Developing CRMSys at SoftTel: Traditional or Agile?

Hyung Koo Lee
HEC Montréal
hyung-koo.lee@hec.ca

Tianjie Deng
University of Denver
tianjie.deng@du.edu

Sumantra Sarkar
Binghamton University, SUNY
ssarkar@binghamton.edu

ABSTRACT

The increasing popularity of Agile development has significantly changed the way organizations plan, design, and implement software applications. However, considering the complexity involved with software development, the decision to choose an Agile approach over the traditional waterfall approach is not straightforward. Organizations should carefully evaluate a variety of factors when choosing a development approach. In this teaching case study, the authors provide a discussion case to facilitate student discussion of using Agile vs. traditional approaches in software development. This case study is designed to reflect the complexities students are likely to encounter in actual software development projects. The case can be used in undergraduate or graduate level management information systems courses that cover software development approaches. Students can gain an understanding of factors to consider when choosing a development approach, which tailors to the organization, the project environment, and the project team.

Keywords

Software development approach, Agile, Traditional, Project management, Teaching case

History: Arturo Castellanos, Accepting Senior Editor. This paper was received October 9, 2020 and was accepted on November 7, 2020 after one round of revisions. Published online November 10, 2020.

INTRODUCTION

Despite the advancements in project management, managing Information technology (IT) projects continues to be challenging for organizations. According to the Standish Group's (2015) CHAOS report, 19% of software projects fail, while 52% of them are challenged in terms of completing on time, on budget, or delivering a satisfactory result. Among the various factors that contribute to the success or failure of software projects, the project management approach taken to execute the project continues to be a critical factor. Projects using an agile approach have demonstrated a success rate that is almost four times greater compared to those using the traditional (i.e., waterfall) approach (Standish Group, 2015). Since the advent of the Agile manifesto (Beck et al., 2001), approaches such as Scrum and Extreme Programming have continued to gain more recognition in the industry due to advantages such as the ability to adapt to changing requirements, improved project visibility, and faster delivery of quality software (Cram & Marabelli, 2018). A recent report found that around 95% of organizations practice Agile approaches in some form or the other for their software development projects (VersionOne, 2020).

The popularity of Agile approaches in the industry has also triggered a significant amount of work in the academia, not only in terms of industry-related research, but also regarding how such approaches can be taught in the classroom. Agile approaches are being introduced in mandatory project management or systems analysis and design courses in the information systems curricula (Sharp & Lang, 2018). Different teaching approaches have been suggested, ranging from teaching Agile concepts and practices in traditional lecture format (e.g., Landry & McDaniel, 2016) to having students engage in using Agile practices through a simulation or class project (e.g., Sibona, Pourreza, & Hill, 2018). Most of these courses focus on introducing what Agile project management is, and cover specific practices mainly pertaining to Scrum (Landry & McDaniel, 2016). Most of the discussion in these courses are centered around the advantages of using an Agile approach. The subject of when to consider Agile approaches or Agile adoption, however, is limited. Considering the complexity involved with software development and the project environment, the decision to choose an Agile approach over the sequential traditional approach is not simple. This decision is driven by various factors. Hence, in this article, we provide a case study that courses covering Agile approaches can use to facilitate the discussion of choosing between Agile vs. traditional approaches in software development.

CASE SUMMARY

This case is about the development of a new customer relationship management (CRM) system in a large manufacturing firm. Students will analyze the case to identify the factors that can influence the choice of a software development approach and decide whether to use the Agile or traditional approach for development. As students explore the case, they will be asked to identify the important factors about the organizational context, the project environment, and the project team that need to be considered for selecting a development approach.

CASE TEXT

SoftTel is a large manufacturing firm and has decided to develop CRMSys, a new CRM system with cutting-edge technology that will replace the current CRM system, which cannot keep up with the company's needs. Daniel, who is in charge with the development and implementation of CRMSys, is sitting in his office thinking about how he should procced with the project. The success of the project is both crucial to the company as well as his career – he cannot afford project failure.

"The software development approach that we use to develop CRMSys is going to have a major impact on the success of the project. Would an Agile approach be appropriate for the CRMSys project? Or should we stick with the traditional development approach that we are accustomed to?"

Company Background

SoftTel Inc. is a mobile phone manufacturing company that has a diverse line of products. The company, headquartered in the US, is well recognized in the industry and has multiple branches worldwide. Currently, SoftTel has approximately 10,000 employees, with an information technology (IT) department that has 100 employees dedicated to providing IT services. These services include not only the maintenance of IT systems and infrastructure, but also the development and implementation of new information systems. The IT function is centralized, but the branches in each country have a few IT personnel to support their local operations.

The company follows a matrix organizational structure with a separate project management office (PMO) that oversees all projects within the organization. The IT department is responsible for most IT projects in conjunction with the relevant functional department (e.g., marketing, accounting, etc.). There are several projects that involve third parties such as consulting agencies, external developers, and software vendors. While the company has a clear hierarchy and a relatively formal atmosphere, it values and fosters collaboration for projects that involve multiple departments. Many of the projects in the past have been successful with minimal conflict due to the high level of trust between employees and departments.

Project Background

The current CRM system was implemented in 2005, which was based on state-of-the-art technology at that time to handle the growth of the company over a long period of time. Few years ago, the system started to have performance issues. While these problems could be dealt with on an ad-hoc basis, recently, they have started to occur more frequently because of the continued growth of the company. Additionally, the system was not capable of supporting the different processes pertaining to the newer marketing channels.

After discussing the situation with the IT department, the VP of marketing and sales decided to initiate a project with the objective of developing CRMSys, a new CRM system that will replace the current legacy CRM system. This was determined due to the frequent down time needed for maintenance activities on the system, which was a huge waste of time and resources. Considering the criticality of the CRM system to the company, the new system needs to be implemented as soon as possible. The project implementation target was decided to be one year with a fixed budget. Given the uniqueness of their CRM operations and the strong IT department at SoftTel Inc., the decision was made to develop CRMSys in-house rather than purchasing an off-the-shelf product. In addition to the standard CRM features, the new system needed to include innovative modules related to CRM such as chatbots, robotic process automation, and artificial intelligence, thereby increasing the overall complexity of the system. Furthermore, due to different data interchange standards used by interfacing software, it is difficult to foresee and obtain all the system requirements up front at the beginning of the project. The new software needs to have a certain level of flexibility to deal with these occasions.

The Project Team

The project team consisted of 27 people: 14 people from the IT department, 6 people from the marketing and sales department, and 2 external consultants who are all collocated on-site, and 5 external developers (who have experience in innovative modules mentioned above) from a software development company located in India. The level of involvement differs between each team member – some of them are fully devoted to the project while others continue with their regular responsibilities within their respective department. The people from the IT department were experienced in that they successfully developed and implemented several different information systems with various departments in the organization. Most projects that were conducted in the past followed the traditional development approach (i.e., the waterfall model with sequential phases), while there were very few recent cases where few agile practices were used in certain phases of the traditional approach. Most of the senior and middle managers in the team are accustomed to the traditional development approach because of their experience with past projects. Only the relatively junior employees had some level of knowledge and minimum experience in agile practices. However, the overall interest in agile development methodologies was increasing at SoftTel due to the limited but successful experience of using certain agile practices, as well as the popularity of the approach in the IT industry.

QUESTIONS FOR DISCUSSION

The CRMSys at SoftTel case can be used to discuss a number of possible subjects related to selecting an appropriate software development approach. These discussions could be done in small groups or with the entire class. The following are the suggested discussion questions to ask after students read the case:

1. What are the important factors that would impact your decision on choosing a software development approach? For each of the identified factors, which development approach is best suited? Why?
2. As the person responsible for the project, which development approach would you adopt for the CRMSys project given the circumstances of the project and the organizational environment? Evaluate the advantages and challenges for SoftTel in adopting the selected approach and provide justifications towards why you deem the selected approach appropriate.

TEACHING NOTES

Teaching Objectives

This case builds on the understanding of basic concepts, values/principles, and practices of traditional and agile development approaches.

The teaching objectives of the case are:

- To understand the basic concepts, principles, and practices of agile approaches
- To understand how the agile approach differs from the traditional software development approach
- To understand the important factors that need to be considered for selecting the appropriate development approach (agile vs. traditional)
- To understand the dilemmas in selecting a software development approach

Teaching Suggestions

This case study is suited for project management or systems analysis and design courses, as well as other courses that cover Agile development approaches. It can be used at both undergraduate and graduate level courses. It will require a few hours for instructor preparation where the instructor reviews the case, the instructor material, and structures the in-class discussion.

Since the case is brief, students can read it and conduct their analysis during class. The discussion normally takes around 30-45 minutes, depending on (1) whether you have students analyze the case individually, open the discussion to the entire class, and provide a debriefing at the end, or (2) whether you have the students analyze the case in small groups, integrate and discuss their analyses with the entire class, and provide a debriefing.

To maximize the effectiveness of the case discussion, we recommend that the case is used after the instructor covers the basic values/principles and the specific practices of Agile and traditional software development approaches. This may depend on the degree of how much Agile approaches are covered (e.g., two or three classes on Agile) and when (e.g., early during the semester or close to the end), but normally, it should be close to the end of the Agile portion of the course.

Suggested readings for students:

- Beck et al. (2001) “Manifesto for Agile Software Development” (https://agilemanifesto.org/)
- Cram, W. A. (2019). Agile Development in Practice: Lessons from the Trenches. *Information Systems Management*, 36(1), 2-14.
- Cram, W. A., & Marabelli, M. (2018). Have your cake and eat it too? Simultaneously pursuing the knowledge-sharing benefits of agile and traditional development approaches. *Information & Management*, 55(3), 322-339.
- Conforto, E. C., Salum, F., Amaral, D. C., da Silva, S. L., & de Almeida, L. F. M. (2014). Can Agile Project Management be Adopted by Industries Other than Software Development? *Project Management Journal*, 45(3), 21-34.

Discussion Questions and Proposed Solutions

1. What are the important factors that would impact your decision on choosing a software development approach? For each of the identified factors, which development approach is best suited? Why?

Both the traditional and agile methodology offer unique benefits. The advantages of the traditional method include its straightforwardness, linear design, and clearly defined milestones and policies (Vidgen & Wang, 2009). In comparison, the agile approach focuses on collaboration and the flexibility to respond to changes (Cram & Marabelli, 2018). A project team should consider and evaluate a variety of different factors, both at the organizational level and the project level, before selecting the appropriate software development approach.

At the organizational level, the following factors can be of consideration:

- ***Organizational culture***: The traditional methods are designed to work in formal, command-and-control management styles while the agile approach works better in a more informal and collaborative culture. Agile development teams are self-governing (Rigby, Sutherland, & Noble, 2018). Therefore, organizations exhibiting a higher level of informality may be more equipped to adopt the Agile approach. By contrast, organizations with strict hierarchy and a formal atmosphere may be more suitable to the traditional methods.
- ***Organizational structure***: The organizational structure can range from the traditional functional organization to the project-based organization, with matrix organizations as a blend of both in between. Among the three, the project-based organizational structure is more aligned with the Agile approach, due to its focus on projects and the high level of integration (Conforto, Salum, Amaral, da Silva, & de Almeida, 2014). Another factor to consider is the existence of the PMO in the organization. If a PMO exists, generally they govern all projects in the organization by providing standard methodologies and monitoring project performance, and thus is more suitable to support the traditional development approach. By contrast, because agile teams work in a self-organizing and self-governing fashion, the need for a PMO is minimum.
- ***Size of organization***: When attempting to adopt to agile approaches, small or midsized companies may face fewer challenges than larger organizations because they are often more flexible (Cram & Marabelli, 2018), and are more likely to learn fast through informal interactions (Nevis, DiBella, & Gould, 1998). By contrast, large organizations may face more challenges in achieving agility due to the established hierarchy and conflicts with existing processes.
- ***Level of trust***: The traditional method has low reliance on trust, due to the establishment of stages and policies (Boehm & Turner, 2003). By contrast, the agile method relies heavily on trust, without which the common practice of agile development (e.g., collective code ownership, collaborative workspaces, and pair programming) is challenging to implement. The trust between the project team and the client is also critical for the agile approach, as it relies on user participation. Thus, the level of trust, both within the team in the organization and external as in between the organization and the software development company’s team, should be evaluated when selecting the development approach. If such trust is low, the traditional approach may be preferred.
- ***Necessary resources***: The traditional approach is more likely to require more resources (both financial and human) to maintain formal documentation and monitor project progress (Cram & Marabelli, 2018). Therefore, when there are limited resources, adopting the traditional development approach may more challenging relative to the Agile approach.

The following factors can be considered at the project level:

- ***Timeline***: The Agile approach can be more appropriate when the project team needs to develop and deliver the software in a short time frame due to its focus on the quick delivery of working software (Cram, 2019). Using Agile approaches for projects with a longer timeline may be more challenging, as it is difficult for team members to maintain a high level of engagement without project fatigue.

- ***Requirements***: Because of its flexibility to respond to changes, the Agile approach is better when user requirements are hard to define and obtain up front (Cram, 2019). In comparison, traditional development emphasizes detailed upfront planning oriented towards a predefined sequence of steps starting from a detailed requirement analysis. Formalized requirements are signed off by stakeholders before the initiation of design and development. Therefore, when the requirements from the stakeholders are relatively clear and stable, traditional approaches are considered.
- ***Complexity of system***: The plan-driven traditional approach generally handles complex system development better, with its clearly defined goals and milestones, monitoring and control, and thorough documentation. When the complexity of the development system increases, the Agile approach may become unsuitable with its limited documentation.
- ***Type of project***: The traditional approach is more suitable in supporting software development projects that require formal controls and stability (e.g., government-funded projects), or projects that involve developing mission critical systems where the stakes of failure are high, as well as projects with a fixed budget and deadline.
- ***Team size and location***: The Agile approach advocates regular and intense team interaction and close collaboration; thus, it is generally suitable for smaller teams where project members are collocated. It can pose additional challenges when certain team members are in offshore locations because of the differences in culture, communication, location, and time zone, etc. However, with the recent advancements in Agile approaches (e.g., Scaled Agile Framework, SAFe), it has been proven that Agile can be extended to distributed and large projects (Bose, 2008; Visitacion, 2016).
- ***User participation***: Agile approaches depend heavily on the continuous engagement of the user. When such engagement is limited, the project team may consider the traditional approach, in which user engagement is generally focused during the early stages of the project for requirements gathering.
- ***Past experience***: Past experience of team members, with each development approach should also be a consideration in deciding on the right approach. Adjusting from traditional to Agile development approaches involves not only the training of Agile practices and adjusting processes related to the management of human resources (HR), but also a change in the mindset that aligns with Agile values and principles. Without the experience and appropriate mindset, the team can face considerable amount of challenges when transitioning from one approach to the other.

2. **As the person responsible for the project, which development approach would you adopt for the CRMSys project given the circumstances of the project and the organizational environment? Evaluate the advantages and challenges for SoftTel in adopting the selected approach and provide justifications towards why you deem the selected approach appropriate.**

Regarding the development approach that SoftTel should proceed with, there is no one correct solution. Given the characteristics of the organization and the project, the development approach selected will differ by student. A good answer should provide necessary assumptions and justifications of his/her decision based on the discussion of the factors considered in the preceding question. The following are only indicative of what key factors could be considered for each option.

Option 1: Continue using the accustomed traditional approach

The first option for the project team would be to continue using the traditional approach. This way, the team can leverage its strengths for developing CRMSys as many of the project team members from the IT department already have experience in successfully developing and implementing different information systems with such approach. Regarding the organizational culture, the strict hierarchy and formal atmosphere aligns better with the traditional approach which emphasizes formal management, control, and monitoring of the project. Also, CRMSys may be considered a mission critical system at SoftTel, hence, the cost of failure for the project may be very high. Given the importance and the complexity of the system, as well as the fixed budget and timeline, the traditional development approach could be a good approach for the project.

However, the difficulties in obtaining system requirements pose challenges to adopting the traditional approach because it will be difficult to plan everything up front. Due to the sequential nature of the traditional approach, the development of the new modules (e.g., chatbots, RPA, and AI) and the complexity of CRMSys can increase project risk – any unforeseen problems with these modules will be dealt with during the later stages of the project, which could have been tackled earlier using an Agile approach. Overall, the team will need to find ways to address certain issues that could be better addressed by using an Agile approach.

Option 2: Adopt an Agile approach and discard the traditional approach

Considering certain characteristics of the CRMSys project, the Agile approach could be a viable option. An Agile approach has advantages in dealing with the challenges to obtaining system requirements up front and providing flexibility through its iterative nature. While SoftTel does have a formal atmosphere, the organizational culture that fosters collaboration, as well as the high level of trust between employees and departments can contribute to increasing the engagement of team members.

Additionally, the agile approach can be suitable for the project because it has relatively short timeline with small team that is collocated on site.

On the other hand, the project team does not have any expertise in using Agile approaches. This can make the transition from traditional to agile very difficult, as it generally involves training team members, adopting a mindset that aligns with Agile values and principles, and adjusting relevant software development and HR processes (e.g., recruitment, performance evaluation, etc.). Such changes are often difficult to make in large enterprises due to conflicts with existing processes and employees' resistance to change. Additionally, having external developers in offshored locations may hinder collaboration within an agile environment.

Table 1 summarizes the organizational and project level factors of the CRMSys project at SoftTel and how each factor is in favor of the traditional vs. Agile approach.

Table 1. Organizational and Project Level factors in Favor of Each Approach

Level	Factors	Traditional	Agile
Organizational level	Organizational culture	Clear hierarchy and relatively formal atmosphere	Value and foster collaboration*
	Organizational structure	Functional or matrix organization structure, existence of PMO*	Project organization structure
	Size of organization	Large organization*	Small or midsized organizations
	Level of trust	Low reliance on trust due to the establishment of stages and policies	High level of trust necessary between employees and departments*
	Necessary resources	Requires more resources to monitor project and maintain documentation*	Requires less resources as the focus is on delivering a working product
Project level	Timeline	Projects with a long timeline	Projects with a short timeline*
	Requirements	Requirements are clear and stable, and can be gathered up front	Changing requirements, difficult to gather all requirements up front*
	Complexity of system	More complex systems*	Less complex systems
	Type of project	Projects that require formal controls and stability, projects that involve mission critical systems	Projects that involve implementing new and innovative modules that could be risky*
	Team size and location	Large team where members can be dispersed	Relatively small team where key members are collocated*
	User participation	Some members are involved less with the project than others*	Team members are dedicated to the project
	Past experience	Prior experience in successful projects using the traditional approach*	Prior experience in using Agile, openness to adopt a mindset that aligns with Agile values and principles

*: indicates those factors that are in favor of the CRMSys project

Option 3: Adopt a hybrid approach based on both traditional and Agile approaches

To bring the best of both worlds, a hybrid approach based on both traditional and Agile approaches could also be considered. Indeed, many of the students will bring this point up at the end of the discussion, given the characteristics of the organization and the project. However very few courses discuss how a hybrid approach mixing traditional and Agile practices can be implemented. In reality, many organizations use some form of hybrid approach for software development (VersionOne, 2020). Hence, the case can lead to a follow-up discussion on different hybrid approaches such as Conforto and Amaral's (2016) framework that combines Agile practices with the stage-gate model, or Hayata and Han's (2011) model that adopts agile practices during the implementation phase (i.e., detailed design, development, and unit test) of the V-model of software development. The following provides a list of suggested readings for the discussion of hybrid development approaches:

- Conforto, E. C., & Amaral, D. C. (2016). Agile project management and stage-gate model—A hybrid framework for technology-based companies. Journal of Engineering and Technology Management, 40, 1-14.
- Hayata, T., & Han, J. (2011). A hybrid model for IT project with Scrum. Proceedings of 2011 IEEE International Conference on Service Operations, Logistics and Informatics.
- Wysocki, R. K. (2019). Hybrid Project Management Framework. In Effective Project Management: Traditional, Agile, Extreme, Hybrid (8th ed., pp. 405-451). Indianapolis, IN: John Wiley & Sons, Inc.

CONCLUSION

The case presented provides opportunities to discuss several different subjects that are often overlooked when teaching agile approaches. Selecting a software development approach is a complex decision where many different factors need to be considered. This case facilitates the discussion on choosing between traditional vs. Agile approaches and can be further extended to discuss hybrid development approaches in undergraduate and graduate courses that cover Agile approaches. While the company SoftTel and the project CRMSys are fictitious, many of the details and parameters of the case come from real-world software development projects and the literature on software development. The current version of the case incorporates the positive feedback received from students that used this case as part of the master's level IT project management course. Students can improve their understanding on how the organization, project, and team environment influences one's decision to select the appropriate development approach for software projects.

REFERENCES

Beck, K., Beedle, M., Bennekum, A. v., Cockburn, A., Cunningham, W., Fowler, M., . . . Thomas, D. (2001). Manifesto for Agile Software Development. Retrieved from https://agilemanifesto.org/

Boehm, B., & Turner, R. (2003). Using risk to balance agile and plan-driven methods. *Computer*, *36*(6), 57-66. doi:10.1109/MC.2003.1204376

Bose, I. (2008). Lessons Learned from Distributed Agile Software Projects: A Case-Based Analysis. *Communications of the Association for Information Systems*, *23*, 619-632.

Conforto, E. C., & Amaral, D. C. (2016). Agile project management and stage-gate model—A hybrid framework for technology-based companies. *Journal of Engineering and Technology Management*, *40*, 1-14.

Conforto, E. C., Salum, F., Amaral, D. C., da Silva, S. L., & de Almeida, L. F. M. (2014). Can Agile Project Management be Adopted by Industries Other than Software Development? *Project Management Journal*, *45*(3), 21-34.

Cram, W. A. (2019). Agile Development in Practice: Lessons from the Trenches. *Information Systems Management*, *36*(1), 2-14.

Cram, W. A., & Marabelli, M. (2018). Have your cake and eat it too? Simultaneously pursuing the knowledge-sharing benefits of agile and traditional development approaches. *Information & Management*, *55*(3), 322-339.

Hayata, T., & Han, J. (2011, 10-12 July 2011). *A hybrid model for IT project with Scrum.* Paper presented at the Proceedings of 2011 IEEE International Conference on Service Operations, Logistics and Informatics.

Landry, J., & McDaniel, R. (2016). Agile Preparation Within a Traditional Project Management Course *Information Systems Education Journal*, *14*(6), 27-33.

Nevis, E. C., DiBella, A. J., & Gould, J. M. (1998). Understanding Organizations as Learning Systems. In D. A. Klein (Ed.), *The Strategic Management of Intellectual Capital* (pp. 121-139). Boston: Butterworth-Heinemann.

Rigby, D. K., Sutherland, J., & Noble, A. (2018). Agile at Scale. *Harvard Business Review*, *96*(3), 88-96.

Sharp, J. H., & Lang, G. (2018). Agile in Teaching and Learning: Conceptual Framework and Research Agenda. *Journal of Information Systems Education*, *29*(2), 45-51.

Sibona, C., Pourreza, S., & Hill, S. (2018). Origami: An Active Learning Exercise for Scrum Project Management. *Journal of Information Systems Education*, *29*(2), 105-116.

Standish Group. (2015). *CHAOS Manifesto 2015*. Retrieved from

VersionOne. (2020). *The 14th annual State of Agile Report*. Retrieved from Alpharetta, GA: https://stateofagile.com/#ufh-c-7027494-state-of-agile

Vidgen, R., & Wang, X. (2009). Coevolving Systems and the Organization of Agile Software Development. *Information Systems Research*, *20*(3), 355-376. doi:10.1287/isre.1090.0237

Visitacion, M. (2016). *Scaling Agile to the Enterprise*: Forrester Research.