# On Using Artificial Intelligence in Software Quality Assurance: A State of the Art

Randa Ouaarous[1,*], Imane Hilal[2], Abdellatif Mezrioui[1]

[1] STRS Lab/CEDOC 2TI, Institut National des Postes et des Telecommunications (INPT), Rabat, Morocco
[2] Information Science School (ESI), Rabat, Morocco

**Abstract**

Artificial intelligence (AI) techniques and models have been applied to assist numerous activities in software engineering, specifically in Software Quality Assurance (SQA), covering both technical and management parts. Several studies have examined the utilization of AI in various tasks within the SQA field, with a particular focus on software testing. This study seeks to investigate the overall impact of AI on SQA, while identifying the prevalent applications and possible areas for further research. We thoroughly examined a selection of articles that study the use of AI approaches in various SQA activities. We have selected relevant research papers that were published in the last 5 years. The analysis was conducted by utilizing established AI and SQA taxonomies and categorizing the chosen papers based on these taxonomies. The resulting mapping and conversations indicate that the use of AI to assist in SQA is a well-established and expanding area of scientific interest with interesting opportunities for future. Evidence of several AI benefits was found when applied to SQA such as cost reduction and process improvement, as well as challenges in case of high complexity and questionable data quality. The discussions of the impact of AI on the roles of the SQA showed better outcomes when adopting a Human-AI collaboration approach.

**Keywords**

Software Quality Assurance, SQA, Artificial Intelligence, AI, Software Testing

## 1. Introduction

The practice of Software Engineering is born around the 1960s, a human activity that involves the process of building software, which remains the dominating industrial field within computer science [1]. In many industries, including retail, healthcare, banking/financial services, business/IT services, and government/defense, Software Engineering practice plays a pivotal role, that's why they are the top 5 industries hiring Software Engineers [2]. However, it is important to note that software, which is designed and created by humans, can contain errors. These errors can have severe consequences, potentially putting businesses, resources, and even human lives at risk [3]. This is why SQA has become an indispensable part of the Software Development Lifecycle. Its primary objective is to ensure that high-quality software is developed that meets the initial requirements set by the software owners [4].

SQA as a practice, has experienced exponential growth lately following the growth of Software Engineering. With growth come new challenges and complexities such as the difficulties to adapt SQA to the organizational features and constraints and the high cost of implementing SQA activities [5]. Luckily, AI is in a fast growth as well, it's currently being used in many fields and sectors to address their challenges. One of the fields where AI is thriving is SQA. According to [6], the percentage of the organizations that are currently investing and using AI to enhance their SQAs processes is at 77% compared to 45% 5 years ago. To understand this trend properly, we seek to answer these questions: How is AI being currently applied to SQA? What are the observed benefits and

*SQAMIA 2024: Workshop on Software Quality, Analysis, Monitoring, Improvement, and Applications, September 9--11, 2024, Novi Sad, Serbia*

ouaarous.randa@doctorant.inpt.ac.ma (R. Ouaarous); ihilal@esi.ac.ma (I. Hilal); mezrioui@inpt.ac.ma (A. Mezrioui)

challenges when applying AI to SQA? And how is the usage of AI impacting the roles involved in the SQA process?
This paper is structured as follows: Section 2 is an overview of the SQA and its activities. Then Section 3, which is a review of the remarkable research where AI was applied and studied in each SQA activity. Last, Section 4 is a summary of the paper reviews we performed emphasizing all the AI techniques applied to SQA activities, the benefits and challenges of using AI in SQA and the impact of using AI on the roles involved in SQA.

## 2. Software Quality Assurance (SQA) - Overview

### 2.1. SQA Definition

According to [7], Software quality is defined as "conformance to established requirements; the capability of a software product to satisfy stated and implied needs when under specified conditions". These requirements need to be questioned by the quality processes as well, and this operation is performed by SQA. According to [8], SQA has been defined as "a set of activities that define and assess the adequacy of software processes to provide evidence that establishes confidence that the software processes are appropriate for and produce software products of suitable quality for their intended purposes".

### 2.2. SQA Activities

To ensure quality, SQA has a set of activities that starts early in a Software project's life cycle. [7] [8] give a detailed description of these activities, we summed them up in Figure 1. The purpose of this section is to explain each SQA activity.

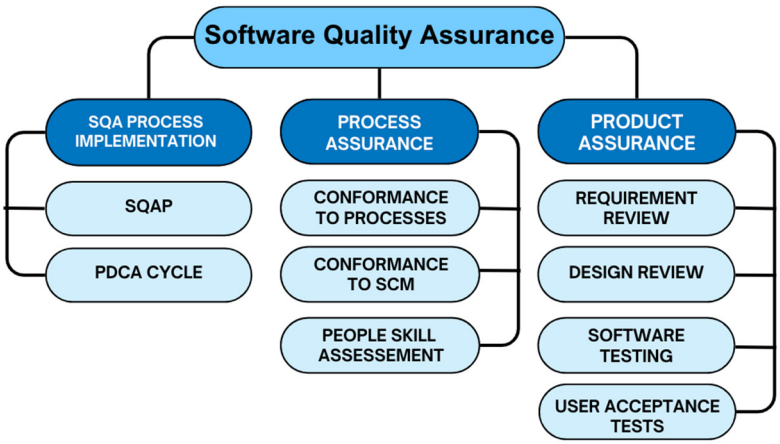


**Figure 1:** SQA activities.

#### 2.2.1. SQA Process Implementation

To prepare for the SQA function, the Software Quality Assurance Plan (SQAP) should be defined and signed off by all stakeholders of the software project as a first step. Once the SQAP is done then the PDCA is launched to guarantee a continuous improvement of the plan.
**SQAP** is a Project Planning artifact that defines the activities and tasks used to ensure that software developed for a specific product satisfies the project's established requirements and user needs within project cost and schedule constraints and is commensurate with project risks [8]. A poorly specified SQAP can have severe consequences on the outcome product [9].
**The PDCA Cycle,** also called The Deming Cycle, is a quality management procedure that ensures a continuous process improvement [10]. This cycle has four steps Plan, Do, Check, Act. These four stages are iteratively and continuously executed through the life cycle of the project for effective and increased SQA [10].

### 2.2.2. Process Assurance

Process assurance seeks to guarantee that the processes adopted by the project are adequate and are followed by all stakeholders. This mitigates the risk of process risks that may lead to delays and extra costs in the project, eventually impacting the quality of the concerned software. Process Assurance is considerably important in case there are external participants in the project [11]. According to [7], The main activities of Process Assurance are:

- Ensure conformance to processes.
- Ensure conformance to Software Configuration Management (SCM).
- Ensure People have the needed skills and knowledge.

### 2.2.3. Product Assurance

It is ensured through product assurance activities that software products are developed in compliance with contractual requirements, project schedules, and established product requirements. These products include not only the software and related documentation but also the plans associated with the development, operation, support, maintenance, and retirement of the software. According to [7], The main activities of Product Assurance are:

- Requirement Review.
- Design Review.
- Software Testing.
- User Acceptance Tests.

## 3. Review of AI applied to SQA

### 3.1. AI – Overview

AI refers to machines capable of simulating and mimicking the human intelligence by thinking and learning from their experiences. This technology includes a set of features such as problem-solving, pattern recognition, understanding natural language and decision making [12].
AI has many definitions and can be seen from many perspectives, The taxonomy outlined in the [12] consists of five primary scientific domains: Reasoning, Planning, Learning, Communication, and Perception. Additionally, there are three transversal domains: Integration and Interaction, Services, and Ethics and Philosophy. In our study we found evidence of the application from AI techniques in SQA that belong to the following AI domain:
**Reasoning** focuses on techniques for processing data into knowledge and determining facts.
**Planning** concentrates on developing and implementing algorithms and strategies that will support in accomplishing tasks that can be carried out by intelligent agents.
**Learning** deals with the capacity of systems, without explicit programming, to learn, decide, forecast, adapt, and respond to changes and improve from experience automatically.
**Communication** relating to the capacities of recognizing, analyzing, comprehending, and producing information from oral and written human communications. Mainly, Natural Language Processing (NLP) covers this field.
**Integration and Interaction** addresses the combination of the previous domains to showcase an intelligent behavior through Agents and bots.
We are classified the findings of our reviews following this taxonomy in Table 1.

### 3.2. Research Methodology

In this paper, we used literature review methodology to investigate the current state of the art of AI application in SQA. The selection method we adopted is based on multi-step approach. First, we build a comprehensive search strategy based on combinations of selected keywords and Boolean operators:

"AI," "Artificial Intelligence," "Machine Learning," "Deep Learning," "Software Quality Assurance," "SQA," "Software Testing," "Process Assurance," "Product Assurance" "Generative AI," and "Software Engineering,". This search was conducted across various academic databases of computer science, engineering and software engineering disciplines. Second, we applied inclusion criteria to the retrieved articles. Articles were mainly selected if they were published within the last 5 years (2019-2024) in journals or conference proceedings, and if they explored the use of AI techniques in SQA tasks. Finally, after screening the title, abstract and full text, our methodology resulted in 16 articles that met our criteria and formed the basis of this review.

## 3.3. AI in SQA Process Implementation

### 3.3.1. AI in SQAP

SQAP is an activity that falls under the Project Planning umbrella, the paper in [13] covered the usage of AI in the software engineering by exploring potential usage of AI in Project Planning phase. This phase has many challenges that exceeds the human capacity such as managing the conflicts for developers and project planners, optimizing tasks, time, and budget allocation [14]. The authors of [13] found that the usage of non-linear and self-optimizing algorithms, such as ant colony optimization, can reduce decision complexity. Furthermore, their work proved that the Bayesian network algorithms, can integrate large amounts of data and handling missing or uncertain information to simultaneously optimize cost and quality outcomes. This AI approach achieved cost & duration optimization, Effective task assignment, Improvement of quality outcomes, improved project planning and many more benefits that will be listed in the last section of this paper.

### 3.3.2. AI in PDCA

Considering this continuous operation directly linked to project management. In general, it has many challenges related to risk management and the accuracy of the decision making. We reviewed numerous studies related to AI applied to Project Management such as [15]. In this study, M El Khatib et al. reviewed literature and interviewed IT Project Managers, they found that AI can improve the quality of decision making as it can manage a significant amount of data. Some AI solutions that the authors found to be used in project management are Chatbots, Machine Learning and Knowledge Management systems with AI features.
There is also another recent study [16] that explores the project planning capabilities of generative AI, the authors specifically used GPT-4 model in project management and compared it to human. The findings of this study are that AI applied to Project management had some hiccups and cannot be fully autonomous. The best results are produced when there is a synergy between human experience and AI.

## 3.4. AI in Process Assurance

### 3.4.1. AI in Evaluating Conformance to Processes

To avoid having issues in the product, the process of developing the product needs to be monitored and tracked. The challenges that are faced in this task are usually related to the complexity of the used processes in the project, the adherence of different contributors to the agreed-on processes and the efficiency of processes. AI can come with great benefits to address these challenges. [17] is a study that explores the usage of AI in Project where the Agile Framework is applied. AI can automate repetitive tasks, enhanced project metric analysis and accelerate team productivity. This benefit impacts directly and positively the process assurance of the project. [14] is another study that was done on Waterfall methodology and found that AI contributes to more efficient and accurate planning and risk assessment.

### 3.4.2. AI in Evaluating Conformance to SCM

Software Configuration Management is an activity that aims to manage the software configurations and versioning to track changes. Ensuring that all parties respect the configuration management is an important activity of SQA, however, there might be some challenges related to the difficulty of tracking all changes and keeping records accessible and exploitable. [18] focused on the usage of AI in the Continuous Integration/Continuous Deployment (CI/CD). Machine learning integrated to DevOps tools, like Jenkins and SonarQube, is primarily used to automate and optimize the pipeline and the deployment in various environment, enhancing efficiency, quality, and responsiveness of software development processes. Moreover, machine learning leverages the CI/CD pipeline by automating the tasks of monitoring and analysis, defect prediction, process improvement insights, and data-driven decision making.

### 3.4.3. AI in People Skill Management

People Skill Management is an important activity of Process Assurance because the skill sets of the people involved in the process of developing the software are key to the quality assurance of the software. This task is challenging as it relies on the human personality traits and knowledge that are relative parameters and difficult to measure. According to [19], AI can be used in People Skill Management by predicting human interactions with their environment in the software project operations. The authors of [19] used Agent Based Modeling and Simulation (ABMS) that considers various human related factors such as personality traits, affective states, competencies, learnability, and individual interactions. The result showed that the prototype composes teams that perform well in terms of output quantity and development speed. The limitation of this model is that it's based on the numeric correlations between human aspects that can be sometime hard to predict with data.

## 3.5. AI in Product Assurance

### 3.5.1. Product Assurance and SDLC

Software Development Life Cycle (SDLC) is a structured process or methodology used in software engineering to build high quality software applications with an optimized cost in the shortest time possible through planned phases and activities [20]. The Product Assurance activities depend on the adopted SDLC Model. There are many SDLC models such as Waterfall, RAD, Spiral, Agile, Prototype. In this paper we are considering the general model presented in Figure 2. In the upcoming 4 sections we are going to explore the usage of AI in each Product Assurance activity in its relationship with the SDLC corresponding stage.

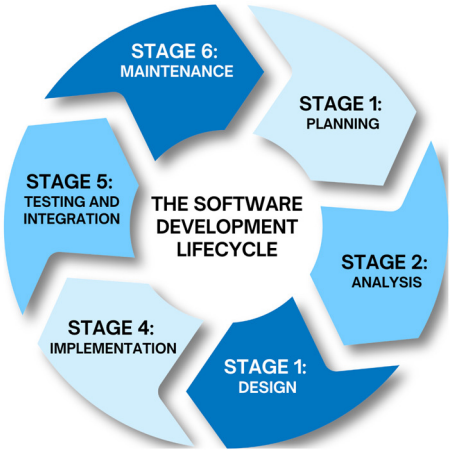


**Figure 2:** Software Development Life Cycle

- Requirement Reviews will be performed in Stage 1,2.
- Design Reviews will be performed in Stage 3.
- Software Testing will be performed in Stage 4, 5, 6.
- User Acceptance Tests will be performed in Stage 5, 6.

### 3.5.2. AI in Requirement Reviews

AI significantly enhances the process of requirement reviews by automating the identification, analysis, and validation of software requirements. According to [21], NLP can retrieve valuable inputs from extensive documentation and identify key capabilities and potential requirements from various sources such as stakeholder interviews, documents, emails, and user feedback. NLP can also provide insights and detect any gaps or omissions in the requirements. [22] Also covered the requirement review by knowledge-based systems. These systems can manage the requirements phase efficiently by capturing and utilizing domain knowledge to ensure that the requirements are complete, consistent, and accurately reflect the stakeholders' needs.

### 3.5.3. AI in Design Reviews

Good software quality is a direct result of a well-defined design, but sometimes, human error and requirement complexity can leave space to gaps and functional misalignments to hide in the design. In this context, [23] discussed the importance of good software design and how the usage of model refactoring and machine learning can help maintain software quality. To identify the presence of functional misalignments and gaps in software models, the authors of [23] proposed the usage of a labeled dataset of metric values of UML class diagrams to train a deep neural network model using an adaptive supervised learning algorithm. After evaluation, the approach showed high accuracy and scalability to large and complex data.

### 3.5.4. AI in Software Testing

#### 3.5.4.1. Software Testing Overview

SQA is commonly mixed with Software Testing. However, Software Testing is one of the main activities of the Product Assurance, which is an activity of SQA [8].
Software Testing is defined by [3] as "a formal process in which the programs are run on a computer and can relate to the analysis of a single software component, a set of integrated software components, or an entire software package."
Software Testing activities are organized and carried out differently in different life cycles. In the upcoming 5 sections, we cover the application of AI in the following 5 components of the Software Testing process:

- Test Analysis and Design
- Test Implementation and execution
- Static Code Analysis
- Software Defect Prediction
- Test Automation

#### 3.5.4.2. AI in Test Analysis and Design

One of the most important artifacts of Software Testing under SQA is the Test Case document [24]. It's known in Software Testing that Test case creation is a challenging task that takes time and analytical skills to produce good quality test cases. Several Machine Learning Models have been subjects of test case generation studies. [25] compared the accuracy of these known models in the case of test case generation based on quality attribute scenarios (QAS). The results show Rain Forest

combined with TF-IDF generated testcases with high accuracy, resulting in significant Test effort optimization in the project.

#### 3.5.4.3. AI in Test Implementation and Execution

Test Implementation and Execution is a phase where we define the priority of each area of the software to be covered by testing. Test Case Prioritization (TCP) is a Software testing activity that is more crucial in the context of regression testing, whether manual or automated, as it reduces the costs by optimizing the testing effort and by detecting issues as early as possible. [26] discusses the various used approaches. According to this study, these are the most used techniques in TCP: Neural Networks, Bayesian Network, genetic algorithm, SVM, K-means. Results show that these techniques can achieve the continuous and adaptive TCP, improving performance of detecting faults earlier, achieve full coverage and faster fault detection.

#### 3.5.4.4. AI in Static code Analysis

Large-scale codebases may include possible flaws that traditional static code analysis methods miss, resulting in delayed troubleshooting and high-cost hotfixes. In [27], the authors introduced an ML prototype to enhance the Static Code Analysis, therefore, leverage SQA in sensitive and complex IT projects. The prototype is based on three ML procedures: the API Mining, the Sequential Pattern Mining, and the Frequent ItemSet Mining. Results show that the prototype can detect complex code patterns and deviations in code and automate error detection.

#### 3.5.4.5. AI in Software Defect Prediction (SDP)

The importance of SDP in Software engineering comes with the challenge of discovering defect in the early stages of the software development to reduce its cost. Recently, this activity has been in the plat of many AI studies. [28] classifies the available SDP models that are based on ML and DL. A large set of techniques are widely used and have each difference advantages, we summed them up in the last section of this paper.

#### 3.5.4.6. AI in Test Automation

According to [29], Test Automation has been a trend in Software Testing for several years. It has been proven that test automation increases quality and delivery time. However, its biggest flaw is that it requires continuous human intervention that sometimes can be costly, time consuming and not efficient. This is where AI can make a difference. [29] presents a collection of AI methods used for software testing activities that are usually automated, such us GUI Testing, Black Box testing, Regression Testing, unit testing. The authors of [29] also highlighted the limitations and challenges that comes with using AI for test automation such as the Test automation complexity that becomes more complex when AI is added.

### 3.5.5. AI in User Acceptance Tests (UAT)

According to [30], UAT is a “Formal testing with respect to user needs, requirements, and business processes conducted to determine whether or not a system satisfies the acceptance criteria and to enable the user, customers or other authorized entity to determine whether or not to accept the system”. The authors of [31] worked on an approach for generating UAT testcases from scenarios that are captured and analyzed using NLP and Task/Method model. This approach was applied in the case of software development in the agricultural domain where the requirements and the scenarios of the agricultural processes change from a region to another. The integration of these AI techniques in the process of UAT resulted in an optimized UAT testing within the agile development environment.

# 4. Results and Discussion

## 4.1. Use of AI Technologies per SQA activity

Table 1 summarizes the used technologies of AI for all the SQA stages as available from the reviewed studies. Rows and columns represent SQA and AI concepts, respectively. Cells include the selected studies from which we extracted the evidence of applications.

**AI in SQA Process Implementation:** We analyzed papers that fall under Project Management in general, according to our analysis we found research interest in using AI techniques that fall under the Planning and Interaction and Integration AI domains such as Probabilistic Planning, Bayesian Network Algorithm and Ant Colony Optimization for SQAP [13] and Intelligent Agents and Chatbots and GPT-4 for PDCA Cycle [15,16].

**AI in Process Assurance:** The usage of AI techniques for this activity included the AI domains: Reasoning, Planning, Learning and Integration and Interaction. We found evidence of usage of Fuzzy logic, Knowledge representation, Probabilistic planning, and Intelligent Agents [14] for Conformance to processes, Machine Learning and ABMS for Conformance to SMC and People Skills Management rsespectively.

**AI in Product Assurance:** For this activity, AI techniques were used from a larger set of AI domains: Reasoning, Planning, Communication, Learning. AI in Software Defect Prediction is the area where a remarkable research interest was found with the application of several ML techniques with supervised and unsupervised learning [28]. User Acceptance Test was the least popular research topics with only the usage of NLP to generate requirement-based testcases [31].

**Table 1:** Review results of AI techniques applied to SQA activities

| | | | Reasoning | | | Planning | | | | Communication | | | Learning | | | | | | | | | | Integration and Interaction | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Fuzzy Logic | Knowledge representation | Knowledge based systems | Probabilistic planning | Bayesian network | Genetic Algorithm | Ant colony Optimization | Natural Language Processing | TF-IDF | Mining | Machine Learning (ML) | Deep Neural Networks (DNN) | Boosting | Classification | Clustering | Decision trees | Ensemble Method | Supervised Learning | Support Vector Machine | Unsupervised Learning | Intelligent Agents and chatbots | Agent Based Modeling and simulation | GPT-4 |
| **SQA Process** | **SQAP** | | | | | [13] | [13] | | [13] | | | | | | | | | | | | | | | | |
| | **PDCA Cycle** | | | | | | | | | | | | | | | | | | | | | | [15] | | [16] |
| **Process** | **Conformance to processes** | | [14] | [14] | | [14] | | | | | | | [14], [17] | | | | | | | | [17] | | [14] | | |
| | **Conformance to SMC** | | | | | | | | | | | | [18] | | | | | | | | | | | | |
| | **People Skills Management** | | | | | | | | | | | | | | | | | | | | | | | [19] | |
| **Product Assurance** | **Requirement Review** | | | | [22] | | | | | [21] | | | | | | | | | | | | | | | |
| | **Design Review** | | | | | | | | | | | | | [23] | | | | | | [23] | | | | | |
| | **Software Testing** | **Test Analysis and Design** | | | | | | | | | [25] | | | | [25] | [25] | | [25] | | | [25] | | | | |
| | | **Test Implementation and execution** | | | | | [26] | [26] | | | | | | [26] | | | [26] | | | | [26] | | | | |
| | | **Static code analysis** | | | | | | | | | | [27] | [27] | | | | | | | | | | | | |
| | | **Software Defect Prediction** | | | | | | [28] | | | | | [28] | [28] | [28] | [28] | [28] | [28] | [28] | [28] | | [28] | | | |
| | | **Test Automation** | | | | | | [29] | | [29] | | | | | | | [29] | [29] | | | [29] | | | | |
| | **User Acceptance Testing** | | | | | | | | | [31] | | | | | | | | | | | | | | | |

## 4.2. Benefits and Challenges of using AI in SQA

AI techniques are introduced in SQA to optimize resources and address common SQA challenges. We extracted many benefits of using AI in SQA such as the cost reduction and the process optimization. We also found evidence of the challenges that necessitate further research and exploration such as the quality and representativeness of the training data. Table 2 summarizes these benefits and challenges and categorizes them by the potential impact on each SQA activity.

**Table 2:** Benefits and challenges of using AI in SQA per SQA Activity

| | Benefits | Challenges |
|---|---|---|
| **SQA Process Implementation** | -Improving project management and decision making [15,16,18,21,22]<br>-Optimizing cost and schedule estimations [13,15,25,26,27,28,29]<br>-Leveraging historical data to forecast future project needs [13,15,16,18] | -Data collection and management can be quite expensive and time-consuming [21][22], this can lead to an impact on the SQA plan. |
| **Process Assurance** | -Employing data-driven techniques, leading to enhanced project metric analysis [14,17,18]<br>-Effective task assignment [13].<br>- Task allocation based on individual strengths and experiences [19]<br>-Accelerating team productivity and communication [14,17] | -Complex, large-scale projects that deviate from conventional processes can lead to AI models generating false results [14,17]<br>-Integration with existing testing tools and workflows [18].<br>-Predicting human aspects such as soft skills through data analysis [19] |
| **Product Assurance** | -Enhanced testing coverage [21,22]<br>-Improved test case design and execution [21,22,25,26,27,28,29]<br>-Accurate defect prediction [19,21,22,23] | -Potential biases within the training data can lead to flawed test outcomes [23,21,22,25,26,27,28,29].<br>-Difficulties in scaling with software complexity resulting in inaccurate testing [25,26,27,28,29]. |

## 4.3. Impact of using AI on different roles involved in the process of SQA

Many of the papers we reviewed highlighted the impact of using AI on human aspects in the context of SQA. On one hand, the positive impacts as AI proved to be efficient in repetitive task automation, leaving more room for the SQA practitioners to focus on more complex tasks [13,15,25,26,27,28,29]. On the other hand, challenges that need to be addressed when integrating AI in the SQA, mainly related to potential job displacement [15] and over-reliance on AI.

**Table 3:** Impact of AI usage approaches on performance during the SQA activities.

| | Without AI | Human-AI collaboration | Only AI |
|---|---|---|---|
| **SQA Process** | Limited use of historical data of the project that can make the decision making subject to individual judgement. | Provides Data-driven insights to help reduce bias and make objective decisions | Some inputs such as the domain expertise can be missed leading to an inadequate planning |
| **Process Assurance** | Subject to individual judgement as there is no efficient use of the project data. | Provides data driven assessment for an objective task assignment and corrective action plan | Limited ability to assess human aspects in the project as it's hard to predict with data. |
| **Product Assurance** | Review of artifacts and testing are time consuming activities when relying only on human effort. | AI Automates repetitive tasks while human focus on complex tasks. | Some defects that require human observation might be missed |

The analysis of these impacts and challenging considerations points to the AI-Human Collaboration [32]. This approach is suggesting using AI techniques to support or automate software engineering tasks and incorporating human domain knowledge as starting points for designing AI techniques and using human feedback to improve these techniques, forming a continuous feedback loop, this approach seeks to reduce human efforts and the burden on human intelligence as shown in Table 3.

## 5. Conclusion and Future Work

SQA is a crucial software engineering activity, that's why organizations are investing efforts and resources to leverage their SQA processes using the latest technologies such as AI. We performed a paper review of AI application to all SQA activities. In these selected papers we also found evidence of its benefits such as cost reduction and Process improvement. AI application in this field also comes with challenges such as the high complexity and the data quality. As well as an impact of AI on the human aspects within the SQA field which was also a topic that we covered and discussed based on the outcomes of some of the reviewed papers that highlighted this aspect. The result of our review unrevealed many research gaps that can be covered by future research, one gap that caught our attention is the lack of research on the application of Generative AI and LLMs in SQA. In our future work we will propose an architecture of an AI Agent that helps functional roles involved in SQA such as the end users, QAs, Product Owners and Project Managers. The agent will bring a great value to SQA as it will save time and reduce test effort.

## References


[1] Batarseh, F. A et al. (2020). The application of artificial intelligence in software engineering: a review challenging conventional wisdom. Data Democracy, 179-232.

[2] The Top 5 Industries Hiring Software Engineers (2023) https://generalassemb.ly/blog/top-5-industries-hiring-software-engineers/

[3] « ISTQB® Certified Tester Foundation Level » Syllabus v4, 2023, [online] Available: https://www.istqb.org/certifications/certified-tester-foundation-level

[4] Zhao, Y.; Hu, Y.; Gong, J. Research on International Standardization of Software Quality and Software Testing. In Proceedings of the 2021 IEEE/ACIS 20th (ICIS Fall)

[5] Zhang, X., Sharma, B., & Koppe, T. (2024). Overcoming Test Debt and Advancing Software Sustainability with Automated Testing: A B2B Trading Platform Case Study.

[6] Buenen M, Natarajan S. World quality report 2023–2024. Capgemini, Sogeti and Opentext

[7] IEEE 730:2014 Standard https://ieeexplore.ieee.org/document/6835311

[8] SWEBOK, 03 2023, [online] Available: https://waseda.app.box.com/v/ieee-cs-swebok.

[9] C. Y. Laporte and A. April, Software Quality Assurance, Wiley, 2018.

[10] ISO 9001 Quality Management System (QMS) standard

[11] Galin, D, Software Process Assurance Activities for External Participants 2018

[12] S. Samoili, et al. 2020. Defining artificial intelligence: Towards an operational definition and taxonomy of artificial intelligence.

[13] Barenkamp, et al. (2020). Applications of AI in classical software engineering.

[14] Kulkarni, R., et al (2017). Integration of AI activities in software development processes and measuring effectiveness of integration. IET Software

[15] El Khatib, M. et Al (2021) Effects of Artificial Intelligence on Decision Making in Project Management. American Journal of Industrial and Business Management

[16] Barcaui, A., & Monat, A. (2023). Who is better in project planning? Generative artificial intelligence or project managers. Project Leadership and Society, 4, 100101.

[17] Salehi, F. (2022). Role of Artificial Intelligence on Agile Planning and Organizational Performance in the ICT Industry. Asian Journal of Economics, Finance and Management.

[18] Nogueira, A. et al. (2018). Improving La Redoute's CI/CD Pipeline and DevOps Processes by Applying Machine Learning Techniques. In 2018 11th QUATIC, IEEE.

[19] Athavale S, Balaraman V. Human behavioral modeling for enhanced software Project Management. In: 7th International Conference on Software Engineering; 2013. p. 15–7.
[20] Viller, S., & Sommerville, I. (2000). Ethnographically informed analysis for software engineers. International Journal of Human-Computer Studies, 53(1), 169-196.
[21] Muhairat, M., et al. (2020). An Intelligent Recommender System Based on Association Rule Analysis for Requirement Engineering. JUCS, Pensoft publishers.
[22] K. Hema Shankari Int. Journal of Engineering Research and Applications
[23] Brahmaleen Kaur Sidhu, et al. (2020): A machine learning approach to software model refactoring, International Journal of Computers and Applications
[24] IEEE Std 829-2008, IEEE Standard for Software and System Test Documentation
[25] A. Worku, et al. (2023). Test Case Generation from Quality Attribute Scenarios Using Machine Learning Approach.
[26] Omri, S., & Sinz, C. (2021). Machine Learning Techniques for Software Quality Assurance: A Survey. arXiv preprint arXiv:2104.14056.
[27] E. Sultanow et al. "Machine Learning based Static Code Analysis for Software Quality Assurance," 2018 ICDIM, Berlin, Germany, 2018
[28] O.E. Emam, et al. Users Review's on Software Defect Prediction Utilizing Machine Learning methods. vol. 6, no 1, p. 34-41.
[29] Hayat, M. A., et al. Artificial Intelligence's Impact, Limitations, Challenges and Prospects in Software Testing.
[30] « ISTQB® Certified Tester Acceptance Testing » Syllabus v1, 2019, [online] https://www.istqb.org/certifications/acceptance-tester
[31] Antonelli, L., et al. (2023). User acceptance test for software development in the agricultural domain using natural language processing. Journal of Decision Systems.
[32] Xie, T. (2013). "The synergy of human and artificial intelligence in software engineering," 2013 2nd International Workshop on (RAISE), San Francisco, CA, USA