

# Chapter XVI

## Software Quality Assurance

**Dawn M. Owens**

*University of Nebraska at Omaha, USA*

**Deepak Khazanchi**

*University of Nebraska at Omaha, USA*

### ABSTRACT

*Successful implementation of IT (information technology) projects is a critical strategic and competitive necessity for firms in all industrial sectors today. However, due to cost overruns, schedule delays, unfilled requirements and poor quality, it is reported that less than 30% of IT projects are perceived to be successful. Much has been written about causes of project failure and many have provided best practices and critical success factors for effective management projects, yet projects still continue to fail. As a first step to overcome systemic causes of project failure we propose a unified definition of software quality assurance (SQA). We use this definition to develop and present an approach to SQA that focuses on controlling risks and provide a framework for assuring the development and project management life cycles.*

### INTRODUCTION

Successful implementation of IT (information technology) projects is a critical strategic and competitive necessity for firms in all industrial sectors today. This is even more important in times of scarce resources needed for other competing strategic initiatives important to a firm. However,

it is well established that a significant numbers of software and infrastructure projects still fail to deliver on time and within target costs and specifications (The Royal Academy of Engineering, 2004). Due to cost overruns, schedule delays, unfilled requirements and poor quality, less than 30% of IT projects are perceived to be successful (Glass, 2006).

Companies spend billions of dollars on IT projects each year, with the average spending at 4 to 5 percent of annual revenue on IT (Charette, 2005). In 2005, organizations spent an estimated \$1 trillion on IT hardware, software, and services worldwide (Charette, 2005). Project failures have cost the US economy at least \$25 billion and maybe as much as \$75 billion (Charette, 2005). Based on the current project failure statistics, companies are faced with billions of wasted dollars on IT projects alone.

Many have written about the causes of project failure and have provided suggestions and critical success factors for effective management of projects. This chapter takes a different approach by examining how *software quality assurance practices* can impact project outcomes. It is the authors' contention that **software quality assurance (SQA)** plays a critical role in the software development lifecycle (SDLC) and can impact a project's overall success. Failure to pay attention to SQA can result in budget overruns, schedule delays, failure to meet project objectives and poor customer satisfaction (Chow, 1985). In fact, quality is considered a vital requirement of software products, a business essential, a competitive necessity, and even a survival issue for the software industry (Murugesan, 1994). Strong quality focus is emerging in all phases of the software development lifecycle with increasing emphasis on product quality, process maturity, and continual process improvements (Murugesan, 1994).

No matter how advanced the tools and techniques – all will come to nothing and the project will fail if the quality management system is not effective (Gill, 2005). The term SQA is often misunderstood, being viewed strictly in terms of software testing activities. However, in our view, SQA is a much broader concept. As we will present in this chapter, SQA is a set of assurance processes that are implemented across all phases of the project and software development lifecycles. Using this definition, we propose in this chapter a comprehensive SQA process. The

SQA phases presented will provide IT professionals with a framework for software practitioners and managers to analyze their quality assurance practices and determine where to initiate changes that will reduce and/or prevent causes of project failure. The rest of the chapter is organized around four main objectives: (1) Discuss the reasons for project failure, (2) Define software quality assurance, (3) Propose a software quality assurance process and discuss how it can be used to assure quality, and (4) Discuss implications for research and practice.

## REASONS FOR PROJECT MANAGEMENT FAILURES

The statistics on project management success are dismal and probably not surprising to most IT professionals. When a project fails it can jeopardize an organization's reputation with its customers and its name in the marketplace (Charette, 2005). In addition, project failures can be costly to organizations in terms of wasted dollars and resources. The bottom line is that projects continue to fail for many reasons.

It is generally accepted that software project failures predominantly occur due to cost or schedule overruns and quality control problems. Table 1 summarizes the most common reasons for project failure documented from published literature.

Many projects fail due to a combination of the factors listed in Table 1 (Charette, 2005). Complicating matters is the fact that there are a number of unique challenges related to software development itself, such as, technology, the development process, scope and complexity of projects, and risk of projects (Reel, 1999). However, since most causes of software failures are predictable and avoidable, premeditated planning and inclusion of proper controls and processes can result in diminishing of these concerns, resulting in project success. Project success requires carefully

17 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

[www.igi-global.com/chapter/software-quality-assurance/21637](http://www.igi-global.com/chapter/software-quality-assurance/21637)

## Related Content

---

### Issues of E-Learning in Third World Countries

Shantha Fernando (2009). *Encyclopedia of Information Science and Technology, Second Edition* (pp. 2273-2277).

[www.irma-international.org/chapter/issues-learning-third-world-countries/13898](http://www.irma-international.org/chapter/issues-learning-third-world-countries/13898)

### How to Disseminate Professional Knowledge in Healthcare: The Case of Skaraborg Hospital

Anne Persson, Janis Stirnaand Lena Aggestam (2008). *Journal of Cases on Information Technology* (pp. 41-64).

[www.irma-international.org/article/disseminate-professional-knowledge-healthcare/3233](http://www.irma-international.org/article/disseminate-professional-knowledge-healthcare/3233)

### Discrete Total Variation-Based Non-Local Means Filter for Denoising Magnetic Resonance Images

Nikita Joshi, Sarika Jainand Amit Agarwal (2020). *Journal of Information Technology Research* (pp. 14-31).

[www.irma-international.org/article/discrete-total-variation-based-non-local-means-filter-for-denoising-magnetic-resonance-images/264755](http://www.irma-international.org/article/discrete-total-variation-based-non-local-means-filter-for-denoising-magnetic-resonance-images/264755)

### Virtual Networking without a Backpack? Resource Consumption of Information Technologies

Justus von Geibler, Michael Kuhndtand Volkar Turk (2008). *Information Communication Technologies: Concepts, Methodologies, Tools, and Applications* (pp. 3500-3513).

[www.irma-international.org/chapter/virtual-networking-without-backpack-resource/22896](http://www.irma-international.org/chapter/virtual-networking-without-backpack-resource/22896)

### The Expert's Opinion

Mehdi Khosrow-Pour, D.B.A. (1989). *Information Resources Management Journal* (pp. 40-42).

[www.irma-international.org/article/expert-opinion/50923](http://www.irma-international.org/article/expert-opinion/50923)