# CHAPTER 4 DISCUSSION, FINDINGS, AND RESULTS

## Introduction

In this chapter, the main conclusions of a research project, or what the project suggested, revealed, or indicated, will be discussed. Presentation findings for this study will be followed by the results and discussion of the findings. In agile methodology, software development is divided into distinct phases that contain activities with the intent of better planning and management The process includes pre-defined deliverables and artifacts that the project team must generate and complete to develop a software project. Most software organizations have implemented the agile methodology process to ease the development process. For the last decade, the long goal is a repeatable and predictable process that can improve productivity and quality. Some programmers have tried to standardize software design work, while others have adopted project management techniques. Software projects can easily be delivered late or over budget if project management is ineffective. Effective project management appears to be lacking in a huge number of software projects that do not reach their goals in terms of functionality, cost, or delivery timeline. Following the study, the findings were discussed simultaneously with the results that were found based on the development of software projects using agile methodology (Albayrak, 2017). Additionally, Agile software development has become a major influence on how software development is conducted. It has become a catch-all word for several improvements in how software engineers organize and coordinate their work, as well as how they communicate with consumers. The agile development method is evaluated as a reaction to plan–based or traditional methods that emphasize a rationale engineering-based approach for extensive planning and addressing challenges of unpredictable values towards software development.

In this study, the theoretical lenses of team adaptation, organizational learning,

and the prior literature on new product development are used to explain the importance

of a team’s ability to process repeated and continuous feedback from the environment. We motivated hypotheses regarding the positive impact of agile methodology use on a multi-dimensional construct representing project success. This construct encompasses the quality of the delivered product, the benefits of the project to the development, and impacts on project management outcomes. In addition, the nature of moderating influences of uncertainty on project success is discussed. The research design for the study utilized a survey that collected responses from 83 agile development teams. Generalized linear modeling was used to test four hypotheses regarding the impact of the extent of agile methodology use on project success, and the moderating influences of uncertainty. It was found that agile methodology use positively impacts project success, while structural complexity negatively moderates the impact of agile use (Nuottila et., 2016). It was also found that environmental dynamism positively moderated the impacts of agile methodologies on project success. Additionally, Agile development methodologies are different from traditional methodologies in several key ways. First, traditional methodologies usually planned to use 33% or more of the project duration in the planning phase. Second, most traditional methodologies are usually planned to deliver the project system at the end of the project. Agile techniques, on the other hand, emphasize the early delivery of business value through an iterative, evolving process. Agile techniques argue that by focusing on early delivery, they can maximize the possibility for business value to be produced while lowering the risk that changes in the environment would limit the system's usefulness before it is delivered.

## 4.2 Analysis Approach

 Because the majority of the variables in this study were either new or modified from their sources, the analysis was exploratory. The tool was first validated using exploratory factor analysis. Following that, the nature of the relationships between the factored structures was looked at. Finally, a generalized linear model was used to test the entire model. The statistics weighed in three factors: environmental feedback, technical feedback, and decreased upfront planning. Agile method utilization was originally projected to contain four aspects. The predicted factor of iterative delivery did not appear in this data set, thus it was removed from the study; however, one of the items that were supposed to load on iterative delivery instead loaded on atmosphere feedback. Although the necessity of well-drafted contracts grows in tandem with the scale of the software project, the developers' and clients' relationships and cooperation are prioritized over tight contracts. Software is developed in small increments and short iterations, which allows identifying risks and problems early on and reacting to them adequately. Each increment brings in new practices to implement and adjusts the previously added practices. Correlation analysis is predicated on the assumptions of linearity in the relationship between the two variables, normality of each variable's distribution, and homoscedasticity across the relationship's range. while running univariate regressions between the quality dependent variable and the other components in the model, after performing numerous assessments of the residuals The assumption of linear relationships between the predictors and the dependent variables was unlikely to hold, according to the homoscedasticity of residuals and a review of enhanced component-plus-residual plots.

## 4.3 Results

 Agile development methodologies have emerged as a major new way of organizing and executing software development projects over the previous decade. Agile practitioners have made numerous claims about the impacts of agile methodologies including associated improvements in team efficiency and performance, higher software quality, and greater organizational benefits. While recent research on agile method use has shown significant results, a number of the normative claims of agile method practitioners have not yet been tested. Software development practices have long been hypothesized to be contingent upon environmental and other factors. This suggests that the performance of a software development technique is a result of interactions between the environment's features and the development method's qualities. Agile practitioners have argued that because the software development process is inherently uncertain and empirical, traditional software methodologies have been unable to successfully develop successful plans of execution, even with the very high initial investment. Instead, agile practitioners have argued that the software development process is much 171 more like the new product development process than an engineering process. Because of this, agile methodologies have adopted practices, techniques, and supporting technologies that are said to enhance delivery success. However, as explained in Chapter 2, many of agile methodologies' normative claims have not been tested, and the methodologies themselves display only partial homogeneity.

 The results show that the extent of the agile method use positively impacts project success. Significant effects were found on the proposed dimensions of project success, including project management metrics, product quality, and perceived organizational impacts. However, the impact of the extent of the agile method use on project success dimensions, on the other hand, was shown to be nonlinear. While the effects of agile use were generally positive, the slope of the effect was considerably lowered at the mean before sharply increasing again. The impact of using the agile method has successfully helped in developing a software project. The results showed that the level of agile method use has a beneficial impact on project success in the majority of models, however, the effects were heterogeneous. Agile impact utilization consistently exhibited positive direct effects on the Quality factor. Improved degrees of agile technique adoption were linked to higher quality in most models, although the impacts were not linear. While the usage of agile methods was consistently substantial, the impacts were quite nonlinear, with some positive and some negative effects occurring even within the same model. All effects were substantial and in the hypothesized direction in the structural complexity tests. The influence on budget was non-significant in the technical complexity model, the effect on time was mixed in direction, and the effect on the scope was as predicted. Finally, the dynamic model had a considerable, but reversed, impact on the budget outcome, while the effect on time and scope outcome was as expected.

## 4.4 Methods

 Agile methods focus on best practices to integrate the development process. It is an approach that defines discipline management of software development projects. Agility recommends an alternative and incremental method to develop software projects although they are popular and successfully applied in software development. With agile methods, development begins before the requirements are well defined. The short iterations of agile development are intended to provide adaptability and developer focus. However, without a detailed planning phase, a fairly accurate estimate of resources and time requirements is virtually impossible, since the developer is enabled to add or remove features during the development process. Agile software development methodologies include Scrum, Extreme Programming (XP), Crystal, Lean Development, Agile UP, and Dynamic Systems Development Method (DSDM). Extreme programming focuses on continuous development and customer delivery that uses intervals to a SCRUM methodology. FDD as an agile method is

one of the agile development methods that manages short incremental iterations which leads to functional software. It is based on a high adaptive software development method that can accept late changes in the software requirements to manage the software development based on the requirement feature list in the business needs. Another agile method for integrating software development is SCRUM that focuses on delivering the highest value in the shortest time.it is known as the most used agile method in the development of software projects due to its simplicity that focuses on the software management issues rather than the technical software development practices which make it widely applicable. However, Scrum is an agile methodology framework it allows the control and management of requirements as well as software development. It is an iterative and incremental basis model for developing software with a defined mechanism, such as a software module that may be iteratively developed in small pieces. Additionally, extreme Programming technique is very helpful when there is constantly changing demands or requirements from the customers or when they are not sure about the functionality of the system. It promotes frequent product releases with short development cycles, which increases the system's efficiency and introduces a checkpoint where any client requirements may be quickly applied.

 Agile software development is distinguishable from the “waterfall” model of sequential software development. Agile development process software, as well as software for tracking and reporting on agile projects, can help developers fully realize the benefits of these methodologies, including flexibility and low cycle times. As a result, there is a demand for user-friendly agile development management software. In terms of the framework employed, agile methodology differs from traditional methodology. The agile technique employs an iterative-incremental approach, with each cycle having its own small system development life cycle. User satisfaction is a major priority, which may be attained by delivering software that is exactly right, not overdeveloped or underdeveloped, early and frequently. Furthermore, because functional software is the essential measure of success in agile methodology, the software developed for each iteration must be functional if this technique is to be used. The functional requirements for each iteration typically grew over time while remaining as simple as possible. Additionally, the use of the practices of agile methodologies is not restricted to agile development projects and has been observed in non-agile methodologies environments. Even so, the previous research literature has usually focused on practices prescribed by a particular agile method. As previously said, the agile technique's internal practices are highly varied and have a wide range of focuses. The requirement for continual input, on the other hand, is universally recognized in agile techniques. Except for Lean/Kanban, every agile method advocates for an iterative delivery cycle. Each development action is considered an iteration in Lean/Kanban, which brings the iterative cycle to its logical conclusion.

## 4.5 Findings

 Based on the findings study on the integration of agile method in developing software projects, the comparison among an old product release developed with traditional methods to a new release developed with agile methods, the developers discovered that 66% increase in productivity for the new agile release compared to the old one. However, in this case, the agile team members had more expertise in software engineering and project management experience than the traditional project team member. A common statement made by the advocates of agile development is, that agile processes lead to better quality in many ways. The arguments in favor of this are frequently related to continuous code testing and integration, rather than doing so after the project as in plan-driven approaches. The data was conducted using a questionnaire and more than 200 respondents on the agile method on the development of software projects as an effective method. This method provides successful refinement and creates a product that is extremely suitable for software development in the software industry due to agile project management iteration and continual feedback. It also decomposes features into smaller increments that require less amount of planning for a short time. Each team member in the project works in every software development for planning, analyzing, designing, developing, testing, and implementing the software development process. However, the findings indicate a significant need for further research on the impacts of agile methodologies and environmental conditions on the successful delivery of software development projects

## 4.6 Discussion

Based on the foregoing, agile methodology has been used in the creation of software projects, requiring developers to collaborate continuously. The selection of this method shows that software development methodologies have the greatest impact when there is a high fit between environmental factors and the practices of the method. Due to the rapid changes requested by software developers, traditional development methodologies such as the waterfall method are no longer used in software project development. As a result, agile methodology is proposed as the best method for how tests are performed and developers adopt this methodology to overcome issues raised by the majority of team members. In an agile environment, the majority of choices are made by the development team, which includes software developers and the client. While the approaches have different enabling practices and different areas of focus, this dissertation explains that they all have a concept of the usefulness of feedback. We conceived agile method utilization at a measurable level across the many agile methodologies in use today, particularly the various feedback processes that are founded on the shared philosophy of the agile manifesto. Therefore, discussion based on approaches of software development such as waterfall, and agile method have some advantages in a real-world work atmosphere of the developers that involves them to equip skills on software project development. The software industry discovered that agile processes are better suited to small, stand-alone projects. Scaling up and integrating agile principles into a company with well-defined traditional processes is difficult for developers and managers. As a result, the industry is looking for a way to combine agile and traditional approaches so that their benefits can be combined.

## 4.7 Conclusions

 Agile methodologies proceed from the philosophy that the software development effort must consistently and efficiently deal with change. These techniques prescribe intricate networks of processes, practices, and procedures, as well as supporting technology, all of which are intended to improve software delivery in unpredictable settings. While these methodologies are extremely heterogeneous in their defined practices, practices that are common to all of the methodologies are those that are designed to elicit feedback cues from the environment. It is only through processing these environmental cues that software development teams can develop adaptability and agility. Numerous claims have been made about the impacts of the agile method use on project success and the environmental conditions for which the methodologies are most well-suited. Agile method use is expected to have the highest impacts in environments that are highly dynamic. Additionally, software development is the backbone of today’s digital economy. Additionally, the findings of this study emphasize the need for feedback in dealing with the unpredictable nature of software development. However, the findings suggest that more research is needed into the effects of agile approaches and environmental factors on the effective delivery of software development projects. In this research, many types and methodologies for agile software development have been explained whereby each methodology has its advantages and disadvantages, so there is no optimal methodology for all types of projects, each project has its specifications, characteristics, and needs to be done. Therefore, selecting the best agile methodology to be used in the project development must be done carefully based on these variabilities, or sometimes there is no agile methodology that can be used on some projects development so the traditional methods can be optimal for these cases, such as the organizations with a large number of teams and employees, and projects with a critical huge budget.

 By concluding, we can say that the agile methodology is an alternative to plan-driven methodology (eg: UML or Waterfall Model) in the development of quality software. Because developers do not frequently focus on models as the main component of the final software, it is also ideal for small to medium-sized projects. Agile approaches feature a distinct approach to each software engineering phase that focuses on feedback and change. The research has shown that traditional plan-driven software development methodologies are not used in practice. It has been argued that the traditional methodologies are too mechanistic to be used in detail. As a result, industrial software developers have become skeptical about "new" solutions that are difficult to grasp and thus remain unused. Agile software development methods, "officially" started with the publication of the agile manifesto, attempt to bring about a paradigm shift in the field of software engineering. Agile methods claim to place more emphasis on people, interaction, working software, customer collaboration, and change, rather than on processes, tools, contracts, and plans.

## 4.8 Recommendations

 It is important to note that before applying the agile methodology to software project development, developers need to undergo training to help them to acquire key agile project management competencies. This project recommends agile methodology as the effective method in the development of software projects. Traditional techniques, such as the waterfall methodology, were created to deliver software after a project. As a result, developers did not engage with the program until after the project was completed. Because of its focus on the early delivery of working software, the agile methodology allowed mismatches between created software and developers to be discovered earlier in the project. When a result of this, agile methodologies may be better equipped to adjust to meet requirements that emerge only as developers interact with the real system. For this reason, agile methodology has been proposed by developers as an effective method that can provide a positive impact on software project performance. Due to its iterative delivery paradigm, agile methodology practitioners have given a positive recommendation that agile methodologies deliver better and more useful software. This increased delivery success is claimed to reduce risk and provide a better return on investment. Agile methods proponents argue that the approach should be implemented in stages, with the major pain points of a team being addressed first. As a result, early in the usage cycle, agile techniques are likely to yield quick results, as the practices that are most likely to improve performance are deployed first. This result is intuitive when considering the recommendations of agile

developers. Additionally, the Agile method use was originally predicted to include four dimensions, however, the data factored into three factors, environmental feedback, technical feedback, and reduced upfront planning. The theorized factor of iterative delivery did not materialize in this data set and was dropped from the analysis, although one of the items that were anticipated to load on iterative delivery loaded instead on environmental feedback. Agile developers, on the other hand, emphasize that, due to the mutually reinforcing nature of agile practices, the method's full performance benefit is only realized when the majority of the practices are implemented. Because the interaction of these techniques is complicated, teams must learn how to use them in order to be successful. This means that, after experiencing the immediate early performance impacts, teams will continue to use agile principles in the hopes of further performance improvements.

# CHAPTER 5: RESEARCH IMPLICATIONS, RECOMMENDATIONS, LIMITATIONS, AND LESSONS LEARNED

## Introduction

In this chapter, it provides an overview of the research implications, recommendations, lessons learned, challenges encountered among others. Additionally, this section provides reiteration of the problem and purpose statements, research questions with reference to results and findings. The primary goal of this research was to perform a comprehensive analysis. Based on the problem statement relating to integration of Agile Methodology in the Development of Software Project, Traditional software development methodologies have been coupled with several problems. Ideally, most companies focus on delivering quality and gaining customer satisfaction with the aim of implementing their projects. The key problem with the traditional project management and software development methodologies are associated the complexity of implementation process. The challenge lies on choosing between the traditional development methodologies and agile development methodologies. Though both approaches have positive and negative sides. Making the right choices play a crucial role while starting a new project. The main points to consider while choosing the methodology has been a major challenge while choosing the software development methodology. Most of the organizations have been found themselves in challenges especially mapping their business needs-which include mapping on the impacts of implementing specified requirements on customer business. Customer perception and perspective of business impact the implementation of project and this delays the delivery of such projects. The application of agile methodology requires the problem to be well defined in advance. The solution to be offered by agile also needs to be well determined in advance. Though with agile methodology there is flexibility, the problem lies on the integration of agile methodology in the development of software project. Agile projects come with several challenges that are distinct from those faced by projects following traditional methodology. Based on the research done, there are various issues affecting the implementation of agile projects in the software development process. Such problems include issues with communicating, management of day today operational problems, customers and team members, changing culture and mindset, among other challenges. Notably, selecting the right methodology and matching the methodology and project needs has been one of the key challenges as far as project management is concerned.

Agile methodology is considered ideal when the product details cannot be defined or agreed in advance with any degree of accuracy. This issue requires a collaborative environment existing between the user or clients and the developer. One of the key challenges associated with agile methodology is based on the availability and popularity of the methodology is critical, customer availability, and organizational culture. In line with Agile manifesto affirms that project managers must deliver working software frequently. Based on the above problems associated with the development of software projects, the proposed solution is based on the integration of agile methodology in the development of the software project. With Agile projects, the idea is to implement software projects incrementally and through iterative approach to development. Integration of agile in the software projects comprises of various stages which include project inception, planning, requirements refinement, product backlog, testing, and transition. The solution realized in the implementation process of agile approach focus on time, cost, quality, and features, which are variable in nature unlike in the traditional methodology that are used in software development. Lastly, the solution derived from the integration of agile methodology in the software development projects is based on identifying factors which makes the transition from traditional to agile methodology necessary. Additionally, this solution can be realized whether the software project is small or big.

## Effectiveness of Agile Methodologies in Software Projects

Notably, agile software development methodologies have emerged to be of great significance in the past few decades. Agile professionals have made several claims relating to the impacts on the application of Agile methodologies including the related improvements in team efficiency as well as performance and software quality. Where recent research on the application of agile methodology have show considerable results, various normative claims by agile methodology practitioners have not yet been tested vividly. Additionally, software development practices have been hypothesized for quite long due to the contingent and environmental factors. This implies that the performance of software development methodology is as a result of function between environmental factors and other factors. Agile software development professionals have argued that since software development process is fundamentally uncertain and empirical, traditional software development methodologies have not been able to develop plans of project execution in a successful manner. In this dissertation, we elaborated that, while agile methodologies have adopted practices and practices that have been significant in software development, it has become necessary to have different point of focus in the realization of the required results. Various studies have shown agile teams executing performing specific agile methods. Implementation of agile methods in software development is mostly based on the application of agile manifesto. Agile methodology, projects are broken down into small portions, which are finished in work- related phases that operates from the design phase, testing to quality-assurance phase (Špundak 2014). The agile development methodology is also referred to as Scrum. Application of Agile method is mostly where rabid feedback is required. The agile method mostly involves four phases, which include Scanning the emerging trends as well as the risks. Analyzing the information led decisions and solutions. The third is responding to the opportunities and risks. The fourth phase is change whereby it involves shaping the future environments.

The results shows that the degree of agile methodology application positively affects the success of project. Significant implications were found on the proposed dimensions of the project success, which include project management measurements, product quality, and organizational impacts. Nevertheless, the effects of the impact of the application of agile method on the project success was found to be non-linear. While the application agile in this research was found to have a positive correlation on software projects. Agile methodologies promote on the application should be adopted. The subduing implication of the technical complexity on the implication to the extent of agile methodology application. As computing technologies become ubiquitous, most of the industries are shifting to agile project management approaches in implementing their IT infrastructure. Automotive industries have continued to expand tremendously as they reduce the cost of enhancing the efficiency as they continue making profits. According to Datrika (2018), the factors that have continued to change in the automotive industry outsourcing for product development, development of specific vehicle software, demands from the customers to improve the reliability of the products. As a result of this, organizations have continued to face several challenges in managing the above factors. Some of these challenges include handling interconnected as well as complex industry. Most people buying automotive products need their vehicles to be equipped with modern technologies.

Agile project methodology an iterative approach that is essential in guiding and project processes. The method is known to be modern and flexible to project management, and it facilitates breaking down a large into manageable tasks that are handled into short iterations. The rationale behind breaking down the project into manageable tasks is to allow the project management team is to adapt quickly to change as well as to deliver the work fast. The idea of and the application of agile was initially coined by manufacturing and automotive industry, which is a software that essentially adopted agile methodology principles and practices. Nayak & Padhye (2017) affirms that over the past few decades, the automotive industry has been growing and evolving, and this growth has prompted to change. The primary focus of the customers is to acquire products that are standardized and satisfy their individual requirements. The application of agile project methodology in the automotive industry has also led to the increased production capabilities. Nevertheless, automotive manufacturers are experiencing stiff competition from their competitors to provide modern technologies in manufacturing their products to their esteemed customers. According to Vasylkov (2019), with the adoption of agile methodology, automotive manufacturers can reduce time to market their products, increased cost efficiency, manufacturing of the product's base on the customer’s needs. Other benefits include improved product quality, handling complexity, and enhancement of production efficiency as well as the performance. It is good to note that that the technological developments in the automotive industry are in line with the application of agile methodology in the manufacturing of vehicles especially when it comes to programming and implementation of software modules. Digital operations and the innovation-driven in automotive industries have enhanced stakes, especially in their processes and value chain. The industry is redefining strategies on what a vehicle can and should do Gauger et al. (2019). The transition from the cars with the combustion engine to electric vehicles is enough to rock the industry through the innovation, whereby software has become the game-changer. As a result of this, there have been increased production capacities by the companies that are in the automotive industry. The rationale behind using agile methodology is because the automotive sector has become increasingly driven by innovation and software. In this case, agile delivers several advantages, as mentioned earlier in this paper.

One of the ways is by delivering like a software company. The industry has grasped functional excellence to develop high-quality products at cheaper costs and within reasonable short periods. The technology-driven revolution in this industry has been changing the rules whereby automakers are expected to be much more dexterous. The application of the agile methodology in the manufacturing of vehicles involves embarking on three primary pillars. The first one is launching three pilots at the corporate level to improve the speed of decision making dramatically. (Gauger et al., 2019). Such pilots focus on optimizing the customer journey from the point of purchase and after purchase, mastering the data-driven marketing and minimizing the dependence of external providers, and lastly is resolving customers' issues quickly. The second key pillar is restructuring the overall organization to facilitate agility. This mostly involves organizing teams to focus on product development. The third and the last primary pillar in the application of the agile methodology in the automotive industry is making upfront planning and gaining momentum with agile ways of working

## Research Implications

Development of Agile Information Systems In the last decade, methodologies have evolved as a new way of managing work and delivering information systems development teams, with a substantial number of companies reporting the results. Agile approaches are being adopted and used. Only a handful of these assertions, however, have been tested in the academic literature to yet. Extreme programming, Scrum, and other agile approaches prescribe a wide range of practices, some of which are incompatible. Furthermore, the application of agile methods techniques is not limited to agile development projects; it has been used in non-agile methodologies situations. Nonetheless, prior research has tended to focus on procedures that have been prescribed. The biggest distinguishing feature of agile techniques, according to this dissertation, is their considerable emphasis on gathering and processing feedback from the outside world. The influence of agile approaches as evidenced by the utilization of these feedback processes is the subject of this dissertation. The importance of a team's ability to process repetitive and continuous feedback from the environment is explained using the theoretical lenses of team adaptation, organizational learning, and earlier literature on new product development in this study. Future studies in information systems should focus on the effects of agile techniques. While this research was mostly exploratory in nature, it did open up a number of new avenues for future investigation. This study contributes to the research stream by providing evidence that more development and pragmatic research is needed.

In terms of software development approaches, practitioners are still looking for "silver bullets." However, as previous research and this study reveal, the employment of techniques labeled as agile methodologies has a variety of consequences is conditional on the environment's suitability. As previously stated, software development methodologies have the greatest impact when there is high with reference to the environmental factors and practice of methods. This was found to be the case in this research. This study backs up previous findings that show that structural complexity has a negative influence on teams in general. When teams function in the midst of increasing degrees of structural complexity, the consequences of the extent of agile method use have a considerable detrimental influence, according to this research. When implementing agile teams, IT professionals should be cautious to keep the structural complexity to a minimum which may have an implication on the teams. Where it is possible, project teams ought to be collocated and the number of stakeholders as well as their reporting relationships needs to be minimized. This study also discovered that the impact of technological complexity on success is very non-linear. There were unfavorable relationships between agile application and success at both low and high levels of technological complexity. This runs counter to earlier practitioner comments that agile approaches are best suited for less technically challenging tasks.

One key reason for this finding could be because if a project is tremendously basic, people's expectations of its success and consequences are likely to be low. Contrarywise, the technical feedback mechanisms may be overwhelmed by a high level of technological implications and complexity a collection of agile teams If the technological effect is hampered by high system complexity, technical feedback, the negative effects of technical feedback would be reduced. For example, if high levels of integration of agile methodology in software projects is implemented with external systems, an agile development team development team may not be able to write and complete tests that are meant to fully tests the system. Based on the findings, the existence of high levels of vitality is the single most important component of uncertainty that would imply a strong fit for the adoption of agile approaches. The capacity to recognize and respond to change has been demonstrated to improve performance in various situations (Tripp, 2016). The effects of agile method ado in situations with lesser degrees of dynamism, on the other hand, are less noticeable. As a crucial pointer of possible high performance for agile methodology utilization, organizations need to assess the level to which the team should be able to adjust to change.

Based on the implications of future research, more empirical and research is needed especially on the integration of agile methodologies to software projects. Further, further empirical and theoretical research on agile methodology needs to be undertaken in a comprehensive manner. The rationale behind this is to adopt a longitudinal research designed to facilitate the researchers investigate the impacts of agile methodologies in software projects over time. Additionally, this would be paramount in understanding the implication of agile methodologies and their uncertainties, Moreover, the non-linear nature of the data 7obtained in this research over time may mean the existence of a recursive process. The implication of the application of experimental research would be of great essence on a cyclical impact on agile method over time. Secondly, with the application of nomological network with the aim of motivating the hypothesis, was notable that network on its own wasn’t tested in this research. With the clear understanding of this network, it would be 7of great significance and interest when furthering this research. Furthermore, the proposed nomological network might be applied to any software development methods. This concept of a nomological network of software development methods components has never been tried before. With proper testing of this concept of agile methods in software projects, it would be of great significance in the field of information systems.

Thirdly, the correlations between different constructs would be contrary to the prior research that has made a proposal on linear negative relationships existing between project complexity and its success and linear positive correlation that exist between complexity and project success. With these findings, they show an indication on the essence of the establishment of new theory. A considerable theoretical has recognized the application of non-linearity in the literature of organization behavior. Nevertheless, information systems theories have though with a few exceptions supported the proposal of establishments of linear relationships and different studies have designed measurements as well as tests that makes assumption on the linear correlation. With such results, they add to previous call for the development pf theories relating to non-linear effects.

## Limitations of this Study

As a result of conceptualization of a comprehensive extent relating to integration of agile methodology application construct, its impact of agile on diverse dimensions of the success of the project has not yet been tested comprehensively. The first identified limitation in this study on theorizing impacts of application of agile methods in software projects which was motivated by team adaptability perspective and organization learning. Such theories are integrally reinforcing due to their cyclical nature. However, this research study was undertaken based on cross-sectional research design that show the state of theoretical model based on particular point in time. Secondly, based on the sample assessed in this study, it comprised of agile teams and was a purposive sample. With this, it created intentional sampling bias through inclusion in the research in the research where only the teams that self-identified as application of agile methodologies. Nevertheless, such bias required as the concepts and constructs relating to agile teams’ process and practices that would not be possible to be interpreted by non-agile professionals. With this, it appears that there is a limitation in terms of the generalization of the study and does not show the applicability of the model of the study or conclusions made outside agile teams

Another limitation relates to the study, which sought to include the constructs and variables that were identified through the review of literature, Ideally, those variables implemented were half the number of variables identifies. Hence, imposing a limitation of this study. This further imposed limitation on the respondents interviewed who were not subjected to all research variables identified in this research. The specific nature of interactions with the client and the frequency as well as the quality of interactions with them was identified to be of extreme importance by the two out of the five focus groups. Ideally, to the extent, that the research model did not consider such variables, we must remain cautious in the application in the research findings

## Conclusion

Integration of Agile Methodologies in software projects advance from the premise that software development effort that should efficiently and consistently deal with change in the software projects. Ideally, such approaches are prescribed to complex network of processes, practices, and procedures. And supporting technologies that are in a similar way are meant to bring a positive implication in the delivery of the software projects within uncertain environments. Such technologies are known to be heterogeneous in nature based on their defined practices. Various claims and assumptions have been made about the application of agile methodology in software projects and their success rate. Agile methodology application in software projects is anticipated to be the most effective in case of low structural complexity barriers. Notably, the results of this research and investigation outlines the nature and significance of feedback for further research on the integration of agile methodologies in software projects and environmental conditions of for the successful completion and delivery of software projects. Additionally, the application of agile project methodologies in software development is paramount and cannot be underestimated. Lastly, with the adoption of agile methodology, automotive manufacturers can reduce time to market their products, increased cost efficiency, development of the products based on the customer’s needs, among other benefits.

## References

Abrahamsson, P., Oza, N., & Siponen, M. T. (2010). Agile software development methods: A comparative review. *Agile software development*, 31-59.

Albayrak, A. (2017). Restructuring of System Analysis and Design Course with Agile Approach for Computer Engineering/Programming Departments. *INTERNATIONAL JOURNAL OF ADVANCED COMPUTER SCIENCE AND APPLICATIONS*, *8*(11), 84-89.

Beerbaum, D. (2021). Applying Agile Methodology to regulatory compliance projects in the financial industry: A case study research. *Applying Agile Methodology to Regulatory Compliance Projects in the Financial Industry: A Case Study Research (April 26, 2021)*.

Boehm, B., & Turner, R. (2005). Management challenges to implementing agile processes in traditional development organizations. *IEEE software*, *22*(5), 30-39.

Datrika, N. K. (2018). Implementing Agile Methodology Techniques in Automobile Industry.

Flora, H. K., Chande, S. V., & Wang, X. (2014). Adopting an agile approach for the development of mobile applications. *International Journal of Computer Applications*, *94*(17), 43-50.

Gauger, C., Heller, K., Tordjman, K., Loh, A., & Rehberg, B. (2019). An Agile Game Plan for Automakers. Retrieved from https://www.bcg.com/publications/2019/agile-game-plan-automakers.aspx

Hass, K. B. (2007). The blending of traditional and agile project management. *PM world today*, *9*(5), 1-8.

Highsmith, J. (2009). *Agile project management: creating innovative products*. Pearson education.

Kaleel, S. B., & Harishankar, S. (2013). Applying agile methodology in mobile software engineering: Android application development and its challenges. *Computer Science Technical Reports*, 1-11.

Katumba, B., & Knauss, E. (2014, December). Agile development in automotive software development: Challenges and opportunities. In *International Conference on Product-Focused Software Process Improvement* (pp. 33-47). Springer, Cham.

Kumar, G., & Bhatia, P. K. (2012). Impact of agile methodology on software development process. *International Journal of Computer Technology and Electronics Engineering (IJCTEE)*, *2*(4), 46-50.

Lufthansa Industry Solutions. (n.d). Using Agile Methods to Become More Competitive. Retrieved from https://www.lufthansa-industry-solutions.com/de-en/solutions-products/automotive/automotive-more-competitive-with-agile-methods/

Nayak, R., & Padhye, R. (Eds.). (2017). *Automation in garment manufacturing*. Woodhead Publishing.

Nuottila, J., Aaltonen, K., & Kujala, J. (2016). Challenges of adopting agile methods in a public organization. *International Journal of Information Systems and Project Management*, *4*(3), 65-85.

Opelt, A., Gloger, B., Pfarl, W., & Mittermayr, R. (2013). *Agile contracts: creating and managing successful projects with Scrum*. Wiley Publishing.

Rahimian, V., & Ramsin, R. (2008, June). Designing an agile methodology for mobile software development: A hybrid method engineering approach. In *2008 Second International Conference on Research Challenges in Information Science* (pp. 337-342). IEEE.

Šmite, D., Moe, N. B., & Ågerfalk, P. J. (Eds.). (2010). *Agility across time and space: Implementing agile methods in global software projects*. Springer Science & Business Media.

Socha, D., Folsom, T. C., & Justice, J. (2013). Applying agile software principles and practices for fast automotive development. In *Proceedings of the FISITA 2012 World Automotive Congress* (pp. 1033-1045). Springer, Berlin, Heidelberg.

Špundak, M. (2014). Mixed agile/traditional project management methodology–reality or illusion? *Procedia-Social and Behavioural Sciences*, *119*, 939-948.

Stamelos, I. G., & Sfetsos, P. (Eds.). (2007). *Agile software development quality assurance*. Igi Global.

Tipnis, P. (2018). Common Challenges Faced by First-Time Agile Organizations. Retrieved from https://www.knowledgehut.com/blog/agile/common-challenges-faced-by-first-time-agile-organizations

Tripp, J. F. (2012). *The impacts of agile development methodology use on project success: A contingency view* (p. 217). Michigan State University. Business Information Systems.

Turk, D., France, R., & Rumpe, B. (2002, May). Limitations of agile software processes. In *Proceedings of the Third International Conference on eXtreme Programming and Agile Processes in Software Engineering* (pp. 43-46).

Vasylkov, O. (2019). How the Automotive Industry Can Benefit from Agile Software Development. Retrieved from https://clutch.co/developers/resources/how-automotive-industry-benefit-agile-software-development.

Wysocki, R. K. (2011). *Effective project management: traditional, agile, extreme*. John Wiley & Sons.