

**AGILE PROJECT MANAGEMENT IN FINANCIAL IT PROJECTS: CHALLENGES  
AND SOLUTIONS**

*Sandeep Rachapudi*  
*Sandeep.Rachapudi@gmail.com*

---

*Abstract*

*Technology has advanced in several key areas of Information Technology. AI, ML and Cloud computing have predominately led to the growth of IT. Along with the fast-paced evolution of IT, managing software development projects in Agile have gained huge acceptance. IT organizations have moved from traditional methodologies like waterfall, spiral model [9] and V-Model to Agile development methodology. This paper aims to address some of the important challenges in Agile project management.*

*Keywords: TDD: Test driven development, IT: Information Technology, RTC: Rational Team Concert, AI: Artificial Intelligence, ML: Machine Learning, V-Model: Verification and Validation Model, Agile, Estimation, Agile Project Management, Agile Coach*

## **I. INTRODUCTION**

Agile project management offers a flexible and iterative approach to managing projects, emphasizing collaboration, adaptability, and customer feedback. It contrasts with traditional project management by focusing on delivering incremental value through short cycles, or sprints, and responding to change rather than following a predefined plan.

Increasingly, there is a lot of interest in moving from traditional software development methodologies towards Agile software development methodology. Various IT survey results [2] have clearly shown that the success rates of projects following Agile is increasing in comparison to projects that adopt traditional methodologies [3] [6].

Agile project management methodologies for software development have been around since the '90s. There are thousands of organizations using them and hundreds of thousands of trained agile coaches.

Agile methodologies are suited for a dynamic and changing environment. This situation calls for a collaborative environment between the user and the developer. While the popularity and availability of the methodology is important, organizational acceptance and project success is based on additional factors such as the project characteristics, organizational culture, and customer availability.

## **II. CHALLENGE**

Many projects fail while following Agile methodology [7], as they start adapting the new methodology without investing enough time in checking the team's readiness and maturity to adapt Agile. Engineering teams sometimes fail to understand or realize lately that Agile is more people dependent than any other traditional methodologies.

Even the most successful teams following traditional software development methodologies fail to succeed in Agile methodology [1]. Some of the simplest project management issues that are easily dealt with in traditional methodologies turn into challenges while adapting Agile methodology.

Going agile requires management awareness. They need to understand the benefits to the change as well as the details of how the change will affect operational aspects of the business.

Furthermore, they need to understand what will be expected from them and what should change in their behavior.

Many cultural and communication problems can be avoided or at least mitigated, by aligning with all levels of management before adopting an agile methodology. The business case for going agile should be defined and agreed upon with the management.

### III. SOLUTION

There are several factors that impact the success of agile projects. Successful adoption and management of Agile projects involves understanding the concept of Agile, training the team, creating the right team mix, and clearly defining the roles and responsibilities.

In Agile, the testing team has a more important role to play than ever. Increasingly, development teams look at them as consultants to help define requirements and suggest improvements in the applications. TDD has gained huge acceptance in IT projects. Here are some of the important factors that influence the success of agile projects.

**Agile Adoption:** Many times, the challenges that come with Agile project management [10] are due to lack of team's readiness to follow the new methodology [4]. For teams that are new to Agile, it is important to have an Agile coach, invest in tools for agile project management, and understand estimation methodology, etc.

**Agile Coach:** Experienced people bring practical knowledge from other situations and environments that can be helpful in avoiding pitfalls, in recommending tips and techniques for executing the project, and in coaching or supporting individual team members. While having many team members with experience is ideal, the Agile Coach, Product Owner, and Agile Tester are three roles where experience is most appreciated.

Investing in an Agile coach for the organization or the team transitioning to Agile is of enormous value. Agile principles sound simple. But to implement them needs a good understanding of retrospectives, sprint ceremonies, definition of a story point, definition of done, estimation using various techniques like planning poker, understanding Agile boards and sprints, optimum documentation, etc.

Agile methodologies promote a participatory design, where the customers or users become part of the team designing the behavior of their software. User stories facilitate participatory design, as the users are required to provide the content for the stories.

Agile coach can help start the project with chartering process. An initial release planning (sprint zero) may be used for working with the Product Owner to perform some initial planning, including defining a high level- architecture. The initial release planning can be used to:

- Produce and estimate the initial product backlog,
- Break down requirements into user stories to produce the product backlog,
- Identify, analyzed dependencies between the product backlog elements
- Produce the product roadmap

Understanding all the above aspects of Agile with the help of a coach is an excellent way to start and manage an Agile project.

### IV. TOOLS

Agile teams need to conduct daily stand-up meetings. Agile board helps the team understand the status of work and move the work items, called stories, on the board. Many favor the physical boards and use post-it notes to represent stories. Moving these stories across the board and discussing the dependencies and blockers every day is a ceremony to be performed. Daily stand-up is critical for the success of Agile projects.

With the COVID pandemic, physical board have slowly disappeared in many IT organizations, and web-based tools have gained popularity for Agile project management. There are several tools

for managing projects in Agile. Some of the most popular ones are Jira, RTC. While the choice of tool varies across organizations, investing in one of them is important for the project.

## V. METRICS

Agile tools help in generating metrics. Many tools provide the ability to create dynamic dashboards that show important metrics such as Burn down chart, sprint velocity, defect count, control chart etc. But many metrics in Agile depend on story points. The process followed to estimate and size story points is an important element of successful agile project management.

## VI. ESTIMATION

There are several ways to perform estimation in Agile. The popular ones are planning poker and T-shirt sizing. It's best to start with planning poker. When the team understands sizing, we can implement T-shirt sizing.

It is important to define a story point estimation mechanism that is quantifiable. Many a times, agile teams don't understand the concept of story points and they give inaccurate estimates. It's best to breakdown the work into clear tasks that formulate the work item. Let's take the example of the financial domain to define an estimation template.

Data management and processing are one of the most critical aspects of financial systems. The following methodology explains how to define story points and helps agile teams in estimation and sizing.

Define story sizes that are acceptable. Since, story points in Agile are in Fibonacci series, it is essential to agree on the size of the stories. The Fibonacci series allows a lot of numbers for sizing. But it is ideal to not have stories that are greater than 8 points.

Teams should learn to break the effort to manageable sizes. Example: Make a 13-pointer story into 8 points and 5 points. We can use one of them for analysis and the other for execution/implementation.

Here is the recommended story size used by a leading US financial services company that has lot of data processing, data manipulation, data retention, data transformation, data analysis, and data distribution in scope.

Story Size	Story Points	Description
Small	1 - 3	For a DB change that involves modification to View/Package/Function or Procedure. Similarly, a Java UI/ML enhancement impacting 1 or 2 functions
Medium	5	When change involved has multiple views/procedures/packages and has considerable design/analysis effort
Large	8	Typical for a large change impacting multiple systems/subsystems

Along with the size, it's also important to identify the complexity. This is one aspect ignored while estimating and sizing stories. Definition of complexity can be defined by engineering teams with the help of Agile coach or the scrum master [5].

Complexity	Description
3 - Low	Well defined requirement with little or no analysis and impacts only a minor feature
5 - Medium	Requires some analysis and can impact 1 or more features
8 - High	Requires more analysis and involves working with multiple teams and has potential to impact multiple functions

Another important aspect that many agile teams fail to understand and define is creating benchmark stories. The goal is to use the benchmark to estimate stories. Without any benchmark, almost every team member tends to give varying estimates making the whole process of estimation very time consuming and inefficient.

Hence, it is recommended to define a template, use that as basis for benchmark stories and use the same template for estimation on need basis.

The idea is not to fill the template for every story. Rather, it must be used for reference on a need basis. Benchmarking will help all the team members have common understanding of estimation.

#### **Benchmarking of a small 3-point story estimation using a custom template:**

**Story ID: XXX - Comma separated numbers for counts**

S. No	Activity	Complexity	% Distribution	Story Points	Description and outcome
1	Analysis	5	15	0.75	Analysis document presented to team during Discovery/Iteration planning
2	Design	NA	10	0	Design document presented to team during Discovery/Iteration planning
3	Logical View Creation	NA	10	0	Simple - Impacts 1-2 tables. Medium Impacts 2+ and Complex impacts several tables
4	Logical View modification	3	5	0.15	Simple - Impacts 1-2 tables. Medium Impacts 2+ and Complex impacts several tables
5	Package/Function Creation	NA	10	0	Simple - Impacts 1-2 tables. Medium Impacts 2+ and Complex impacts several tables
6	Package/Code Modification	5	5	0.25	Simple - Impacts 1-2 tables. Medium Impacts 2+ and Complex impacts several tables
7	Test Case preparation	3	5	0.15	Complexity Defined based on functionality
8	Test Execution	5	5	0.25	Complexity Defined based on functionality
9	Code Review	3	5	0.15	Document review outcome - in Quip?
10	Test Case Review	3	5	0.15	Document review outcome - in Quip?
11	Deployment	3	5	0.15	Document in Quip
12	Documentation	NA	5	0	Knowledge gaining/Links to Test Cases/
13	ETL/Informatica Job Setup	NA	10	0	Creation of Param files and Informatica Mapping
14	UAT	3	5	0.15	Effort and coordination
	Total Points			~3 Pt	

**Benchmarking a medium 5-point story estimation:**

**Story ID : XXX - Download full list of tables**

S. No	Activity	Complexity	% Distribution	Story Points	Description and outcome
1	Analysis	8	15	1.2	Analysis document presented to team during Discovery/Iteration planning
2	Design	NA	10	0	Design document presented to team during Discovery/Iteration planning
3	Logical View Creation	NA	10	0	Simple - Impacts 1-2 tables. Medium Impacts 2+ and Complex impacts several tables
4	Logical View modification	5	5	0.25	Simple - Impacts 1-2 tables. Medium Impacts 2+ and Complex impacts several tables
5	Package/Function Creation	NA	10	0	Simple - Impacts 1-2 tables. Medium Impacts 2+ and Complex impacts several tables
6	Package/Code Modification	8	5	0.4	Simple - Impacts 1-2 tables. Medium Impacts 2+ and Complex impacts several tables
7	Test Case preparation	5	5	0.25	Complexity Defined based on functionality
8	Test Execution	5	5	0.25	Complexity Defined based on functionality
9	Code Review	5	5	0.25	Document review outcome - in Quip?
10	Test Case Review	5	5	0.25	Document review outcome - in Quip?
11	Deployment	5	5	0.25	Document in Quip
12	Documentation	NA	5	0	Knowledge gaining/Links to Test Cases/
13	ETL/Informatica Job Setup	NA	10	0	Creation of Param files and Informatica Mapping
14	UAT	5	5	0.15	Effort and coordination
	Total Points			~5 Pt	

**Benchmark of a large 8-point story estimation:**

**Story ID: XXX - Application Performance improvement**

S. No	Activity	Complexity	% Distribution	Story Points	Description and outcome
1	Analysis	8	15	1.2	Analysis document presented to team during Discovery/Iteration planning
2	Design	5	10	0.5	Design document presented to team during Discovery/Iteration planning
3	Logical View Creation	NA	10	0	Simple - Impacts 1-2 tables. Medium Impacts 2+ and Complex impacts several tables
4	Logical View modification	8	5	0.4	Simple - Impacts 1-2 tables. Medium Impacts 2+ and Complex impacts several tables
5	Package/Function Creation	NA	10	0	Simple - Impacts 1-2 tables. Medium Impacts 2+ and Complex impacts several tables
6	Package/Code Modification	8	5	0.4	Simple - Impacts 1-2 tables. Medium Impacts 2+ and Complex impacts several tables
7	Test Case preparation	8	5	0.4	Complexity Defined based on functionality
8	Test Execution	8	5	0.4	Complexity Defined based on functionality
9	Code Review	8	5	0.4	Document review outcome - in Quip?
10	Test Case Review	8	5	0.4	Document review outcome - in Quip?

11	Deployment	8	5	0.4	Document in Quip
12	Documentation	NA	5	0	Knowledge gaining/Links to Test Cases/
13	ETL/Informatica Job Setup	5	10	0.5	Creation of Param files and Informatica Mapping
14	UAT	5	5	0.15	Effort and coordination
	Total Points			~8 Pt	

Story points are calculated as the product of complexity and effort distribution. So, a large 8-point story has 15% effort allocation for analysis, making it worth  $8 \times 15 / 100 = 1.2$  story points. Similarly, not all stories need design, hence there is no effort calculated in the above examples for 3-point and 5-point stories. A summation of all the activities and their corresponding story points will determine the final size of the story.

## VII. CONCLUSION

Successful agile project management is a journey. It involves understanding the Agile manifesto, adopting the new methodology, and using Agile management tools to the advantage of the team.

Adopting Agile project management can lead to more responsive, efficient, and customer-focused projects. It's particularly beneficial in environments where change is frequent and customer needs are evolving. However, successful implementation requires commitment from all team members, an openness to change, and a supportive organizational culture.

Investing in Agile coach, Agile management tools, and defining estimation techniques are the key to successful Agile projects.

## REFERENCES

1. E Octavio. "Agile Is Getting Us Nowhere Fast". Forbes. <https://www.forbes.com/sites/frog/2022/04/20/agile-is-getting-us-nowhere-fast/> (accessed May 25, 2022)
2. "Agile adoption and development trends" <https://www.advisedskills.com/about/news/233-agile-adoption-and-development-trends-surprising-statistics-2022> (accessed May 25, 2022)
3. C Danijela, L Bojan, G Danijela, T Nemanja, D Milan, Medic N. "Agile vs. Traditional Approach in Project Management". <https://www.sciencedirect.com/science/article/pii/S2351978920303814> (accessed Jan 10, 2022)
4. J Miller. "Agile problems, challenges, & failures". [https://www.researchgate.net/profile/Gloria-Miller-2/publication/335475075\\_Agile\\_problems\\_challenges\\_failures/links/5d683a6d299bf1d599449143/Agile-problems-challenges-failures.pdf](https://www.researchgate.net/profile/Gloria-Miller-2/publication/335475075_Agile_problems_challenges_failures/links/5d683a6d299bf1d599449143/Agile-problems-challenges-failures.pdf) (accessed April 5, 2022)
5. M Maarif, M Yosof, N Satar. "The Challenges of Implementing Agile Scrum in Information System's Project". [https://papers.ssrn.com/sol3/papers.cfm?abstract\\_id=3786550](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3786550) (accessed Dec 5, 2021)
6. G Peggy, B Leonor, S Helen, D Advait, Taylor K. "The challenges that challenge: Engaging with agile practitioners' concerns". <https://www.sciencedirect.com/science/article/pii/S0950584916300623> (accessed Feb 10, 2022)
7. S Aljaz. "Agile Project Management in Product Development Projects". <https://www.sciencedirect.com/science/article/pii/S1877042814021259> (accessed Mar 15, 2022)
8. C Lan, M Kannan, R Balasubramaniam, S Sumantra. "Adapting funding processes for agile IT projects: an empirical investigation" <https://www.tandfonline.com/doi/abs/10.1057/ejis.2012.9> (accessed Oct 28, 2021)
9. C John, Project Management the Agile Way: Making it Work in the Enterprise. Fort Lauderdale, FL, USA. J. Ross Publishing 2010.

10. S Saya. "Mapping Agile Project Management Practices to Project Management Challenges for Software Development"  
[https://webweb.ams3.cdn.digitaloceanspaces.com/data/simmcdev.webweb.ai.in/MCAdigitalbook/software-project-management-20240629T072455Z-001/MAPPING%20AGILE%20PROJECT%20MANAGEMENT%20PRACTICES%20TO%20PROJECT%20MANAGEMENT%20%20\(%20PDFDrive%20\).pdf](https://webweb.ams3.cdn.digitaloceanspaces.com/data/simmcdev.webweb.ai.in/MCAdigitalbook/software-project-management-20240629T072455Z-001/MAPPING%20AGILE%20PROJECT%20MANAGEMENT%20PRACTICES%20TO%20PROJECT%20MANAGEMENT%20%20(%20PDFDrive%20).pdf) (accessed Jan 12, 2021)