Your Vision - Brought to Life.

White Paper

Managing Information Technology Risk

Disclaimer

Although Informulate LLC takes great care to ensure the accuracy and quality of these materials, all material is provided without any warranty whatsoever, including, but not limited to, the implied warranties of merchantability or fitness for a particular purpose.

Trademark Notices: All product names and services identified throughout this document are trademarks or registered trademarks of their respective companies. No use of any such trade names in this document is intended to convey endorsement by Informulate or affiliation with Informulate.

TABLE OF CONTENTS

Introduction 4
Information Technology (IT) in Business 4
Unaddressed Information Technology needs 4
Why do Project Implementations fail? 4
Knowing where real risks lie 4
Symptoms of Project Failure and Root Causes 6
Cost overruns 6
Scope change 6
Timeline slippage 7
Product functionality that does not meet user expectations 7
The "Perfect Storm" 8
Risk Insurance through Informulate 9
Having a well-defined standard and proven process 9
Looking for help beyond the local area 9
Leveraging the power of globalization 10
Having an iterative development approach 10
Conclusion 11
Feedback 11

Introduction

This white paper is intended as a guide for understanding and mitigating project risk. The intended audience is small to mid-sized businesses, project managers, or business leaders planning to fund an information technology project.

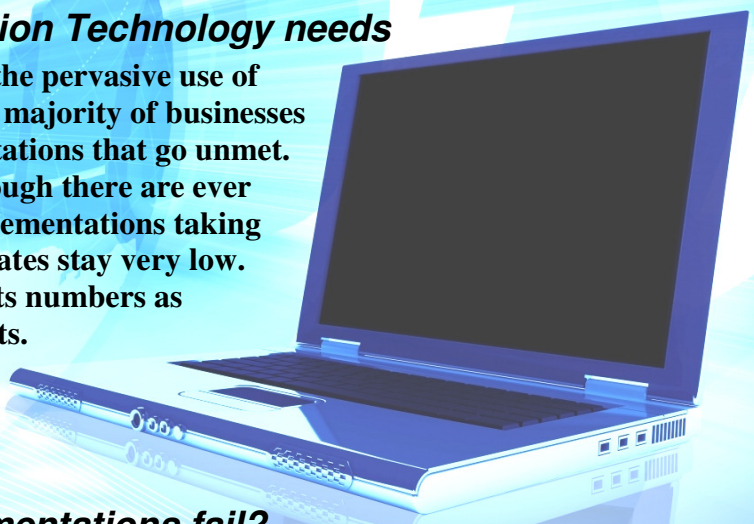
It also provides insight into how Informulate mitigates risk in its projects.

Information Technology (IT) in Business

All businesses use the power of software in some form or another. Whether as basic word processing or as a database, whether as competitive tool or as infrastructure, from marketing to accounting, from tracking inventory to assisting in decision-making, it is impossible to imagine a business that does not depend on IT.

Unaddressed Information Technology needs

Research shows that despite the pervasive use of computers in business, a vast majority of businesses still have IT needs and expectations that go unmet. Statistics also show that although there are ever greater software project implementations taking place each year, the success rates stay very low. InformationWeek.com reports numbers as high as 70% for failed projects.



Why do Project Implementations fail?

These high failure rates beg the question, are we asking too much of custom software development? What are the causes of project failures? How can a business leader hedge her technology bets to minimize risk while maximizing value?

Knowing where real risks lie

As most experienced project campaigners know, project risks are everywhere. Teams engaged in software development (including both clients and vendors) when interviewed before, during, and after implementations reveal many different reasons for project failure. Reasons such as technical limitations, code quality, testing, lack of communication, lack of vision, changing requirements, etc.

At Informulate, we recognize the reality of the software projects and vendor engagements. In our experience with small and mid-sized business, the risk of a project is rarely in the technical aspect. It lies largely in project management and

business analysis. We think it is time these critical activities are given their due especially vis-à-vis the role of coding. So yes, an installed application is the final deliverable of a software project. And though it comprises mainly of custom code, that code is only as good as the requirements and success parameters defined for it.

So what do we do to mitigate these risks? Apart from using well-defined processes (Scrums, Iterative development), and standard methodologies (Unit testing, automated testing) we believe it is critical that a strong trust relationship exists with our clients. Our PMs are trained to quickly become specialists on the client's business. Understanding why clients are making the decisions they are is key to aligning project direction with client vision. When you talk to our PMs, you will soon find that you can talk your business with them and leave it to them to figure out the technical aspects. Most of our clients are relieved to note that they do not need to become technical overnight or hire an intermediary or expert to initiate technical projects.

Symptoms of Project Failure and Root Causes

So what causes failures? First, let us look at a partial list of symptoms or shared attributes that we can use to identify failure.

Cost overruns

Cost overruns can be defined as project costs exceeding budget by over 30%. It is a known fact that making estimations on a piece of work that is being done for the first time (which is the definition of custom software development) is not an exact science.

Here are some reasons that commonly show up in post-implementation reviews:

- Improper estimating with insufficient buffer,
- Bad management of resources,
- External dependencies, etc.

The Informulate View: The issues mentioned above and others will certainly affect projects but might we suggest that perhaps one overlooked reason for cost overruns are… costs. Given the lack of supply of skilled computer professionals at inexpensive rates it is only natural that the afore-mentioned problems tend to end up being project-breaking, instead of simply presenting manageable challenges. With Informulate, you have the ability to quickly add or augment onshore resources with offshore or swap as needed based on whether the need is for speed/agility or for costs/long term maintenance.

Scope change

Also referred to as "Scope Creep", this is defined as unpredictable and unplanned change in the functional requirements of a product during the course of a project implementation. Typically this refers to sizeable change in requirements *after* the requirements phase has officially completed.

Root causes for this include:

- Inadequate time planned for Business Analysis (BA)
- Arbitrary time-boxing of the requirements gathering phase
- Not making the business experts/users available at the early stages
- Not setting business user expectations on their role in the project

The Informulate View: There is no substitute for up-front high level requirements gathering, scoping and giving this phase the importance it deserves. BA work also provides valuable artifacts that show the business process and can be re-used for process improvement or future product enhancement. A telling metric of the importance of this phase is that errors made in this phase will get more and

more expensive to fix as each phase is kicked off until finally when the product is close to production, even minor errors become project-breaking.

Timeline slippage

We can define this in quantitative terms as delays 30% greater than the original planned deadline. This is probably the most externally visible and biggest project-crushing symptom of failed projects.

Root causes for this include:

- Bad estimations
- Unplanned outages
- Bad resource planning
- Scope creep
- Unavailability of backup resources
- Under-skilled resources

Informulate view: Timeline slippages are such a generic symptom that they can be caused by any number of root causes. Unfortunately though, the high hourly rates of local resources make it very difficult for businesses to throw additional resources at the problem in order to catch up. Also, lack of experience causes many teams to attempt to encompass the entire product functionality in one release rather than manageable iterations. This results in the final product being painted into an all-or-nothing corner, thus setting it up for failure.

Product functionality that does not meet user expectations

When a product fails to meet user expectations by such a wide margin that substantial rework is required, it falls into this category. This event is usually quickly followed by lack of trust between the users and the project team and its all downhill from there.

Root causes for this include:

- Bad requirements gathering or specifications
- Bad testing or quality control of features
- Insufficient involvement of users in the initial phases
- Under-skilled programmers who were unable to code to specifications

Informulate View: Product “dysfunctionality” is probably one of the easiest problems to avoid by ensuring early and continuous user involvement in the project. Iterative development is a byword in modern software development and it

has been proved to be very effective. Coupled with good business analysis and clear specifications this is an avoidable failure.

The "Perfect Storm"

The worst case scenario for any business sponsor of an IT project would be a lethal combination of the afore-mentioned causes. Typically a failed implementation will exhibit more than one of the symptoms discussed. Often while in "fire-fighting" mode, project teams will fail to see the root causes that led them to such a visible symptom.

In addition to root causes mentioned, here are some more:

- Unrealistic expectations of project team
- Ad hoc project management that does not set clear targets
- Putting people in the wrong roles
- Taking short-cuts on critical activities
- Insufficient or inefficient communication between project members
- Users that "do not know what they want"
- Resource leakage or turnover

Informulate view: There are many risks to a project's well being and certainly a lot of these play into the "human factor" of software development. In an ideal world, a project team would have a single highly experienced project manager, an indefinitely scaleable team of inexpensive coders that can be quickly assembled or disassembled, an experienced team of business analysts, testers, designers who are highly motivated to get the job done and a support team to care for the project once in production…

But should this be considered unattainable? At Informulate, we think not.

Risk Insurance through Informulate

We believe that a lot of these problems and the risks they present can be mitigated.

As business leaders know, managing risk is never about simply avoiding risky endeavor (or we would be questioning the risk-return model which is the very foundation of capitalism).

No, we need to work towards the project objective/s while having a framework to deal with disruptions and issues as they come up. This dynamic feedback loop of predicting/identifying and mitigating risk in an ongoing project is made much easier when you have certain key competitive advantages as listed below.

Having a well-defined standard and proven process

A clear process provides teams a guideline on which to base their activities and allows a quick ramp up for new team members. Also, it places emphasis on the early and critical phases such as requirements gathering and design before arriving at the estimation stage.

The Informulate Application Development Methodology or other standard methodologies ensure that due importance is given to the various stages of the software development lifecycle. For more information, please refer to http://www.informulate.net/docs/WhitePaper-IADM.pdf

Looking for help beyond the local area

Since finding talented, local people is a problem for most businesses, they can open up the search area for suitable candidates by allowing them to work remotely. Given the huge strides in security services such as VPN, Remote Desktop and such technologies, this is not risky proposition anymore. The cost-benefit analysis of this is a no-brainer when teams realize that you need less infrastructure when teams work remotely and also studies have shown that

remote working employees are less stressed and more productive as long as the goals are clearly defined. However, there is a definite learning curve with remote work and requires experience to facilitate and manage efficiently.

Informulate leverages remote teams dispersed across the U.S. in California, Florida, Oregon, and Virginia. This minimizes infrastructure costs and allows lower hourly rates that are passed on to clients. Simultaneously, the flexibility that this option provides team members in scheduling their work around their own lives provides a powerful incentive that leads to motivated work.

Leveraging the power of globalization

The prevailing wages for highly qualified candidates in lesser developed countries are very low compared to local wages. Until recently these resources could only be engaged by getting them to enter the country using a limited "work visa" which also meant that the labor cost differential was lost or to send work overseas to offshore companies.

Informulate leverages remotely working teams and the "labor arbitrage" that exists in pay scale differentials, while engaging local project managers. This provides a best-of-both-worlds option to businesses that did not exist previously. Businesses can now work with a local company with a global presence. Thus they benefit from low costs while getting high quality work, all while maintaining a direct line of communication with a local project manager who encapsulates the development process from the client. Risk factors are further mitigated by Informulate's ability to quickly access a large pool of onshore, near-shore and offshore resources.

Having an iterative development approach

In order to prevent product development from going off track, it is best to put many and frequent revisions of the product in front of stakeholders so that issues are identified early and so have a better chance of being rectified. This has been popularized through various standards such as the RUP, Agile etc.

Informulate employs an iterative approach to software development as described in our Informulate Application Development Methodology or IADM. For more information, please refer to http://www.informulate.net/docs/IADM.pdf

Conclusion

Application development and managing information technology projects is a challenging task at the best of times. This paper provides an overview of common risk factors that teams must be aware of and plan to mitigate. However, project teams still need to do due diligence by analyzing their own risk factors apart from the ones mentioned in this paper.

Informulate brings a high level of expertise to its engagements that reduces risk for our clients by allowing flexibility and scalability within the team, while delivering high quality at low cost.

Feedback

If you have feedback on this White Paper or on Informulate in general, please contact us at clients@informulate.net or visit our website at http://www.informulate.net for more details.