Chapter 13
QPLAN:
A Tool for Enhancing Software Development Project Performance with Customer Involvement

Marco Antônio Amaral Féris
University of Wollongong, Australia

ABSTRACT

As business competition increases, there is pressure on software development projects to become more productive and efficient. Previous research has shown that quality planning is a key factor in enhancing project performance. Thus, this article reports on the successful development and implementation of a tool (QPLAN) that enhances software development project performance by evaluating the planning quality of any type of software project and introducing best planning practices (such as references from historical data) that suggest how to manage projects in an appropriate manner, including encompassing lessons learned and involving the customer in the development process. This is applied research aimed at solving a real problem; thus, Design Science Research was adopted as the research methodology and the design science research process (DSRP) model was selected to conduct it. This artifact was designed for the project management literature, and implemented and validated in 11 organizations in five countries.

INTRODUCTION

Software organizations are taking over large slices of the economy from other sectors (Krishnan, Kriebel, Kekre, & Mukhopadhyay, 2000). For example, Google is the largest direct-marketing platform, while Netflix is the largest video service based on number of subscribers (Andreessen, 2011). In the automotive industry, cars have been launched on the market with software to control their engines and safety features, entertain passengers, and guide drivers to their destination. In the oil and gas industry, software has been used for the automation and control of operations that are essential for exploration and refining efforts. The defense industry has planes that do not require human pilots and missiles that achieve their targets guided by software. In some cases, software organizations have become leaders in traditional

DOI: 10.4018/978-1-5225-2084-9.ch013

industries—for example, Amazon is currently the world's largest bookseller. More than one decade ago, Borders sold its online business to Amazon because Borders believed that online book sales were unimportant (Andreessen, 2011).

Despite the significant influence of software around the world, the low performance of software development projects has plagued the IT industry for years (Krishnan et al., 2000). In 2000, only 28 percent of software projects were considered successful—that is, were completed on time and on budget and offered all features and functions as initially specified. However, 23 percent failed and, of the remaining fraction, the projects had higher costs than the original estimates, were completed behind schedule, or offered fewer features or functions than originally specified (Standish Group, 2013). For customers, unsuccessful projects may lead to a lack of productivity or loss of business, and the implications are equally problematic for organizations (Moløkken-Østvold & Jørgensen, 2005). In 2013, the results were slightly better; however, the success rate was still low, with only 39 percent of projects completed successfully (Stojanov, Dobrilovic, & Stojanov, 2013). Motivated by the significance of the software industry in the contemporary world and the frequently low performance of software development projects, the following research question was formulated to guide this research:

How can the planning quality effectiveness of software development projects be evaluated and improved?

BACKGROUND

Planning to Enhance Project Performance

As business competition grows increasingly difficult, there is pressure on software development projects to become more productive and efficient. Many factors affect the performance of software development projects, including a high level of complexity (Wohlin & Andrews, 2005), the level of project management knowledge, project managers' characteristics and level of technical expertise, the level of top management support, effective communication, enterprise environment factors, and the quality of the methods and tools used (Bechor, Neumann, Zviran, & Glezer, 2010). To complicate matters further, it is usually not obvious how all of these factors interact (Obiajunwa, 2012; Wohlin & Andrews, 2001).

To overcome these difficulties, researchers have sought to enhance project performance, with many researchers focusing on project planning, which is a critical phase of software development projects (Conforto & Amaral, 2010; IBM Staff, 2003; Sudhakar, 2012). Among other advantages, planning allows one to obtain a better understanding of the project requirements (Goldstein, Petrie, & Sherif, 2010; IBM Staff, 2003) and business context (Zwikael & Smyrk, 2011), reduce the inherent uncertainty of the project at this stage, and provide a basis for the next project phase (Zwikael, 2009).

Planning involves establishing a more formalized set of plans to ensure that the problem domain, architecture solution, requirements analyses, and project plans are sufficiently mature to conduct the project through the next phases and achieve the desired goals (IBM Staff, 2003). Among other activities, planning deals with the project's requirements (IBM Staff, 2003), time and cost estimations, identification of the critical path of a project (Dvir & Lechler, 2004), alternative solutions (Alblas & Wortmann, 2012; Bannerman, 2008), risks (Tesch, Kloppenborg, & Frolick, 2007), and risk mitigation (IBM Staff, 2003). This project phase is undertaken before the funder makes their major investment. Here, the level of effort steadily increases, while the level of uncertainty remains high, yet tends to decrease toward the