software design," in 27th Annual NASA Goddard/IEEE Software Engineering Workshop, 2002. Proceedings, 2002, pp. 24–29.

[5] S. Kumaresh and R. Baskaran, "Defect analysis and prevention for software process quality improvement," Int. J. Computer. Appl., vol. 8, no. 7, 2010.

[6] C. C. Jones, Software quality: Analysis and guidelines for success. Thomson Learning, 1997.