Chapter 7
Motivation behind Agile Software Development over Traditional Development

Karun Madan
SD College, India

ABSTRACT

Out of the many revolutions in software development methodologies over the past years, no new methodology withstands completely in one way or the other. The failure rate of projects is still very high in spite of so much of revolutionary methodologies come in existence. Unsuccessful projects not only mean the incomplete projects even after deadline or outdated projects but there may be several other scenarios like project did not meet up the real requirements or lack of ability to deal with the changing requirements etc. In this kind of circumstances, projects were never successfully utilized, and another high percentage of projects again required massive rework to be utilizable. Factors like changing requirements and late testing and integration are few of the main causes of this high percentage of failure.This paper reveals how agile development is a way out to the issues linked with traditional software development. Agile development primarily focuses on the rapid delivery of enterprise worth in the form of working software.

INTRODUCTION

Over the years, traditional software development proved to result in huge failure percentage in one or the other way. Most of the projects entirely fail, or may be of no use due to several reasons. Generally software are developed in waterfall-style methodology and this methodology is proved to be one of the prime contributing factor for almost all types of failures. Failures may be total failures or mostly software does not meet the actual needs in spite of the fact that they met on-paper specifications. So they are never successfully utilized. Even if one wants to re-tailor them, it requires extensive amount of rework to make them usable. Most of the time, re-tailoring decision goes wrong as compare to making a fresh start.

In contrast to traditional software development, agile development has shown promising results in the past few years. The Manifesto for Agile Software Development was agreed and signed by experienced

DOI: 10.4018/978-1-4666-8510-9.ch007

and recognized software development "gurus", inventors and practitioners (Cusumano et. al., 2003). This manifesto declares the main values of agile software development (Murauskaite et. al., 2008).

Agile software development is based on the methodology in which at first attempt, working software is given to the client. Client takes it as complete software and gives feedback to the team. This feedback may be some shortcomings or some new features or some change requirements. The new features were those which were put into notice at the time of requirements specifications or some change requirements. In Agile software development, development team takes these change requirements in a positive way even at late stages of development. In contrast to this, in traditional software development, team can only present documents at this stage as first attempt to test or run software would be at last stage. Actual working software is far more useful than just offering documents to product owners in the meetings.

Agile software development is termed as lightweight development methods as an answer to heavy-weight waterfall-based methodology. In Agile software development requirements need not be fully gathered at the beginning stages of the software development, rather continuous customer involvement is vital. Customer should be called in each and every meeting along with development team. As a result, customer satisfaction increases a lot by rapid delivery of useful working software. This rapid delivery increases customer's interest in development process. This results in daily cooperation between business clients and developer team.

AGILE PRINCIPLES

The Agile Manifesto is based on 12 principles (Beck et. al., 2001):

1. Customer satisfaction is the prime priority by means of rapid and continuous delivery of useful workable software.
2. Welcome changing requirements throughout the development process, even late in development to oblige customer's competitive arena
3. Working software is delivered frequently not only at the end, so weekly releases are preferred over releases after months of wait
4. Close collaboration among the business people and developers on the daily basis
5. Projects must be built around motivated individuals, Provide them the environment and support, more importantly show trust in them.
6. Face-to-face discussion is the excellent form of conveying information among the teammates, so try to assemble them together under one roof frequently
7. Working deployable software is the key measure of progress as compare to less effective documentation
8. Sustainable development must be supported, all the team members of agile software development ought to maintain a constant pace
9. Continuous attention to the technical brilliance and good design boosts the effect of agile software development.
10. Simplicity- the art of make the most of the quantity of work not done is necessary.
11. Self-organizing teams results in best architectures, requirements, and designs for the project.
12. Regular adaptation to ever changing situations, then modify its behavior accordingly at regular intervals.