



Why do large and complex IT and digital transformations fail?

A framework for managing success and failure factors



midagon

Contents

Background	1
PART 1: SUCCESS AND FAILURE FACTORS	2
Critical success and failure factors	2
PART 2: IMPACT OF DEVELOPMENT METHODOLOGY	3
Is the development methodology the key to project success?	3
The promises of agile and scaled agile	4
The promises of Continuous Delivery	6
lessons learned from agile and continuous development	7
PART 3: FRAMEWORK FOR SUCCESS	9
Understanding the success and failure factors	9
Embedding success factors into methodology	10
Project maturity assessment with critical factors approach	12
Conclusions	13
References	14

Background

Large-scale IT and digital transformation projects have a high likelihood of failure. According to (Shaul & Tauber 2013):

- 90% of ERP implementations are delivered late or are over budget,
- Enterprise initiatives show a 67% fail rate in achieving corporate goals and are considered negative or unsuccessful,
- More than 40% of all large-scale projects fail.

This is indicative of a real issue that ultimately influences the vitality of organisations. To improve this situation, Midagon wanted to get an objective understanding of the reasons underlying success and failure and to find ways to increase project success rates. We requested Professor **Matti Rossi** from Aalto University School of Business and Professor **Kari Smolander** from Lappeenranta University of Technology to study the success and failure factors of complex digital transformations. As part of the study, we also wanted to understand if new methodologies (namely agile) have changed the success rates and if they ensure the success of digital transformations.

Based on the findings from academic research, we introduce a Midagon framework to systematically address the success and failure factors in IT and digital transformations. This framework can be applied in IT and digital transformations, independent of the methodology used.

These issues are covered in three parts:

1. The first part looks at large-scale system projects through academic studies. The goal is to identify critical success factors, pitfalls and why so many transformation projects fail. As a result, success and failure factors are identified.
2. The second part examines the impact of project methodologies. We study how new implementation approaches, especially agile methods and continuous delivery, can limit the risks of these endeavours. This section is based on academic research and Midagon's experience.
3. The third part introduces a Midagon framework through which:
 - The existence and maturity level of success and failure factors can be assessed and
 - Success and failure factors can be systematically addressed throughout the project lifecycle

PART 1: SUCCESS AND FAILURE FACTORS

CRITICAL SUCCESS AND FAILURE FACTORS

What does the academic research state about factors increasing the likelihood of success or failure? The following table presents a slightly condensed list based on the studies of Somers and Nelson, 2001 and Wong et al., 2005. The lists have been very stable over the years. More recent studies and articles have identified the very same themes around critical failure factors;

for example, an article about lessons learned from failed projects (Mishra, 2017), a collection of project management statistics (Bonnie, 2018) and an article about mistakes to avoid in core banking systems transformation (Frieze, Kube, Schöbl and Schöller, 2022). Therefore, these success and failure factors can be seen as definitive.

Critical success factors (CSF)	Critical failure factors (CFF)
1. Top management support and CEO sponsorship	1. System misfit
2. Project team competence	2. High turnover rate of project team members
3. Inter-departmental cooperation and open communication	3. Over-reliance on heavy customization
4. Clear goals and objectives	4. Poor consultant effectiveness
5. Project management	5. Poor IT infrastructure
6. Management of expectations	6. Poor knowledge transfer
7. Vendors' support	7. Poor project management effectiveness
8. Data quality and conversion	8. Poor quality of Business Process Re-engineering (BPR)
9. User training on software and education on new business processes	9. Poor quality of testing
10. Dedicated resources	10. Poor top management support
11. Enough time and resources for business process redesign	11. Too tight project schedule
12. Minimal customisation	12. Unclear concept of the nature and use of the system from the users' perspective
13. Change management	13. Unrealistic expectations from top management concerning the System
14. Use of third-party consultants	14. Users' resistance to change
15. Working partnership with vendors and consultants	

Table 1: Success and failure factors. These lists are compiled based on a large number of empirical studies of success factors and are presented in order of importance.

The well-known issue related to these Critical Success Factors lists (later CSFs) is that not following them almost certainly leads to failure, whereas following them will not guarantee success. Having all the CSFs in place means that there are good reasons to believe in successful delivery. However, even with all the CSFs in place, the likelihood of delivering on time and budget is still far less than 50 %.

There are several reasons for failure. The comprehensive review by Shaul and Tauber (2013) provides the following key reasons for the high failure rate.

- First, systems are often **promoted as a unified platform for all business processes**. Since the system involves a large portion of the organisation, companies can experience difficulties convincing employees to commit to the implementation process, who then fail to implement the system effectively (Davenport, 2000).
- Second, many organisations start complex projects, even though near-term success and long-term survival are difficult to predict.
- Third, most enterprise systems follow best practices, such as a maximum integration of information flows and standardisation, that are **less suitable for firms with decentralised, non-hierarchical structures and non-uniform cultures** (Mattila et al., 2017).
- Fourth, organisations increasingly find they must **accept the project outcomes that emerge from compromises** between an installed consultancy base or software vendor solutions and the local context (Wagner & Newell, 2004).
- Fifth, **legacy systems and shadow systems are still being used**. (Tanriverdi et al., 2007; Mattila et al., 2017).

PART 2: IMPACT OF DEVELOPMENT METHODOLOGY

CRITICAL SUCCESS AND FAILURE FACTORS

Methodologies are an integral part of the project setup because all companies using project methodologies expect greatly improved project performance (Joslin et al., 2014). In the following section, we review new development and project methodologies and evaluate whether selecting the right methodology alone is enough to eliminate critical failure factors and establish critical success factors.

Traditional waterfall approaches, which have been widely used for decades, have become a target for criticism. The long planning and development cycles attributed to these approaches are not able to predict and consider the changes coming from the fast-changing and chaotic business environment. High customisation is often a consequence of unexpected changes in the environment.

If an IT project takes two to three years from the planning table to the first go-live, the business environment will have changed, and the result will most likely no longer fully or sufficiently match the needs. The development and implementation process is also a significant learning process, where the involved stakeholders learn and start understanding the potential and the possibilities of the new solution, as opposed to having enough knowledge upfront and being able to control all of the variables in advance.

Over the past twenty years, new approaches have emerged in managing development and development projects. These methods are especially rooted in software development, but they have also been adopted more widely in IT system implementation projects and environments. The methods referred to in this white paper primarily include Agile / SCRUM, Scaled Agile Framework (SAFe), and continuous delivery or DevOps.

THE PROMISES OF AGILE AND SCALED AGILE

The emergence of agile methodologies is, without a doubt, the most important change in project management practices in recent decades. They are influenced by lean manufacturing in Japan but have since been developed independently. However, elements such as Kanban boards, with their origin in lean manufacturing, are widely used by agile teams.

Agile practices emphasise direct interaction and co-located development teams to solve the immediate problem of software developers working together. In the modern, fast-changing business environment, responding to change should be valued more than following a pre-defined plan until the bitter end. The customer is seen as an integral part of the development team. The benefits are obvious, and therefore, agile methods make a significant contribution to answering how development teams should work and reflect together. However, in large, complex, and geographically distributed projects, the essential problems may be totally different from the idea of agile.

To address the problem of project and organisation size, complexity, and geographical distribution, during the past ten years, much of the progress in agile development has occurred in distributed and large-scale contexts. Scaled Agile (SAFe) is perhaps the most prominent agile methodology to address these issues. According to Scaled Agile Inc. (2018) (<https://www.scaledagileframework.com/why-safe/>), the

business benefits of SAFe are improved quality, employee engagement, productivity, faster time to market and program execution. Indeed, SAFe, and other similar methods aim at bridging the gaps between how one co-located development team operates and how to operate agile on an organisational level.

Whether agile works or not is difficult to study. How much of the success can be attributed to agile and how much to other factors involved? Large quantities of detailed and correct project data across industries are difficult to gather. The development context is often extremely complex, which makes projects incomparable. For example, Clarke and O'Connor (2012) identify from the literature 48 essential factors that can influence project success.

Despite the difficulty, some attempts have been made to evaluate the effects of agile on project success. Serrador and Pinto (2015) tested the effect of agile use in more than 1,000 projects on two dimensions of project success: efficiency and overall stakeholder satisfaction against organisational goals. The findings suggest that agile has a positive impact on efficiency, stakeholder satisfaction, and perception of overall project performance. On the other hand, in the study by Budzier and Flyvbjerg (2013), agile methods seemed to reduce schedule risk but not the other risks, such as benefit or cost risk.

Standish Group International (2015) compared more than 10,000 software projects from 2011–2015, segmented by the agile process and waterfall method. The results for all projects showed that agile projects have almost four times the success rate as waterfall projects, and waterfall projects have three times the failure rate as agile projects.

The results are also broken down by project size: large, medium, and small. The results showed that waterfall projects do not scale well, while agile projects scale better. However, they also noted that the smaller the project, the smaller the difference is between the agile and the waterfall process.

CHAOS RESOLUTION BY AGILE VERSUS WATERFALL

SIZE	METHOD	SUCCESSFUL	CHALLENGED	FAILED
All Size Projects	Agile	39%	52%	9%
	Waterfall	11%	60%	29%
Large Size Projects	Agile	18%	59%	23%
	Waterfall	3%	55%	42%
Medium Size Projects	Agile	27%	62%	11%
	Waterfall	7%	68%	25%
Small Size Projects	Agile	58%	38%	4%
	Waterfall	44%	45%	11%

The resolution of all software projects from FY2011-2015 within the new CHAOS database, segmented by the agile process and waterfall method. The total number of software projects is over 10,000.

Table 2: Comparison of success between different methods in different size projects. Source: Standish Group International Inc, CHAOS Report 2015.

Some studies have found agile approaches to be positive for health and well-being at work (e.g., Syed-Abdullah et al., 2006) through enthusiasm and autonomy. Other studies have found that agile increases stress in the workplace (Laanti, 2013). Laanti explains that it is management’s responsibility to create a working environment suitable for agile. If that does not happen, increased stress will result.

Moving from waterfall to agile itself is a major transformation. The challenges of such a transformation should not be overlooked. Since organisations, their practices and cultures are quite different, there is no single recipe for implementing agile transformation successfully. Agile and large-scale agile methodologies are frameworks. Each organisation must design how to apply it in their context and how to transition from the old way of working

to the new. A stepwise approach to transition from waterfall to agile is possible and often recommended.

Dikert et al. (2016) point out that in a large organisation, contexts beyond development must be involved in the transformation. Functions ranging from marketing and sales to human resources must also align with agile. If this does not happen, it might cause trouble for the transformation. To enable successful large-scale agile deliveries (for example, by using SAFe), a major top-down agile transformation must occur on the enterprise level – from top management sponsorship to individual development teams. A common failure factor for agile deliveries is the mismatch between organisational culture and delivery methods. If the underlying structures are based on waterfall processes, the odds of efficiently leveraging the benefits of agile methods are limited.

In many organisations that use waterfall, development teams are used to doing several projects simultaneously with partial allocations to each project. When moving to agile, these resourcing principles need to be fundamentally changed. Agile methods should provide more flexibility for parallel development. However, the principles for work allocation and resourcing can differ greatly from the waterfall deliveries. Agile teams and release trains are typically specialised around certain competence areas and, therefore, a major transformation may require a comprehensive reallocation of ongoing tasks and projects.

THE PROMISES OF CONTINUOUS DELIVERY

While agile development is iterative and gradual, modern software applications and information systems are also expected to be continuously developed, enhanced, and delivered. continuous delivery (or DevOps) is an extension of agile, with short iterations and emphasised interaction. However, the more profound difference is that continuous delivery is a change from time-limited project delivery to a continuous activity that is happening in parallel with business development. In this regard, continuous delivery goes beyond the definition of a project, which, by its nature, is a temporary and time-limited process with progressive elaboration towards a goal. However, many common components of continuous delivery (for example, short build cycles and test automation) can be implemented under any development methodology (and provide similar benefits as with agile methods).

Continuous delivery is important as companies build digitalised capabilities and services which continuously interact with the company's customers. In this future environment, Fitzgerald and Stol (2017) point out that the systems development operations and business contexts cannot be as separate as they used to be. Their continuous nature goes together through continuous planning and budgeting (biz) to continuous integration and development (dev) and to continuous improvement and experimentation (ops).

The other major reason for using continuous delivery goes back to the "non-functional" needs that may emerge suddenly and require immediate system changes. These non-functional needs may include information security updates and technical upgrades to fix patches. Atlassian writes: "In the DevOps community, those with Agile experience acknowledge that SCRUM is useful for tracking planned work. Some work in operations can be planned: releasing a major system change, moving between data centres, or performing system upgrades. However, much of the work of operations is unplanned: including performance spikes, system outages, and compromised security. These events demand an immediate response. There is no time to wait for the items to be prioritised in a backlog or for the next sprint planning session.

For this reason, many teams that have come to embrace DevOps thinking look beyond Scrum to Kanban. This helps them track both kinds of work and to understand the interplay between them. They may also adopt a hybrid approach, often called Scrumban or Kanplan (kanban with a backlog)." For more information, please see: <https://www.atlassian.com/agile/devops>

Holmström-Olsson et al. (2012) provide a process for adopting continuous delivery. They propose a ladder model, with actions that must be taken in a certain order: 1) from traditional to agile R&D; 2) from agile R&D to continuous integration; 3) from continuous integration to continuous deployment; and 4) from continuous integration to innovation system including continuous experimentation.

When continuous delivery spreads through business, development, and operations, the lifecycle principle of projects and software and systems development is really challenged. The project is no longer the dominant principle for organising development. At the same time, many organisations will find that the nature of digitalised services, which directly interact with companies' customers, requires nothing less.

LESSONS LEARNED FROM AGILE AND CONTINUOUS DEVELOPMENT

Academic research does not provide a definitive answer on the impact of emerging methodologies on digital transformation success. Agile methodologies have important positive effects on development and transformation outcomes, but they do not give a universal answer to handling success and failure factors. There are important takeaways to be acknowledged.

Agile methods' most important principle is that responding to change is more important than following the plan. More profound planning does not solve the problems because the business environment continuously changes. This requires continuous learning. Many of the current development approaches are based on this urgent need to be able to respond and be flexible. Agile and continuous practices may help make objectives and conflict resolution more open and transparent. Short iterations and continuous deliveries keep developers and other stakeholders more aware of risks related to conflicting objectives.

However, complex and large IT projects are never only about systems development. Business cases need to be calculated and business benefits realised, sometimes long after the deployment phase itself has ended. They often involve selecting and buying new technology and selecting implementation partners. They change underlying business processes, alter peoples' roles and responsibilities, and require people to obtain new skills. In the case of a new product or service introduction, marketing and sales must time their go-to-market efforts.

Simply put, complex and large IT projects involve a substantial number of changes outside systems, changes that must be managed in a structured way to succeed. These broad changes cannot be managed simply based on the prioritised product backlog and visibility of the next three sprints. It is not enough to plan what these changes need, but equally important is to be structured and plan carefully upfront how to deploy these changes. e.g., what resources will be allocated to these activities, what kind of organisation model supports change deployments etc.

Different business problems and different projects require different solutions and methodologies. There is no one right way of doing everything. Each methodology has its advantages and disadvantages. If the expected project outcome is well-known in advance or can be predicted with high accuracy, waterfall is well-suited. This is often the case, for example, when you are replacing something existing with a new solution. On the other hand, if there is considerable uncertainty about the expected outcome, e.g., what customers will value, agile is the obvious choice. This is often the case, for example, when developing something new. When digital services development becomes an ongoing process with no clear start or end, continuous delivery methods like DevOps may provide the best tools for continuous and synchronised business development and technical development execution. Finally, there is a possibility to combine waterfall and agile into what we call a hybrid methodology.

Combining waterfall with agile is typically called disciplined agile. This is often used in large IT system implementations since they require certain logical and progressive paths to be followed (e.g., large-scale ERP solutions). For more information, please see <http://www.disciplinedagiledelivery.com/>. There are also two rather well-known hybrid models that combine the good parts from SCRUM and the continuous delivery from Kanban / DevOps: Scrumban and Kanplan. <https://www.atlassian.com/agile/kanban/kanplan>. In this white paper, we address the combination of waterfall and agile.

Especially in complex and large ICT projects, many organisations would benefit greatly by combining the best parts of waterfall and agile methods. Hybrid methodology takes the structured planning elements from waterfall methodology to establish a milestone or gate-based approach. Various practices from agile methods, like shorter development cycles, are then added to the framework: within the milestones, each phase of the project or program can be structured into sprints, e.g., design sprints and system development sprints. Other agile practices to be adapted are typically related to DevOps, team collaboration and measurement. Finally, the overall solution is tested end to end and deployed or launched in desired phases or sequence.

The main benefits of using agile as a key component of hybrid methodology are:

- First, agile provides more built-in tools and processes on how the team should work together daily. Waterfall methods focus less on these things, mainly assuming that a phase has a starting and end point, but how work gets accomplished is another problem. In many complex IT projects, shared understanding and purpose are becoming increasingly important.
- Second, agile provides more tools for ensuring that there is transparency and visibility to progress across all organisation levels. It provides more flexibility to adjust the plan, as the team and key stakeholders learn what adds value as the business environment changes. The challenge will be to “slice and dice the elephant” into a minimum viable product that can be further built on. Especially when teams are geographically distributed in several locations, building co-located agile teams is still a task to be resolved and may require restructuring or re-purposing existing teams.

Success and failure factors are largely universal and independent of the methodology chosen. The selection of the most suitable methodology offers a partial solution. However, it does not systematically address all the success and failure factors academic research identifies. An over-focus on methodology easily creates biases that methodology solves everything or is the answer to existing problems. In many organisations, the discussion about methodology is dogmatic. The old methodology is bad, while the new methodology is good. This is not the case, and over time, organisations learn that each methodology has its pitfalls and strengths.

PART 3: A FRAMEWORK FOR SUCCESS

In the previous section, we observed that new methodologies bring elements that positively affect the success of digital transformations and that the optimal

methodology is dependent on the context and method of application. We also learned that selecting the optimal methodology alone does not ensure success.

UNDERSTANDING THE SUCCESS AND FAILURE FACTORS

Due to the limitations of the available methodologies, a systematic approach is needed to address the success and failure factors identified by the academic research. A knowledge of these factors alone is pointless unless it leads to actions. What should then be done to:

- Put the missing success factors in place and strengthen the existing ones
- And eliminate or mitigate the failure factors?

First, one must understand what success and failure factors (listed in Table 1) mean in practice. The number one success factor, “Top management support and CEO

sponsorship”, can be used as an example. In the various projects we have run, the level of top management support has varied greatly. Still, the top management has typically believed that the support they are giving is on a good level, even when this is not actually the case. A better way to objectively assess whether the success and failure factors are properly addressed is needed.

The following table provides a framework for assessing the maturity of top management support and CEO sponsorship. It describes typical indicators from Stage 1 with limited or no support to Stage 5 with excellent support.

Maturity assessment: Typical indicators of stage				
1	2	3	4	5
<ul style="list-style-type: none"> • Top management and CEO are not aware of the project. • There is a conflict between the organisation's strategic intent and project objectives. • Top management is not involved in project planning or steering the project. • There is no connection between expected project outcomes and top management incentives. 	<ul style="list-style-type: none"> • Top management and CEO are aware of the project but do not actively support it. • There is no strategic fit between organisations' strategic intent and the project objectives. • Top management is aware of project planning and progress but does not contribute to it. • There is limited connection between expected project outcomes and top management incentives. 	<ul style="list-style-type: none"> • Top management and CEO are aware of the project. Some executives support it, but not all. • There is some strategic alignment between the project and the organisation as a whole, but it is not obvious. • Top management comments on project planning and progress. • There is some connection between expected project outcomes and top management incentives. 	<ul style="list-style-type: none"> • Top management and CEO have approved the initiation of the project. • Top management and CEO are involved in project steering group. • Top management publicly supports the project. • Project is aligned with top management strategic intent. • Top management contributes to project planning and progress. • There is a strong connection between expected project outcomes and top management incentives. 	<ul style="list-style-type: none"> • Top management and CEO initiated the project. • Top management and CEO are involved in project steering group and daily operations. • Top management very visibly and publicly support the project. • There is a strong strategic fit between project and organisation as a whole. • Top management directs project planning and progress. • There is very strong connection between expected project outcomes and top management incentives.

Table 3: Example of maturity assessment - Top management support and CEO sponsorship

The maturity of the other success and failure factors can be assessed similarly. Midagon has created a maturity assessment framework systematically covering the success and failure factors.

A maturity assessment serves two purposes:

- 1) Highlight potential problem areas. A low maturity score for an individual factor highlights a potential problem. If the score is lower than expected, areas to improve can be selected, and corrective actions taken.

- 2) Indicate the overall likelihood of success. The total score, whether low or high, gives an indication of the overall likelihood of success and failure. If the total score is lower than expected, one should consider postponing the project and taking corrective actions.

EMBEDDING SUCCESS FACTORS INTO METHODOLOGY

Once success and failure factors are understood, a systematic approach is needed to ensure they are proactively considered and acted on. Our point of view is that success and failure factors should be addressed and be an integral part of an organisation's project culture and methodologies. We also see that these factors are universal, i.e., independent of the development methodology applied.

Let us use Midagon's concept of project phases as a reference point:

- Initiation
- Project planning
- Execution
- Stabilisation
- Project closing
- Business as usual

Using these project phases as a basis, the table below indicates in which phase the maturity of each success and failure factor should be evaluated. It is important to acknowledge that maturity assessment is not a one-off exercise. Maturity may change over time for the better or for the worse. The assessment is needed in each phase where the factor in question is relevant.

The responsibility for assessment and the forum in which the results are reviewed varies depending on the methodology. The evaluation can be done – for example – at defined project check points or in sprint reviews.

CSFs/CFFs		Initiation	Project planning	Execution	Stabilization	Project closing	Business as usual
Critical Success Factors	1. Top management support	x	x	x	x	x	x
	2. Project team competence		x	x	x		
	3. Cooperation and open communication across organization	x	x	x	x	x	x
	4. Clear goals and objectives	x	x	x	x	x	x
	5. Effective project management		x	x	x		
Critical Failure Factors	1. System misfit	x		x	x	x	x
	2. High turnover of team members		x	x	x		
	3. Over-reliance on heavy customization		x	x	x		
	4. Poor consultant effectiveness		x	x	x		
	5. Poor IT infrastructure		x	x	x	x	x

Table 4: A subset of the success and failure factors identified in the study done by Rossi and Smolander.

As a result, we have a framework that helps us monitor the maturity of key success and failure factors continuously and systematically. Deviations can be quickly identified, and corrective actions taken. Follow-up and regularly repeated assessments ensure that the impact of corrective actions is measured and understood.

Organisations with large project portfolios gain additional benefits from these assessments. This is due to the accumulation of experience and data from several different projects, their maturity scores, and their success. This allows organisations to learn more about the relevance of different factors and start consciously considering them from the very early stages of new projects.

PROJECT MATURITY ASSESSMENT WITH CRITICAL FACTORS APPROACH

The Critical Success and Failure Factors (Table 1) can be aggregated into twenty generally critical factors in any transformation project. These neutral factors can logically be categorised into five main themes:

- Realistic expectations
- Proper preparation and planning
- Strong leadership
- Execution quality
- Smooth transition to “business as usual”

Each neutral factor under these main themes can turn into a success factor when it is taken into focus and managed well throughout the project. They can also turn into a failure factor if neglected or managed poorly. In the picture below (Table 5), the themes are connected to a project phases timeline to indicate in which phase of the project each theme and factors within the theme must be taken into focus to make it stay on the safe side. As the picture shows, the basis for project success is mostly set already in the early phases of the project. This makes it important to reserve enough resources and allow enough time for preparation – even though the temptation to kick off the project as fast as possible might be strong.

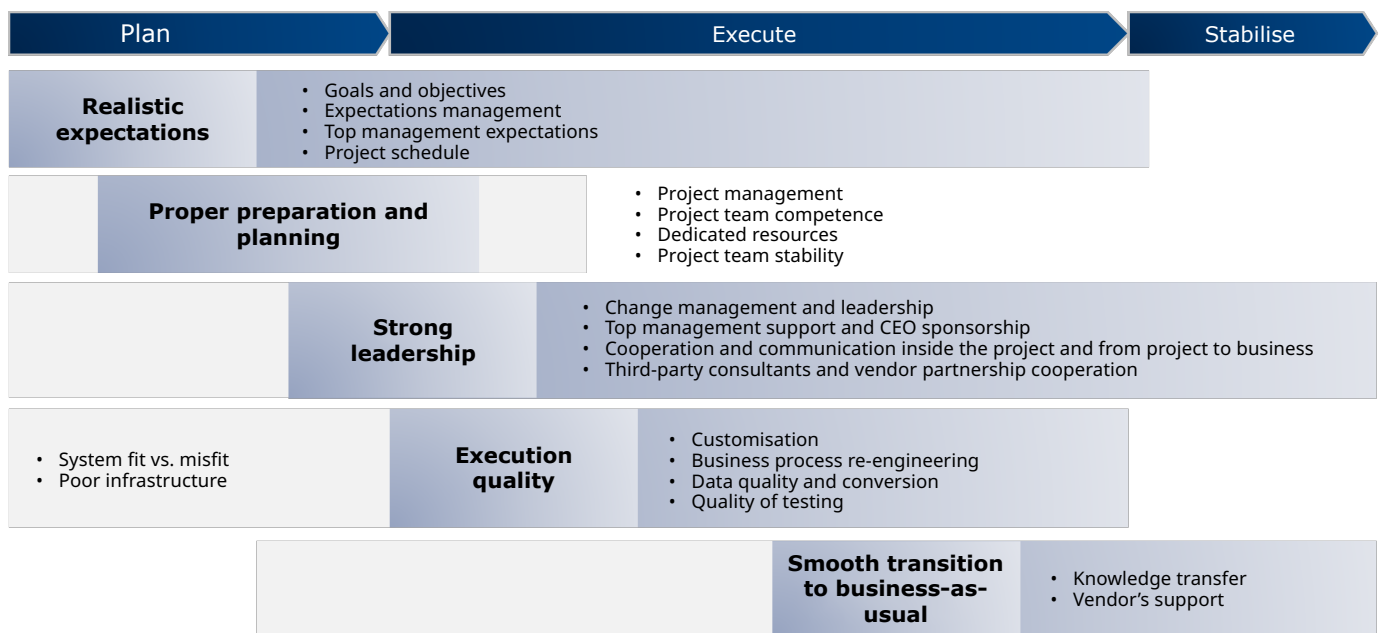


Table 5: Critical factors for project success and failure connected to project overall phases on timeline.

Midagon has developed an MPMA tool (Midagon Project Maturity Assessment), a qualitative survey, to systematically evaluate the status and maturity of these critical factors in a project. The survey scans all the critical factors and gives a maturity overview of each theme, showing how the involved organisation and people see the project situation. In an ideal situation, this assessment is targeted to everyone involved in the project work, and it is repeated a few times during the project to follow the trend and react to any negative changes. The number and interval of assessment rounds varies between projects depending on the project's overall timeline.

It also makes sense and adds value to combine the MPMA assessment with a regular CRA assessment (Change Readiness Assessment), providing this way a regular “full health check” not only for the ongoing project but also for the expected change in target. While MPMA's target group is all people involved in the project team and work, CRA's target group is all people who will eventually need to face and deal with all the changes the project is meant to make happen. MPMA indicates how people in the project see and trust the project, while CRA indicates how the people in the project's target perceive the change and if they trust they will be handling it.



Table 6: Overview of a project's "full health check." The Project Maturity Assessment tells how well the project is managed and progressing, and the CRA indicates how the change is managed and perceived by the people it involves.

CONCLUSIONS

It is important to keep your eye on the ball. Based on our experience, critical success and failure factors identified by academic research are not systematically addressed in the development and project methodologies. Different methodologies often focus mainly on what needs to be done (e.g., top management needs to support the project) but do not give concrete guidance on how to do that (how management supports the project). Solving this issue can easily improve the likelihood of success.

We propose methodologies and acceptance criteria for revising critical project milestones. A systematic and regular approach to evaluating the critical factors is needed. Regardless of the chosen methodology, pay close attention to securing success factors and removing failure factors. You should also make sure that you apply the selected project methodology prudently to the project at hand.

Reported failures of large-scale complex systems development and contemporary approaches to development, e.g., agile and continuous development, emphasise that most problems underlying success and failure factors are managerial. They relate to the way of working, communicating, and interacting with stakeholders. We recommend that you specifically focus on these aspects.

A successful program requires strong and visible sponsorship from top management. The leadership's role is vital in creating success. The company's executives, who are sponsors, need to create the circumstances for success. In many cases, management's objectives destroy any chance of success by setting overly ambitious targets compared to existing circumstances and conditions.

An experienced and competent project manager is essential. Leading a project is different from administration. Project plans need to be created, and progress tracked, which is administration. In a small and trivial project, administration may be enough. The bigger and more complex the project, the more it requires leadership in addition to project administration. People need to be led, and the value of relevant subject matter expertise becomes crucial.

Having the right team is also critical, while the cost of an incompetent team can be detrimental. If you have the wrong people on your project team, it is possible that nothing will save the project. On the other hand, when everything fails in a project, it is the key individuals who rise to the challenge and create success. This has been proven in many projects.

Great results are seldom realised without inspiring leadership, a high-performing team and a clear future vision. Take sports teams and teams that win championships as an example. You seldom hear them saying that the key success factor was tactics, a playbook or well-managed weekly practice routines. It is true that sometimes a team invents new tactics that had not been thought of before. However, other teams quickly learn and catch up. Exceptional teams are made up of individuals who want to contribute to each other's and their team's success. This is called team spirit. They typically have coaches who find ways to put the right players into the

right roles, unlock the true potential of the team and go beyond the potential. Truly exceptional teams grow together. Creating and leading high-performing project teams isn't that much different.

To conclude: 1) Select an optimal methodology for the context in question and adjust it to fit your environment and project, 2) actively and systematically evaluate and manage success and failure factors in every stage of the transformation, and 3) focus on people and leadership. This is what success ultimately depends on.

REFERENCES

- BONNIE, E., 2018. *Complete Collection of Project Management Statistics 2015*.
- DZIER, A. & FLYVBJERG, B., 2013. *Making Sense of the Impact and Importance of Outliers in Project Management Through the Use of Power Laws*. *Proceedings of IRNOP*, 11.
- BUDZIER, A. & FLYVBJERG, B., 2013. *Making Sense of the Impact and Importance of Outliers in Project Management Through the Use of Power Laws*. *Proceedings of IRNOP*, 11.
- CLARKE, P. & O'CONNOR, R.V., 2012. *The situational factors that affect the software development process: Towards a comprehensive reference framework*. *Information and Software Technology*, 54(5), 433–447.
- DAVENPORT, T. 2000. *Mission Critical: Realizing the Promise of Enterprise Systems*, Boston, Massachusetts, Harvard Business School Press.
- DIKERT, K., PAASIVAARA, M. & LASSENIUS, C., 2016. *Challenges and success factors for large-scale agile transformations: A systematic literature review*. *Journal of Systems and Software*, 119, 87–108.
- FITZGERALD, B. & STOL, K.-J., 2017. *Continuous software engineering: A roadmap and agenda*. *Journal of Systems and Software*, 123, 176–189.
- FRIESE R., KUBE H., SCHÖBL S. & SOLLER H., 2022. *How to get a core banking transformation right: Eight mistakes to avoid*. [How to get a core banking transformation right: Eight mistakes to avoid | McKinsey & Company](#)
- HOLMSTRÖM-OLSSON, H., ALAHYARI, H. & BOSCH, J., 2012. *Climbing the "Stairway to Heaven" -- A Multiple-Case Study Exploring Barriers in the Transition from Agile Development towards Continuous Deployment of Software*. In *2012 38th Euromicro Conference on Software Engineering and Advanced Applications*. 2012 38th Euromicro Conference on Software Engineering and Advanced Applications. 392–399.
- JOSLIN, R. & MÜLLER, R. (2014). *The impact of project methodologies on project success in different contexts*. Paper presented at *Project Management Institute Research and Education Conference*, Phoenix, AZ. Newtown Square, PA: Project Management Institute.
- LAANTI, M., 2013. *Agile and Wellbeing -- Stress, Empowerment, and Performance in Scrum and Kanban Teams*. In *2013 46th Hawaii International Conference on System Sciences*. 2013 46th Hawaii International Conference on System Sciences. 4761–4770.
- MATTILA, M., NANDHAKUMAR, J. & ROSSI, M. 2017. *Balancing of fluid and cemented routines in Enterprise Systems*.
- MISHRA, S., 2017. *Project Management: Lessons Learned from Failed Projects*. [Project Management : Lessons Learned from Failed Projects | LinkedIn](#)
- SHAUL, L. & TAUBER, D. 2013. *Critical Success Factors in Enterprise Resource Planning Systems: Review of the Last Decade*. *ACM Computing Surveys*, 45, 55- 55:39.
- SERRADOR, P. & PINTO, J.K., 2015. *Does Agile work? — A quantitative analysis of agile project success*. *International Journal of Project Management*, 33(5), 1040–1051.
- SOMERS, T. & NELSON, K. *The Impact of Critical Success Factors across the Stages of Enterprise Resource Planning Implementations*. *Hawaii International Conference on Systems Sciences*, 2001. 1-10.
- SYED-ABDULLAH, S., HOLCOMBE, M. & GHEORGE, M., 2006. *The Impact of an Agile Methodology on the Well Being of Development Teams*. *Empirical Software Engineering*, 11(1), 143–167.
- STANDISH GROUP INTERNATIONAL. *CHAOS Report 2015*. [CHAOSReport2015-Final.pdf \(standishgroup.com\)](#)
- TANRIVERDI, H., KONANA, P. & GE, L. 2007. *The choice of sourcing mechanisms for business processes*. *Information Systems Research*, 18, 280-299.
- WAGNER, E. L. & NEWELL, S. 2004. *'Best' for whom? the tension between 'best practice' ERP packages and diverse epistemic cultures in a university context*. *Journal of Strategic Information Systems*, 13, 305-328.
- WONG, A., SCARBROUGH, H., CHAU, P. & DAVISON, R. 2005. *Critical failure factors in ERP implementation*. *Pacis 2005 Proceedings*, 40.



midagon

Keilaniementie 1

02150 Espoo

Business ID: 2058234-3

www.midagon.com