Agile Project Management and Organisational Performance: A Study of Systemspec Holdings Limited

Olota Oluwayomi Omotayo (Ph.D)
University of Ilorin, Nigeria
olota.oo@unilorin.edu.ng

Ayedebinu James Adegoke
University of Ilorin, Nigeria
jamesayedebinu@gmail.com

Popoola Oluwadamilare James
University of Ilorin, Nigeria
popoolaojames@gmail.com

Adetoro Promise Iranlowo
University of Ilorin, Nigeria
adetoropromise@gmail.com

DOI: 10.56201/ijssmr.vol.12no7.2026.pg84.123

Abstract
This study examined the impact of agile project management on organisational performance at Systemspec Holdings Limited, a Nigerian information technology and software firm. Despite the growing adoption of agile methodologies across the technology sector, empirical evidence grounding their performance impact in specific African IT contexts remains limited. The study focused on three agile project management factors — lead time, team morale, and bugs and defects — and assessed their influence on corresponding organisational performance outcomes: customer satisfaction, employee turnover rate, and software quality respectively. A quantitative research design was adopted, and data were collected through structured questionnaires administered to 174 employees, of which 138 valid responses were returned, yielding a response rate of 79.3%. Data were analysed using Partial Least Square Structural Equation Modelling (PLS-SEM) via SmartPLS 3.3.3. The findings revealed that lead time had a significant positive effect on customer satisfaction ($\beta = 0.488$, $p = 0.000$); team morale had a significant positive influence on employee turnover rate ($\beta = 0.646$, $p = 0.000$); and bugs and defects had a significant negative effect on software quality ($\beta = 0.763$, $p = 0.000$). All three null hypotheses were rejected. The study concludes that agile project

management, when practiced thoughtfully, supports better organisational performance across key dimensions. Recommendations are made for improved lead time management, investment in team morale, and stronger defect management practices.

Keywords: *Agile Project Management, Organisational Performance, Lead Time, Team Morale, Bugs and Defects, Software Quality, PLS-SEM, Nigeria*

Introduction

1.1 Background to the study

Information technology and software organisations operate in an environment defined by rapid change. Technological breakthroughs emerge faster than traditional management frameworks can absorb them, and customer expectations shift with equal speed. For decades, many IT organisations relied on structured, sequential approaches like the Waterfall model, which offered a sense of control through upfront planning and detailed documentation (Smith, 2018). In stable environments, this worked reasonably well. In the fast-moving world of software development, it frequently did not. Projects ran late, market windows were missed, and the rigidity of the process left little room for the kind of creative adaptation that the industry demands (Jones, 2019). The Agile Manifesto, introduced in 2001, marked a turning point. It reoriented project management around people, collaboration, and the continuous delivery of working software rather than adherence to a fixed plan. Methodologies like Scrum and Kanban gave these principles a practical structure, breaking projects into short development cycles that allowed teams to deliver value incrementally, respond to feedback quickly, and course-correct without derailing the entire project (Leffingwell, 2018). Cross-functional teams replaced traditional hierarchies, decision-making became faster, and the emphasis shifted firmly toward what customers actually needed rather than what had been planned months in advance. The organisational benefits of this shift have extended well beyond faster delivery. Agile adoption has been associated with improved efficiency, higher productivity, stronger customer satisfaction, and greater overall business performance (Anderson, 2010). The approach has become less a methodology choice and more a strategic necessity for IT firms competing in markets where disruption is constant and adaptability determines survival.

1.2 Statement of the Research Problem

Despite the widespread adoption of agile project management in the IT sector, important questions remain about how and to what extent it actually improves organisational performance in practice. The promise of agile — faster delivery, better quality, higher customer satisfaction — is well documented in theory, but empirical evidence grounded in specific organisational contexts is still relatively limited (Anderson, 2010).

Balancing innovation with operational efficiency remains a genuine tension. IT organisations are under pressure to continuously develop new solutions while managing

costs, timelines, and resources responsibly. Agile frameworks offer a path through this tension, but their effectiveness depends heavily on how well they are implemented, and on factors such as team capability, leadership support, and organisational culture (Leffingwell, 2018). These contextual variables are rarely examined in depth, particularly within African IT organisations where the operating environment differs meaningfully from the Western contexts where most agile research has been conducted.

This study therefore investigated the impact of agile project management on organisational performance at Systemspec Holdings Limited, a Nigerian IT firm. Specifically, it examined how lead time affects customer satisfaction, how team morale influences employee turnover, and how bugs and defects affect software quality. The aim was to generate empirically grounded insights that are relevant both to Systemspec Holdings Limited and to the broader community of IT organisations operating in similar contexts.

1.3 Research Questions

In attempting to assess the impact of agile project management methodologies on organization performance, the following research questions are pertinent:

i. To what extent does lead time affect customer satisfaction?
ii. To what extent does team morale influences employee turnover rate?
iii. To what extent does bugs and defects affect software quality?

1.4 Research Objectives

The main objective of this study is to examine the impact of agile project management on organization performance in Systemspec Holdings Limited. Other objectives are to:

i. assess the impact of lead time on customer satisfaction;
ii. evaluate the influence of team morale on employee turnover rate; and
iii. investigate the effect of bugs and defects on software quality.

1.5 Research Hypotheses

To provide empirical answers to the research questions above, the following hypotheses were developed. They are stated below:

Ho_1: Lead time does not significantly impact customer satisfaction.
Ho_2: Team morale does not significantly influence employee turnover rate.
Ho_3: Bugs and defect does not significantly affect software quality.

1.6 Significance of the Study

This study holds relevance at both the organisational and industry levels. For Systemspec Holdings Limited, the findings provided concrete, evidence-based insights into how agile project management shapes key performance outcomes, equipping leadership with a clearer basis for strategic decisions about the continued use and refinement of agile practices within the organisation. Beyond the single organisation, the study contributes to

the growing body of empirical literature on agile project management in African IT contexts, where such evidence remains sparse. Most existing research has been conducted in Western settings, and findings do not always translate directly to organisations operating under different economic, structural, and cultural conditions. By grounding the analysis in a Nigerian IT firm, this study offers a perspective that other organisations in similar environments can learn from and apply.

For practitioners and managers considering agile adoption, the study identified which specific factors — lead time, team morale, and defect management — most meaningfully drive customer satisfaction, staff retention, and software quality. These findings offer practical reference points for organisations designing or refining their own agile implementation strategies.

1.7 Scope of the Study

This study focused exclusively on Systemspec Holdings Limited, a Nigerian information technology and software company, examining how agile project management influenced organisational performance within that specific context. Three performance variables were investigated: customer satisfaction, employee turnover rate, and software quality. Data collection covered a ten-year period from 2012 to 2022, reflecting the organisation's active experience with agile practices. Systemspec Holdings Limited was selected for several reasons. It operates in a sector defined by rapid technological change and shifting customer demands, making it a relevant setting for studying agile project management. It also held a decade of historical data across the key variables of interest, enabling a sufficiently deep analysis. The study's conclusions are therefore most directly applicable to Systemspec Holdings Limited, though the insights drawn carry lessons for IT organisations operating in comparable contexts.

Literature review

2.1.1 Concept of Agile Project Management

Agile project management originated in the software development domain as an adaptable and iterative approach to managing projects. At its core, it prioritises collaboration, continuous improvement, and the ability to respond to changing requirements and customer needs. The Agile Manifesto, introduced in 2001, laid the foundation for this approach by establishing key values such as placing individuals and interactions above rigid processes, delivering working software over comprehensive documentation, collaborating closely with customers, and staying open to change throughout a project.

One of the things that makes agile project management work well in practice is how it encourages frequent communication among cross-functional teams. Rather than waiting until the end of a long project to show results, teams deliver working software in short cycles, keeping stakeholders involved and informed along the way.

Several widely used methodologies bring these principles to life. Scrum is perhaps the

most popular, particularly for complex projects. Schwaber and Sutherland (2017) described it as a lightweight framework that helps teams tackle difficult, adaptive problems while still delivering high-value products consistently. Work in Scrum is organised into sprints, which are short, fixed-length development cycles typically lasting between one and four weeks. During each sprint, teams plan, build, test, and deliver a product increment that is ready or close to ready for use (Cervone, 2017).

The product owner plays a central role in keeping this process on track. This person is responsible for making sure the team is always working on the features that matter most to customers and to the business (Schwaber and Sutherland, 2017). Alongside this, agile teams are expected to be self-organising, meaning they take collective responsibility for their work rather than waiting for top-down direction. This sense of shared ownership tends to produce better outcomes and keeps motivation high.

The Project Management Institute (PMI) captures the essence of agile neatly, defining it as "a value-driven, iterative approach for managing projects, with an emphasis on delivering products incrementally to enable more flexibility and a faster response to change" (PMI, 2021). What this looks like in practice can be broken down into a series of steps that form the agile project cycle. It begins with establishing a **Project Vision and Product Backlog**. The project vision gives the team a clear sense of purpose and direction, while the product backlog serves as a living, prioritised list of everything the project needs to deliver, from new features and user stories to bug fixes and technical tasks.

Next comes **Release Planning**, where the whole team, including product owners and relevant stakeholders, comes together to agree on priorities and map out how they will achieve the project goals within a given release period.

Sprint Planning follows, where the team looks at the product backlog and decides what can realistically be achieved in the coming sprint. Tasks are broken down into manageable pieces so that every team member knows what they are responsible for.

During **Sprint Execution**, the team gets to work. Time-boxed iterations keep everyone focused, and team members collaborate daily to complete the agreed tasks and move toward the sprint goal.

Daily Stand-up Meetings keep the team aligned throughout the sprint. These are brief, structured check-ins where members share what they have done, what they plan to do next, and whether anything is blocking their progress. By the end of each sprint, the team aims to have a **Potentially Shippable Product Increment**, meaning a piece of the product that has been built, tested, and is ready to be reviewed or released. The **Sprint Review** is where this increment gets presented to stakeholders. Feedback gathered here directly informs what the team prioritises next, keeping the product closely aligned with real user needs. Equally important is the **Sprint Retrospective**, which turns the lens inward. The team reflects on how they worked together, what went well, and what could be improved. This habit of honest reflection is one of the reasons agile teams tend to get better over time.

This cycle is built around **Incremental Delivery**, the idea that value should be released steadily throughout a project rather than all at once at the end. Stakeholders get to see and benefit from progress early, and the team can adjust based on what they learn.

Running alongside all of this is **Backlog Refinement**, a regular activity where the team revisits and updates the product backlog to reflect changing requirements, new feedback, or shifting priorities. Items near the top of the backlog get refined in detail so they are ready to be picked up in future sprints.

Finally, the principle of **Continuous Feedback and Adaptation** ties everything together. Agile teams actively seek input from users and stakeholders and use it to improve with every iteration. Rather than following a fixed plan from start to finish, the approach is to stay responsive and keep learning throughout the life of the project.

2.1.2 Benefits of Agile Project Management

Layton (2012) identified several benefits that agile project management brings to organisations and teams. First, agile promotes **better product quality** by building quality checks into the process rather than treating them as an afterthought. Requirements are refined progressively, daily testing catches issues early, and continuous integration ensures that problems are resolved while the work is still fresh in the team's mind.

Second, **higher customer satisfaction** is achieved by keeping customers actively involved throughout the project. A knowledgeable product owner, a well-maintained backlog, frequent releases, and working demonstrations during sprint reviews all ensure that what gets built genuinely reflects what customers need. Third, agile fosters **higher team morale**. Self-managing teams give members real ownership over their work, encouraging creativity and initiative. The scrum master supports this by removing obstacles and protecting the team from unnecessary external disruption, while cross-functional working creates opportunities for continuous learning and growth. Fourth, **increased collaboration and ownership** are built into agile structure through daily stand-ups, sprint reviews, and close interaction between the development team, product owner, and stakeholders. These touchpoints keep everyone aligned and ensure that feedback shapes the product throughout rather than only at the end.

Fifth, agile allows for **customised team structures**. Small, self-managing teams of five to nine members are easier to coordinate, and larger projects can simply run multiple scrum teams in parallel, with each team retaining the autonomy to organise its own work.

Sixth, agile produces **more relevant metrics**. Estimates are made by the people actually doing the work, revised as understanding deepens, and tracked through tools like the daily burndown chart, giving a far more accurate picture of progress than traditional plan-based reporting. Finally, agile offers **improved performance visibility and greater project control**. Because progress is tracked openly and reviewed regularly, no one is left guessing how a project is going. The frequent opportunities for inspection and adaptation across all levels of the team mean that course corrections happen early and often, resulting in a final product that is much more likely to meet its goals.

2.1.3 Agile Manifesto

The Agile Manifesto, introduced in 2001, provides the foundational values and principles that guide agile software development. Rather than reproducing all twelve principles in full, what follows is a summary of the core ideas they represent, as these are what shape agile practice in organisations.

At the heart of the manifesto is a commitment to delivering working software early and often, rather than waiting until a project is complete. Teams are expected to welcome changing requirements at any stage, recognising that customer needs and market conditions shift over time and that adapting to this is a strength, not a disruption. Frequent delivery in short iterations keeps stakeholders engaged and creates regular opportunities to gather feedback and improve. The manifesto also places great emphasis on people and communication. Business representatives and developers are expected to work closely together throughout a project, not just at the beginning or end. Face-to-face conversation is regarded as the most effective way to share information, and teams are encouraged to be motivated, trusted, and given the support they need to do their best work. Progress is measured by working software rather than documentation or plans, keeping the focus on outcomes that are tangible and useful.

Technical discipline is equally valued. Good design, clean code, and sound architecture are treated as ongoing responsibilities rather than one-time activities, because they are what allow a team to keep moving quickly as a project evolves. Simplicity, in the sense of avoiding unnecessary work and complexity, is seen as essential to maintaining that pace. Self-organising teams are trusted to make the best technical decisions because they are closest to the work. Finally, the manifesto calls for regular reflection. Teams are expected to periodically step back, assess how they are working, and make adjustments. This habit of continuous improvement is what allows agile teams to grow stronger over time rather than simply repeating the same patterns. Underpinning all of this are five core values that run through agile methodologies: a customer-centric focus that keeps delivery aligned with real needs; iterative and incremental development that allows for regular course correction; cross-functional teams that bring diverse skills together; adaptability that allows teams to respond to change without losing momentum; and a commitment to continuous improvement at every level of the team and its processes (Schwaber and Sutherland, 2017).

2.1.4 Agile Methodologies and Frameworks

Several agile methodologies and frameworks are available to IT organisations seeking to improve how they manage and deliver projects. This study draws on five of the most widely recognised, namely Scrum, Kanban, Extreme Programming, Lean, and the Scaled Agile Framework (Kiguolis and Linas, 2023). **Scrum** is arguably the most widely adopted agile framework in software development (Schwaber and Sutherland, 2017). It organises work into short, fixed cycles called sprints, typically lasting two to four weeks, within which a team plans, builds, reviews, and improves. The process begins with sprint

planning, where the team selects priority items from the product backlog and agrees on what can realistically be delivered. Throughout the sprint, daily stand-up meetings keep everyone aligned and surface any obstacles quickly. At the end of each sprint, a review is held to demonstrate completed work to stakeholders and gather feedback, followed by a retrospective where the team reflects on how to work better in the next cycle. This rhythm of plan, deliver, review, and improve is what gives Scrum its strength. **Kanban** takes a different approach, focusing on visualising work and controlling the flow of tasks rather than working in fixed iterations (Anderson, 2010). Work items are represented on a Kanban board, typically moving through columns such as To-Do, In Progress, and Done, giving the whole team a clear, real-time picture of where things stand. A defining feature of Kanban is the use of Work-in-Progress limits, which cap the number of tasks allowed in any stage at once. This prevents teams from taking on more than they can handle and helps identify bottlenecks before they become serious problems. Decisions in Kanban are driven by data, with metrics like lead time and cycle time used to guide ongoing improvements.

Extreme Programming (XP) was developed to address the challenges of rapidly changing requirements and the constant pressure to deliver high-quality software (Anchit Shrivastava et al., 2021). It centres on four values: communication, simplicity, feedback, and courage. Three of its most distinctive practices are pair programming, where two developers work together at one workstation to improve code quality and share knowledge; test-driven development, where tests are written before the code itself to catch defects early and keep the codebase maintainable; and continuous integration, where code changes are merged into a shared repository frequently and tested automatically to prevent problems from accumulating.

Lean originated in Toyota's manufacturing operations and has since been adapted for software development and other service environments (Leffingwell, 2018). Its central idea is straightforward: deliver maximum value to the customer while eliminating everything that does not contribute to that value. Lean uses tools like value stream mapping to trace the full journey of work from request to delivery, identifying steps that add value and those that represent waste. It categorises waste into seven types, including overproduction, defects, and unnecessary waiting, and works systematically to reduce each one. Pull systems ensure that work is only started in response to actual demand, keeping inventory low and delivery timely.

The Scaled Agile Framework (SAFe) was designed for organisations that need agile to work not just within a single team but across many teams working on large, complex programmes (Ciancarini et al., 2022). It does this by organising teams into Agile Release Trains, which are groups of teams aligned around a shared value stream and working toward common goals. A central event in SAFe is Agile Release Train planning, where all

teams come together every eight to twelve weeks to plan their work for the upcoming Programme Increment. This shared planning session ensures that dependencies are identified, priorities are agreed upon, and every team understands how their work connects to the broader organisational objectives.

2.1.5 Drawbacks in Agile Methodologies and Frameworks

While agile methodologies offer considerable advantages, their implementation is not without challenges. Understanding these drawbacks is important for organisations seeking to adopt agile practices realistically and sustainably.

One of the most commonly cited difficulties is cultural resistance. Moving from traditional, hierarchical structures to the self-organising, collaborative teams that agile requires can meet significant pushback, particularly in organisations with deeply entrenched ways of working (Schwaber, 2017). This transition demands genuine leadership commitment and deliberate change management, not just a rebranding of existing processes.

Resource constraints present a related challenge. Agile depends on dedicated cross-functional teams, and in organisations where staff are stretched across multiple priorities, assembling and maintaining such teams consistently can be difficult (PMI, 2021). Without adequate resourcing, the iterative cycles that give agile its strength begin to break down.

Scaling agile beyond a single team adds further complexity. Coordinating multiple agile teams on a large programme while maintaining alignment, communication, and a consistent pace of delivery requires careful structural support (Leffingwell, 2018). Frameworks like SAFe and LeSS were developed to address this, but their adoption introduces its own overhead and learning curve. Skills and experience gaps also slow agile adoption. Agile calls for a particular mindset and set of practices that many teams have not been trained in. Without investment in proper onboarding and ongoing development, teams tend to apply agile superficially rather than effectively (Masood and Farooq, 2017). Similarly, sustaining meaningful stakeholder involvement throughout the project lifecycle can be difficult when clients and sponsors have limited time to engage with regular reviews and feedback sessions (Highsmith, 2009).

Finally, measuring progress and return on investment in agile environments is not straightforward. Traditional project metrics such as percentage completion against a fixed plan do not translate well. Agile-specific measures like velocity and burndown charts provide more relevant signals, but interpreting them correctly and using them to communicate value to senior stakeholders requires a shift in how success is defined and reported (Ambler, 2017).

2.1.6 Factors of Agile Project Management

Conforto et al. (2014) describe agile project management enablers as internal or external factors that are directly or indirectly related to the implementation of agile practices and

that can influence the quality of software delivered. This study focuses on three such factors: lead time, team morale, and bugs and defects.

Lead Time

Lead time refers to the total time it takes for a task or feature to move from the moment it is requested to the point it is completed and delivered. Blackstone (2013) defines it simply as the span of time required to perform a process or series of operations. In an agile context, lead time is a useful indicator of how efficiently a team is working. Shorter lead times mean value reaches the customer faster, which is one of agile's central promises. De Treville et al. (2014) note that reducing lead time can itself become a source of competitive advantage, particularly in environments where customers are waiting on delivery.

In practice, lead time in agile covers everything from when a user story enters the backlog to when it is marked as done, passing through planning, development, testing, and deployment. Bringing this time down has several benefits: customers receive value sooner, feedback loops become faster, and bottlenecks in the development process become easier to spot (Cohn, 2010). That said, speed should never come at the cost of quality. The goal is efficiency without cutting corners. Managing lead time well requires regular retrospectives, honest process analysis, and tools like WIP limits in Kanban to keep work flowing smoothly without overloading the team (Anderson, 2010; Leffingwell, 2011).

Team Morale

Team morale sits at the heart of agile's success. Agile relies heavily on motivated, self-organising teams that take ownership of their work, communicate openly, and adapt quickly to change. When morale is high, these qualities tend to emerge naturally. Team members contribute ideas, support one another, and stay engaged with the project goals. When morale is low, the opposite tends to happen, and the collaborative spirit that agile depends on begins to erode (Funk and Segrest, 2017; Cohn, 2010). Beyond productivity, morale also shapes how resilient a team is when things get difficult. Agile projects involve constant change, and a team that feels supported and valued is far better placed to handle that pressure constructively (Deemer et al., 2010). There is also a direct link between morale and retention. People who feel recognised, challenged, and cared for are less likely to leave, and high turnover is particularly disruptive in agile environments where team continuity matters greatly. Several strategies help sustain team morale in agile settings. Recognising and celebrating achievements, both big and small, reinforces a positive team culture. Involving team members in decisions and giving them genuine autonomy builds a sense of ownership (Cervone, 2017). Setting clear and meaningful goals keeps people focused and motivated (Locke and Latham, 2019). Investing in learning and professional growth signals that the organisation values its people for the long term (Ambler, 2012).

Bugs and Defects

Bugs and defects are an unavoidable part of software development, particularly in the early stages of a project. In agile methodologies, they are not treated as failures but as expected occurrences that need to be managed carefully and resolved quickly (Ambler, 2012). The emphasis is on catching them early, before they compound and become harder to fix. Agile's continuous testing and integration practices are designed with exactly this in mind (Schwaber and Beedle, 2001). Bugs can arise from many sources including coding errors, incomplete requirements, miscommunication between team members, or scope changes. Whatever the cause, how a team responds to them matters enormously. The most effective approach is to fix defects as close to their point of origin as possible. As Kelly (2014) points out, the longer a bug is left unaddressed, the more difficult and riskier the fix becomes, because other code tends to be built on top of it. Beyond prompt resolution, agile teams manage bugs and defects through a combination of backlog prioritisation, where defects are ranked alongside new features based on their severity and impact; sprint allocation, where time is deliberately set aside within each sprint to address outstanding issues (Leffingwell, 2018); automated testing, which allows teams to catch regressions early and verify that fixes hold (Humble and Farley, 2010); and a culture of transparency, where team members take personal accountability for the defects they introduce or discover (Ambler, 2017). Collaborative practices like pair programming also help by reducing the likelihood of bugs making it through undetected in the first place.

2.1.7 Organizational Performance

Organisational performance refers to how effectively a company achieves its goals across all areas of its operations, from financial health and operational efficiency to customer satisfaction, employee engagement, and overall competitiveness (Dragomir and Panzaru, 2014; Richard et al, 2009). It is not a single measure but a combination of indicators that together reveal whether an organisation is functioning well enough to sustain and grow its market position. Operational efficiency sits at the core of this, capturing how well an organisation uses its resources, whether people, money, or technology, to produce outputs with minimal waste. When operations run efficiently, productivity rises, processes become leaner, and customers tend to receive better and faster service. Customer satisfaction and employee engagement reinforce this further. Satisfied customers are more loyal and more likely to recommend the organisation to others, while engaged employees work harder, stay longer, and contribute more meaningfully to the organisation's goals (Rowold, 2011). Taken together, these dimensions define an organisation's competitiveness, which ultimately determines its ability to hold its ground in the market and withstand external pressures.

2.1.8 Factors of Organizational Performance

Boru and Chen (2020) identified several measurable dimensions of organisational

performance. This study focuses on three of them: customer satisfaction, employee turnover rate, and software quality.

Customer Satisfaction

Customer satisfaction is one of the most important drivers of business performance. Retaining an existing customer is far easier and less costly than acquiring a new one, and satisfied customers tend to spend more, stay longer, and actively refer others to the business (Kotler and Keller, 2016). Kotler and Keller (2016) describe it as the aggregate of a customer's experience across all their interactions with a company, covering everything from the product itself to the service received.

Beyond loyalty and referrals, customer satisfaction also feeds back into operations. Customers who are happy tend to provide useful feedback that helps organisations improve their products and processes. There is also a human dimension to this: when employees see that their work is genuinely making customers happy, it tends to boost their own morale and motivation, which in turn drives further improvements in service quality (Gupta and Chathoth, 2019). For an IT company like Systemspec Holdings Limited, where software delivery is the primary product, customer satisfaction is a direct reflection of how well the team is performing.

Employee Turnover Rate

Employee turnover measures how frequently staff leave an organisation over a given period, and high turnover is a significant drag on performance. The reasons people leave are varied. Al-suraihi et al (2021) point to unmet expectations, poor fit between the person and their role, lack of training and feedback, limited growth opportunities, feeling undervalued, overwork, and a breakdown of trust in leadership. Structural factors also play a role, including poor compensation, weak human resource policies, and conflict with colleagues (Al Mamun and Hasan, 2017; Alshammari et al., 2016).

The costs of turnover extend beyond recruitment. When experienced employees leave, they take institutional knowledge with them, and the organisation loses the continuity that good teamwork depends on. High turnover also tends to damage culture over time, as remaining employees begin to question the organisation's stability (Allen et al., 2013). Addressing it requires a thoughtful response across several fronts, including better training and development, clearer career pathways, stronger management practices, and conditions that allow people to balance work with their personal lives (Ahmed and Kolachi, 2013).

Software Quality

Software quality is a multidimensional concept that can be understood from several perspectives. From the customer's point of view, it is the extent to which a product meets their requirements (Singh and Kassie, 2018). From a product standpoint, it relates to how well the software performs its intended functions. From a value perspective, it is about

delivering what the customer expects at a reasonable cost (Zakaria, Hamdan, and Deraman, 2016).

Whatever the perspective, the importance of software quality to organisational performance is clear. High-quality software satisfies customers and builds loyalty (Pressman, 2014). It reduces maintenance costs because fewer defects mean less time and money spent on fixes. It gives organisations a competitive advantage by establishing a reputation for reliability and excellence. It also reduces risk, since poorly built software tends to be more vulnerable to security threats and compliance failures (Mellado et al., 2014). At the level of day-to-day operations, well-designed software that is intuitive and easy to use helps employees work more efficiently, which amplifies productivity across the organisation (Zhang et al., 2011). For an organisation operating in the software industry, quality is therefore not just a technical concern but a direct determinant of how well the business performs overall.

2.1.9 Agile Project Management and Organizational Performance

The relationship between agile project management and organisational performance has attracted growing attention in both academic and professional circles. The core argument is that agile's emphasis on flexibility, collaboration, and continuous improvement does not just make projects run better; it also strengthens the broader organisation over time.

One of the most direct contributions agile makes to performance is through efficiency. By breaking work into short, focused iterations and delivering a working product increment at the end of each one, agile teams maintain a steady and visible pace of progress. This keeps motivation high and reduces the kind of prolonged uncertainty that tends to drain productivity in traditional project environments (Cervone, 2017). The iterative approach also compresses the time it takes to bring products to market. Delivering minimum viable products early and improving them through successive cycles means organisations can respond to customer needs and market shifts far more quickly than competitors using longer, more rigid development processes (Schwaber and Sutherland, 2017).

Adaptability is another dimension where agile positively influences performance. Rather than treating change as a problem to be avoided, agile teams are structured to absorb it. When requirements shift or new information emerges, the team adjusts without losing momentum. This responsiveness allows organisations to keep their products aligned with what the market actually needs, which over time strengthens their competitive position (Highsmith, 2009).

Customer satisfaction is perhaps the most consistent performance outcome associated with agile adoption. Involving customers at every stage of development, through reviews, feedback sessions, and collaborative prioritisation, means the final product is far more likely to meet real expectations rather than assumed ones. Chow, Cao, Leiva, and Shi (2016) highlight this customer-centric orientation as one of agile's defining characteristics, and its effect on loyalty and repeat business is well documented.

Finally, agile's culture of continuous improvement has a compounding effect on

organisational performance. Teams that regularly reflect on how they are working, identify what can be done better, and implement changes in the next iteration are organisations that are constantly learning (Chen, Song, and Tsai, 2021). Over time, this habit of structured reflection and adaptation raises the overall capability of the team and produces progressively better outcomes, making agile not just a project delivery method but a genuine driver of long-term organisational growth (Schwaber and Sutherland, 2017).

2.2 Theoretical Review

This study focused on three theories namely; Lean Thinking, Agile Maturity Model, Information Systems Success Model. The theories are reviewed in terms of their applicability to the study objectives.

2.2.1 Lean Thinking

Lean Thinking originated in manufacturing, where Toyota pioneered its use as a systematic approach to eliminating waste and delivering maximum value with minimum effort. Its core principles centre on understanding what customers truly value, removing everything in the process that does not contribute to that value, and building a culture of continuous improvement at every level of the organisation. When applied to software development, these principles translated naturally into what became known as Lean Software Development, an approach that shares much of its DNA with agile methodologies.

The overlap between Lean and agile is not coincidental. Both prioritise delivering value efficiently, both treat waste as something to be actively identified and removed, and both rely on iterative cycles and ongoing feedback to keep improving. Scrum, for instance, reduces unnecessary work by focusing teams on the highest-priority tasks within each sprint (Schwaber and Sutherland, 2017), while Kanban directly applies Lean's principle of optimising flow by visualising work and limiting the amount in progress at any one time (Anderson, 2010). It is this alignment that makes Lean Thinking a particularly relevant theoretical lens for this study. It provides a framework for understanding why agile practices improve efficiency and quality, grounding the study's variables — lead time, defect management, and team performance — in a well-established body of thinking about how processes can be designed to deliver consistent value. For these reasons, Lean Thinking is adopted as one of the theoretical foundations of this study.

2.2.2 Agile Maturity Model

The Agile Maturity Model describes how an organisation's agile practice evolves over time, moving through progressively higher levels of capability and effectiveness (Fontana et al., 2015). At each level, the organisation becomes more proficient in applying agile principles, more consistent in its delivery, and better at identifying and closing gaps in its processes. The model essentially provides a structured way to measure where an

organisation stands on its agile journey and what it needs to do to advance.
Its relevance to agile project management is direct. The model assesses the same qualities that agile depends on: continuous improvement, collaboration, customer focus, and the ability of teams to self-organise and adapt (Ambler and Lines, 2012; Schwaber, 2017). By regularly evaluating performance against these dimensions, organisations can move beyond simply adopting agile practices on paper and work toward genuinely embedding them in how they operate. This leads to better project outcomes, stronger customer satisfaction, and greater organisational adaptability over time (Niazi and Wilson, 2007).
For this study, the Agile Maturity Model provides a useful framework for understanding organisational performance not as a fixed state but as something that develops incrementally as agile practices deepen and mature within the organisation.

2.2.3 Information Systems Success Model

The Information Systems Success Model, proposed by DeLone and McLean (1992), offers a framework for evaluating how well an information system achieves its intended purpose within an organisation. It assesses dimensions including system quality, information quality, user satisfaction, and organisational impact, all of which together determine whether a system can be considered genuinely successful. The model aligns closely with agile project management in several ways. Agile's emphasis on delivering high-quality, functional software in every iteration directly corresponds to the model's focus on system quality (Schwaber and Sutherland, 2017). Its customer-centric orientation, built around continuous feedback and user involvement, maps onto the model's dimension of user satisfaction. And agile's goal of producing software that is not just functional but genuinely usable reflects the model's concern with effective system use (Cohn, 2010). Together, the three theoretical frameworks reviewed in this section — Lean Thinking, the Agile Maturity Model, and the Information Systems Success Model — each offer a distinct but complementary lens for understanding agile project management and organisational performance. While all three are relevant, Lean Thinking is adopted as the primary theoretical foundation of this study given its direct alignment with the core agile principles under investigation and its particular applicability to efficiency, waste reduction, and continuous improvement in IT environments.

2.3 Empirical Review

Lalsing, Kishnah, and Pudaruth (2012) examined people factors in agile software development and project management, finding that team composition is one of the most overlooked determinants of agile success. Their study showed that carefully matching individuals to teams based on skills, collaboration ability, and adaptability was critical to achieving strong project outcomes. The implication is that agile effectiveness is as much a human resource question as it is a process one. Dybå, Dingsøyr, and Moe (2014) characterised agile project management as a shift away from direct supervision and rigid control toward informal collaboration, self-managing teams, and continuous learning.

Drawing on extensive studies of industrial projects, they identified four guiding principles of agile project management: minimum critical specification, autonomous teams, redundancy, and feedback-driven learning. Their work established a useful conceptual foundation for understanding how agile differs from traditional management in practice. Serrador and Pinto (2015) provided one of the most rigorous empirical tests of agile effectiveness to date, analysing data from 1,002 projects across multiple industries and countries. Their findings confirmed that agile methods positively influenced both project efficiency and overall stakeholder satisfaction. The quality of the project vision emerged as a moderating factor, suggesting that agile works best when teams have a clear sense of direction from the outset. Todorović et al (2018) focused on the human dimension of agile adoption, using Scrum as their primary reference point. Comparing agile with traditional methodologies, they found that agile fosters self-organisation, learning, and communication more effectively. Key individual-level factors enabling successful agile adoption included trust, openness to learning, and genuine internalisation of agile values rather than surface-level compliance. Scholz, Sieckmann, and Kohl (2020) examined agile implementation in the context of an Industry 4.0 learning factory in China, where the pace of technological change and cultural specificity made conventional project management inadequate. Their case study demonstrated that agile approaches were better suited to managing both the structured and unstructured dimensions of rapidly evolving, context-sensitive projects, underscoring the importance of adaptability in technology-driven environments. Muhammad et al. (2021) investigated the impact of agile management on project performance in Pakistan's IT sector, surveying 176 respondents across multiple cities. They found that agile practices effectively offset the negative effects of project complexity on performance, and that leadership competence played a significant moderating role. The study is particularly relevant as one of the few empirical contributions from a developing country context, highlighting that agile's benefits extend beyond Western settings. Marnada et al., (2022) examined how agile teams manage scope and change, a challenge that intensified during the COVID-19 pandemic. Their analysis identified eighteen distinct issues grouped into seven categories, with people and organisational factors, requirement prioritisation, and communication breakdowns emerging as the most significant sources of project delays and cost overruns. Their findings reinforced the importance of clear governance structures even within flexible agile environments. Sharma., et al (2022) studied agile implementation challenges in emerging economies during the pandemic period. They found that scepticism toward new ways of working was the single most influential barrier to agile adoption, with a cascading effect on other implementation challenges. Their work provides a timely reminder that the behavioural and cultural dimensions of agile adoption are just as important as the technical ones. Shi., et al (2022) assessed the relationship between agile project management and the competitiveness of software development firms in China. Using explanatory factor analysis, they found that agile practices positively influenced firm performance, with innovation capability and responsiveness to

environmental dynamics emerging as key differentiators. Their study adds to the growing evidence that agile is not just a delivery methodology but a strategic driver of competitive advantage in high-tech industries.

2.4 Research Gap

These studies all suggest that agile project management can be an effective tool for improving organizational performance. However, it is important to note that these studies were conducted in different countries, and the results may not be generalizable to all organizations. Additionally, the studies were conducted at different times, and the results may not be reflective of the current state of the market. Though, the evidence suggests that agile project management can be a valuable tool for improving organizational performance. However, more research is needed to confirm these findings and to identify the specific benefits of agile project management for different types of organizations. Hence, this study examined agile project management and organizational performance in System spec Holdings Limited.

Methodology

3.1 Research Design

The influence of Agile project management on organizational performance at Systemspec Holdings Limited is investigated using a quantitative research approach in this study. The quantitative technique is chosen for its capacity to provide statistically significant insights into variable interactions and draw generalizable conclusions. The quantitative technique is used because it enables the systematic gathering and analysis of numerical data linked to customer satisfaction scores, employee turnover rate, and software quality, allowing for a thorough assessment of the research topic inside Systemspec Holdings Limited.

3.2 Population of the Study

This study's population includes all Systemspec Holdings Limited personnel and stakeholders who use Agile project management approaches. Individuals from various departments, roles, and degrees of experience are included in the population.
According to the most recent organizational data available, Systemspec Holdings Limited employs roughly 309 people. This statistic indicates the total organization's workforce.

3.3 Sample Size and Sampling Technique

The sample was drawn from a population of 309 employees using a stratified random sampling technique. Departments, jobs, and levels of expertise within Systemspec Holdings Limited were used to stratify employees. Stratified sampling guarantees that the sample chosen is representative of the organization's diversified workforce. The study can capture a complete range of viewpoints and experiences connected to Agile project management and its impact on organizational performance by including personnel from all strata. The sampling procedure involved randomly picking individuals from each

stratum in proportion to their population representation. This method will aid in the prevention of bias and will ensure that all segments of the workforce are sufficiently represented in the study. The sample size adopted in this study was determined using Taro Yamane (1967) formula. The formula is as follows:

$$n = \frac{N}{1+N(e)^2}$$

Where:

- n represents the required sample size.
- N is the total population size, which, in this case, is 309 employees.
- e is the desired level of precision, represented as a decimal (e.g., 0.05 for a 5% margin of error).

$$n = \frac{309}{1+309(0.05)^2}$$

$$n = \frac{309}{1+0.7725}$$

$$n = \frac{309}{1.7725}$$

$$n = 174.34$$

Sample size = 174.34

Rounding up to the nearest whole number, the recommended sample size would be approximately 174 employees.

3.4 Method of Data Collection

Data were collected through electronic questionnaires administered to employees and stakeholders at Systemspec Holdings Limited who were involved in agile project management. Each questionnaire was accompanied by clear completion instructions, and respondents were informed that participation was entirely voluntary and that all responses would be treated as confidential. Informed consent was obtained from all participants, and personal data were aggregated for analysis to protect individual privacy.

3.5 Research Instrument

The research instrument was a structured questionnaire designed to capture respondents' perceptions of agile project management practices and organisational performance at

Systemspec Holdings Limited. It opened with a brief introduction explaining the study's purpose and assuring respondents of anonymity. The questionnaire comprised three sections: demographic information including age, educational qualification, and years of experience; agile project management practices covering methodology adoption, agile ceremonies, cross-functional collaboration, and iterative development; and organisational performance measures covering customer satisfaction, employee turnover rate, and software quality. Items were primarily Likert-scale based, enabling systematic measurement of respondents' perceptions, with provision for additional qualitative comments where relevant.

3.6 Reliability and Validity of Research Instrument

The validity and reliability of the instrument were ensured through expert validation and pilot testing prior to full deployment. Internal consistency was assessed using Cronbach's alpha, with all constructs returning values above the accepted threshold of 0.7 (see Table 4.5). Potential response bias was minimised by clearly communicating the study's purpose to participants and guaranteeing the confidentiality and anonymity of their responses.

3.7 Method of Data Analysis

The collected data was examined statistically using relevant procedures. To describe the quantitative data, descriptive statistics was employed, offering insights into the frequencies, means, and variability of the responses. To investigate the link between agile project management methods and organizational performance, inferential statistics was used which include Partial Least Square (PLS), Partial Least Square Path Modelling (PLS-PM) and Partial Least Square Structural Equation Modelling (PLS-SEM).

3.8 Model Specification

Hypothesis I

Ho_1: Lead time does not significantly impact customer satisfaction.

$Y = \alpha + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \varepsilon$

$CS = f(PC + RA + TS)$

Where:

Y = Customer Satisfaction (CS)

X_1 = Project Complexity (PC)

X_2 = Resource Allocation (RA)

X_3 = Team Size (TS)

ε = Error Terms

Hypothesis II

Ho_2: Team morale does not significantly influence employee turnover rate.

$Y = \alpha + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \varepsilon$

ETR = *f*(WLB + CP + CDO)
Where:
Y = Employee Turnover Rate (ETR)
X_1 = Work-Life Balance (WLB)
X_2 = Compensation Package (CP)
X_3 = Career Development Opportunity (CDO)
ε = Error Terms

Hypothesis III

Ho_3: Bugs and defect do not significantly affect software quality.

$Y = \alpha + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \varepsilon$

SQ = *f*(CC + BF + DP)
Where:
Y = Software Quality (SQ)
X_1 = Code Complexity (CC)
X_2 = Bug Frequency (BF)
X_3 = Development Process (DP)
ε = Error Terms

3.9 Ethical Considerations

This study was conducted in accordance with established ethical principles governing academic research involving human participants. Prior to data collection, ethical approval was obtained from the Research Ethics Committee of the University of Ilorin, and permission to conduct the study within the organisation was granted by the management of Systemspec Holdings Limited.

All participants were fully informed of the purpose of the study before completing the questionnaire. Participation was entirely voluntary, and respondents were made aware that they could withdraw at any point without consequence. Informed consent was obtained from every participant prior to their involvement in the study.

To protect the privacy and confidentiality of respondents, no personally identifiable information was collected. All responses were anonymised and reported only in aggregate form, ensuring that no individual participant could be identified from the data or findings. The data collected were used solely for the purposes of this research and were stored securely throughout the study period.

The study did not expose participants to any physical, psychological, social, or economic harm. Respondents were not subjected to deception at any stage of the research process, and the findings have been reported accurately and without misrepresentation. The researcher maintained professional integrity throughout the data collection, analysis, and reporting phases of the study.

Data Presentation and Analysis

4.0 Preamble

This chapter deals with the presentation and analysis of data collected through administered questionnaire. Out of 174 copies of questionnaires administered to the employees of the selected companies, 138 administered questionnaires were filled and returned. This brings the response rate to 79.3%. Tabular presentation and interpretation of data on respondents' demographic data are also given attention. The last section covers the presentation and interpretation of estimated parameters from the structural equation model. From the models estimated, the research hypotheses formulated in chapter one was tested to achieve the objectives of the study.

4.1 Response Rate

In this study, questionnaire was used to obtain the needed data from employees of the SystemSpec Holdings Limited. The total of 309 employees were recorded in SystemSpec Holdings Limited which represent the population of the study. From this population, sample size of 174 was drawn through the use of Taro Yamane (1967) sample size determination formula. Questionnaires were filled and returned to the researcher. At the end, 36 of the filled and returned questionnaire were discarded due to incomplete response, failure to meet the criteria set by researcher. Finally, 138 questionnaire responses were used for the analysis as shown in the table 4.1 below:

Table 4.1 Summary of Response Rate of the Respondents

Validity	Frequency	Percentage	Valid Percentage	Cumulative Percentage
Valid Questionnaires	138	79.3	79.3	79.3
Invalid Questionnaires	36	20.7	20.7	100.0
Total	174	100	100	

Source: Field Survey, 2023

4.2. Preliminary analysis

The essence of preliminary analysis is to ensure that data inputted and used for analysis is adequate, proper, and satisfactory for the intended purpose. An inappropriate data can be misleading and give a wrong direction or signal for inferences. It is necessary to double-check and test inputted data for normality tests to ensure that data were properly cleansed for the result to be reliable; acceptable and also generalized. The overriding goal of managing all screening operations is to eliminate model estimation failure and fitting software crashes especially in a multivariate analysis. Data editor was used to crossed check that data were accurately inputted.

4.2.1 Normality test

The normalcy of data measurement that is regularly dispersed throughout a population

sample which ensures that no extremely high or low scores from few participants affect the overall results is the main goal of normality test (Hair et al, 2010). This is important because it is capable of affecting the appropriateness and standard error if the data is not normal. Data normalcy is usually measure by kurtosis and skewness. The peakedness of distribution by kurtosis and symmetry distribution of data is revealed by skewness with an acceptable threshold of -1 to +1 depending on the number of sample size which might rise to 1.5 while the standardize kurtosis index values ranging from -3 to +3 and absolute kurtosis index values might range from 8.0 to over 20.0 for severe kurtosis which called for concern. A positive kurtosis is indicated by peaked or taller distribution while flat distribution implies negative kurtosis. A positive skew indicates a shifted or skewed distribution to the left and a negative skew indicate right skewed distribution.

Table 4.2 Normality Results

	Excess Kurtosis	Skewness	Number of Observations Used
Bugs and Defect	-0.163	-0.229	138.000
Customer Satisfaction	-0.447	0.064	138.000
Employee Turnover Rate	-0.526	-0.039	138.000
Lead Time	-0.324	-0.291	138.000
Organizational Performance	-0.424	-0.168	138.000
Software Quality	-0.315	-0.041	138.000
Team Morale	-0.313	-0.309	138.000

Source: Authors Compilation (SmartPLS 3.3.3 Output) 2023

The normalcy results of the distribution revealed that the sample size is than 200 which imply that an absolute value of skewness of +1.0 or below is expected for the data to be normal. Also, for kurtosis, an absolute value of +3.0 is expected for a normal peakedness as any figure outside the threshold may be severe signaling a concern. The normalcy results show that all the variables were within the threshold of the absolute value of ±1.0 and the kurtosis results were also within the absolute value of ±3.0. The implication from the normality test results shows that all the data inputted for the analysis are normally distributed and can be used for further analysis and inferences.

4.3 Demographic Characteristics of the Respondents

This section detailed out the demographic characteristics of the respondents measured by attributes such as age, gender, educational qualifications, and the likes. Here, each of these attributes was related to the concept of the study as it affects the variables

measured.

Table 4.3: Demographic Characteristics of Respondents

	Category	Frequency	Percentage
Variable			
Gender	Male	41	29.6%
	Female	97	70.4%
	Total	**138**	**100.0%**
Age	18-23 years	41	29.8%
	24-29 years	50	36.4%
	30-35 years	31	22.4%
	36-42 years	10	7.4%
	42-47 years and above	7	5.0%
	Total	**138**	**100.0%**
Marital Status	Single	23	16.9%
	Married	84	60.6%
	Divorced	10	7.0%
	Widowed/Widower	21	15.5%
	Total	**138**	**100.0%**
Highest Educational Level	SSCE	29	20.8%
	OND/NCE	33	23.6%
	B.Sc/HND	47	34.4%
	MSc/MBA	20	14.2%
	Ph.D.	10	7.0%
	Total	**138**	**100.0%**
Year of Experience	Less than 2 years	21	15.2%
	2 to 5 years	99	71.8%
	6 to 10 years	15	11.0%
	11 years and above	3	2.0%
	Total	**138**	**100.0%**

Source: Field Survey, 2023

Table 4.3 indicates that 70.4% of the respondents were female and 29.6% of them were male. This implies that more than half of the total respondents were female. The world statistics has shown that there are more female gender than the male gender in the world. Hence, the more reason female employees are more in SystemSpec Holdings Limited than male employees and female employees are more than male generally.

Age group of the respondents is also presented in table 4.3. The result shows that 29.8% of respondents are between 18-23 years, 36.4% are within the ages of 24 - 29 years, 22.4% are within the ages of 30 – 35 years, 7.4% are within the ages of 36 – 42 years and 5.0% are within the ages of 42-47 years and above. This implies that the selected

employees are full of largely homogenous, young employees (18 - 29 years) which could be due to the fact that achieving better agile project management method young and vibrant employees.

Marital status of the respondents is also presented in table 4.3. The result shows that 16.9% of respondents are single, 60.6% are married, 7.0% are divorced, and 15.5% are widowed or widower. This implies that most of those employees who work in SystemSpec Holdings Limited are married (60.6%). This means they have sense of high responsibility due to their family nature. This makes them have full sense of responsibility at work place.

Educational qualification of the respondents as shown in the table 4.3 revealed that 20.8% of the respondents are at least SSCE holders, 23.6% of the respondents hold Ordinary National Diploma (OND) or Nigeria Certificate in Education (NCE), 34.4% hold B.Sc/HND degrees, 14.2% hold Masters degrees, while only 7.0% hold Ph.D. degrees. This implies that respondents have prerequisite qualifications to understand and respond to the study questionnaire as it is needed to be learned to be able to read and write when been employed in organizations such as SystemSpec Holdings Limited.

Year of the respondents was also displayed in table 4.3. 15.2% of the respondents have less than 2 years' experience, 71.8% of the respondents have 2-5 years of experience with SystemSpec Holdings Limited, 11.0% of the respondents have 6-10 years of working with SystemSpec Holdings Limited, while the remaining 2% has 11 years of experience with SystemSpec Holdings Limited. This implies that SystemSpec Holdings Limited recruits highly skilled employees regularly and more often who are capable of producing higher level of performance.

4.4 Descriptive Statistics of the Responses Relating to the Questions/Variables

Table 4.4 Descriptive Statistics on Questions Relating to the Study

	Mean	Standard Deviation	Number of Observations Used
Bug Frequency	3.180	0.520	138.000
Career Development Opportunity	4.357	0.650	138.000
Code Complexity	3.918	0.572	138.000
Compensation Package	3.811	0.529	138.000
Customer Satisfaction 3	4.025	0.552	138.000
Customer_Satisfaction 1	3.640	0.525	138.000
Customer_Satisfaction 2	3.235	0.466	138.000
Development Process	3.429	0.591	138.000
Employee Turnover Rate 2	3.165	0.461	138.000
Employee Turnover Rate 3	3.368	0.522	138.000
Employee _Turnover Rate 1	4.456	0.505	138.000
Project Complexity	2.942	0.457	138.000

Resource Allocation	3.928	0.496	138.000
Software Quality	3.690	0.544	138.000
Software Quality 1	3.435	0.568	138.000
Software Quality 2	3.710	0.516	138.000
Team Size	4.140	0.594	138.000
work-life Balance	3.513	0.522	138.000

Source: Authors Compilation (SmartPLS 3.3.3 Output) 2023

Table 4 shows the mean and standard deviation of the variables/indicators used in the study, these were derived from the study's questionnaire. Criterion mean of 2.50 was set to determine respondents' agreement with items possessing mean quite above the 2.50 benchmark considered as strongly agree and items below the 2.50 benchmark considered as strongly disagree. Determining the items that are strongly agreed on is dependent on identifying the items above the set benchmark and as shown in the table, Overall, all the respondents strongly agree to the indicators used to pose questions to them in the questionnaire. This suggest that the responses are valid to probe further and make valid conclusions on the effect of agile project management method on organizational performance.

4.5 Objectives and Hypotheses Restatement

Objectives:

Objective One: Examine the impact of lead time on customer satisfaction;
Objective Two: Assess the influence of team morale on employee turnover rate; and
Objective Three: Examine the effect of bugs and defect on software quality

Hypotheses:

Ho1: Lead time do not have significant effect on customer satisfaction
Ho2: Team morale do not have influence on employee turnover rate
Ho3: Bugs and defect do not have significant effect on software quality

4.5.1 Assessment of Measurement Model

To assess the effect of agile project management method on organizational performance, three objectives and hypothesis were formulated, the first has to do with lead time measured by project complexity, resource allocation and team size against customer satisfaction. The second has to do with team morale measured by work-life balance, compensation package and career development opportunity against employee turnover rate. The third has to do with bugs and defect measured by code complexity, bug frequency, development process against software quality.

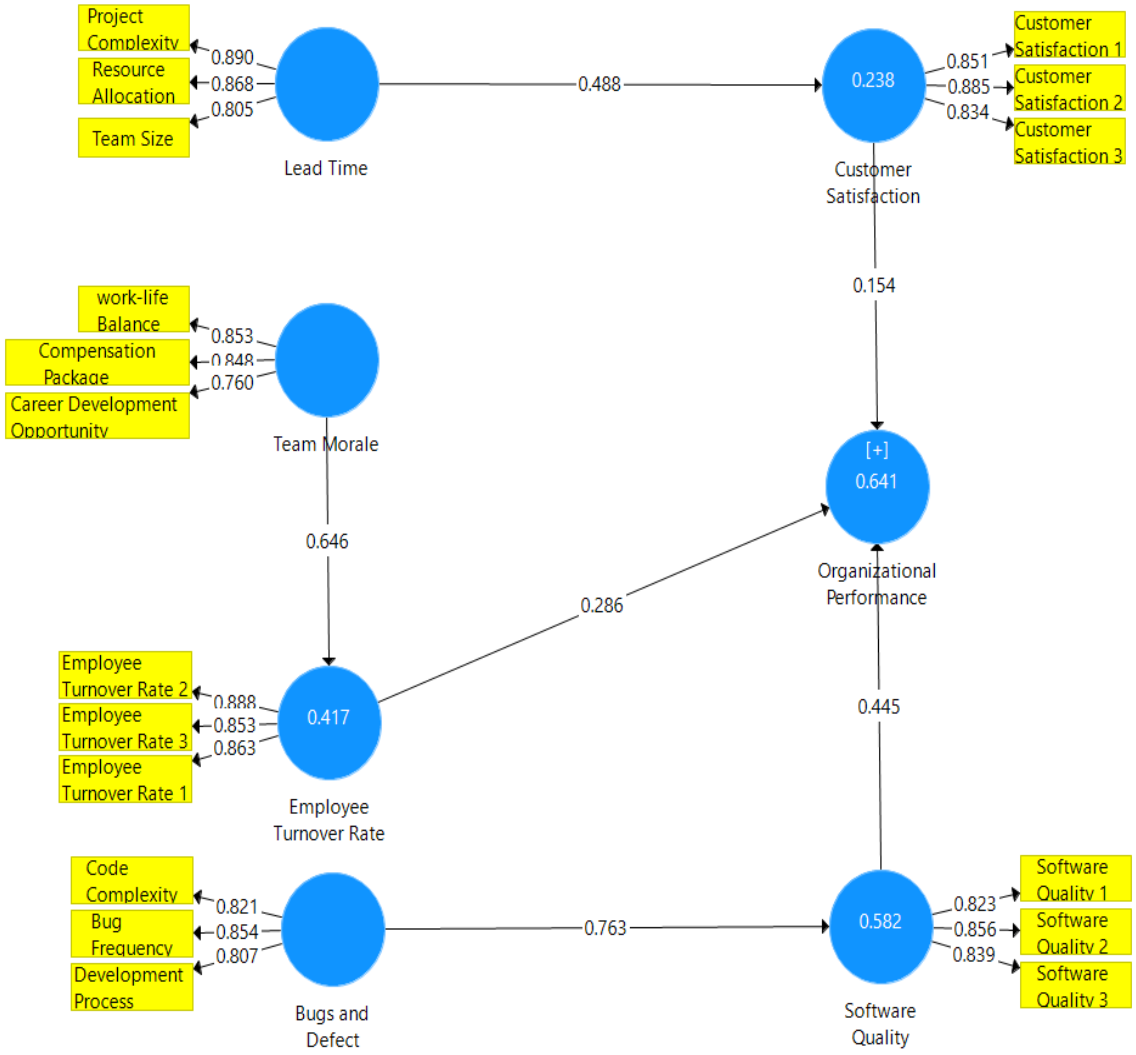

Figure 4.1: A path model of agile project management method and organizational performance

Source: SmartPLS Output, 2023

Figure 4.1 showed the path model of the agile project management method and organizational performance. The figure presents interaction effects where more than single variables contribute to the latent variables. The outer weight model varies from zero to an absolute maximum lower than 1, it has been established that the more the indicators for a latent variable, the lower the maximum and the lower the average outer model weight. The results of outer model weights justify why the weak loading could not be dropped as all the loading weights were greater than or close to 0.50. Also, these variables were major constituents of the latent variables from the literature.

Table 4.5 Construct Reliability and Validity

	Cronbach's Alpha	Composite Reliability	Average Variance Extracted (AVE)
Bugs and Defect	0.770	0.867	0.685
Customer Satisfaction	0.819	0.892	0.734
Employee Turnover Rate	0.836	0.902	0.753
Lead Time	0.815	0.891	0.731
Organizational Performance	0.795	0.880	0.710
Software Quality	0.791	0.878	0.705
Team Morale	0.757	0.861	0.675

Source: Authors Compilation (SmartPLS 3.3.3 Output) 2023

Table 4.5 displays important statistical measures related to the construct reliability and validity of latent variables in this study. These measures help assess the robustness of these variables in measuring the underlying concepts they are intended to represent. Construct reliability is evaluated through two key metrics: Cronbach's Alpha and Composite Reliability. Cronbach's Alpha assesses the internal consistency of a latent variable by measuring the degree to which all items within the variable are correlated with one another. All latent variables exhibit good internal consistency with Cronbach's Alpha. All values are well above the commonly accepted threshold of 0.7, suggesting that the items within each variable are reliable measures of their respective constructs. Composite Reliability is another measure of construct reliability that takes into account both the internal consistency and the relationships between items and the latent variable. It provides a more robust estimate of reliability, and in this study, all variables show excellent composite reliability, with all values above 0.7. These high values indicate that the latent variables are reliable indicators of the constructs they represent. Additionally, the table presents the Average Variance Extracted (AVE), which assesses the convergent validity of each latent variable. Convergent validity indicates the degree to which items within a variable are related to each other, and to what extent they measure the same underlying concept. AVE values in the table surpass the recommended threshold of 0.5. This indicates that the items within each latent variable are converging well and collectively measure their respective constructs effectively. The results suggest that the latent variables in this study exhibit strong construct reliability and validity. They demonstrate high internal consistency, robust composite reliability, and satisfactory convergent validity, supporting the use of these variables as reliable and valid measures in the research study.

Table 4.6 Discriminant Validity

	Bugs and Defect	Customer Satisfaction	Employee Turnover Rate	Lead Time	Organizational Performance	Software Quality	Team Morale
Bugs and Defect	0.828						
Customer Satisfaction	0.784	0.857					
Employee Turnover Rate	0.648	0.704	0.868				
Lead Time	0.440	0.488	0.530	0.855			
Organizational Performance	0.734	0.694	0.694	0.505	0.842		
Software Quality	0.763	0.761	0.674	0.458	0.755	0.840	
Team Morale	0.646	0.693	0.646	0.709	0.662	0.640	0.821

Source: Authors Compilation (SmartPLS 3.3.3 Output) 2023

The results of the discriminant validity analysis in table 4.6 indicate that there is strong evidence of discriminant validity among the latent variables: Bugs and defect, Lead time, Organizational performance, Customer satisfaction, Software quality, Employee turnover rate, and Team morale. Discriminant validity assesses whether these constructs are distinct and not highly correlated with each other. Looking at the correlations between these variables, it's evident that the diagonal values (the correlations of each variable with itself) are all substantially higher than the off-diagonal values (the correlations between different variables). This suggests that each latent variable is more strongly related to itself than to the other constructs, supporting the idea that they are distinct and measure different aspects of the overall construct. Team morale has a high correlation of 0.821 with itself, which is higher than its correlations with other latent variables. Similarly, Lead time has a strong correlation with itself (0.855) compared to its correlations with the other variables. While it is the same for other variables in their respective situations.

These results provide evidence that the latent variables in this analysis are measuring distinct concepts and are not just different manifestations of the same underlying construct. This suggests that the measurement model is suitable for the purpose of this study, as it effectively differentiates between these key factors: Bugs and defect, Lead time, Organizational performance, Customer satisfaction, Software quality, Employee turnover rate, and Team morale.

4.5.2 Multicollinearity

This assesses the correlation between the independent variable. It is to know if two independent variables are not correlated and producing the same result. The variance inflation factor (VIF) is used in this study to assess likely correlation between the independent variables.

Table 4.7 Inner VIF Values

	Customer Satisfaction	Employee Turnover Rate	Organizational Performance	Software Quality
Bugs and Defect				**1.000**
Customer Satisfaction			**2.827**	
Employee Turnover Rate			**2.179**	
Lead Time	**1.000**			
Organizational Performance				
Software Quality			**2.612**	
Team Morale		**1.000**		

Source: Authors Compilation (SmartPLS 3.3.3 Output) 2023

Table 4.7 depict the VIF values for the latent variables related to Organizational performance. The VIF values for Bugs and defect, Lead time, and Team morale are 1.000, 1.000, and 1.000, respectively. These values are all well below the threshold of 10, which is a positive sign. It suggests that there is no severe multicollinearity among these latent variables. In other words, these variables are not highly correlated with each other, and are therefore included in this analysis without significant concerns about multicollinearity.

4.5.3 Test of Hypothesis

Table 4.8 Coefficient of Determination Score

	R Square	R Square Adjusted
Customer Satisfaction	0.238	0.234
Employee Turnover Rate	0.417	0.414
Organizational Performance	0.641	0.635
Software Quality	0.582	0.580

Source: Authors Compilation (SmartPLS 3.3.3 Output) 2023

The coefficient of determination is depicted in table 4.8, which denote R-squared which is used to assess the goodness of fit of a model. The R-squared values of 0.238, 0.417, and 0.582 for the Customer satisfaction, Employee turnover rate, and Software quality respectively, indicates that approximately 23.8%, 41.7% and 58.2% of the variability in the dependent variables (Customer satisfaction, Employee turnover rate, and Software quality), is explained by the independent or latent variables included in the model. This

suggests that the model captures and explains the observed variations in Customer satisfaction, Employee turnover rate, and Software quality. While the model shows an overall R-squared value of 0.641 on organizational performance.

The adjusted R-squared value are 0.234, 0.414, and 0.580 for Customer satisfaction, Employee turnover rate, and Software quality and 0.635 for organizational performance. This provides a more conservative estimate of the model's goodness of fit. The adjusted R-squared value is very close to the regular R-squared value, indicating that the inclusion of the independent variables in the model doesn't seem to be leading to overfitting or excessive complexity. This suggests that the model's explanatory power remains strong even when considering potential issues related to model complexity. The R-squared and adjusted R-squared values suggest that the organizational performance model is fairly effective at explaining organizational performance variability, and the model's performance doesn't appear to suffer from the inclusion of multiple latent variables.

Table 4.9 Assessment of the Effect Size (f^2)

	Customer Satisfaction	Employee Turnover Rate	Organizational Performance	Software Quality
Bugs and Defect				**1.391**
Customer Satisfaction			0.023	
Employee Turnover Rate			0.105	
Lead Time	**0.313**			
Organizational Performance				
Software Quality			**0.211**	
Team Morale		**0.716**		

Source: Authors Compilation (SmartPLS 3.3.3 Output) 2023

The effect size, often denoted as f-square is depicted in table 4.9, this measures the magnitude of the relationship or impact of independent variables on a dependent variable in statistical analysis. This study assesses the effect sizes of various latent variables on "Organizational performance." Lead time have an f-square value of 0.313 on customer satisfaction. This suggests a large effect size, indicating that Lead time have a noticeable impact on customer satisfaction. In other words, changes or differences in Lead time can explain the variability in customer satisfaction largely. Team morale, on the other hand, has an f-square value of 0.716 on employee turnover rate. This indicates a relatively large effect size, suggesting that Team morale has a strong influence on employee turnover rate. Changes in Team morale explain the variability in employee turnover rate largely. Also, bugs and defect have an f-square value of 1.391 on software quality, indicating a large effect size. This suggests that bugs and defect also have a noticeable impact on software quality, explaining the variability in software quality largely. The effect sizes suggest that Lead time, Team morale as well as Bugs and defect, have relatively stronger

influences on customer satisfaction, employee turnover rate and software quality with Team morale having the largest impact.

Table 4.10 Bootstrapping Results Showing Path Coefficient for Structural Model

	Original Sample (O)	Sample Mean (M)	Standard Deviation (STDEV)	T Statistics (|O/STDEV|)	P Values	Hypotheses
Bugs and Defect -> Software Quality	0.763	0.764	0.032	23.721	**0.000**	**Supported**
Customer Satisfaction -> Organizational Performance	0.152	0.148	0.077	1.967	**0.050**	**Supported**
Employee Turnover Rate -> Organizational Performance	0.285	0.282	0.069	4.150	**0.000**	**Supported**
Lead Time -> Customer Satisfaction	0.488	0.490	0.061	8.058	**0.000**	**Supported**
Software Quality -> Organizational Performance	0.446	0.454	0.085	5.246	**0.000**	**Supported**
Team Morale -> Employee Turnover Rate	0.646	0.649	0.043	15.051	**0.000**	**Supported**

Source: Authors Compilation (SmartPLS 3.3.3 Output) 2023

The bootstrap path coefficient analysis depicted in table 4.10 was conducted to test the direct null hypotheses that; Lead time does not significantly affect customer satisfaction; Team morale does not significantly affect employee turnover rate; and bugs and defect does not significantly affect software quality.

The results indicate that there are significant effects of Lead time, Team morale, and bugs and defect as factors of agile project management method on customer satisfaction, employee turnover rate and software quality as a factor of organizational performances. A look at the path from Lead time, Team morale, and bugs and defect to customer satisfaction, employee turnover rate and software quality shows that the relationship between Lead time, Team morale, and bugs and defect against customer satisfaction,

employee turnover rate and software quality are statistically significant. The p-values are less than the conventional significance level of 0.05 and the T statistics are greater than 1.96, suggesting strong evidence to reject the null hypothesis. Therefore, the null hypotheses that; Lead time does not significantly affect customer satisfaction; Team morale does not significantly affect employee turnover rate; and bugs and defect does not significantly affect software quality are all rejected and the alternative hypotheses that Lead time significantly affect customer satisfaction; Team morale significantly affect employee turnover rate; and bugs and defect significantly affect software quality are all accepted.

4.6 Discussion of Findings

The objective one was to determine the effect of Lead time on customer satisfaction with the hypothesis been; Lead time does not significantly affect customer satisfaction. The result of the study revealed that all three factors, namely project complexity, resource allocation and team size, have statistically significant effects on customer satisfaction. The null hypothesis that Lead time does not significantly affect customer satisfaction is rejected based on this finding, indicating that these factors do influence customer satisfaction in a meaningful way. The implication of this is that ensuring employees' project complexity, resource allocation and team size guarantees Lead time and strengthens customer satisfaction.

The objective two was to determine the effect of Team morale on employee turnover rate with the hypothesis been; Team morale does not significantly affect employee turnover rate. The result of the study revealed that all three factors, namely work-life balance, compensation package and career development opportunity, have statistically significant effects on employee turnover rate. The null hypothesis that Team morale does not significantly affect employee turnover rate is rejected based on this finding, indicating that these factors do influence employee turnover rate in a meaningful way. The implication of this is that ensuring work-life balance, compensation package and career development opportunity guarantees Team morale and strengthens employee turnover rate among employees.

The objective three was to determine the effect of bugs and defect on software quality with the hypothesis been; Bugs and defect does not significantly affect software quality. The result of the study revealed that all three factors, namely code complexity, bug frequency, development process, have statistically significant effects on software quality. The null hypothesis that bugs and defect does not significantly affect software quality is rejected based on this finding, indicating that these factors do influence software quality in a negative way. The implication of this is that ensuring efficient and fast code complexity, bug frequency, development process guarantees bugs and defect reduces software quality.

Summary, Conclusion and Recommendations

5.1 Summary of Findings

Lead time and customer satisfaction**:** The study found a clear and significant link between lead time and customer satisfaction (b = 0.488, p = 0.000). This tells us that how well a team handles project complexity, allocates resources, and maintains the right team size directly shapes how quickly work gets done and how satisfied customers are with the results. Team morale and employee turnover rate: The results showed a strong positive relationship between team morale and employee turnover (b = 0.646, p = 0.000). When employees enjoy a good work-life balance, feel fairly compensated, and see room to grow in their careers, morale stays high and staff are less likely to leave the organisation. Bugs, defects and software quality: The analysis confirmed a significant negative effect of bugs and defects on software quality (b = 0.763, p = 0.000). In plain terms, the more complex the code, the higher the bug frequency, and the weaker the development process, the poorer the quality of software that gets delivered to end users.

5.2 Conclusion

This study set out to understand how agile project management affects organisational performance at Systemspec Holdings Limited, and the findings paint a fairly clear picture. Managing lead time well, through the right mix of project complexity control, resource planning, and team sizing, makes a real difference to how satisfied customers are. Keeping team morale high by taking care of employees through fair pay, career growth, and a healthy work environment helps hold teams together and reduces costly turnover. At the same time, poor handling of bugs and defects takes a measurable toll on software quality, which means that building strong development processes and keeping code clean is not optional but necessary. Taken together, the three findings confirm that agile project management, when practiced thoughtfully, supports better performance across the key areas that matter most to an IT organisation like Systemspec Holdings Limited.

5.3 Recommendations

i. Systemspec Holdings should pay closer attention to how lead time is managed across projects. Reducing unnecessary complexity, ensuring the right people and resources are available from the start, and keeping team sizes appropriate will go a long way in keeping customers happy and deliveries on time.

ii. The organisation should take deliberate steps to protect team morale. This means looking seriously at compensation, creating genuine opportunities for career growth, and making sure that the demands of work do not consistently spill into employees' personal lives. Teams that feel valued tend to stay, and staying teams perform better.

iii. Systemspec Holdings should invest in better defect management practices. Writing cleaner code, catching bugs earlier through regular automated testing, and continuously improving how the development process is run will reduce the drag that

bugs and defects currently place on software quality.

5.4 Contributions to Knowledge

This study is unique in the sense that review of past studies revealed that researchers made use of tools of analysis from content analysis, systematic literature review and case analysis but this current study made use of partial least square-structural equation modelling (PLS-SEM) which has not been so commonly used, to analyze the data obtained for the study. This tool of analysis is more robust and highly advanced when compared to previous ones This study has contributed to existing body of knowledge by using various aspects of agile project management method, which are Lead time, Team morale and bugs and defect which are scarce in the study of agile project management method. This study examined this element to examine new view and contribute new dimensions to the study which is a clear departure to previous works. The use of these factors' standout this study from previous ones. Hence, the study has contributed to existing knowledge by introducing Lead time, Team morale and bugs and defect as a sub-variable of agile project management method in enhancing organizational performance. This present study recognizes the important of these variables as one of the major sub-variables for agile project management method. The study has also contributed to existing knowledge by developing a comprehensive model that presents a clearer understanding of the effect of agile project management method on organizational performance. In this model are the sub-variables of managing each latent variable used to represent agile project management method as depicted in figure 4.1. This model type is unique of previous studies.

5.5 Areas for Further Study

This current study focused on the effect of agile project management method on organizational performance in Systemspec Holdings, with particular reference to employees of Systemspec Holdings. Future study could focus on agile project management method and organizational performance in specific states to encourage in-depth focus and also other industry apart from Systemspec Holdings.

References

Ahmed I., & Kolachi N. A (2013) "Employee Payroll and Training Budget: Case Study Of A Non-Teaching Healthcare Organization," *J. Bus. Econ. Res.,* vol. 11, no. 5, p. 229, doi: 10.19030/jber.v11i5.7838.

Al Mamun, C. A., & Hasan, M. N. (2017). *Factors affecting employee turnover and sound retention strategies in business organization: A conceptual view*. **Problems and Perspectives in Management, 15**(1), 63–71. https://doi.org/10.21511/ppm.15(1).2017.06

Al-Suraihi, W. A., Samikon, S. A., Al-Suraihi, A. A., & Ibrahim, I. (2021). *Employee turnover: Causes, importance and retention strategies*. *European Journal of Business and Management Research, 6*(3), 1–10. https://doi.org/10.24018/ejbmr.2021.6.3.893

Allen, D. G., Bryant, P. C., & Vardaman, J. M. (2013). Retaining Talent: Replacing Misconceptions with Evidence-Based Strategies. *Academy of Management Perspectives*, 27(2), 104-108.

Alshammari M. A., Qaied A. L., Al-Mawali H., & Matalqa M (2016). "What drives employee's involvement and turnover intentions: Empirical investigation of factors influencing employee involvement and turnover intentions?", *Int. Rev. Manag. Mark*., vol. 6, no. 2, pp. 298–306.

Ambler, S. W., & Lines, M. (2012). Introduction to the Agile Maturity Model. *Retrieved from* https://www.agilemodeling.com/essays/agileMaturityModel.htm

Ambler, S. W. (2017). Choose Your WoW!: A Disciplined Agile Delivery Handbook for Optimizing Your Way of Working. *Project Management Institute*.

Ambler, S. W., & Lines, M. (2020). Introduction to Disciplined Agile Delivery 2nd Edition: A Small Team's Journey from Scrum to Disciplined DevOps. *Project Management Institute.* Scott

Ambler, S. W. (2017). The Agile Data Method: A Disciplined Approach to Agile Database Development. *Project Management Institute*.

Anderson, D. J. (2010). Kanban: Successful Evolutionary Change for Your Technology Business. *Blue Hole Press.*

Beck, K. (2000). Extreme lean Explained: Embrace Change (2nd ed.*). Addison-Wesley*. Kent Beck's book is a foundational resource that explains the principles and practices of Extreme Programming, including Pair Programming, TDD, and Continuous Integration.

Beck, K., et al. (2001). "Manifesto for Agile Software Development." Agile Alliance.

Beedle, M., et al. (2001). "Agile Manifesto Principles." Agile Alliance. This reference pertains to the underlying principles of the Agile Manifesto, which provide a guiding framework for Agile software development.

Blackstone, J. H., Jr. (Ed.). (2013). *APICS dictionary: The essential supply chain reference* (14th ed.). APICS.

Biffl, S., et al. (2006). Software Quality: The Future of Systems- and Software

Development. *Springer*.

Cascio, W. F. (2006). The Cost of Employee Turnover. Workplace Health & Safety, 54(10), 397-399.

Cervone, H. F. (2017). Agile project management with Scrum. *CRC Press.*

Cervone, H. F. (2017). Essentials of Software Engineering. *Jones & Bartlett Learning.*

Chen Y., Wang C., & Cheng W (2010) "Structural investigation of the relationship between working satisfaction and employee turnover," *J. Hum*. vol. 6, no. 1, pp. 41–50.

Ciancarini, P., Kruglov, A., Pedrycz, W., Salikhov, D., & Succi, G. (2022, May). Issues in the adoption of the scaled agile framework. In *Proceedings of the 44th International Conference on Software Engineering: Software Engineering in Practice* (pp. 175-184).

Cohn, M. (2005). "Agile Estimating and Planning." *Pearson Education*. Mike Cohn's book focuses on Agile estimation and planning techniques, crucial aspects of Agile project management for effective delivery.

Cohn, M. (2010). Succeeding with Agile: Software development using Scrum. *Addison-Wesley.* Mike Cohn's book provides an in-depth understanding of Scrum, including the Product Backlog, Sprint Planning, Daily Stand-ups, Sprint Review, and Sprint Retrospective.

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. https://doi.org/10.1002/pmj.21410

Crispin, L., & Gregory, J. (2009). Agile Testing: A Practical Guide for Testers and Agile Teams. *Addison-Wesley.*

De Treville, S., Bicer, I., Chavez-Demoulin, V., Hagspiel, V., Schürhoff, N., Tasserit, C., & Wager, S. (2014). Valuing lead time. *Journal of Operations Management*, 32(6), 337-346.

Deemer, P., et al. (2010). "The Distributed Scrum Primer." DSDM Consortium. The Distributed Scrum Primer, authored by Pete Deemer and others, explores the distributed implementation of Scrum, an essential aspect of Agile project management in modern, global work environments.

Deemer, P., Benefield, G., Larman, C., & Vodde, B. (2010). Agile Retrospectives: Making Good Teams Great. *Pragmatic Bookshelf.*

DeLone, W. H., & McLean, E. R. (1992). Information systems success: The quest for the dependent variable. *Information Systems Research,* 3(1), 60-95.

Dragomir, C.-C., & Pânzaru, S. (2014). *The managerial performance*. *Review of General Management, 19*(1), 43–54.

Dybå, T., Dingsøyr, T., Moe, N.B. (2014). Agile Project Management. In: Ruhe, G., Wohlin, C. (eds) Software Project Management in a Changing World. *Springer, Berlin, Heidelberg*. https://doi.org/10.1007/978-3-642-55035-5_11

Fontana, R. M., Meyer, V., Jr., Reinehr, S., & Malucelli, A. (2015). *Progressive outcomes: A framework for maturing in agile software development*. **Journal of Systems and Software, 102**, 88–108. https://doi.org/10.1016/j.jss.2014.12.032

Fuggetta, A. (2000). Software Process: A Roadmap. International Conference on Software Engineering (ICSE).

Funk, C., & Segrest, B. (2017). The Power of People: Learn How Successful Organizations Use Workforce Analytics to Improve Business Performance. *Pearson FT Press*.

Gupta, A., & Chathoth, P. (2019). The Mediating Role of Employee Engagement in the Relationship Between Transformational Leadership and Employee Performance. *The International Journal of Human Resource Management,* 30(18), 2605-2629.

Hausknecht, J. P., Rodda, J., & Howard, M. J. (2009). Targeted Employee Retention: Performance-based and Job-related Differences in Reported Reasons for Staying. *Human Resource Management,* 48(2), 269-288.

Highsmith, J. (2009). "Agile project management: Creating innovative products (2nd ed.)." *Addison-Wesley*.

Holtom, B. C., Mitchell, T. R., Lee, T. W., & Eberly, M. B. (2008). Turnover and Retention Research: A Glance at the Past, a Closer Review of the Present, and a Venture into the Future. *The Academy of Management Annals,* 2(1), 231-274.

Humble, J., & Farley, D. (2010). Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation. *Addison-Wesley.*

Jaffari A. R., Hussain Z., Akhtar N., & Ur-Rehman K (2011) "Prime and sub-prime factors of employee voluntary turnover in boom phase of industry: Empirical evidence from banking sector of Pakistan," *African J. Bus. Manag.,* vol. 5, no. 15, pp. 6408–6414, doi: 10.5897/AJBM10.1556.

Jalote, P. (2008). An Integrated Approach to Software Engineering. *Springer*.

Janes, A., & Mathews, B. (2004). The Agile Organization. *Cutter IT Journal.*

Janes, A., & Mathews, S. (2004). Extreme Programming Installed. *Addison-Wesley.*

Jeffries, R., Anderson, A., & Hendrickson, C. (2000). Extreme Programming Installed. *Addison-Wesley*.

Kan, S. H. (2002). Metrics and Models in Software Quality Engineering. *Addison-Wesley.*

Kelly, A. (2014). *Bugs: Management Strategies*. Software Strategy Ltd.

Kiguolis, Linas. (2023). Software Development Frameworks.

Kotler, P., & Keller, K. L. (2016). Marketing Management. *Pearson*.

Lalsing, V., Kishnah, S., & Pudaruth, S. (2012). *People factors in agile software development and project management*. *International Journal of Software Engineering & Applications,* *3*(1), 117–137. https://doi.org/10.5121/ijsea.2012.3109

Leffingwell, D. (2018). SAFe: The Scaled Agile Framework: Lean Software and Systems Consortium. Dean Leffingwell's book is a comprehensive guide to SAFe, covering its principles, practices, and components, including Release Trains and

ART Planning.

Leffingwell, D. (2011). Agile Software Requirements: Lean Requirements Practices for Teams, Programs, and the Enterprise. *Addison-Wesley.*

Liker, J. K. (2004). The Toyota Way: 14 Management Principles from the World's Greatest Manufacturer. *McGraw-Hill Education*.

Manns, M. L., & Rising, L. (2005). "Fearless Change: Patterns for Introducing New Ideas." *Addison-Wesley.*

Marnada, P., Raharjo, T., Hardian, B., & Prasetyo, A. (2022). *Agile project management challenge in handling scope and change: A systematic literature review*. *Procedia Computer Science, 197*, 290–300. https://doi.org/10.1016/j.procs.2021.12.143

Masood, Z. A., & Farooq, S. (2017). *The benefits and key challenges of agile project management under recent research opportunities*. *International Research Journal of Management Sciences, 5*(1), 20–28.

Mellado, D., et al. (2014). Secure Software Development: A Security Programmer's Guide. *Jones & Bartlett Learning.*

Muhammad, U., Nazir, T., Muhammad, N., Maqsoom, A., Nawab, S., Fatima, S. T., Shafi, K., & Butt, F. S. (2021). *Impact of agile management on project performance: Evidence from I.T sector of Pakistan*. *PLoS ONE, 16*(4), e0249311. https://doi.org/10.1371/journal.pone.0249311

Niazi, M., & Wilson, D. (2007). Evaluating the Levels of Agile Maturity and Methods through the Agile Maturity Model. *In 40th Annual Hawaii International Conference on System Sciences* (pp. 1-10).

Osborne S., & Hammoud M. S (2017) "Effective Employee Engagement in the Workplace," *Int. J. Appl. Manag. Technol.,* vol. 16, no. 1, pp. 50–67, doi: 10.5590/IJAMT.2017.16.1.04.

Pfleeger, S. L., & Atlee, J. M. (2006). Software Engineering: Theory and Practice. *Pearson*.

PMI (Project Management Institute). (2021). The PMI Guide to Business Analysis. *Project Management Institute*

Poppendieck, M., & Poppendieck, T. (2003). Lean Software Development: An Agile Toolkit. *Addison-Wesley.*

Pressman, R. S. (2014). Software Engineering: A Practitioner's Approach. *McGraw-Hill Education.*

Reichheld, F. F., & Schefter, P. (2000). E-loyalty: Your Secret Weapon on the Web. *Harvard Business Review,* 78(4), 105-113.

Richard, P. J., Devinney, T. M., Yip, G. S., & Johnson, G. (2009). *Measuring organizational performance: Towards methodological best practice*. *Journal of Management, 35*(3), 718–804.

Rowold, J. (2011). *Relationship between leadership behaviors and performance: The moderating role of a work team's level of age, gender, and cultural heterogeneity. Leadership & Organization Development Journal, 32*(6), 628–647.

https://doi.org/10.1108/01437731111161094

Rust, R. T., & Thompson, D. V. (2006). How Does Customer Satisfaction Drive Loyalty? *Journal of Marketing*, 69(4), 32-51.

Scholz, J., Sieckmann, F., & Kohl, H. (2020). Implementation with agile project management approaches: Case Study of an Industrie 4.0 Learning Factory in China. *Procedia Manufacturing*, 45, 234-239.

Schwaber, K., & Beedle, M. (2001). Agile Software Development with Scrum. *Prentice Hall.*

Schwaber, K., & Sutherland, J. (2017). "The Scrum guide." *Scrum.org*. The Scrum Guide, authored by Ken Schwaber and Jeff Sutherland, provides a detailed understanding of Scrum, a widely adopted Agile framework, emphasizing roles, events, and artifacts in Scrum.

Serrador, P., & Pinto, J. K. (2015). *Does Agile work? A quantitative analysis of agile project success*. *International Journal of Project Management, 33*(5), 1040–1051. https://doi.org/10.1016/j.ijproman.2015.01.006

Shahzad K., Rehman U., Shad I., Gul A., & Amnulah Khan (2011) "Work-life policies and job stress as determinants of turnover intentions of customer service representatives in Pakistan," *Eur. J. Soc. Sci.,* vol. 19, no. 3, pp. 403–411.

Sharma, M., Luthra, S., Joshi, S., & Joshi, H. (2022). Challenges to agile project management during COVID-19 pandemic: an emerging economy perspective. *Operations Management Research,* 15, 461 - 474.

Shi, W. W., Cao, W. C., & Xun, J. L. (2022). *Impact of agile project management on the competitiveness of Chinese software development organizations*. *Journal of Entrepreneurship & Project Management, 6*(4), 1–13. https://doi.org/10.53819/81018102t4067

Shrivastava, A., Jaggi, I., Katoch, N., Gupta, D., & Gupta, S. (2021). *A systematic review on extreme programming*. *Journal of Physics: Conference Series, 1969*(1), 012046. https://doi.org/10.1088/1742-6596/1969/1/012046

Singh J., & N. B. Kassie (2018) "User's perspective of software quality*," in 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), Piscataway, NJ, USA.*

Sohrabi, B., Tetard, F., Mayrhofer, U., & Ferreira, P. (2019). Agile project management and decision-making in project-based organizations: A review and research agenda. *International Journal of Project Management, 37(3)*, 559-576.

Spinellis, D., & Vlachopoulou, M. (2020). Agile project management principles in practice: An empirical investigation. *Journal of Systems and Software, 160*, 110549. doi:10.1016/j.jss.2019.110549

Todorović, M., Toljaga-Nikolić, D., & Bjelica, D. (2018). *People-oriented principles and values of agile project management*. *European Project Management Journal, 8*(2), 3–8. https://doi.org/10.18485/epmj.2018.8.2.1

Womack, J. P., Jones, D. T., & Roos, D.(1990). The Machine That Changed the World:

The Story of Lean Production. *Free Press*.

Zakaria N. H., Hamdan A. R., Yahaya J., & Deraman A (2016) "User centric software quality model for sustainability: a review," *Lecture Notes on Software Engineering, vol*. 4, no. 3, p. 199.

Zhang, P., et al. (2011). HCI in Business*: First International Conference*, *Springer*.

IJSSMR