Chapter 83

Business Risk Analysis:
Obsolescence Management in Requirements Engineering

Jasbir Virdi
University of Toronto, UK

ABSTRACT

Up until now, efficient and applicable project methodologies, process improvement methodologies, product improvement methodologies, and relevant software engineering best practices have been developed and successfully implemented in many software projects in a variety of industries. However, projects still fail. Most of the project failure factors are diagnosed; however, some may fall through holes or gaps and get overlooked due to unknown reasons. The basis of this research study is a comprehensive questionnaire with relevant and probing questions that have collected data on obsolescence of requirements. Through analysis of the gathered data and information, this study aimed to present these risk factors, their impact, and a possible way to measure them. The critical factors identified and the result of this research project can be a trigger to conduct further in-depth analysis of project risks based on categories of projects (maintenance & support projects, development projects, data conversion projects, etc.), or rather, analysis of projects based on the business area/function. This research study is another attempt to assist in turning IT project failures into project successes.

INTRODUCTION

When we talk about software development projects, it is hard to ignore the requirements phase of it. For any software project to be successful, the requirements discovery, definition and documentation phase is very essential. However

software projects become very expensive when the requirements are not nailed down and firmed up at the beginning but are rather discovered during the later stages of the system development life cycle. Obsolete requirements, missing requirements, conflicting requirements, or ambiguous requirements, along with the factors than can cause requirements tin o be any of these

categories can pose risks to businesses especially if the software projects involve big budgets and expensive resources. Most businesses these days depend on and grow with large IT projects. Understanding requirements obsolescence can differ from business to business, project to project, and also from the stakeholders' point of view to the business users' perspective involved in a project. The major reason a requirement or requirements would become obsolete is due to change in business reason or business process. However there could be many other factors that can remain unnoticed until they pose serious negative implications to the project, in turn causing significant cost runs and immeasurable grief. Requirements engineering is the foremost and a pertinent phase of a properly managed software project. Clearly specified requirements at the business as well as the functional (detailed level) can help reduce the amount of rework at the development, testing or other later phases of the SDLC. Although it is usually the business analysts' obligation to gather and document requirements, however the project team usually plays or at least should play an active part in defining and validating requirements in regular meets with the stakeholders. It is a spiral process.

A vital activity in any software project after conception and planning is to gather, specify, and validate requirements; however this practice may be conducted differently across organizations. The objective of this research project is to survey, study, analyze, and report on how requirements obsolescence is understood in the IT community and what are its causes. The scope definition in this project is to bring awareness to requirements obsolescence as a term itself as well as identify and understand the factors that may cause requirements to become obsolete one way or the other. The term 'requirements obsolescence' will be discussed in context to the various aspects of project management in IT. The deliverable of this research project is to design, develop, and produce final results based on the analysis and findings

calculated from the survey results and face-to-face interviews which contain questions relevant to the research topic. It is not the objective of this research project to formalize a new or refine an existing requirements engineering framework in the IT industry, since various different paradigms and exhaustive models currently exist and are used quite widely in the market place.

PROBLEM STATEMENT

Meeting business needs, client requirements, or business requirements is the focal objective of any software project, however a significant number of projects fail to meet those requirements. The three main visible causes are summarized to be a change in budget, time, and/or scope of the project (Schwalbe, 2006).The importance of assessing the cause has actually to do with a deeper analysis of the failure. Software systems can become highly complex to build and developing complex systems must start with the right approach. The foremost step in undertaking a big IT venture is to account for obsolete requirements during the planning, analysis, and requirements elicitation phase so that a reduction if not total elimination of both financial as well as technical risks can be attained. If obsolete requirements are identified and addressed in the initial phases of the project, they will become less costly and more manageable in the later phases. The problem statement of this research project as a result of the above discussion is to understand what obsolescence is and how it can originate in context to IT projects. Data collected as part of this research will assist the investigation of requirements obsolescence in answering what, how, when, and why requirements become obsolete and the relevancy to the size of projects, project management methodologies implemented or otherwise, strength of audit controls and compliances and communication amongst project team members and stakeholders, to name a few.

Related Content

A Blur Classification Approach Using Deep Convolution Neural Network
Shamik Tiwari (2020). *International Journal of Information System Modeling and Design (pp. 93-111).*
www.irma-international.org/article/a-blur-classification-approach-using-deep-convolution-neural-network/250315

Intelligent Software Agents with Applications in Focus
Mario Jankovic-Romano, Milan Stankovicand Uroš Krcadinac (2009). *Software Applications: Concepts, Methodologies, Tools, and Applications (pp. 1426-1433).*
www.irma-international.org/chapter/intelligent-software-agents-applications-focus/29454

Adaptive Modeling and Dynamic Targeting for Real Time Analytics in Mobile Advertising
Donald Kridel, Dan Dolkand David Castillo (2017). *International Journal of Systems and Service-Oriented Engineering (pp. 24-39).*
www.irma-international.org/article/adaptive-modeling-and-dynamic-targeting-for-real-time-analytics-in-mobile-advertising/190411

Exploring Knowledge Engineering in Cognitive Skills Transfer for Small and Medium-Sized Companies Using Eye Tracking
Jun Nakamuraand Sanetake Nagayoshi (2022). *International Journal of Systems and Service-Oriented Engineering (pp. 1-16).*
www.irma-international.org/article/exploring-knowledge-engineering-in-cognitive-skills-transfer-for-small-and-medium-sized-companies-using-eye-tracking/297138

Validating the INTERPRETOR Software Architecture for the Interpretation of Large and Noisy Data Sets
Apkar Salatian (2013). *Integrated Models for Information Communication Systems and Networks: Design and Development (pp. 135-148).*
www.irma-international.org/chapter/validating-the-interpretor-software-architecture-for-the-interpretation-of-large-and-noisy-data-sets/79662