# Defect Prevention

## Assignment 4

Sai Sandeep Chikkala
930617-5051
saca14@student.bth.se

Ravichandra Kumar Kollu
920525-6895
rako14@student.bth.se

*Abstract*—**This paper comprises of opinions of authors on various aspects of defect prevention that are based on literature knowledge and experience. The report outlines the team interpretation of defect prevention and provides advantages, disadvantages of few common defect prevention techniques. Also cost benefit analysis for these techniques and recommendation of these techniques in an industrial setting is mentioned.**

## I. INTERPRETATIONS OF THE DEFINITION OF DEFECT PREVENTION

Defect Prevention Process is a set of measures taken to prevent the occurrence of defects in the final product. Rather than fixing the defects after the product development, the defect prevention techniques helps to avoid them in first place. Adopting defect prevention methods ensure the quality of the delivered product. As team interprets defect prevention technique is not the silver bullet for achieving defect free products, but these techniques combined in proper proportion with the other defect detection methods will generate more than 95% defect free product. Also, it is important to choose the type of technique which suits the context. Various factors like available resources, product nonfunctional requirements, phase of development etc., play an important role in the process of adopting a defect prevention technique. There exist variety of techniques that are specified in the literature [1], like JAD, prototyping, Reusable code, structured tools etc., But it's important to note that a combination of these techniques with other defect detection methods yield best possible results i.e. high probability of having low defects in the final product [1].

## II. TECHNIQUES COMMONLY USED FOR DEFECT PREVENTION

There seem to exist various kinds of defect prevention techniques which have been specified in the literature, but not all of them are used in practice [1][2]. This section highlights three of the most common techniques used for defect prevention, their advantages and disadvantages. JAD (Joint Application Development), Prototyping and Test- Driven Development.

JAD:

Joint Application Development is generally used in the initial stages of development. This approach allows developers and analysts and other user participants come together and share common views on the gathered requirements [3]. This has been proved as an efficient defect prevention technique for requirements phase.

PROTYPING:

Prototyping is developing a sample model or a partial implementation of the software components so that it serves as a guideline for developers to know beforehand what they implement and for customers to check if implementation is going according to their needs[4].

TDD:

TDD is practice of writing tests before writing code. Initially the test should fail and then a minimal amount of code has to be written to make the test pass. TDD is into practice from a decade and the existing empirical research proves the effectiveness of TDD in terms of product quality [5]. Since tests are written beforehand and code is then written accordingly to pass the test, it is obvious that the functionality is tested during implementation. Continuous testing and then refactoring the code enable the usage of TDD to realize the expected product quality. This practice helps to maintain quality in development phase i.e. it lowers defects that arise from development phase.

Advantages and Disadvantages:

| Technique | Advantage | Disadvantage |
|---|---|---|
| JAD | • Enables to have multiple perspectives on a topic.<br>• Enables thorough validation of requirements by all the available stakeholders [3]. | • It needs planning and crucial amount of effort.<br>• Not very well suitable for large and complex projects[6]. |

| Prototyping | • Better understanding of requirements and design by development team and testers [4]. | • Less software quality when developed a throw away kind of prototype [4]. |
|---|---|---|
| TDD | • Increased code quality and hence decreased defects in code [5]. | • Takes more time in the beginning, reduces developer productivity [5]. |

## III. SOLUTIONS TO DEFECT PREVENTION TO AVOID DEFECTS IN THE FINAL PRODUCT

As mentioned earlier combination of both defect prevention methods and defect detection method complement to produce nearly defect free product. Many companies do not successfully adopt prevention methods but follow alternative methods i.e. detection methods to avoid defects in the products. Some of most common defect detection methods are testing, inspections, reviews. Application of defect detection methods and their results provide implications for calculating the defect removal efficiency of the method. Testing is the main method used in organizations to detect and fix defects [2]. Various dimensions of testing like Unit testing, system testing, regression testing, integration testing are performed to identify the defects after production.

## IV. COST AND BENEFIT ANALYSIS OF THE DEFECT PREVENTION TECHNIQUES

Cost benefits analysis of techniques:

Cost incurred for any method or process depends upon the effort that is put into that process. For defect prevention methods the cost benefit analysis could be interpreted as costs incurred for implementing a technique and it is related to the quality of the product that is obtained. Benefit can be measured either with means of quality or reduced effort of maintenance/ debugging. Fixing the defects in the early stages of development, can save exponential amount of money than detecting and fixing them in later stages of development [7]. The return of investment, effort and the effectiveness of the techniques are mentioned below:

JAD:

JAD is defect prevention technique mainly used for detecting the requirements defects [1]. According to [7]

majority of defects that incur major costs for fixing the defects occur in requirements phase. Also, to perform JAD all the stakeholders like developers, testers, analysts, and customer has to participate. This calls for more effort from human resource. But given the context of project JAD can provide high return of investment and is efficient to use in the early stages of development.

PROTOTYPING:

Prototyping is used to identify code defects and design defects. Prototype typically guides the implementation of the design. It helps to align the requirements to implement the desired functionality. There are various types of prototypes [4]. The return of investment is low if the prototype is throw away and is relatively high if it is evolutionary. The effort for prototype is medium. This technique is effective if used in design phases.

TDD:

Test driven development addresses the prevention of defects in development phase. Effort required for TDD implementation depends on the experience of the developers. Compared to traditional method TLD, TDD is proved to be less productive i.e. takes more time upfront of the development [5]. It is because of need to write effective tests. Comparative return of investment is high in terms of the quality of the product.

## V. RECOMMENDATION WITH MOTIVATION ABOUT USE OF TECHNIQUE

Industry setting: A company X develops software products and offers services and it is has good reputation. Customer who wanted a software product especially emphasizes on quality of the product. The product is relatively small and less complex. Company has enough resources to deliver product on estimated time. And the organization follows iterative incremental development cycle.

Generic recommendations: These are the very generic recommendation for any organization which is adopting defect prevention methods. The four key activates that a software development organization need to follow when implementing defect prevention process are: [2]

Casual analysis meeting: this meeting is held after completion of a development stage after fixing the identified defects. It typically involves 5 developers and involves identifying defects and categorizing them. The cause of every defect is explored in this meeting and provides suggestions for next stages.

Action Team: This team is a subset of development team whose responsibility is to make sure the above suggestions are implemented in the development. Responsibilities are assigned depending on the developer's expertise.

Stage kickoff meetings: These kickoff meetings are held before the beginning of a stage. This is led by the stage

technical leader and provides information regarding the upcoming stage i.e. about the process to be followed, the common errors that happen in this stage etc.,

Data Collection and tracking: The identified defects and suggestions in the above activities are collected and stored. The suggestion and necessary actions to be performed by the action team are tracked.

Now, Recommendations that are specific to the company are outlined below:

Since different phases of development include different processes and there will be defects that are very much specific to particular phase, the defect preventive actions are recommended that are effective in the context of the particular development phase [2].

Requirements:

JAD is the effective defect prevention technique to be used in requirements phase. Since JAD involves discussions of all stakeholders while finalizing the requirements, it will help all people from various roles i.e. developers, architect and tester to have an overview of the requirements and what has to be implemented.

Design:

For design, prototyping is the efficient defect prevention method. Since, company follows iterative and incremental approach having prototypes will perfectly suit for representing the product through design. Developers can have a prototype to see visually and understand what functionality they need to implement. Although there exist several other competent preventive measures, prototyping particularly evolutionary type will fit the given industry setting.

Implementation:

Since there are enough resources and also customer emphasizes on quality of the product, test driven development is the best fit defect preventive method. Although the developer productivity will decrease in the initial stages, the customer satisfaction in the end will complement it.

Testing:

Various kinds of testing like unit testing and integration testing, system testing has to be performed in this stage. Also, the product is handled to the customer after this stage, so it has to be done in efficient way so that the defects are detected and fixed.

## REFERENCES

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

[2] R. G. Mays, "Applications of defect prevention in software development," *Sel. Areas Commun. IEEE J. On*, vol. 8, no. 2, pp. 164–168, 1990.

[3] E. J. Davidson, "Joint application design (JAD) in practice," *J. Syst. Softw.*, vol. 45, no. 3, pp. 215–223, 1999.

[4] V. S. Gordon and J. M. Bieman, "Reported effects of rapid prototyping on industrial software quality," *Softw. Qual. J.*, vol. 2, no. 2, pp. 93–108, 1993.

[5] N. Nagappan, E. M. Maximilien, T. Bhat, and L. Williams, "Realizing quality improvement through test driven development: results and experiences of four industrial teams," *Empir. Softw. Eng.*, vol. 13, no. 3, pp. 289–302, 2008.

[6] E. Carmel, R. D. Whitaker, and J. F. George, "PD and Joint Application Design: A transatlantic comparison," *Commun. ACM*, vol. 36, no. 6, pp. 40–48, 1993.

[7] B. Boehm, V. R. Basili, H. D. Rombach, and M. V. Zelkowitz, *Foundations of empirical software engineering: the legacy of Victor R. Basili.* Springer, 2005.