# Proposed an Integrated Model to Detect the Defect in Software Development Projects

**Dr. Ayman E. Khedr** *Ayman_khedr@helwan.edu.eg*
*Department of Information System*
*Helwan University*
*Cairo, Egypt Post*

**Ahmed Fouad Omran** *Ahmed_Fouad2025@yahoo.com*
*Department of Sciences*
*Helwan University*
*Cairo, Egypt Post*

**Abstract**

Currently, software quality assurance does not apply completely on the development of software industry and this leads to some challenges in the industry of software, specially, concerning cost and time consuming. While, the success of software development project depends on the application quality assurance standards, which starts by the pre-project and continue through the project till it reaches the user at the end. The aim of this model is the challenge to produce not only the software development projects without defects but also the customer's acceptance and satisfaction of the software. Thus to prevent software development projects failure, this defect should be detected at an early stage. This study will deal with; the software quality assurance inspection and testing, in the whole stages of software development projects. Then proposing the quality assurance inspection and testing model for the software development projects process, which known as "Integrated Model".

**Keywords:** Software Quality Assurance, Software Inspection, Software Testing, Software Development Projects

## 1. INTRODUCTION

Software quality assurance is "A planned and systematic pattern of all actions necessary to provide adequate confidence that an item or product conforms to established technical requirements. And, a set of activities designed to evaluate the process by which the products are developed or manufactured" [1]. Mostly, the success of any software development projects depends on three major factors, which are, quality, cost and time. As it was found, that one of the reasons for the failure of software development projects is that, some organizations hesitate to place a portion of the project's budget on the quality assurance process [2]. Software quality assurances are responsible to reduce the defect, and repair this defect before it reaches the production environment, because repairing the defect in the production environment is more expensive than repairing it, in the test environment [3].

No doubt that the projects had improved from 16% in 1994 to 32% in 2006, that refers to the better training, and better tools and techniques, on the contrary project complexity and environments have increased while the time to deliver has been reduced [4]. Therefore, the software quality assurance is trying to reach the best product for users by reducing the cost and time. The aim of this model is the challenge to produce not only the software development projects without defects but also the customer's acceptance and satisfaction of the software.

## 2. BACKGROUND AND RELATED WORK

Early studies, dealt with some notions concerning the research proposed topic, the researcher discovered that, they only dealt with software testing and software inspection as an optional

choice by which time and cost constrains play a very important role [5]. So far, the following subsections will illustrate the software inspection and software testing notions.

### 2.1 Software Inspection

It is a proven method that enables the detection and removal of defects in software artifacts as soon as these artifacts are created [6] [7] .


**FIGURE 1**: Structure of software inspection process.

### 2.2 Software Testing

Software testing is a process, or a series of processes, designed to make sure computer code does what it was designed to do and that it does not do anything unintended. Software should be predictable and consistent, offering no surprises to users. In this book we will look at many approaches to achieving this goal [8].

There are different types of software testing such as [9].

- **Black box testing,** this test does not only focus on the internal design but also depend on the functionality of the requirement.
- **White box testing**, this test depends on the structure of the program testing on the source code and coding style.
- **Unit testing,** this test depends on specific functions of the program and its progress to do a test, as it requires knowledge of interior design details of the program.
- **Integration testing:** tests the joint parts of the program to ensure that it is working well together, as due to software components, when one of these components work individually it works well, but when collected with other components they fail.
- **System testing:** is one of black – box testing which cover all the components of the system.
- **End to end testing or user acceptance testing:** the test contains a test environment which is very similar to the real environment which the customer ensures if the software meets his requirements or not.
- Regression testing: re – testing after the repairing of the software problems.

## 3. RESEARCH PROBLEM

The failure of software projects was due to the incomplete stage or stages of these projects, which lead the software projects to end with defects. For example, in 2003 the American ministry of Treasure sent 50 thousand social security checks without beneficiary name which was due to a defect in the program. [10]

While, the most important obstacles facing the software industry is the failure of software development project due to some reasons which could be summarized in the following [11]:

1- Not enough time, because the deadline dates are decided before the projects starts and are non-negotiable, while it is important to spend time to create a good design.
2- Budget constraints, as many projects have “lowest price most successful candidate” policy, or an unrealistically low budget, not based on the true requirements.
3- Neglecting the quality assurance, causes incomplete documentation for code changes, the design flaws and implementations.

The previous reasons lead to the failure of software development projects; accordingly this failure leads to re-work on all stages of the software development projects in order to detect the defect. Also, the reports of The Curious Case of the CHAOS Report (2009) showed that [4]:

1- Software development projects success was 32%.
2- Software development projects which faced challenges were 44%.

3- Software development projects failed 24%.

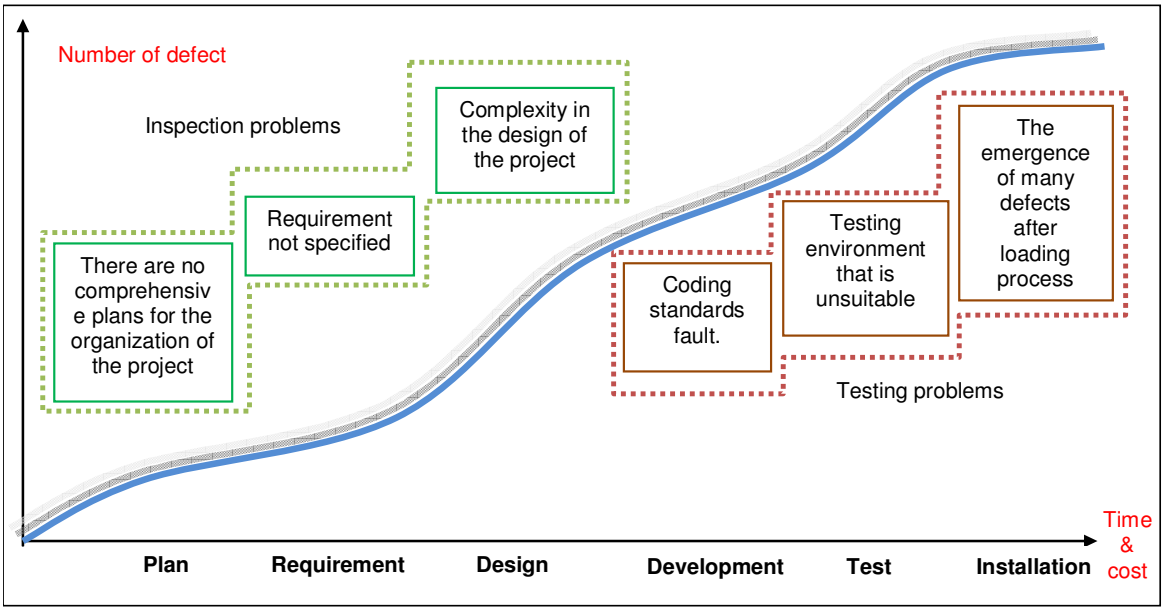


**FIGURE 2:** Changes of the number of defects with time and cost.

Source: Adapted from Quality Improvement by the Real-Time Detection of the problems, [12]

## 4. RESEARCH METHODOLOGY

The software development projects consist of a series of stages; these stages are [13] [14]. (1) The project plan stage this stage begins before the project starts. (2) The requirement specification this stage is responsible for the specific requirement which will be used during the project. (3) Design stage starts when the component is selected. (4) Development stage, this stage implements the design document. (5) Testing and integration stage, which responsible for the software testing, (6) Installation stage which installs the software into the production environment.

The software development project identifies six stages; each stage contains a set of functions. The following table shows these six stages and their functions.

| Phase | Function |
|---|---|
| Project Plan | - General view of indented software product.<br>- Establish the basic project structure.<br>- Evaluate the risk of the project. |
| Requirement Specification | - Software requirement document.<br>- Component schedule.<br>- Work structure. |
| Design | - Software design document.<br>- Describe design element.<br>- Object diagram. |
| Development and Implementation | - Describe of program operations to meet the Requirement.<br>- make sure coding stander.<br>- Operations of introduction of source code.<br>- implementation of the program design by the documentation |
| Integration and Testing | - check to migrate data from the development Environment to test environment.<br>- Application procedures for acceptance test. |
| Production and Installation | - install the software to production environment.<br>- Training plan.<br>- Customer accepts the software. |

**TABLE 1:** Functions of software development projects stages

## 5. THE PROPOSED INTEGRATED MODEL (IM)

The proposed integrated model proposes a general model to detect the defection in each stage of software development project before entering the next stage. This integrated model is based on two methods namely, inspection (static test) and testing (dynamic test), where we apply the inspection on the first three stages then apply the testing on the other three stages of the software development projects.

The three stages of the first phase of software development projects are project plan, requirement specification, and design, respectively, in such stages the programmer deals with software inspection (static test). The static test depends on inspecting and reviewing, after finishing the process of these stages, the integrated model prepares the report to ensure the end the first phase of the model.

While the other three stages of the second phase of the model contains the development, integration testing, and installation, respectively, in these stages the programmer can deal with software testing (dynamic test). The implementation of the program starts from documenting the data then to migrate data from the development environment which tests the environment and applies the needed procedures in order to accept the test or not. Finally, the software will be installed according to the production environment. Therefore, the software inspection (static test) will be used in the first phase of the model and the software testing (dynamic test) will be used in the other phase of the model.

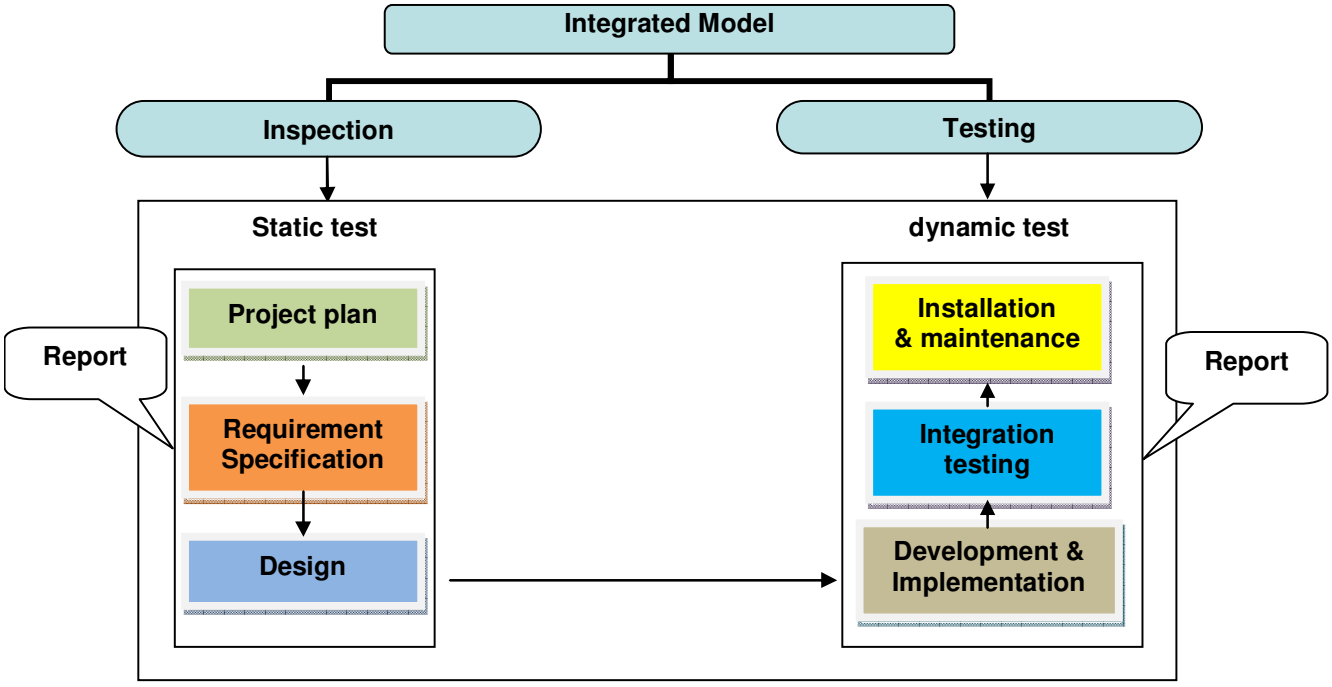


**FIGURE 3:** "Integrated Model" for software development projects

## 6. RESEARCH CASE STUDY

The case study was chosen according to the analysis of an Islamic bank case in Cairo, Egypt, to review the defects in the whole stages of its software development projects. As shown below, in figure 4. The figure shows the relation between time, cost and quality, the defects detected before adopting the Integrated Model indicates that when the quality decreases the time and costs increase in the whole stages of the software development projects. As in the whole six stages, many defects appear.

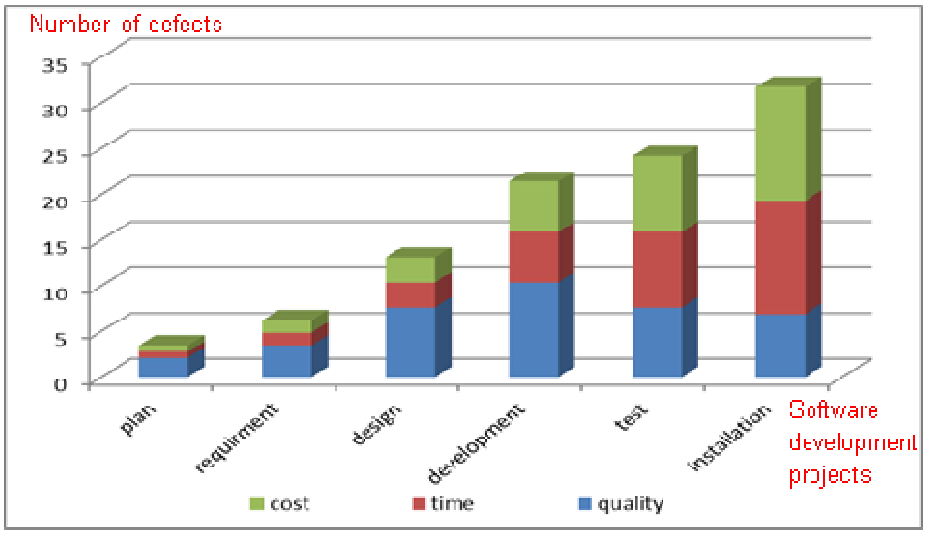


**FIGURE 4:** The relation between cost, time, and quality in software development projects in Islamic bank.

While figure 5 represents the same case study but after adopting the proposed Integrated Model, as it shows the difference between both cases in which the quality increased totally in the first four stages without taking over the projects time or costs, even though it affects the last two stages of the software development project by a small percentage which if compared to the figure 3 could be neglected due to the high quality of the project results.

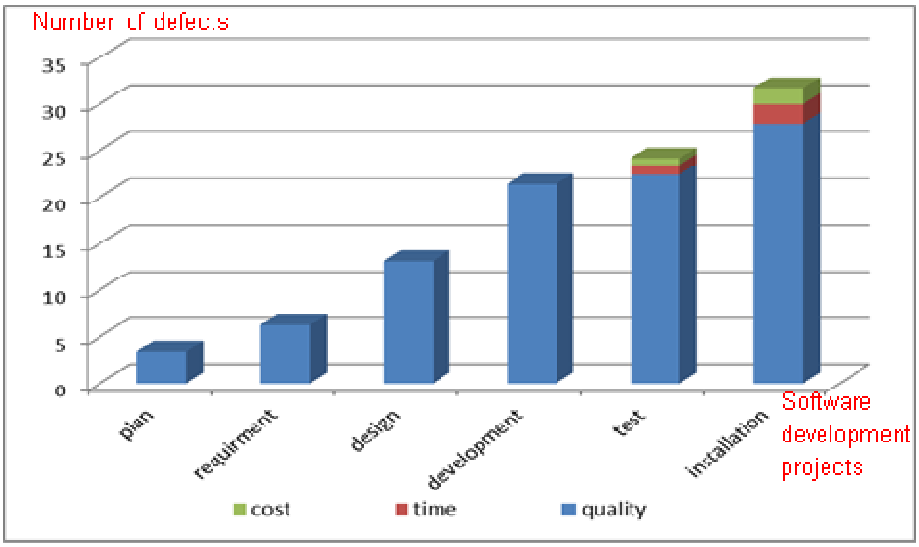


**FIGURE 5:** The relation between cost, time, and quality in software development projects after adopting the integrated model.

## 7. FINDINGS ANALYSIS AND DISCUSSION

So after adopting the Integrated Model of software development projects on the three stages of the inspection process, it indicated that; firstly, the Integrated Model revised the project's plans and prevented the project's defects, as it was able to complete correctly to the next stage. Secondly, the model reviewed the requirements, removed all defect, and prepared the report; then was ready to engage in the other stage. Thirdly, the design document and the design

element were checked and prepared a report that confirms the completion of the design stage then ready to be implemented.

While, on the three stages of the testing process, the Integrated Model indicated that; fourthly, debug the resulted errors, ensured that the working code is correct and prepared the report which underscores, the end of the development stage and the start of the testing and integration stage, fifthly, all components and procedures were tested to ensure that the software works well and prepared the end of stage report, while the software is ready to be installed in the production environment. Finally, the installation stage, the software was installed and few defects were found, which do not affect the software project, latterly customer accepts the software.

To be brief, in the first four stages (plan, requirement, design, and development stages) of the software development projects, the Integrated Model achieved a noticeable success, while in the testing and installation stages some defects appeared, but it did not affect the software project and the user accepted the software.

## 7. CONCLUSIONS

There are many methods and techniques to detect the defects in software development projects. But in spite of the of whole these methods there are still obstacles facing the software development project, and the most important obstacle, is that, it exceeds most of the projects' time and costs identified.

In this study, we found that the lack of participation of quality assurance in every stage of the software development project is one of the key factors for the failure of these projects. The quality assurance process advances, while the cost overruns and time overruns decrease. So the quality assurance considered to be more important in software development projects.

In this study we proposed an integrated model to detect the defect and prevent it in software development projects; this model is based on inspection and testing.

## 8. REFERENCES

[1] GALIN, D., Software Quality Assurance: from theory to implementation, (2004), 50(1), pp.14-34.

[2] Ashrafi, N. The impact of software process improvement on quality: In theory and practice. Information & Management, (2003), 40(7), pp.677–690.

[3] Heusser, M., An ounce of prevention. Better Software, 3, 48, (2004).

[4] Jorge Dominguez, The Curious Case of the CHAOS Report (2009), Project Smart, 2009

[5] Evertsso, G. (2002). Inspection vs. Testing, Blekinge Institute of Technology Software Verification and Validation (PAD001) from http://www.guzzzt.com/files/coding/inspectionvstesting.pdf

[6] Ackerman, A. F., Buchwald, L. S., and Lewsky, F. H., (1989). Software Inspections: An Effective Verification Process. IEEE Software, 6(3), pp.31-36.

[7] Department of Computer Science University of Toronto (2001), Inspections & Reviews from http://www.cs.toronto.edu/~sme/CSC444F/slides/L09-Inspections.pdf

[8] Glenford J. M., (2004). The Art of Software Testing, Second Edition, 5(21), pp.8

[9] Types of software Testing, from

http://www.softwaretestinghelp.com/types-of-software-testing/

[10] John E. Bentley, Software Testing Fundamentals—Concepts, Roles, and Terminology, Wachovia Bank, Charlotte NC, pp. 141 – 30.

[11] Haughey, D., PMP, project smart (2000-2011), Why software projects fail and how to make them succeed, 1-2

[12] Suzuki, T., (2011), Quality Improvement by the Real-Time Detection of the Problems, 15: 1-6

[13] Senyard A., and Michlmayr M.(2004). How to Have a Successful Free Software Project, from http://www.cyrius.com/publications/senyard_michlmayr-successful_project.pdf

[14] Zrie, M., (2010), Software Project plan Structure, from http://www.zriel.com/lamp/49-software-project-plan-structure