The Impact of Regulatory Compliance on Agile Software Processes with a Focus on the FDA Guidelines for Medical Device Software

Hossein Mehrfard, Concordia University, Canada
Abdelwahab Hamou-Lhadj, Concordia University, Canada

ABSTRACT

The difficulty of complying with different regulations has become more evident as a large number of regulated businesses are mandated to follow an ever-increasing set of regulations. These regulations often drive significant changes in the way organizations operate to deliver value to their customers. This paper focuses on the impact of the Food and Drug Administration (FDA) regulations on agile software development processes, which in many ways can be considered as just another type of organizational processes. Particular focus is placed on the ability for Extreme Programming (XP) to support FDA requirements. Findings show that XP fails to meet many of the FDA guidelines for medical device software, which increases the risks of non-compliance for organizations that have adopted XP as their main software process. The results of this study can lead the work towards designing an extension to XP for FDA regulations.

Keywords: IT Compliance, Organizational Dynamics, Organizational Processes, Regulatory Compliance, Software Processes

INTRODUCTION

Recently, there has been a significant increase in attention to regulatory compliance and its impact on the way organizations are managed and controlled. This increase is driven by several factors including the recent corporate scandals such as the ones that involved some of the major U.S. organizations (e.g., Enron, WorldCom), the new challenges that Information Technology (IT) pose on protecting and securing sensitive information, and a higher need for business continuity in an ever-changing business world.

As a result, more regulations, laws, standards, and guidelines are introduced every year driving significant changes in the way companies are managed (Hamou-Lhadj & Hamou-Lhadj, 2007). These changes vary in...
scope and impact ranging from the introduction of new business processes to changes at the governance and strategic level. Hamou-Lhadj et al. characterize these changes in the form of a compliance support framework that can help effective handling of regulatory compliance requirements (Hamou-Lhadj & Hamou-Lhadj, 2007). The framework is composed of four main components: Governance, People, Process, and Technology. The aim of the governance component is to provide the strategic direction that will guide an effective delivery of end-to-end compliance support activities, while ensuring that these activities are aligned with the company’s vision and business objectives. The people component revolves around the proper selection, training, and retention of human potential that will operate the compliance support framework. The process component (the topic of this paper) is concerned with the need to adapt existing business processes (or creating new ones) for the handling of compliance requirements at the operational level. Finally, the technology component emphasizes the need for the proper tools and techniques in order to automate the delivery of compliance support activities.

In this paper, we particularly focus on the impact of regulatory compliance on the process by which software systems, used by regulated companies, are developed, maintained, and tested. Software processes can be seen as just another type of organizational processes since they are used by software companies to carry on the development of software products. As such, the paper has the broader objective of looking into the issue of how regulatory compliance impacts organizational processes used by software companies during product development.

More specifically, we target software systems used to control medical devices. These systems are subject to heavy regulations from government organizations to ensure that their design is carried out based on sound software engineering practices. One of the most predominant set of regulations in North-America that regulate the way software systems used to control medical devices should be developed is the Food and Drug Administration (FDA) regulations.

The FDA is a U.S. government agency that protects consumers by enforcing the U.S. Federal Food, Drug, and Cosmetic Act (FDA, 2009b). It regulates more than $1 trillion worth of consumer goods, about 25% of consumer expenditures in the U.S. (FDA, 2009b). The cost of not complying with FDA regulations can be considerably high, which makes its regulations some of the most important ones that should be on the priority list of a strategic compliance management initiative of any organization subject to the FDA laws.

The FDA also regulates the design and use of medical devices. There are several guidelines that have been issued by the FDA (2002) on how to monitor the manufacturing of safe and reliable medical devices. This also includes the software systems that control these medical devices. Due to complexity and criticality of medical devices, the FDA sets high demands on how to develop software for medical devices. Most of the FDA requirements are directly related to the process activities (e.g., requirement analysis, design, implementation, etc.) used by an organization to develop software. In addition, the FDA expects sufficient level of auditability within the software process itself. In other words, certain aspects of the development life cycle need to be tracked to allow external auditors to assess whether the system is FDA-compliant or not.

However, the requirements imposed by the FDA on the development process are very stringent, and may not be easily attainable. These requirements have been developed to overcome the difficulty of assessing the safety and reliability of software through traditional testing techniques. Additional verification and validation techniques as part of a broader and systematic process have to be applied. As such, The FDA requirements often translate into documenting and following specific guidelines to certify that the system is built, verified, and validated in a systematic manner and according to proven software engineering practices (FDA, 2002).
Related Content

Applying AHP for Collaborative Modeling Evaluation: Experiences from a Modeling Experiment
[www.igi-global.com/article/applying-ahp-collaborative-modeling-evaluation/75462?camid=4v1a](www.igi-global.com/article/applying-ahp-collaborative-modeling-evaluation/75462?camid=4v1a)

Design-Based Research with AGILE Sprints to Produce MUVES in Vocational Education
[www.igi-global.com/chapter/design-based-research-with-agile-sprints-to-produce-muves-in-vocational-education/188226?camid=4v1a](www.igi-global.com/chapter/design-based-research-with-agile-sprints-to-produce-muves-in-vocational-education/188226?camid=4v1a)
Reverse Engineering of Object-Oriented Code: An ADM Approach
[www.igi-global.com/chapter/reverse-engineering-of-object-oriented-code/117934?camid=4v1a](www.igi-global.com/chapter/reverse-engineering-of-object-oriented-code/117934?camid=4v1a)

A Comparative Study of Bayesian and Fuzzy Inference Approach to Assess Quality of the Software Using Activity-Based Quality Model
[www.igi-global.com/chapter/comparative-study-bayesian-fuzzy-inference/74876?camid=4v1a](www.igi-global.com/chapter/comparative-study-bayesian-fuzzy-inference/74876?camid=4v1a)