South African Journal of Information Management
ISSN: (Online) 1560-683X, (Print) 2078-1865

A contextualised model of the use of agile technique in South African software development team

Authors:
Mmaphuti C. Teffo[1]
Khuliso Sigama[2]
Fred Kanobe[3]

Affiliations:
[1]Department of Informatics, Faculty of Information and Communication Technology, Tshwane University of Technology, Pretoria, South Africa

[2]Department of Computer Science, Faculty of Information and Communication Technology, Tshwane University of Technology, Pretoria, South Africa

[3]Department of Mathematics and Statistics, Faculty of Science, Kyambogo University, Kampala, Uganda

Corresponding author:
Khuliso Sigama,
sigamak1@gmail.com

Dates:
Received: 22 Sept. 2022
Accepted: 20 Feb. 2023
Published: 24 Apr. 2023

How to cite this article:
Teffo, M.C., Sigama, K. & Kanobe, F., 2023, 'A contextualised model of the use of agile technique in South African software development team', *South African Journal of Information Management* 25(1), a1617. https://doi.org/10.4102/sajim.v25i1.1617

Background: Software development teams are challenged with the adoption of the most appropriate software development methodology. Despite their acceptance of agile methodologies, still many countries use traditional software development methods.

Objectives: The major objective of this study was to develop a contextualised model for the adoption of agile methodologies in software development teams in South Africa.

Method: Our study identified the factors that contributed to the adoption of agile software development teams in South Africa by conducting a factor analysis. We used descriptive statistics to determine the frequencies of the participant's demographics and situational variables. A composite research model based on technology, organisation and environment, individual characters and culture constructs was constructed and analysis was performed by applying the statistical techniques of correlation analysis, regression analysis and structural equation modelling.

Results: The result of the analysis revealed that many teams do not have a specific model that they use in agile development methodology. Technology, organisation, environment and culture constructs were found to have an impact on teams in adopting agile methodologies.

Conclusion: The study demonstrated that organisations lack appropriate models of agile methodologies adoption to inform their decision making. Therefore, adopt the developed model in taking informed decisions for their software development methodologies. The study makes a practical contribution to management and practice.

Contribution: This study improves on software development approaches in developing countries. The developed model can be adopted as a theory in future research. It contributed literature to the body of knowledge.

Keywords: agile; adaptive software development; rapid application development; Scrum; Kanban system; development teams.

Introduction

Software is developed to meet the specific needs of the organisations, to perceive the needs of potential users and for personal use. However, software development is a complex task that involves coordination of different disciplines and skill sets (Sedano, Ralph & Peraire 2017). Therefore, it is important for the software development teams to select the appropriate methodology to carry out their tasks. Software development is a diverse phenomenon with organisation required to adopt new technologies to remain competitive. Thus, neither software development methodology is a silver bullet nor a 'one size fits all' (Kuhrmann 2017). The success of software development is generally determined by how effectively knowledge is shared between development teams (Field, Anderson & Eder 2014).

Software development teams can choose from two main software development methodologies, namely traditional and agile. Each methodology consists of various approaches in developing software. According to Alqudah and Razali (2016), the traditional approach includes waterfall method, V-model, rational unified process (RUP), spiral, prototyping and rapid application development (RAD). On the other hand, agile approaches include Extreme Programming (XP), Scrum (Iterative Incremental Development), Function Driven Development (FDD), Lean Software Development (LSD) and Crystal Development Method (CDM) (Kamil 2018). These methodologies assist the software development team in developing the software required by the business. They are not themselves software development solution.

Organisations are moving towards adopting agile methodologies that have several benefits that include but are not limited to higher product quality, faster return on investment, reduced risk, cheaper solutions in a short time and higher customer satisfaction (Jadhav, Kaur & Akter 2022). Agile methodology can also be used to make changes in designs and improve communications (Kamil 2018). Agile methodologies encourage software development teams to work in collaboration (Kaleshovska et al. 2015).

Agile has a higher impact on team productivity and employee satisfaction compared with traditional software methodologies such as waterfall and others (Rai & Dhir 2014). In addition, Holgeid et al. (2021) found that agile minimises repetitive planning, unnecessary meetings and excessive documentation while reducing quality defects and enhancing product features. Moreover, agile offers the potential to adapt to changing priorities, improve customer engagement, bring teams together and reduce risks (Kuhrmann 2017).

Background

Organisations need agility as cornerstones of success to achieve this objective they must flatten their structures and transfer decision making from upper management lower operational levels through team-based autonomous structures. On the one hand, software development teams need to leverage this experience to have a success with projects and better performance evaluation. However, traditional software development teams are challenged with complex problems that hinder their progress and quick completion of projects (Wiesche 2021). Some of these challenges include effective communication, cultural differences, coordination, trust, asymmetry in processes, policies and standards, people management/conflict resolution as well as identification of roles and responsibilities (Sedano et al. 2017). This led to increasing need of finding better and quicker ways of developing software.

In South Africa, for example, Edison, Wang and Conboy (2022), have seen an increasing number of large companies wishing to remain relevant as they compete with leaner, more technology-driven competitors. However, this desire has been challenged by the traditional organisational culture that conflicts with modern software development methodologies such as agile. As a result, there has been resistance to change and a lack of management support for adopting these methodologies within organisations (Kuhrmann 2017). For instance, despite the extensive growth and acceptance of agile methodologies in many developed countries, there is still limited literature that supports agile adoption in the developing countries context (Da Silva 2010) Surprisingly, there are no comprehensive studies on how agile implementation fails and derailments occur (Jadhav et al. 2022).

Currently, software developing teams are faced with the challenges of ever-increasing software complexity, dynamic user requirements, low budgets and tight schedules (Da Silva 2010). For software developers to fully understand user requirements, frequent interaction with customers is required. However, this interaction is lacking in current teams that are still using traditional software development methods, and it leads to delays in signing off developed software because of changing software demands (Matharu et al. 2015). Another challenge is that in most large organisations, employees are hesitant to take accountability of their daily operations and it creates a lot of pressure and resistance among them. In an agile organisation, for example, if someone is not contributing, the fact will come to light very quickly, as opposed to a complex, waterfall organisation (Al-Saqqa, Sawahla & Abnelnabi 2020).

The main objective of the study was to develop a contextualised model for the adoption of agile methodologies in software development teams in South Africa. The key question addressed by the study was: How can a contextualised model be developed for the adoption of agile methodologies in software development teams in South Africa?

Related work

Field et al. (2014) conducted a study to establish the influence of individuals' behavioural in software engineering teams. The study observed that engineers, software developers and all other stakeholders play a crucial role in determining project outcomes and success. Their study revealed that most of the role individuals play in software development is critical, research has been focusing mainly on the technology and processes while paying little attention to individual characteristics and behaviour.

Agile-based software development provides an effective solution to the challenges presently faced by the software industry including ever-increasing software complexity, dynamic user requirements, tight schedules, low budgets as well as frequent interaction between developers and customers. The top five reasons for adopting agile methodology were the need for accelerated product delivery, better management of changing priorities, improved software maintenance, greater deliverable predictability and simplified development process (Matharu et al. 2015).

Noteboom et al. (2021) conducted a study on project management practice called agile project management (APM) that went beyond traditional processes to involve how projects are managed. The study emphasised the adaptive processes that respond to uncertainty. The study identified 11 factors that drive adoption and 13 factors that influence success dimensions of projects, cultures and teams. The study's findings provide insights into current APM adoption and usage for project managers and researchers alike.

Agile methodology can be used to manage software development projects using literature and industry best practices. The study looked at specific characteristics such as capability, adequacy risks and effectiveness. Furthermore, the study also highlighted that there are no mathematical

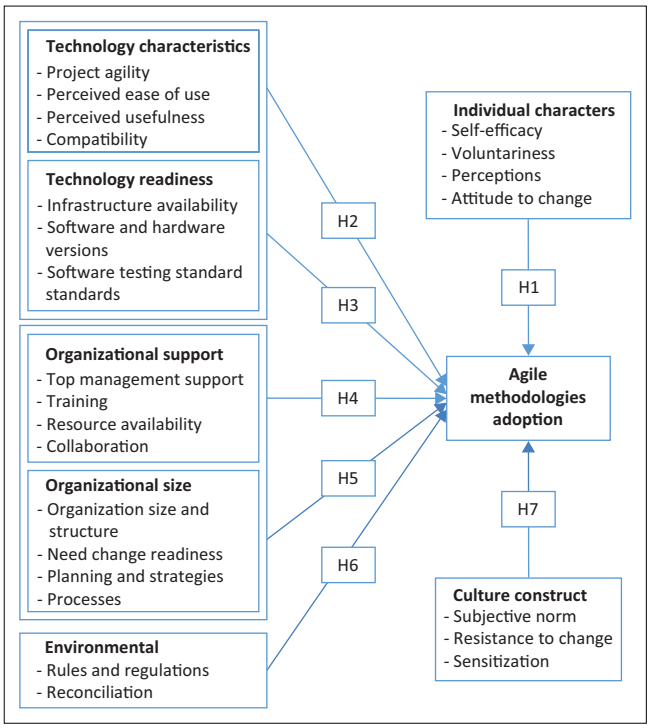


FIGURE 1: The conceptual model (Rogers 1995; Depietro, Wiarda & Fleischer 1990; Davis 1989).

models and frameworks addressing agile methodology specifically for managing software development projects in IT industries, which are usually used for traditional methods. Moreover, their research findings suggest that in the future, software development projects in the IT industry should use agile methodology (Dhole & Kumar 2018).

Agile methodologies

In the lifecycle of a project, agile methodology encourages continuous iterations of development and testing of software during the software development phase (Zia 2018). With agile methodology, both software development and testing occur concurrently. Al-Saqqa, Sawala and Abdelnabi (2020) explain that the aim of agile is to quickly deliver the software with complete and functional components that can be made available to the users after completion. As defined by Da Silva (2010), agile methodology is a set of iterative and incremental methods for developing software, with requirements and solutions developing through collaboration between self-organising, cross-functional teams. It is flexible and allows changes to be made easily even after the project starts (Kumar, Maheshwary & Malche 2019).

According to Eason (2016), an agile methodology is aimed at responding effectively to change and delivering the software as quickly as possible. He alludes that, unlike traditional methodologies, agile is not document oriented and that instead of creating the functional specification, software developers rely on storyboards that are created. Zia (2018) asserts that with agile methodology there is no detailed requirements specification but only users' stories that are iteratively written during the software development process.

In order to achieve agile software development's ultimate value, teams must deliver high-quality functional software faster (Panesar et al. 2021). The study also adds that most software development teams aim to deliver a software in a short period and at a minimal cost. Some of the agile methodologies used worldwide are discussed next.

Scrum

This methodology is characterised by its use of self-organised, cross-functional and empowered teams that are divided into small, manageable work cycles called sprints (Bhavsar, Shah & Gopolan 2020). In any project, Scrum brings the benefits of continuous improvement, transparency, adaptability, continuous feedback, continuous delivery of value, early delivery in high quality, faster problem resolution, sustainable pace, customer centricity, efficient development process and effective deliverables (Alqudah & Razali 2016).

Kanban

Kanban originated in the automobile industry and has been a method that optimises every stage of production and ensures that production is on time. Despite its origins in manufacturing, Kanban has evolved into an important project management tool today (Alaidaros, Omar & Romli 2021). A Kanban project management approach allows you to visualise your work and ensure the smooth flow of your work. A Kanban board visualises your work into three columns using the Kanban method. The columns are To Do, In Progress and Complete. It is an ideal system for knowledge work that is not easily measurable (Kumar et al. 2019).

Lean

In the process of developing software, lean development reduces the number of unnecessary steps. It allows development to be faster and cheaper, saving the two most important resources – time and money (Umar, Janga & Shaba 2021). It prevents any losses. It is based on five principles, namely optimising value streams, creating value for the customer, eliminating waste, empowering people and continually improving (Zia 2018).

Extreme programming

Through XP, development cycle processes are broken down into manageable smaller segments, thus solving the traditional asymmetry in lengths of development processes (Noteboom et al. 2021). Each segment undergoes all the phases of the development cycle. As a result, change costs are lowered. Its fundamentals include communicating simply, getting feedback, going with the flow and respecting others (Panesar et al. 2021). With XP, small to midsize teams can easily create high-quality software and adapt to changing and evolving requirements. Exploration, planning, iterations to release, productionising, maintenance and death are the six phases of XP (Jadhav et al. 2022).

Adaptive software development

Adaptive software development (ASD) works in iterations such as other agile methodologies mentioned here. However,

it differs from the rest by focusing on components rather than tasks. It is composed of a collection of features delivered in parallel. They are classified as primary components, technology components and support components (Kumar et al. 2019). It allows projects to be completed on time and, in some cases, even earlier. Users must be involved heavily in the process. There are three stages to the realisation of an ASD project: Speculate, collaborate and learn (Jadhav et al. 2022).

Theoretical background

This study used a composite model of integrated constructs from diffusion of innovation (DOI) (Rogers 1995), technology, organisation and environment (TOE) (Depietro, Wiarda & Fleischer 1990) and technology acceptance model (Davis 1989). The model is conceptualised to suit this research study. The conceptualised model indicates that individual characteristics, technology characteristics, technology readiness, organisational support, organisational size, environmental and culture constructs influence agile methodologies adoption. The constructs are explained as follows:

Individual characteristics

In the majority of scenarios for technology uptake and use, individual characteristics are crucial. Given the potential of mistakes, ambiguity and opacity in this particular situation, personality traits connected to these technological features may be especially pertinent. Agile methodologies are more likely to be adopted by people who are more likely to be risk takers, tolerant of uncertainty and eager to learn (Venkatesh & Davis 2000). Individual characteristics can be explained using sub-constructs: self-efficacy, voluntariness, perception and attitude to change:

> **Hypothesis 1 (H1):** Individual characters has a positive contribution towards the agile methodologies' adoption.

Technology characteristics

Depending on the type of research, technology characteristics can be investigated as either objective attributes or employee perspectives. The specific traits of agile methodologies adoption in relation to the many already outlined possible difficulties may be important. Project agility, perceived ease of use, perceived usefulness and compatibility (Depietro et al. 1990). Thus, we postulate the hypothesis:

> **Hypothesis 2 (H2):** Technology characteristics have a positive contribution towards the agile methodologies' adoption.

Technology readiness

Technology readiness construct can be seen of as an all-encompassing mental state that results from a gestalt of mental facilitators and inhibitors that together influence a person's propensity towards technologies (Parasuraman 2000). In this study, technology readiness include infrastructure availability, software and hardware version and software testing standards. Hence, the hypothesis:

> **Hypothesis 3 (H3):** Technology readiness has a positive contribution towards the agile methodologies' adoption.

Organisational support

How well a company supports its employees technical innovation will be influenced by the use of a specific technology or system. Providing incentives for innovation adoption and ensuring the availability of financial and technical resources for innovation have positive effects on the adoption of technical innovation (Depietro et al. 1990). Hence the Hypothesis:

> **Hypothesis 4 (H4):** Organisational support has a positive contribution towards the agile methodologies' adoption.

Organisational size

Given that it gives an organisation more chances to find an innovation, organisational scale may be a significant factor in the early adoption of disruptive technologies. Given that organisations are interconnected by a web of networks, which affects their behaviour, this may be the case (Pavitt 1999). Hence the hypothesis:

> **Hypothesis 5 (H5):** Organisational size readiness has a positive contribution towards the agile methodologies' adoption.

Environmental factors

According to Lee and Chen (2019), environmental factors directly influence individuals' self-efficacy to perform good behaviours. Depietro et al. (1990) also emphasised that environmental factors are believed to directly influence self-efficacy, which, in turn, influences subsequent interests, choice of goals and actions: Hence, the hypothesis:

> **Hypothesis 6 (H6):** Environmental construct has a positive contribution towards the agile methodologies' adoption.

Culture

Subjective norm, resistance to change, sensitisation and attitude are some of the factors that influence any technology adoption. A habit that applies to an organisation is organisational culture. Every organisation has its own traditions and routines. The fundamental presumptions and beliefs of an organisation's workforce are then established and transmitted in order to address issues with external adaption and internal integration (Limaj & Bernroider 2019). Hence, the hypothesis:

> **Hypothesis 7 (H7):** Culture construct has a positive contribution towards the agile methodologies' adoption.

Materials and methods

This study adopted positivism paradigm. In positivism research, answers to the research question or hypothesis are precise, verifiable, systematic and theoretical (Park, Konge & Artino 2020). This study required precise and systematic answers to address its research questions. A quantitative survey questionnaire to collect data was needed for the study. A questionnaire survey helps to reach many participants (Pandey & Pandey 2015). In this situation, the study sampled

out the participants from software development teams of organisations that develop software for other small and corporate enterprises.

Close-ended questionnaires were used to collect data for this study. Closed-ended questionnaires are measuring instruments with predetermined responses from which participants can choose for each item to be measured (Walliman 2011). Data were collected by distributing the questionnaires to stakeholders who were sampled out from the population of software development teams. To ensure anonymity, we got a contact person from each of the participating organisations who distributed the questionnaires to the participants.

Using the sample tool of Krejcie and Morgan (1970), a sample size of 150 participants was achieved. The sample population for this research study included 150 participants from Road Accident Fund (RAF) and Electronic Toll Collections (ETC). This study used purposive sampling to select only those organisations that deal with software development for other organisations. The online survey monkey questionnaires were sent to 150 participants combined from RAF and ETC. These participants were working at the software development teams. We had 101 combined responses from the software development teams both at RAF and ETC.

IBM SPSS 24.0 was used for descriptive statistics and for testing the validity and reliability of data collected, and Amos 24 was used to run the structural equation modelling (SEM) to test the model fitness. Out of the 150 questionnaires distributed, 101 were found to be useful for analysis. In order to design the survey questionnaire, the guidelines as provided by Babbie (2005) were followed.

The questionnaire was structured as a multiple-choice closed statement item where participants were asked to select their level of agreement with each statement. The instrument was designed primarily to assess the adoption of agile methodologies in software development teams in South Africa.

Ethical considerations

Ethical clearance was obtained from Research Ethical body. The study also ensured that participants' privacy and confidentiality were maintained and all issues raised by them were respected and taken into consideration.

Results

The results reveal that 59% of respondents aged between 31 and 40 years, 27% are between were ages of 41 and 50 years and 13% of the respondents are between the ages of 21 and 30 years. In this study, they were few people from the age of 51 and above, which is only 1% of the population.

The study further revealed that there were 62% of male respondents and 38% of female respondents. On level of education, 61% of the respondents had obtained their BTech Degree in IT, 20% had the National Diploma, 10% had master's degree, 5% had certificates and 4% had PhD degrees.

The results shows that participants were from different employment positions. Thirty nine per cent of the respondents were the project managers, 25% were the test analysts and 22% were from the software developers and 14% were system analysts. Respondents had different working experience. Forty per cent of the respondent's had a work experience between 11 and 15 years, 23% had between 16 and 20 years, 17% had between 0 and 5 years, 12% had between 6 and 10 years, 8% had between 21 and 25 years and only 1% had a work experience of 26 years and above.

Five factors, namely technology factors, organisational factors, environmental factors, individual characters and culture were extracted using the EFA with high reliability coefficients between 0.55 and 0.90. Factor loadings close to 5 and above were used in this study. Technology factors had a very high reliability coefficient of 0.84. Environmental factors and individual characters had a low reliability value of 0.60. Table 1 shows the factor loadings.

The five factors extracted from the EFA explained only 43.6% of the variation of the data. This is an indication that the extracted factors are not adequate in explaining the variation in the sample. Figure 2 illustrates this as well.

An analysis of SEM was conducted to assess the model hypotheses and measure the relationships between the five constructs. This study used SEM to answer the research

TABLE 1: Factor loadings.

Variable	F1	F2	F3	F4	F5	Cronbach's alpha
QAQ8	0.591	-	-	-	-	0.84
QAQ9	0.526	-	-	-	-	-
QAQ10	0.637	-	-	-	-	-
QAQ11	0.626	-	-	-	-	-
QAQ15	0.682	-	-	-	-	-
QAQ16	0.444	-	-	-	-	-
QAQ4	0.422	-	-	-	-	-
QAQ23	0.455	-	-	-	-	-
QAQ25	0.536	-	-	-	-	-
QAQ7	-	0.570	-	-	-	0.79
QAQ27	-	0.439	-	-	-	-
QAQ29	-	0.552	-	-	-	-
QAQ30	-	0.737	-	-	-	-
QAQ31	-	0.669	-	-	-	-
QAQ37	-	0.503	-	-	-	-
QAQ1	-	-	0.573	-	-	0.60
QAQ5	-	-	0.520	-	-	-
QAQ22	-	-	0.570	-	-	-
QAQ32	-	-	-	0.497	-	0.60
QAQ34	-	-	-	0.544	-	-
QAQ35	-	-	-	0.655	-	-
QAQ21	-	-	-	-	0.759	0.67
QAQ24	-	-	-	-	0.547	-
QAQ20	-	-	-	-	0.696	-

question, 'What factors influence the adoption of the agile methodologies within the software development teams?' SEM was performed using the lavaan package in *R* (Rosseel 2012).

Figure 3 illustrates an SEM fit diagram, which shows that technology factors are positively correlated with organisational factors, environment factors and culture, while technology factors are negatively correlated with individual factors. The perceived ease of use of information technology has been widely studied and is not an exception when it comes to the adoption of agile methodologies (Tahar et al. 2020). There have been several studies conducted on the perceived ease of use of technologies in developed and developing countries. The objective of this study was to explore the perceived ease of use of agile methodologies. In addition, it was confirmed that perceived usefulness might not predict adoption of agile methodologies in developing countries. Individual characters still believe using a new technology requires effort (He 2018).

Organisational factors have a positive correlation with the environment factors, individual factors and culture. Fernandes (2018) found that organisational culture has a positive and significant impact on environmental, individual and cultural factors. Environmental factors have a positive correlation relationship with technology, individual factors and cultural factors (Jian & Qin 2020). Technology factors negatively affect individual factors although there are positive correlations between technology and culture, environment and a very strong positive correlation with organisation (He, Chen & Kitkuakul 2018). Culture is positively correlated with technology, organisational, environment and individual. Individual factors have a very strong relationship with the organisational factors and cultural factors have a positive relationship with all factors. However, individual factors have a negative relationship with technology factors.

Table 2 shows the accepted and rejected hypotheses. Ogee et al. (2015) indicate that a *p*-value of ≤ 0.05 indicates strong evidence in relation to hypothesis; it indicates that there is 95% confidence level concerning the results. A *p*-value of > 0.05 indicates no significance between the constructs, which means the hypothesis is rejected.

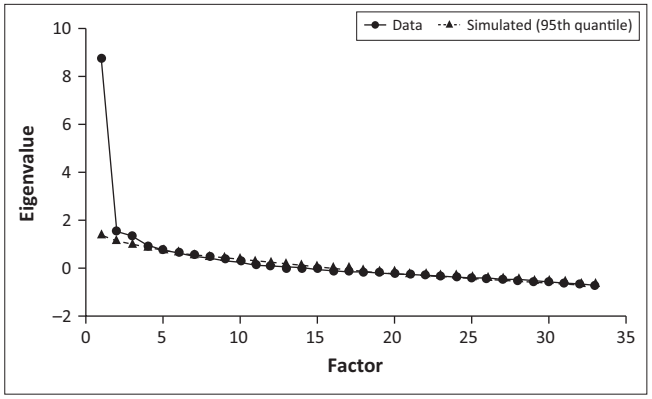


FIGURE 2: Scree plot.

Discussion

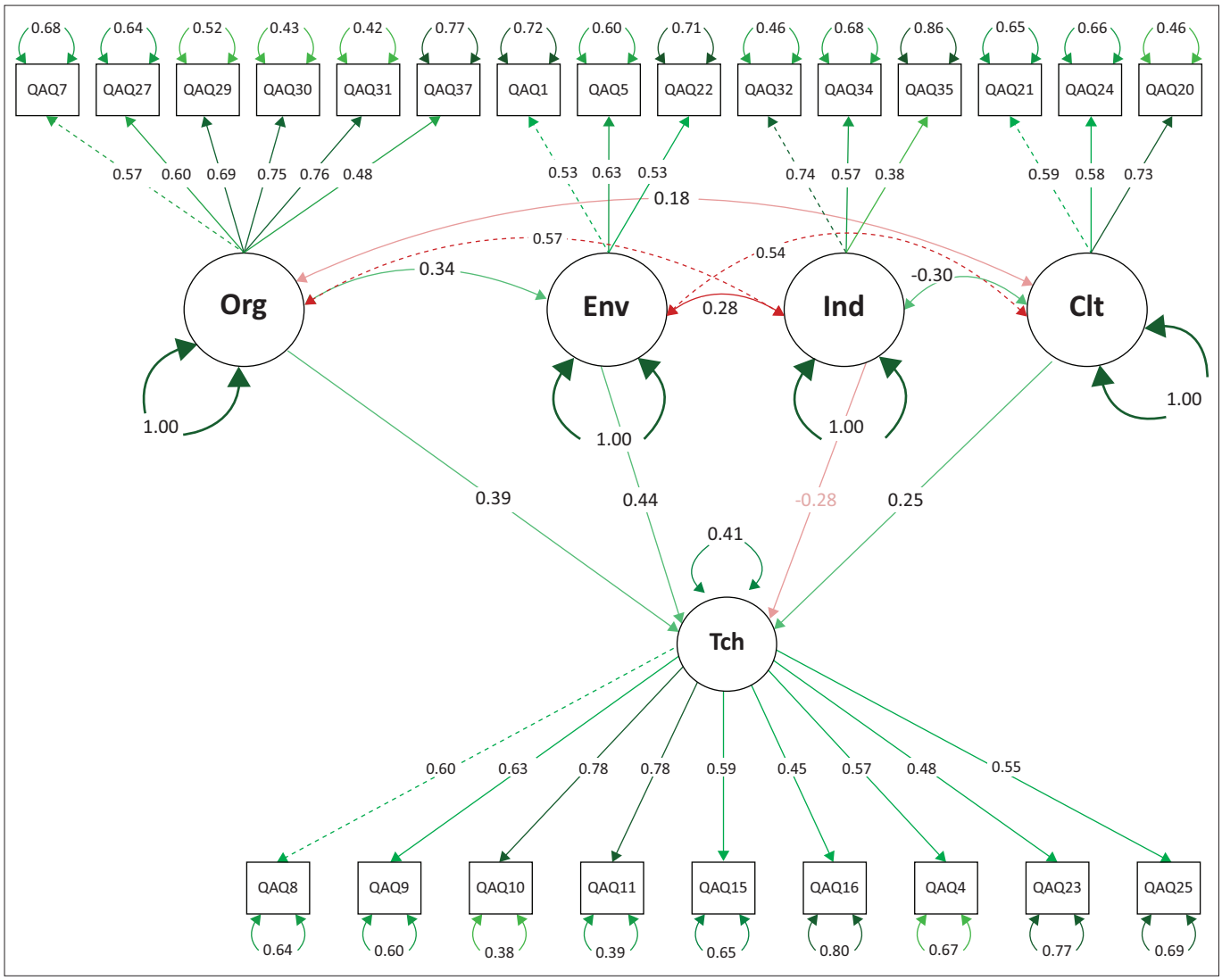
Various strategies are implemented by organisations to remain relevant in their respective industries and every industry is different. In order to remain competitive and ahead of their competitors, organisations must constantly evolve. Nevertheless, organisations can achieve competitiveness in other ways. There is no absolute solution to competitiveness based on the findings of this study. Therefore, future studies should focus on other methods used by organisations to become competitive.

Based on this research study, a contextualised model of agile methodologies adoption in software development teams in South Africa was created. Nevertheless, industries may have different challenges and levels of technology usage and development. For this reason, the findings of this study may not be generalisable across industries. Therefore, this study recommends that participants from more than one industry participate in future research and that a comparative analysis be used to compare the findings across these different industries.

A limited research has been conducted in relation to the performance of the agile methodology. Future researchers could wrench on this study as literature source to extend studies in determining the adoption of agile methodologies. The study also educates and informs software development in terms of the agile methodology benefits and implementations. The factors identified by this study can be leveraged to adopt agile software development methodologies. These factors can inform policymakers and top management of private and public institutions when adopting agile methodologies.

A key limitation of the research study is that it was undertaken only in the private sector, so the research needs to be expanded to include public software companies as well; the study followed a quantitative approach and data were analysed statistically. The mixed methods approach could have worked better because it uses strategies such as comparative analysis that could have been explored too.

This study used the lavaan package in R (Rosseel 2012) to conduct SEM. To examine model fit indications in the research, fit indices such as absolute, incremental and parsimonious were used. Therefore, the goodness of fit is guaranteed for this research, even though one category does not meet the criteria. The results show that organisational factor is significant with a *p*-value of 0.020, which is less than 0.05 and environment factor is also significant with a *p*-value of 0.031. However, individual and culture seem not to be significant factors of agile as their *p*-values are greater than 0.05. Individual is bigger with a *p*-value of 0.108 as well as culture is bigger with a *p*-value of 0.141. Therefore, our final model is shown is Figure 4.



Org, Organisational; Env, Environmental; Ind, Individual; Clt, Cultural; Tch, Technological; QAQ, Quantitative Attitude Questionnaire.

FIGURE 3: Structural equation modelling fit final model.

TABLE 2: Hypothesis testing

Hypothesis	*P*	Accepted/ Rejected
Hypothesis 1 (H1): Individual characters have a positive contribution towards the agile methodologies' adoption.	0.243	Rejected
Hypothesis 2 (H2): Technology characteristics have a positive contribution towards the agile methodologies' adoption.	0.000	Accepted
Hypothesis 3 (H3): Technology readiness has a positive contribution towards the agile methodologies' adoption.	0.014	Accepted
Hypothesis 4 (H4): Organisational support has a positive contribution towards the agile methodologies' adoption.	0.023	Accepted
Hypothesis 5 (H5): Organisational size readiness has a positive contribution towards the agile methodologies' adoption.	0.000	Accepted
Hypothesis 6 (H6): Environmental construct has a positive contribution towards the agile methodologies' adoption.	0.000	Accepted
Hypothesis 7 (H7): Culture construct has a positive contribution towards the agile methodologies' adoption.	0.000	Accepted

Conclusion

The findings of this study contribute to the software development and project management body of knowledge. This knowledge could assist software development teams in the adoption, use and implementation of agile methodologies. The designed framework could be used to assist software

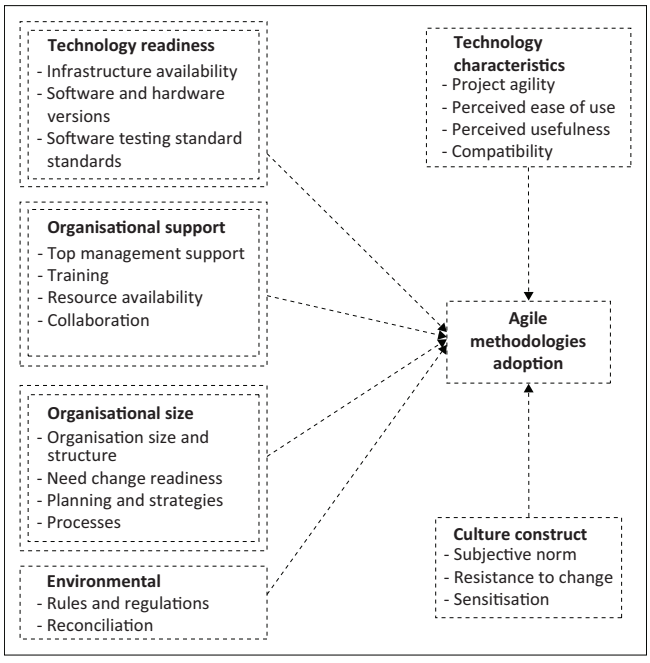


FIGURE 4: Final deduced model.

development teams to adopt agile methodologies with a success and enjoy the benefits. The designed conceptual framework helps in determining factors influencing the adoption of agile methodologies. This study found that technology, organisation, environmental and culture factors were significant, whereas individual factors were found to have no significance in influencing the adoption of agile methodologies. This is significantly important in understanding the relationship between these factors and the adoption of agile methodologies.

Acknowledgements

The authors would like to thank the Road Accident Fund (RAF) and Electronic Toll Collections (ETC) for participating in the study and answering the questionnaire.

Competing interests

The author(s) declare that they have no financial or personal relationship(s) that may have inappropriately influenced them in writing this article.

Authors' contributions

M.C.T. conducted the research as part of Master of Computing degree at Tshwane University of Technology under the supervision of F.K. and K.S.

Funding information

This research study received no specific grant from any funding agency in the public, commercial or not-for-profit sectors.

Data availability

The data that support the findings of this study are available on request from the corresponding author, K.S.

Disclaimer

The views and opinions expressed in this study are those of the authors and do not necessarily reflect the official policy or position of any affiliated agency of the authors.

References

Al-Saqqa, S., Sawalha, S. & Abdelnabi, H., 2020, 'Agile software development: Methodologies and trends', *International Journal of Interactive Mobile Technologies* 14(11), 246–270. https://doi.org/10.3991/ijim.v14i11.13269

Alaidaros, H., Omar, M. & Romli, R., 2021, 'The state of the art of agile Kanban method: Challenges and opportunities', *Independent Journal of Management & Production* 12(8), 2535–2550. https://doi.org/10.14807/ijmp.v12i8.1482

Alqudah, M. & Razali, R., 2016, 'A review of scaling agile methods in large software development', *International Journal on Advanced Science, Engineering and Information Technology* 6(6), 828–837. https://doi.org/10.18517/ijaseit.6.6.1374

Babbie, E., 2005, *The basics of research*, Thomson Wadsworth, Belmont, CA.

Bhavsar, K., Shah, V. & Gopalan, S., 2020, 'Scrum: An agile process reengineering in software engineering', *International Journal of Innovative Technology and Exploring Engineering (IJITEE)* 9(3), 840–848. https://doi.org/10.35940/ijitee.C8545.019320

Da Silva, F., 2010, 'Challenges and solutions in distributed software development project management: A systematic literature review', in Davis, F.D., (eds.), *2010 International Conference on Global Software Engineering*, IEEE.

Davis, F.D., 1989, 'Perceived usefulness, perceived ease of use, and user acceptance of information technology', *MIS Quarterly* 13(3), 319–340. https://doi.org/10.2307/249008

Depietro, R., Wiarda, E. & Fleischer, M., 1990, 'The context for change: Organization, technology and environment', *The Processes of Technological Innovation* 199(0), 151–175.

Dhole, R.D. & Kumar, K.S., 2018, 'Adoption models for agile software development projects', *Advances in Computational Sciences and Technology* 11(1), 69–76. https://doi.org/10.37622/ACST/11.1.2018.69-76

Eason, O.K., 2016, 'Information systems development methodologies transitions: An analysis of waterfall to agile methodology', University of New Hampshire, *Honors Theses and Capstones.*

Edison, H., Wang, X. & Conboy, K., 2022, 'Comparing methods for large-scale agile software development: A systematic literature review', *IEEE Transactions on Software Engineering* 48(8), 2709–2731. https://doi.org/10.1109/TSE.2021.3069039

Fernandes, A.A.R., 2018, 'The effect of organization culture and technology on motivation, knowledge asset and knowledge management', *International Journal of Law and Management* 60(5), 1087–1096. https://doi.org/10.1108/IJLMA-05-2017-0105

Field, H., Anderson, G. & Eder, K., 2014, 'EACOF: A framework for providing energy transparency to enable energy-aware software development', in Y. Cho & S.Y. Shin (eds.), *Proceedings of the 29th Annual ACM Symposium on Applied Computing*, Association for Computing Machinery, Gyeongju Republic of Korea, March 24–28, 2014, pp. 1194–1199.

He, Y., Chen, Q. & Kitkuakul, S., 2018, 'Regulatory focus and technology acceptance: Perceived ease of use and usefulness as efficacy', *Cogent Business & Management* 5(1), 1459006. https://doi.org/10.1080/23311975.2018.1459006

Holgeid, K.K., Jprgensen, M., Sjpberg, D.I.K. & Krogstie, J., 2021, 'Benefits management in software development: A systematic review of empirical studies', *IET Software* 15(1), 1–24. https://doi.org/10.1049/sfw2.12007

Jadhav, A., Kaur, M. & Akter, F., 2022, 'Evolution of software development effort and cost estimation techniques: Five decades study using automated text mining approach', *Mathematical Problems in Engineering* 2022, 5782587. https://doi.org/10.1155/2022/5782587

Jian, Z. & Qin, L., 2022, 'The application of big data network crawler technology for architectural culture and environment protection', *Concurrency and Computation: Practice and Experience* 34(9), e5769. https://doi.org/10.1002/cpe.5769

Kaleshovska, N., Josimovski, S., Pulevska-Ivanovska, L., Postolov, K. & Janevski, Z., 2015, 'The contribution of serum in managing successful software development projects', *Ekonomiski Razvoj* 17(1–2), 175–194.

Kamil, N., 2018, 'Exploring the perceived success factors of agile software projects in Indonesian startups', Dissertation, viewed 3 September 2022, from http://urn.kb.se/resolve?urn=urn:nbn:se:uu:diva-355431.

Krejcie, R.V. & Morgan, D.W., 1970, 'Determining sample size for research activities', *Educational and Psychological Measurement* 30(3), 607–610. https://doi.org/10.1177/001316447003000308

Kuhrmann, M., 2017, *Teaching empirical software engineering using expert teams*, pp. 20–31, SEUH, Kyiv.

Kumar, R., Maheshwary, P. & Malche, T., 2019, 'Inside agile family software development methodologies', *International Journal of Computer Sciences and Engineering* 7(6), 650–660. https://doi.org/10.26438/ijcse/v7i6.650660

Lee, J.C. & Chen, C.Y., 2019, 'Investigating the environmental antecedernts of organizations' intention to adopt agile software development', *Journal of Enterprise Information Management* 32(5), 869–886. https://doi.org/10.1108/JEIM-06-2018-0119

Limaj, E. & Bernroider, E.W., 2019, 'A systematic analysis and synthesis of case study based agile scaling research in the context of digital transformations', in W.M.P., Van der Aalst, S., Ram, M., Rosemann, C. Szyperski & G. Guizzardi (eds.), *Research and Practical Issues of Enterprise Information Systems: 13th IFIP WG 8.9 International Conference, CONFENIS 2019*, Springer International Publishing, Prague, December 16–17, Proceedings 13, pp. 74–84.

Matharu, G.S., Mishra, A., Singh, H. & Upadhyay, P., 2015, 'Empirical study of agile software development methodologies: A comparative analysis', *ACM SIGSOFT Software Engineering Notes* 40(1), 1–6. https://doi.org/10.1145/2693208.2693233

Noteboom, C., Ofori, M., Sutrave, K. & El-Gayar, O., 2021, 'Agile project management: A systematic literature review of adoption drivers and critical success factors', in *Proceedings of the 54th Hawaii International Conference on System Sciences*, p. 6775, Honolulú Manoa.

Ogee, A. & Ellis, M., 2015, *What can you say when your P-value is greater than 0.05*, p. 2018, The Minitab Blog, Pennsylvania, PA.

Pandey, P. & Pandey, M.M., 2021, *Research methodology tools and technique*, Bridge Center, Buzau, Al. Marghiloman.

Panesar, G.S., Venkatesh, D., Rakhra, M., Jairath, K. & Shabaz, M., 2021, 'Agile software and business development using artificial intelligence', *Annals of the Romanian Society for Cell Biology* 25(2), 1851–1857.

Parasuraman, A., 2000, 'Technology readiness index (TRI) a multiple-item scale to measure readiness to embrace new technologies', *Journal of Service Research* 2(4), 307–320. https://doi.org/10.1177/l094670500240001

Park, Y.S., Konge, L. & Artino, A.R., 2020, 'The positivism paradigm of research', *Academic Medicine* 95(5), 690–694. https://doi.org/10.1097/ACM.0000000000003093

Pavitt, K., 1999, *Technology, management and systems of innovation*, University of Sussex, Edward Elgar Publishing, Brighton.

Rai, P. & Dhir, S., 2014, 'Impact of different methodologies in software development process', *International Journal of Computer Science and Information Technologies* 5(2), 1112–1116.

Rogers, E., 1995, *Diffusion of innovations*, 4th edn., Free Press, New York, NY.

Sedano, T., Ralph, P. & Peraire, C., 2017, 'Software development waste', in *2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE)*, pp. 130–140, May, IEEE.

Umar, I., Janga, A. & Shaba, M.A., 2021, 'Lean and agile thinking in small and medium foundry shops: Benefits and challenges', *Journal of the Institute for Empirical Research and Sustainable Development Worldwide* 16(1), 65–75.

Venkatesh, V. & Davis, F.D., 2000, 'A theoretical extension of the technology acceptance model: Four longitudinal field studies', *Management Science* 46(2), 186–204. https://doi.org/10.1287/mnsc.46.2.186.11926

Walliman, N., 2021. *Research methods: The basics.* Routledge, New York.

Wiesche, M., 2021, 'Interruptions in agile software development teams', *Project Management Journal* 52(2), 210–222. https://doi.org/10.1177/8756972821991365

Zia, A., 2018, 'Preference in using agile development with larger team size', *International Journal of Advanced Computer Science and Applications* 9(7), 116–123. https://doi.org/10.14569/IJACSA.2018.090716