

Driving Continuous Improvement in Engineering Projects with AI-Enhanced Agile Testing and Machine Learning

Abhishek Goyal

Independent Researcher

abhishekgoyal2005@gmail.com

Abstract: *Continuous improvement projects follow unique selection, deployment, and tracking processes. In today's fast-paced engineering projects, it has become crucial to drive actionable change for continuous improvement to support the delivery of success. Software development has been a major adopter of agile approaches because of the iterative nature of the process and how well it fits the ever-changing goals of engineering projects. This article delves into the topic of how software development and engineering project management may be improved by using AI and ML in Agile testing procedures. With the help of AI, working in teams, it is possible to sort out routine processes, create more test cases, and use obtained outcomes to reasonably distribute resources and time frames. Data mining, for example, helps in predictive analysis in order to detect risks and any possible slowdown for further action to be taken. The ability to receive constant feedback from AI reinforces the Agile testing process to thereby enable fast identification of problems, better organisational cooperation and successful implementation of intricate projects. This paper discusses the methodologies of Agile testing, a role of AI in enhancing testing efficiency, and best practices for incorporating AI in Agile engineering projects*

Keywords: Continuous improvement, Agile, Artificial Intelligence (AI) and Machine Learning (ML), Engineering Projects

I. INTRODUCTION

The use of AI is on the rise in engineering project management, which is good news for management since it helps to promote the disciplined practice of management and lessens the effect of destructive management practices. The engineering project management community is currently discussing how to successfully integrate artificial intelligence technology with engineering project management practices due to the extensive research on artificial intelligence technology in the field of computers and its social influence[1].

Figure 1 indicates that continuous improvement is an essential component of quality management and has been a major idea for scholars and practitioners for more than 30 years. The integration of CI operations into the strategic objectives at all organisational levels and across all borders is one of the management techniques that are crucial to the success of quality management in construction projects. CI is more comprehensive, methodical, and strategic in character than improvement work generally[2].

AI has changed our lives in ways they don't yet fully understand. Advances in algorithmic design and falling computing costs have allowed AI to dominate fields like NLP, pattern recognition, and decision-making[3]. Consequently, software engineering has seen a rise in the integration of AI, which helps to reduce the likelihood of human mistakes. Team management, decision-making, risk reduction, and customer satisfaction are all pillars of project management, which aim to maximise a team's efforts and resources throughout a project's growth using a variety of approaches[4].

Different approaches to software development encourage effective, reliable, and flexible development. The agile development approach is the most widely used of these, and it may be used in almost any field where it is necessary to be able to build or manage a solution fast, effectively, robustly, and constantly[5]. The primary goal of the agile development method is software development. Thus, it's important to remember that all ML algorithms are software, as is any system made up of several software components.



Figure 1: Continuous improvement in various sectors

Scrum, feature-driven development, and extreme programming are just a few aspects of agile software development. It encompasses more than just stand-ups, sprints, planning meetings, pair programming, test-driven development, and other techniques. The Agile Manifesto lays out the concepts and ideals that should guide software development projects. The word "agile" may be used to describe a variety of approaches to software development [6]. Consequently, sticking to these ideas and values may help you choose the best course of action to take in software development, regardless of the specifics of your situation. The emphasis on collaboration is a key differentiator between Agile and traditional software development methodologies[7][8].

This paper is motivated by the need to improve an efficiency and adaptability of engineering projects through an integration of AI and ML with Agile testing practices. Traditional engineering project management often struggles with complex coordination, evolving requirements, and timely defect detection[9]. By combining AI's automation and data analysis capabilities with Agile's iterative approach, this paper aims to drive continuous improvement, optimise resource allocation, predict risks, and enhance decision-making[10][11]. The goal is to showcase how AI and ML can streamline processes, accelerate feedback loops, reduce defects, and improve overall project outcomes, providing a modern solution for the challenges faced in engineering projects.

The following paper is organised as follows: Section II provides an overview of the agile software development method; Section III gives an overview of the role of AI in agile testing and machine learning; Section IV provides an AI-driven approach to agile testing with machine learning for improvement engineering projects; Section V discusses the previous Literature Review; and Section VI provides the conclusion and future work.

II. OVERVIEW OF AGILE SOFTWARE DEVELOPMENT METHOD

Agile represents one of the project management and product development approaches, built from the basis of flexibility, partnership, and customer-focused iterative progress. Its use is taken across a wide variety of industries-not only software development-but instead comes in handy since it could allow adaptation to changing requirements as well as the delivery of value incrementally. Agile methodologies, which include both widely accepted and contentious software engineering techniques, are becoming more and more common in the business. The realisation that conventional testing methods often led to longer release cycles, delayed feedback, and higher project expenses gave rise to agile testing. Agile testing emphasises early defect identification, continuous integration, and developer-tester cooperation while integrating testing operations throughout the software development lifecycle[12].

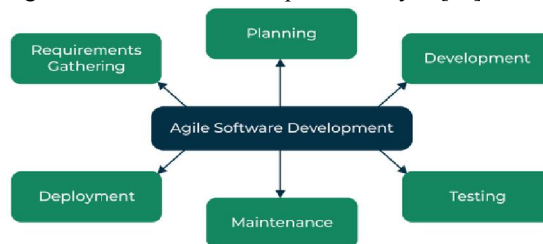


Figure 2: Agile software development phases

Agile Software Development is a flexible approach that emphasises collaboration shown in Figure 2, customer feedback, and continuous improvement. It has been divided into important phases such as requirements, design, coding

and implementation, integration and testing, deployment, and maintenance phases. Flexibility, which is a general characteristic of Extreme Programming, is a key benefit of implementing Scrum and Kanban approaches in solving complex tasks with alterations in customers' requirements.

- **Requirements Gathering:** This is done in a view to assimilating the various needs and expectations as expressed by the various stakeholders. It can be done by interviewing the users, questionnaires and employing the workshop with the users.
- **Planning:** Consequently, a broad schedule is developed that organises the project in terms of objectives, time frame, and resources to be employed in the course of its implementation.
- **Development:** In the development process, the project is broken down into smaller, easier to accomplish segments called sprints. The general objective of each sprint is to provide a given range of features or performance capabilities.
- **Testing:** A large number of tests are conducted to identify defects right through the development stage. This ensures that developed software has the quality on or above the required standards.
- **Deployment:** After testing it is implementing into production environment in computer system or information technology. This means making it available for use by its intended customers, users or clients.
- **Maintenance:** Maintenance work is always required to either patch up security flaws, implement fresh functionality or resolve bugs after a launch.

B. Methodologies of Agile Testing

Agile testing methodologies such as Test-Driven Development, Behavior-Driven Development, Acceptance Test-Driven Development, and Exploratory Testing[13].

- **Test-Driven Development (TDD):** This methodology fosters the development of good, efficient, and reliable code since focus is placed on what should be produced as well as the specification that each increment of code has to meet prior to release to testing.
- **Behavior-Driven Development (BDD):** In addition, BDD reduces the probability of confusing and error-prone situations because the work is carried out with reference to a shared understanding of the requirements for a specific system.
- **Acceptance Test-Driven Development (ATDD):** ATDD is an Agile testing technique for ensuring that a system meets the expectations of the stakeholder as the acceptance criteria and test cases are defined upfront of development.
- **Exploratory Testing:** Exploratory testing is really beneficial in Agile development since it allows fast testing and rectification each time new requirements emerge or freshly identified defects are identified.

C. Characteristics of Agile

The following are categories of the agile SDLC: requirements gathering and analysis process. Creation, structure, development, evaluation and maintenance. Software risks are decreased when customers and software team management are involved. Iterative in nature, this agile method allows for adjustments based on consumer satisfaction. The following are some examples of new features that may be readily introduced to an agile process over several iterations:

- **Iterative:** Agile software techniques concentrate on a single need with several iterations since their primary goal is customer satisfaction.
- **Modularity:** The whole system is broken down into manageable components known as modules using an agile methodology. A key component of software development methods is modularity.
- **Time Boxing:** Agile has time limitations for each module with its own cycle since it is an iterative process.
- **Adaptive:** Agile processes are iterative, which means that new risks may arise at any point. Agile procedures are able to respond to new risks and changes in real-time needs because of their adaptable features.
- **Convergent:** The iterative and incremental nature of the agile process ensures that all risks connected with each increment converge.

- **Collaborative:** The agile methodology relies on strong communication among the software development team due to its modular design. As a software development project nears completion, it is necessary to combine several components.
- **People Oriented:** Customer satisfaction takes precedence over procedure and technology in an agile environment. The efficiency and effectiveness of software is enhanced by a competent software development team.

D. Core Principles of Agile Development

The concepts outlined in the Agile Manifesto form the basis of Agile software development:

- Customer satisfaction via timely and consistent delivery of important software is the top goal.
- Frequent, functional software delivery is preferred, with a timeframe ranging from a few weeks to a few months.
- Throughout the project, business executives and developers are required to collaborate everyday.
- Create projects with motivated folks. Provide them with the necessary atmosphere and support, and believe in their ability to complete the task.
- Face-to-face communication is the most effective and efficient way to communicate with a development team.
- The most important indicator of progress is functional software.
- Sustainable growth is facilitated by agile processes. It should be possible for the developers, sponsors, and users to keep up a steady pace forever.
- Constant focus on quality design and technical proficiency improves agility.
- The ability to maximise the quantity of work that is not done is crucial for simplicity.
- The most effective designs, specifications, and architectures come from self-organising teams [14].

III. THE ROLE OF AI IN AGILE TESTING AND MACHINE LEARNING: AN OVERVIEW

Automating intricate jobs and cutting down on testing time are two ways that AI and ML might improve the efficacy and efficiency of software testing procedures [15][16]. ML and AI are two fast-expanding technologies that have a significant influence on how organisations run[17] [18][19]. AI and ML, for instance, are often utilised in Agile Project Management to stimulate creativity and enhance project results. The use of AI is shown in Figure 3.

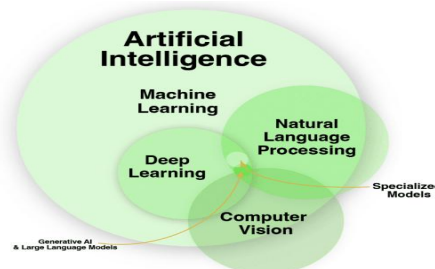


Figure 3: Artificial intelligence

Among the most significant technological approaches to AI, machine learning provides the foundation for several recent developments[20][21] as well as commercial AI applications[22]. The statistical method of modern ML aids in defining the results and potential applications of data.

A. Types of Machine Learning in AI:

- **Following types of learning:** Supervised learning, Unsupervised/predictive learning and Reinforcement learning are as follow:
- **Supervised learning:** The researchers may instruct the machine on the right response for a certain input in this procedure. It is the method most often used to train ML architectures, including neural networks[23].

- **Unsupervised/predictive learning:** This approach gives the learning algorithm no labels, allowing it to discover structure in its data on its own. It may be a goal unto itself, such as data and hidden patterns.
- **Reinforcement learning:** The AI agent uses this algorithm to determine how to act in order to complete the majority of the task. In order to defeat an opponent, a computer program must complete a certain job in a dynamic environment.

B. AI And Its Role in Enhancing Agile Testing

AI is starting to emerge as a disruptive force in Agile project management, providing creative answers to improve productivity, teamwork, and flexibility [24][25]. AI may support teams during brainstorming sessions by analysing patterns and producing original ideas, assisting in overcoming creative obstacles and promoting a continual innovation culture [26]. AI algorithms may evaluate past project data, team performance, and outside variables to provide wise suggestions for allocating resources optimally and adjusting work priorities [27]. This facilitates decision-making and helps to make project management more flexible and responsive.

In addition to helping teams overcome obstacles brought on by human error, generative AI may foster a culture of constant, unbroken invention by using its capacity to recognise patterns and produce deterministic ideas[28]. Agile project management technologies like JIRA, Assemble, and Axe Soft support sprints, user stories, product backlogs, and sprint backlogs. In order to make agile software project development and management a smoothly integrated process with few instances of error, the current tools do not go beyond the purview of management[29]. The proliferation of generative software platforms makes it possible to translate written instructions into code. Improvements may be made to development tools like Visual Studio Code, JetBrains IDE, and others to make it easier for programmers to integrate GitHub Copilot. An AI assistant called GitHub Copilot was created by Open AI and is based on their generative AI framework called Open AI Codex[30].

C. Key feature of AI for agile software testing

- **Accurate Agile planning and estimation:** It therefore causes significant importance in the planning and estimation process of the software development projects. The intricacies and uncertainties present in contemporary software development are often too much for traditional approaches to handle [31]. When supported by AI-based predictive analytics, Agile teams can improve the level of planning and estimation. Machine learning algorithms process huge datasets, combining knowledge from the past to assist in resource management and risks.
- **Improving requirement gathering:** This paper presents the importance of requirements gathering and management in order to provide the best solutions for users. AI mainly through its NLP[32], improves these processes analysing user inputs as well as enabling precise and fast determination of requirements for Agile teams.
- **Analyzing project data:** Project managers may make good decisions owing to insights that can be offered by algorithms, which are based on predictive analysis tools. Agile teams may find possible obstacles, streamline procedures, guarantee project success, and make empowered decisions thanks to this data-driven decision-making. Machine learning algorithms and AI tools can analyses big datasets and identify trends and patterns while providing priceless insights at the same time.
- **AI-powered automation streamlines repetitive tasks,** allowing Agile teams to focus on high-value activities. Automation of project tracking, resource allocation, and progress reporting enhances team productivity and agility.
- **Continuous development:** AI-powered analytics identify optimisation opportunities, patterns of success, and actionable insights for enhancing Agile processes. AI algorithms are constantly evolving, and the Agile team members need to adopt a learning mindset.

D. Best Practices for Implementing Agile in AI and ML projects

The following are some methods for applying agile to projects:

- **Ensure the solution** satisfies user demands and has a positive effect by consistently seeking feedback.
- **Use iterative development:** Construct and improve the solution step-by-step, beginning with a minimal viable product and gradually adding functionality.
- **Embrace flexibility:** This means being prepared at any time to change and modify direction according to new information or changing needs.
- **Foster a data-driven culture:** People in the team should be encouraged to make their decisions based on facts instead of their judgment.
- **Continuously monitor performance:** For improvements and controlling of the overall performance of an AI/ML system, metrics need to be utilised.
- **Encourage collaboration:** Help your team members to feel free to share information with their counterparts.
- **Automate testing:** By implementing the automated testing it is possible to ensure that solution is working properly and in case aspires issues must be detected rapidly.
- **Keep security in mind:** Security should be the first consideration in the development lifecycle, from data collection through to deployment.

E. Impacts of AI on Agile Software Testing

Test automation has emerged as one of the significant effects that AI has had on software testing, as it created a new generation of effectiveness, precision, and expansiveness in the approaches to evaluate quality. Here's an exploration of how AI has influenced agile software testing:

- **Automation of Repetitive Tasks:** AI in automated testing solutions helps reduce manual work and back clarion work relating to test case generation, test execution, and test reporting. This automation helps to speed up testing, decrease the amount of work done by people and, thereby, improve the effectiveness of testing.
- **Enhanced Test Coverage:** One of AI system's major advantages is its ability to analyse vast amount of data and find patterns that could remain unnoticed to testers. This capability provides better test coverage by forcing the discovery of conditions that might not be produced by normal test situations.
- **Performance Optimization:** AI based performance testing tools help in analysing the application performance in continuous mode, the identification of problems and suggesting resolution. This capability allows the applications to function correctly when it is in use and at multiple scales.
- **Code Quality Analysis:** Static analysers identify issues such as bugs & vulnerabilities, coding standard compliance & more. They are useful to developers mainly in providing recommendations for the enhancement of the quality of codes before advanced stages of development.
- **Predictive Analytics for Testing:** AI enables predictive analytics to forecast the impact of changes on software quality. AI systems may predict dangers and direct decision-making throughout testing stages by examining past data and patterns.

IV. AI-DRIVEN IN AGILE TESTING WITH MACHINE LEARNING FOR IMPROVEMENT ENGINEERING PROJECTS

Project management was essential, necessitating careful resource distribution and team collaboration. In order to keep the diverse project team on track to finish tasks, project managers had to oversee everyone, maintain good communication, and make important choices [33]. Engineering project management includes creating a thorough plan, establishing goals, and assigning management resources and expertise. The development of comprehensive engineering plans is made easier by project management, which gives managers the ability to deconstruct tasks, allocate duties to specific team members, and establish and adhere to reasonable deadlines. Engineering project management involves coordinating various stakeholders, including: Engineers, the project manager, other team members even without engineering background clients, and suppliers. Figure 4 shows the continuous improvement in projects.



Figure 4: Continuous Improvement in Projects

Additionally, data-driven forecasting may be facilitated by engineering software for project management, which helps project managers foresee obstacles, calculate expenses, and modify plans as necessary. These forecasting skills, which are accessible to anyone without an engineering degree, aid in decision-making, enhance engineering activities and facilitate non-engineering duties like cost estimates.

A. The role of AI in agile testing for Engineering projects

Any engineering project may be transformed by AI and deep learning. The predictive skills established with the Deep Learning AI technique may automate difficult operations and reduce time-consuming ones from hours to seconds. This is great news for engineering project management. Design engineers and their engineering management may benefit from AI's predictive analytics and intelligent automation, which improve team collaboration regardless of team members' technical backgrounds and encourage innovation in the engineering profession.

B. Benefits of AI in Engineering project

There are some benefits of AI in Agile Testing for Improvement of Engineering Projects are as follows [34]:

- Improved efficacy, AI greatly improves efficiency by automating repetitive tasks. Tasks can be handled by AI tools, reducing the need for manual input and minimising human errors.
- AI improves decision-making by providing data-driven insights and analytics that are predictive in nature. AI tools use machine learning algorithms to analyse large volumes of data. Management of projects may benefit from predictive analytics, which use past data to foretell future results.
- AI can automatically generate test cases based on system requirements, reducing manual effort and saving time.
- Machine learning models identify defects earlier in the testing cycle, improving the quality of engineering projects.
- AI predicts potential areas of failure or risk, enabling proactive measures and reducing downtime.
- Automated testing minimises human error, ensuring consistent test results across iterations.
- Reduced manual testing efforts and early defect detection lead to significant cost savings.
- AI ensures broader test coverage by simulating diverse scenarios and edge cases.
- AI ensures broader test coverage by simulating diverse scenarios and edge cases.
- AI-based analytics provide actionable insights, fostering better communication among teams and stakeholders.

C. Applications of AI and ML in software Testing for engineering project

Software performance testing is being revolutionised by AI and ML, which are paving the way for smarter, faster, and more predictive testing. AI can instantly through mountains of test data, revealing patterns that engineers may use to proactively fix performance problems, simplify testing processes, and decrease testing time and effort while increasing software quality, there are some applications of AI in software testing for projects are as follow:

- AI models analyse requirements of the project for generating test cases thoroughly and automatically.

- Reduces the time and efforts in comparison with the manual process to design tests.
- AI relies on historical data along with an assessment of risk so that the testing can be prioritised through critical test cases.
- High Risk aspects are tested first so that potential cases of failure can be minimised while handling the projects.
- Machine learning algorithms determine patterns in defect data in order to forecast likely bug-prone areas.
- AI enables the execution of real-time testing and continuously update test strategies based on the outcome.
- Enables adaptability in Agile domains that keep changing; AI enables automation of test data generation, extraction, and anonymisation.
- Ensures that data privacy regulations are met while speeding up the test preparation process.

V. LITERATURE REVIEW

This section provides related work, a systematic review of Driving Continuous Improvement in Engineering Projects with AI-enhanced Agile Testing and Machine Learning. The summaries of the literature reviews mentioned below are also shown in Table I:

In, Esa et al. (2006) the presentation focusses on the level of efficacy of internal audits conducted on active learning methods in an endeavour to continuously enhance teaching and learning. The purpose of this study was to analyse how the TL strategy may benefit from audit reports. Successful audits help guarantee that microwave engineering education is always becoming better[35].

In, Singla, Bose and Naik, (2018) data collected from Scrum to identify project challenges in various machine learning initiatives. They look for patterns in the execution of ML initiatives that vary from regular software engineering projects by comparing this data with similar data from non-ML projects. They discover that ML projects contain a larger number of research-oriented or exploratory activities than implementation tasks, and that challenges are described using various sorts of language[36].

In, Dam et al. (2019) the framework where AI may be used to help with the management of agile projects, which are gaining popularity in the business world. AI can automate high-volume, repetitive operations, which in turn allows project analytics for risk prediction and assessment, gives practical suggestions, and even makes judgements, all of which may be helpful to project managers and their teams. AI has the ability to revolutionise project management by boosting efficiency and the percentage of successful projects[30].

This, Puleio (2006) to implement and ensure the correct operation of the many testing procedures. Specifically, the author delves into the difficulties they encountered in reaching a consensus on the practices, determining the time needed to run automated tests within a feature-driven culture, and bringing together a diverse team of individuals to grasp the concept of applying difficult-to-implement principles[37].

In, Snider et al. (2019) an innovative method and structure for tracking engineering projects using data-driven and computational analytics, which subsequently aid in the development of higher-level, situation-specific knowledge for managers. EPHM is a product of the first integration of vehicle health management with engineering management, a strategy that has so far been used for predictive maintenance and machine monitoring [38].

This study, Saleh and Philip (2019) in order to justify the decision of Agile to the subjective experience of undergraduates studying software engineering, who, after finishing lab projects according to Agile principles, were attempting to implement Agile, the philosophy of software engineering is gaining popularity in contrast to the traditional approaches. Graduates armed with state-of-the-art software engineering methodology expertise are a must-have for software organisations[39].

This study, Butgereit (2019) a method that makes use of ML to prioritise automated testing, such that the tests with the highest failure probability are performed first, providing programmers with early warning of issues. This is accomplished by collecting a number of metrics, including Cyclomatic, Halstead-based, and Chidamber-Kemere values, about the tested program. Engineering projects that make use of AI-Enhanced Agile Testing and ML may find an overview of related work in Table I[40]

Table 1: Related work summary for AI-Enhanced Agile Testing and Machine Learning for engineering projects

Ref. No.	Focus	Methods	Key Findings	Limitations / Challenges
[35]	Effectiveness of internal audits in active learning practices for continuous improvement in teaching (microwave engineering)	Analysis of audit reports for TL strategy value-addition	Effective audits ensure continuous improvement in microwave engineering education	Limited to teaching practices in a specialised field (microwave engineering)
[36]	Differences in project issues tracking between ML and non-ML projects in Scrum	Comparison of issue-tracking data from ML and non-ML projects	ML projects involve more exploratory/research tasks than non-ML projects; distinct vocabulary in issue descriptions	Limited generalizability; may vary by specific project requirements or Scrum practices
[30]	Application of AI to support agile project management	Framework development for AI in project analytics and task automation	AI can improve productivity by automating repetitive tasks, aiding risk prediction, and offering recommendations	Adoption challenges, especially in tasks requiring significant human judgment
[37]	Challenges in establishing effective testing practices in agile, feature-driven development	Case study on implementing testing practices	Highlighted the need for shared understanding, proper effort estimation, and time allocation for automated tests	Difficulty in aligning diverse team members; complexities in time management and testing effort estimation
[38]	Framework for data-driven monitoring of engineering work to support managerial insights (EPHM)	Framework design using computational analytics adapted from integrated vehicle health management	EPHM supports managerial interpretation of data for better decision-making	Adaptation to broader engineering fields may require customisation and validation.
[39]	Perception of Agile principles among Software Engineering undergraduates	Observation and analysis of students' project experience following Agile principles	Agile methods are increasingly preferred over traditional methods by students	Limited to educational context; may not fully reflect industry practices
[40]	Machine learning technique for prioritising automated testing based on probability of test failure	Collection of software metrics (Cyclometric, Halstead, and Chidamber-Kemere values)	ML helps prioritise tests likely to fail early, giving quicker feedback	Dependency on accurate metric collection may not capture all potential failure factors

VI. CONCLUSION AND FUTURE WORK

Therefore, the Agile approaches incorporated Artificial Intelligence, thereby evoking a dramatic shift in the current Software Development practices. Thus, the purpose of this study is to demonstrate how AI is important in automating routine work, enhancing decision-making, and detecting flaws. This has enhanced the software development efficiency

and, at the same time, boosted project efficiency. AI and Agile can nurture development cycles and impact continuous delivery as they reduce workload. That can let teams quickly respond to change and provide better-quality software products. The use of AI and Agile is an important strategy in modern software engineering because of the critical role that it plays in today's innovation and agility. This integration implies that the engineering profession will also attain enhanced operation efficiency, velocity and flexibility with a view to addressing future requirements in the profession. Therefore, in the future, Future, there should be a need to improve the transparency and interpretability of the AI models especially with Agile framework. When problems are presented in a simpler form, they are easier to solve or debug and so are the working models herein described. This clear understanding can help to establish higher levels of trust in the decisions made by artificial intelligence as well as guarantee the correct interaction of the stakeholders with the software, which means a better inclusion of AI in software development processes. Getting over this challenge will be essential for the integration of AI in Agile methodologies and its ability to enhance decision-making and forge strong bonds between different participants in development projects.

REFERENCES

- [1] A. Gao, N. L. Guillaou, and Z. Yu, "Modern Engineering Project Management Based on Deep Artificial Intelligence Technology," 2022. doi: 10.2991/978-94-6463-044-2_30.
- [2] F. Backlund and E. Sundqvist, "Continuous improvement: challenges for the project-based organization," *Int. J. Qual. Reliab. Manag.*, vol. 35, no. 7, pp. 1306–1320, 2018, doi: 10.1108/IJQRM-12-2016-0229.
- [3] P. Khare, "The Impact of AI on Product Management : A Systematic Review and Future Trends," *Int. J. Res. Anal. Rev.*, no. December 2022, 2022.
- [4] P. Callier and O. Sandel, "Introduction to Artificial Intelligence," *Actual. Pharm.*, 2021, doi: 10.1016/j.actpha.2021.10.005.
- [5] R. Goyal, "Software Development Life Cycle Models: A Review Of Their Impact On Project Management," *Int. J. Core Eng. Manag.*, vol. 7, no. 2, pp. 78–87, 2022.
- [6] R. Goyal, "THE ROLE OF REQUIREMENT GATHERING IN AGILE SOFTWARE DEVELOPMENT: STRATEGIES FOR SUCCESS AND CHALLENGES," *Int. J. Core Eng. Manag.*, vol. 6, no. 12, pp. 142–152, 2021.
- [7] D. Trivedi, "Agile Methodologies," *VerisonOne. Agil. Made Easier*, vol. 48, no. 5, pp. 73–78, 2013.
- [8] A. P. A. Singh, "Streamlining Purchase Requisitions and Orders : A Guide to Effective Goods Receipt Management," *J. Emerg. Technol. Innov. Res.*, vol. 8, no. 5, pp. g179–g184, 2021.
- [9] K. Patel, "Quality Assurance In The Age Of Data Analytics: Innovations And Challenges," *Int. J. Creat. Res. Thoughts*, vol. 9, no. 12, pp. f573–f578, 2021.
- [10] K. Patel, "An Analysis of Quality Assurance Practices Based on Software Development Life Cycle (SDLC) Methodologies," *J. Emerg. Technol. Innov. Res.*, vol. 9, no. 12, pp. g587–g592, 2022.
- [11] S. A. and A. Tewari, "AI-Driven Resilience: Enhancing Critical Infrastructure with Edge Computing," *Int. J. Curr. Eng. Technol.*, vol. 12, no. 02, pp. 151–157, 2022, doi: <https://doi.org/10.14741/ijcet/v.12.2.9>.
- [12] A. Smith and D. Manna, "Lean Principles and Optimizing Flow: Case Studies," 2019.
- [13] M. Ijaz Khan *et al.*, "Analysis of Agile Software Testing, It's Impact on Software Quality and Cost," *Int. J. Comput. Intell. Control Copyrights @Muk Publ.*, vol. 14, no. 1, pp. 974–8571, 2022.
- [14] M. Von Rosing, J. Von Scheel, and A. Q. Gill, "Applying agile principles to BPM," in *The Complete Business Process Handbook: Body of Knowledge from Process Modeling to BPM*, 2014. doi: 10.1016/B978-0-12-799959-3.00027-6.
- [15] R. Bishukarma, "The Role of AI in Automated Testing and Monitoring in SaaS Environments," *IJRAR*, vol. 8, no. 2, 2021, [Online]. Available: <https://www.ijrar.org/papers/IJRAR21B2597.pdf>
- [16] M. S. Rajeev Arora, "Applications of Cloud Based ERP Application and how to address Security and Data Privacy Issues in Cloud application," *Himal. Univ.*, 2022.
- [17] R. Bishukarma, "Adaptive AI-Based Anomaly Detection Framework for SaaS Platform Security," *Int. J. Curr. Eng. Technol.*, vol. 12, no. 07, pp. 541–548, 2022, doi: <https://doi.org/10.14741/ijcet/v.12.6.8>.

- [18] M. T. Arora, Rajeev and Kumar, Shantanu and Jain, Nitin and Nafis, "Revolutionizing Healthcare with Cloud Computing: Superior Patient Care and Enhanced Service Efficiency," *SSRN*, 2022, doi: <http://dx.doi.org/10.2139/ssrn.4957197>.
- [19] R. Arora, S. Gera, and M. Saxena, "Mitigating Security Risks on Privacy of Sensitive Data used in Cloud-based ERP Applications," in *2021 8th International Conference on Computing for Sustainable Global Development (INDIACom)*, 2021, pp. 458–463.
- [20] M. Gopalsamy, "Advanced Cybersecurity in Cloud Via Employing AI Techniques for Effective Intrusion Detection," *Int. J. Res. Anal. Rev.*, vol. 8, no. 01, pp. 187–193, 2021.
- [21] M. Gopalsamy, "Artificial Intelligence (AI) Based Internet-ofThings (IoT)-Botnet Attacks Identification Techniques to Enhance Cyber security," *Int. J. Res. Anal. Rev.*, vol. 7, no. 4, pp. 414–420, 2020.
- [22] Mani Gopalsamy, "An Optimal Artificial Intelligence (AI) technique for cybersecurity threat detection in IoT Networks," *Int. J. Sci. Res. Arch.*, vol. 7, no. 2, pp. 661–671, Dec. 2022, doi: 10.30574/ijrsra.2022.7.2.0235.
- [23] M. Gopalsamy, "Scalable Anomaly Detection Frameworks for Network Traffic Analysis in cybersecurity using Machine Learning Approaches," *Int. J. Curr. Eng. Technol.*, vol. 12, no. 06, pp. 549–556, 2022, doi: <https://doi.org/10.14741/ijcet/v.12.6.9>.
- [24] S. Bauskar, "BUSINESS ANALYTICS IN ENTERPRISE SYSTEM BASED ON APPLICATION OF ARTIFICIAL INTELLIGENCE," *Int. Res. J. Mod. Eng. Technol. Sci.*, vol. 04, no. 01, pp. 1861–1870, 2022, doi: DOI: <https://www.doi.org/10.56726/IRJMETS18127>.
- [25] H. S. Chandu, "A Survey of Memory Controller Architectures: Design Trends and Performance Trade-offs," *Int. J. Res. Anal. Rev.*, vol. 9, no. 4, pp. 930–936, 2022.
- [26] M. S. Rajeev Arora, Sheetal Gera, "Impact of Cloud Computing Services and Application in Healthcare Sector and to provide improved quality patient care," *IEEE Int. Conf. Cloud Comput. Emerg. Mark. (CCEM), NJ, USA, 2021*, pp. 45–47, 2021.
- [27] Mani Gopalsamy, "A review on blockchain impact on in cybersecurity: Current applications, challenges and future trends," *Int. J. Sci. Res. Arch.*, vol. 6, no. 2, pp. 325–335, Aug. 2022, doi: 10.30574/ijrsra.2022.6.2.0146.
- [28] M. R. S. and P. K. Vishwakarma, "An Efficient Machine Learning Based Solutions for Renewable Energy System," *Int. J. Res. Anal. Rev.*, vol. 9, no. 4, pp. 951–958, 2022.
- [29] S. A. and A. Tewari, "Security Vulnerabilities in Edge Computing: A Comprehensive Review," *Int. J. Res. Anal. Rev.*, vol. 9, no. 4, pp. 936–941, 2022.
- [30] H. K. Dam, T. Tran, J. Grundy, A. Ghose, and Y. Kamei, "Towards effective AI-powered agile project management," in *Proceedings - 2019 IEEE/ACM 41st International Conference on Software Engineering: New Ideas and Emerging Results, ICSE-NIER 2019*, 2019. doi: 10.1109/ICSE-NIER.2019.00019.
- [31] R. Goyal, "THE ROLE OF BUSINESS ANALYSTS IN INFORMATION MANAGEMENT PROJECTS," *Int. J. Core Eng. Manag.*, vol. 6, no. 9, pp. 76–86, 2020.
- [32] S. Bauskar, "Predictive Analytics For Sales Forecasting In Enterprise Resource," *Int. Res. J. Mod. Eng. Technol. Sci.*, vol. 04, no. 06, pp. 4607–4618, 2022, doi: <https://www.doi.org/10.56726/IRJMETS26271>.
- [33] P. S. Woolfolk (T. Aruna Bharathi, "International Journal of Science and Research," *Int. J. Sci. Res.*, vol. 5, no. 02, p. 512, 2020, doi: 10.30574/ijrsra.
- [34] R. Mahato, "ARTIFICIAL INTELLIGENCE, WHAT IS IT?," in *Outcomes of Best Practices in Classroom ResearchPublisher: L' Ordine Nuovo Publication, India, 2022*.
- [35] M. Esa, R. A. Rashid, R. Abdullah, and M. S. Masodi, "Value-adding through internal audits: A continuous improvement process in microwave engineering education," in *2006 International RF and Microwave Conference, (RFM) Proceedings*, 2006. doi: 10.1109/RFM.2006.331097.
- [36] K. Singla, J. Bose, and C. Naik, "Analysis of Software Engineering for Agile Machine Learning Projects," in *INDICON 2018 - 15th IEEE India Council International Conference*, 2018. doi: 10.1109/INDICON45594.2018.8987154.
- [37] M. Puleio, "How not to do agile testing," in *Proceedings - AGILE Conference, 2006*, 2006. doi: 10.1109/AGILE.2006.34.

[38] C. Snider, J. A. Gopsill, S. L. Jones, L. Emanuel, and B. J. Hicks, "Engineering Project Health Management: A Computational Approach for Project Management Support through Analytics of Digital Engineering Activity," *IEEE Trans. Eng. Manag.*, 2019, doi: 10.1109/TEM.2018.2846400.

[39] S. M. Saleh and S. R. Philip, "Students Evaluation in AGILE after Completing Software Engineering Course," in *1st International Conference on Advances in Science, Engineering and Robotics Technology 2019, ICASERT 2019*, 2019. doi: 10.1109/ICASERT.2019.8934745.

[40] L. Butgereit, "Using machine learning to prioritize automated testing in an agile environment," in *2019 Conference on Information Communications Technology and Society, ICTAS 2019*, 2019. doi: 10.1109/ICTAS.2019.8703639.