

TALES OF (GOOD TEAMS') FAILURES



Case Studies, Root Causes and Recommendations.



About the article

This article is a collection of short case studies where teams failed to meet the expected results. The underlying companies were of different sizes, culture and industries. The only thing common across the board was the quality of people – all teams had bright individuals that had a track record of success.

About the Author

Rajeev Kumar is a senior partner at Mirketa Inc. Rajeev specializes in managing large business and IT programs, lean processes and governance. He has over 18 years of experience working in executive, middle management and individual contributor roles at startups and fortune 500 companies from different industries and countries. Prior to Mirketa, Rajeev founded 2 startups in financial planning and event management space.

UNPREPARED

BACKGROUND

- ▶ A successful pre-IPO company bought cloud based solution to manage a process. The company was already using the best in class cloud solutions for Sales, Services and Accounting.
- ▶ The chosen product was not a good fit. The company had to make significant changes to their process and overall business model to align with the capabilities of the new software.
- ▶ The company couldn't keep their market commitments for the new product/services launch. A lot of customization to manage integration as out of the box integration didn't help.
- ▶ Ended up making significant customization, as out of the box integration didn't help.
- ▶ The solution is partially integrated and team is looking for a different long term solutions.
- ▶ The integration consultants played waiting game on company's money while major business and architectural decisions were being made.



What went wrong?

- ▶ Started the project pre-maturely.
- ▶ The product team didn't define their core needs upfront and did not evaluate the tools' capabilities with respect to their needs before signing up for the tool.
- ▶ Made external commitments for new product releases without knowing what they were getting into.



How to avoid it?

- ▶ Identify the core business needs and see if a cloud solution makes sense before committing to one.
- ▶ Factor in the time for integration and customization as most cloud solutions require that. Make commitments after going through some level of release planning.
- ▶ Have a solution architect who can lay out high level business and technology solution before getting the full delivery team onboard.

BLIND LEADING THE BLIND

BACKGROUND

- ▶ In late 90s a large company started a project to transition from legacy systems to a web based solution.
- ▶ Outsourced their product vision, strategy and the development of the new system to a technology strategy company. The strategy company in turn hired a large offshore development vendor to develop the application.
- ▶ Actual time and cost were order of magnitude higher than the estimates.
- ▶ The new product had design flaws that created serious performance and scalability issues. Some of the critical functionality could not be deployed due to performance issues.
- ▶ The product had some serious usability issues.



What went wrong?

- ▶ Team members from the strategy firm had very little experience developing high performance web systems. Folks developing the vision and requirements had no development background. There were no feasibility assessments.
- ▶ The program team had missing roles from the user experience discipline. We did not have anyone who specialized in designing high touch web experience. The product was good looking but was hard to use.
- ▶ The offshore vendors had developers that were good at coding but had limited experience with architecture and design. Bad architecture and design drove project costs, delays and performance issues with the systems
- ▶ The client leadership did not have experience with large web development programs. The program vision, management and the design leadership was 'outsourced' to the strategy firm which in turn outsourced the decisions to their technology partner. It was a classic case of 'a blind leading the other blinds'.
- ▶ Project team spent about 8-10 months writing requirements and creating design documentation.



How to avoid it?

- ▶ Team up product manager with seasoned technical lead, architect and an interaction designer. Check feasibility and usability of all product modules before passing it on to the developers.
- ▶ Hire FTEs with good hands on experience with the technology for development and delivery lead roles. Have them create/review architecture, design specs and the code from the vendor teams.
- ▶ Hire a seasoned program manager as FTE to manage the overall program and hire FTEs or independent contractors to manage different project teams. This puts the ownership of the program and accountability for success onto the employees who have vested interest in the project.
- ▶ Screen the key team members from vendors' teams
- ▶ Follow Agile development practices.

OVER-ENGINEERED SOLUTIONS

BACKGROUND

- ▶ Two very large companies tried rolling out Rational Unified Process (RUP) as defined by the book
- ▶ Pretty much everything went wrong – productivity went down for a long period of time and the teams were utterly confused and frustrated with the process as most people found it way too difficult and/or cumbersome.
- ▶ There was visible hostility and/or strained relationships among the team members from different disciplines – developers always blamed on requirements, BA blamed product managers & product managers, testers blamed everyone for humongous documentation needs from every project etc.



What went wrong?

- ▶ RUP for software development was/is a classic example of an over engineered process. It over prescribes the details on what needs to be done by who and completely undermines the human piece for the process.
- ▶ Ill-Equipped Drivers/Sponsors - These companies committed to a process without knowing how it will solve their problems and what problems it will create. There was no process expertise from the owners and drivers of the change. Instead of adapting to the process to the organization, the approach was to change the organization to make the process work (which by the way is way too complex for any team)
- ▶ The change management focused only on skill upgrade and did not include changes to organizational structure, culture and values.



How to avoid it?

- ▶ Don't go crazy on RUP. :)
- ▶ Have a clear understanding of the problems and how the new process will solve it.
- ▶ Hire change owners/drivers with strong knowledge of the new process and organizational change management. Don't give a machine gun to a monkey (you get the point)
- ▶ Have a very good change plan that has three dimensions – people, process and tools, and very clearly defined metrics. BTW, it helps to identify the folks who will never change. Instead of trying to change the last 20% who won't change, give them options where they don't have to be a part of the change (worst case, manage them out)

NOT A LEAN START-UP

BACKGROUND

- ▶ Company: myTafi.com – This was my first attempt to build a product company. MyTafi.com is a personal financial planning application designed to help people make better financial decisions.
- ▶ Developed great set of product features but could not drive user adoption and could not convert strong interest from large banks and investment management firms into sales.



What went wrong?

- ▶ We were internally focused. We did not systematically collect 'real user' feedback that could have helped us with usability and product marketing. Instead, we relied on the feedback from friends and family (which was always good by the way).
- ▶ We had a very good development plan and we executed well on that. However, our sales and marketing plans were weak and we our execution of those plans was even weaker.
- ▶ On the user interface, we focused mostly on the visual design and missed the mark on interaction design and overall content which are critical for better usability.



How to avoid it?

- ▶ Get product validation through paper prototypes or light weight prototypes.
- ▶ Focus on interaction design and content/messaging. Hire experts from the beginning.
- ▶ Find a partner(s) who specialize(s) in sales and marketing. You got to have a detailed plan and flawless execution.
- ▶ Focus on four things – 1) Concept 2) Usability 3) Sales and marketing 4) Delivery

TOO MANY "TOP" PRIORITIES

BACKGROUND

- ▶ I have seen many good teams struggle with too many top priorities or shifting priorities. The examples that I have in mind are from the times where the top leaders were exceptionally sharp and ambitious.
- ▶ Teams worked on too many things at the same time and couldn't complete anything with exceptional quality.
- ▶ Highly dissatisfied internal customers/business partners.
- ▶ Top leaders always talked about fixing priorities but their actions never showed it
- ▶ Team members were mostly tired, frustrated and intolerant.



What went wrong?

- ▶ Management team's inability of say "No" to wider stakeholders and/or sort out priorities; most requests got added as 'Must do now'
- ▶ Lack of clear prioritization/portfolio management process/function
- ▶ Prioritization was left onto middle management and/or the individuals who had no control and /or experience
- ▶ Delivery/execution plans/commitments disregarded the organizational sustainable velocity



How to avoid it?

- ▶ Have a clear and transparent prioritization and portfolio management function that takes organizational constraints into account in planning
- ▶ Show priorities through actions; ***"Action expresses priorities"*** - Gandhi ;
- ▶ Say 'No' to what can't be done with a sustainable pace; ***"The key is not to prioritize what's on your schedule, but to schedule your priorities."***
- Stephen Covey;

INTERESTED IN LEARNING MORE?

Contact Rajeev Kumar to learn more about these case studies and how Mirketa Inc can partner with you to design and rollout business solutions.



www.mirketa.com



[info \(at\) mirketa.com](mailto:info@mirketa.com)



1-855-Mirketa

