The Relationship between Risk Management and the Success of Software Development Projects

Muhammad Muhsin

1. Introduction

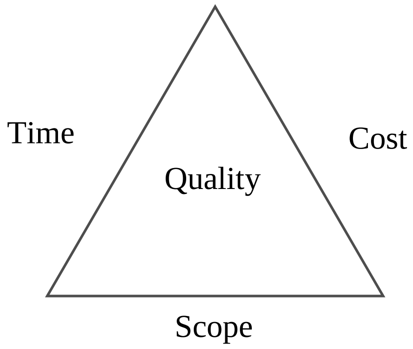
Software Projects are different from other types of projects due to certain characteristics (Hughes & Cotterell, 2009). One of the realities of this difference is that the risks associated with Software Projects are varied and novel. Therefore, software development projects require effective and careful Risk Management.

The context of this report is of a business that regularly undertakes software development projects. Recently, it has become evident that the business has seen some project failures and various reasons have been attributed to this. This report is a study of the reasons why Risk Management is a key factor to the success or failure of a software development project. The report concludes with a list of recommendations for practising Risk Management in the said organisation.

2. Risk Management

Risk is defined as the possibility of loss or injury (Merriam-Webster Online , n.d.). Project Risk refers to an uncertain event or condition that, if it occurs, has an effect on at least one project objective (Project Management Institute, Inc., 2008).

Relating to the project management triangle, project risks can be thought of as factors that cause one or more of the elements in the triangle to be compromised.



Accordingly, the factors that lead to project failure could be:

- failing to meet the schedule and deadline of the project
- not being able to keep the project under budget
- unable to fulfil the key scope of the project

which would eventually lead to a lower quality project.

Risk Management is about identifying these risks, assessing them and having strategies to minimise the risks so that there is minimum negative impact on the project (Wiley, et al., 2016). There is no such thing as a risk-free project, since the possibilities of compromise to a project are endless. Our goal here, therefore, is not to eliminate but rather effectively manage risks on a project. Risk Management is vital since it keeps one's attention on the factors which, when overlooked, could lead to an unfavourable outcome. It is something that needs to be conducted from the onset of any project and monitored throughout the stages of the project, while ensuring involvement from team members.

In the context of software development, Risk Management includes the tools, processes and methodologies employed to handle the risks in the Software Development Life Cycle (SDLC) process of a software project. It can also be defined as a project management tool which can measure and moderate the impact of events that might affect the project negatively. The purpose of this is to improve the probability of success. (Zardari, 2009)

3. Risk Management in relation to project success

Project managers, or any individual for that matter want their projects to succeed. There is no one in the world who wants to fail. Yet risks, in the context of project management, are things that could jeopardize the success of the project in varying levels. It is imperative that the project stakeholders actively perform Project Management. Thus saying 'Risk Management is the new Project Management' wouldn't necessarily be an overstatement.

When performed properly, risk management measures the possibility of success/failure of a project when we embark upon it. Moreover, it allows the project manager to gain insight on how to handle risks and unexpected outcomes well. For example, suppose a key member of the team has left his job or in a serious condition that he cannot be a part of the team anymore. This is a risk that can be managed by foreseeing this situation and not having all responsibilities tied to a single person. This is also known as 'mitigating risks'.

However, all risks are not negative! Suppose that a new hire is twice as good as expected and therefore finishes the tasks assigned in half the time (Roberts, 2016). These are termed as 'opportunities' and the organisation has to effectively manage this employee to overcome the 'risk' of underutilising this hire. Therefore, effective project Risk Management is two sided: preventing negative risks from becoming issues to the project and maximising the opportunities (also known as positive risks).

Developing and maintaining a software is risky. The vast majority of the running affairs of modern enterprises are controlled by software. Things like excess cost, delays and non-functioning software can have a greater impact. Moreover, it is a given that projects do not turn out to run as smoothly as we planned. All this means software Risk Management is vital to any project. It keeps the stakeholders of a project focused on the things that could lead to a disaster when overlooked.

4. How to manage risks

Risks are inevitable in any software development project. Taking a ‘positive approach’ to risks by ignoring them and hoping that they won’t turn out into issues (Mangrum, 2015) is not the wisest move. Rather, such an approach is naïve and could lead to project failure or at least largely hinder its success.

Risks Management which actively looks at the possibility of risks occurring, the type of risks and their impact before they take place is known as ‘proactive risk management’. However, the second approach, when organisations ‘respond’ to risks that have occurred by taking appropriate measures is termed as ‘reactive risk management’. (Zardari, 2009). While organisations want to perform proactive risk management, sometimes totally unexpected events take place, forcing them to opt to reactive risk management.

Many frameworks have been developed for dealing with risk management. They generally have similar characteristics; even if they don’t, the goal that is desired by all frameworks is the same – manage and mitigate project risks.

M Harding Roberts mentions the 5 M’s in his report: Measure, Minimise, Mention, Monitor, and Modify as a framework for Risk Management.

However, what follows is a brief summary of how to effectively manage risks broken down into 4 main parts: Identify, Analyse and Prioritise, Plan and Track. (Hughes & Cotterell, 2009)

1. Identify

   Risk Identification being the initial step aims to proactively consider risks before they turn out to be issues. This information is gathered and then turned into tangible risks which can then be descried and measured. This is done using techniques such as using checklists and brainstorming. Another method is using a standard format of risks statement, which contains the condition and the consequence.

2. Analyse and Prioritise

   The second stage is about examining the identified risks and assessing them to see which ones are significant and how they relate to each other. Analysing comprises of three main activates (Zardari, 2009). 1. Evaluating the characteristics of the risk to better understand them (by establishing values for Impact. Probability and Timeframe) 2. Classifying them and 3. Prioritising the risks by sorting through a large list of risks to identify which are important and which need to be dealt with first. Calculated values like Risk Exposure are used to get a score for the risks.

3. Plan

   The focus here is to be forward looking in order to avoid project risks from turning into issues. It’s about deciding what is to be done about a risk. Planning should be done effectively and high priority risks must be taken first. According to the book Software Project Management, risks can be dealt with any of the following ways: risk acceptance, risk avoidance, risk reduction, risk transfer and by taking risk mitigation/contingency measures.

Contingency planning is the developing of alternative methods for completing a project goal when a certain risk would jeopardize the successful completion of that goal. Suppose a certain member of the project team is sick and cannot make it to work. In this situation, other team members should be motivated enough so that at least the minimal requirements are fulfilled.

4. Track

   Once the risk plans are developed and scheduled, they are monitored with respect to their action plans. The motivation of risk tracking and monitoring is that the stakeholders of the project risks are aware of the status of the risks and the plans to mitigate them.

Lastly, it is essential that the team is continually learning from this process. This is done by documenting the risks and the process carried out as well as communicating it to the team.

5. What happens when Risk Management fails: a case study

J. P. Morgan Chase & Co. is one of world's leading and most powerful financial organisations. However, it faced a major blow losing billions of dollars in an incident in which it failed to actively perform Risk Management.

Microsoft Excel is a tool with remarkable functionalities which assists financial organisations. J. P. Morgan was no exception. However, they failed to consider the flaws of Excel. Once a formula is wrong, it becomes difficult to notice where the calculations are going wrong. The beginning of the issue was when J. P. Morgan developed a 'Synthetic Credit Value at Risk (VaR) Model' which would allow them to know about the risks they were exposed to and make more informed decisions.

The tool was developed by J. P. Morgan themselves using Excel spreadsheets. The major flaw that they made was relying on the copy-paste feature of Excel. J. P. Morgan, in their report following the major setback said, '*the model operated through a series of Excel spreadsheets, which had to be completed manually, by a process of copying and pasting data from one spreadsheet to another...*' This means that a minor mistake could case the data in the spreadsheets to change or delete the formula while the user in unaware of the situation.

In their report, J.P Morgan describes how the reason for this major failure was due to failures in the software project that developed the tool. The report states that the Chief Investment Office (CIO) Risk Management played '*a passive role in the model's development, approval, implementation and monitoring*'. It also mentions that CIO Risk Management personnel '*saw themselves as consumers of the model'* rather than taking responsibility for the development and operation of the software tool.

The International Project Leadership Academy is a project leadership school. It mentions that a key contributing factor to the failure of this project is a lack of Risk Management in deciding the best way to implement the new tool.
(International Project Leadership Academy , 2011)

6. Is Risk Management everything?

However much an organisation plans and manages risks, there is always an exception. For instance a bank could have the best security possible, spending a lot on investment of technology and people. However, there is no guarantee that a more-talented hacker with the motivation won’t be able to break it.

Brad Egeland shares a story of his past experience working for a ‘great organisation’. It had many great individuals and were working on amazing projects. However, the CEO of the same company turned out to be a fraud. A few of his friends who supported him with the fraud made sure they received fake loans from banks. The rest of the organisation didn’t know anything until it was too late. In this case, one person was responsible for the downfall of a whole organisation. Not one but every project in the business and hundreds of project customers. (Egeland, 2015) This case study illustrates that planning is important, yet there is always a limit of how far a planning and risk management can go.

Does this mean that organisations completely ignore Risk Management? No, rather it should encourage business to be even more prudent regarding risks and not get complacent with the status quo.

7. Recommendations

Based on studies from research in the context of my organisation in question, I propose the following recommendations:

- At the beginning of each project, thoroughly plan how the software development team is going to handle risks; how the team will approach risks and manage them during the project.
- Communicate more. Two way-communication is vital. Involve all members of staff responsible for the project throughout all the stages when identifying project risks. Don’t think you are ‘wasting their time’ by doing so.
- Experience is the best teacher. Review risks from previous projects and look at where we went wrong and how could that be prevented in future projects.
- Maintain a project Risk Register (Gary Hamilton, n.d.) on a regular basis. This is done by having a list of risks and moving them up and down based on priority.
- Incorporate training and make it a priority. Senior level staff ‘are always busy’, but taking some of their time (or having external trainers) to train staff at other levels is an investment that will pay off.
- Finally, I recommend that the organisation adapt Agile methodologies for their projects. The benefits of Agile are too many to enumerate here and thus I will suffice by saying this much.

8. Conclusion

The organisation would now realise that Risk Management is an effective and useful tool which must be incorporated within their software development projects. At the same time every single risk cannot be controlled and certain factors exceed the authority of the managers. Nevertheless, the methods of Risk Management are effective in identifying problems upfront. This has to be coupled with retrospection to see ‘what went right’ and what did not.

Risk Management is among the best controls for software development. It can provide huge benefits to the business that adopts it by cutting down costs and ensuring that projects are delivered on schedule. The business will definitely face risks that take them by surprise and impact their project. However, the more they actively practise Risk Management, the higher the chance they have to see a favourable outcome.

Reference list

Egeland, B., 2015. *The challenge of project risk management.* [Online]
Available at: http://www.cio.com/article/2925775/risk-management/security-is-a-myth-your-project-is-never-safe.html
[Accessed 26 January 2016].

Gary Hamilton, G. B. a. J. H. (., n.d. *Risk management and project management go hand in hand.* [Online]
Available at: http://www.cio.com.au/article/385084/risk_management_project_management_go_hand_hand/
[Accessed 26 January 2016].

Hughes, B. & Cotterell, M., 2009. *Software Project Management.* 5th ed. London: McGraw-Hill Education.

International Project Leadership Academy , 2011. *J.P. Morgan Chase & Co..* [Online]
Available at: http://calleam.com/WTPF/?p=5517
[Accessed 27 January 2016].

Mangrum, J., 2015. *CIO Insights: Managing Project Risks and Issues.* [Online]
Available at: http://www.techwire.net/news/cio-insights-managing-project-risks-and-issues.html
[Accessed 26 January 2016].

Merriam-Webster Online , n.d. *risk.* [Online]
Available at: http://www.merriam-webster.com/dictionary/Risk
[Accessed 21 January 2016].

Project Management Institute, Inc., 2008. *A Guide to the Project Management Body of Knowledge (PMBOK Guide).* 4th ed. Newtown Square, PA: Newtown Square, PA:.

Roberts, M. H., 2016. *Project Management Book.* 5th ed. s.l.:Project Management Book.

Wiley, et al., 2016. *Project Management for Instructional Designers.* Pressbooks: Wiley, et al..

Zardari, S., 2009. *Software Risk Management.* Kuala Lumpur , IEEE.