JOURNAL OF ENGINEERING RESEARCH & SCIENCES

JENRS

www.jenrs.com
ISSN: 2831-4085

Volume 3 Issue 12
December 2024

Editorial

Technological advancement continues to drive innovation across industries from software engineering and industrial automation to healthcare. These studies exemplify how critical thinking, strategic design, and computational technologies converge to address modern challenges in quality assurance, operational safety, and medical diagnostics. By drawing from structured methodologies and harnessing the power of digital tools, these works lay the foundation for future-ready systems that prioritize reliability, accessibility, and intelligent performance.

Ensuring the integrity and quality of software is an indispensable task in the software development life cycle (SDLC). This study examines the intricate dynamics of software testing through a qualitative lens, guided by Structuration theory. By analyzing the social and organizational structures influencing testing practices, it identifies five key insights: the centrality of testing for software quality, the importance of communication and collaboration among project stakeholders, the influence of power hierarchies on decision-making, the necessity of adherence to standards and processes, and the legitimization of the testing role within teams. The study demonstrates that software testing is not merely a technical activity but a socially embedded process requiring structural support, clear role recognition, and collaborative governance to ensure defect-free and requirement-aligned products [1].

Precision in nuclear power infrastructure is paramount, especially in maintenance tasks involving inaccessible or hazardous environments. This paper presents the creation of a digital twin for a robotic inspection system designed to assess pipe wall thickness within turbine buildings. Utilizing Process Simulate software, the system integrates mobile, robotic, and service units to achieve automated ultrasound-based inspections. The research details the virtual commissioning process and the design innovations of the robotic arm's end-effector, enabling flexible, interpolated movement around complex geometries. Through laboratory prototyping and simulation, the system's operability across diverse configurations is validated, emphasizing the digital twin's value in improving commissioning accuracy, safety, and efficiency in industrial robotics applications [2].

Peripheral artery disease (PAD), a widespread vascular condition, demands timely and accurate diagnosis for effective treatment. This paper introduces the DECODE platform, a cloud-based diagnostic tool that combines machine learning, 2D/3D visualization, and data warehousing for non-invasive PAD evaluation. The system enables artery segmentation, reconstruction, and drug-coated balloon simulation within a scalable infrastructure, supporting clinicians with actionable insights. Notably, the multiplanar and 3D rendering modules achieve high performance scores, validating the platform's technical robustness and clinical utility. The DECODE system redefines PAD diagnostics by enhancing accessibility, improving decision-making accuracy, and setting a precedent for the integration of digital health solutions in vascular medicine [3].

These studies represent distinct but complementary perspectives on how technological systems are transforming essential sectors. From ensuring software quality and industrial safety to revolutionizing healthcare diagnostics, they underscore the value of combining technical innovation with strategic insight. As digital systems grow more interconnected and intelligent, the lessons and models presented here provide essential blueprints for the development of reliable, scalable, and human-centric solutions in an increasingly complex world.

References:

[1] T. Gordon Sekgweleo, P. Makovhololo, “Exploring Challenges in Software Testing: A Structuration Theory Perspective,” *Journal of Engineering Research and Sciences*, vol. 3, no. 12, pp. 1–13, 2024, doi:10.55708/js0312001.

[2] R.A.P. Vitalli, J.M.L. Moreira, “Advanced Digital Twin of a Industrial Robotic System for Measuring Pipe Wall Thickness in Nuclear Power Plants,” *Journal of Engineering Research and Sciences*, vol. 3, no. 12, pp. 14–23, 2024, doi:10.55708/js0312002.

[3] M.A. AboArab, V.T. Potsika, D.I. Fotiadis, “Advanced Cloud-Based Solutions for Peripheral Artery Disease: Diagnosis, Analysis, and Visualization,” *Journal of Engineering Research and Sciences*, vol. 3, no. 12, pp. 24–35, 2024, doi:10.55708/js0312003.

Editor-in-chief

Prof. Paul Andrew

JOURNAL OF ENGINEERING RESEARCH AND SCIENCES

Volume 3 Issue 12 **December 2024**

CONTENTS

Received: 30 October, 2024, Revised: 23 November, 2024, Accepted: 23 November, 2024, Online: 19 December, 2024

DOI: https://doi.org/10.55708/js0312001

Exploring Challenges in Software Testing: A Structuration Theory Perspective

Tefo Gordon Sekgweleo *[1], **Phathutshedzo Makovhololo** [2]

[1] Eskom, Department, Research, Testing & Development, Johannesburg, 2095, South Africa

[2] Cape Peninsula University of Technology, Informatics, University, Cape Town, 8000, South Africa

E-mails: Ts330ci@gmail.com / phathuts@gmail.com

*Corresponding author: Dr Tefo Gordon Sekgweleo, Lower Germiston Rd, Rosherville, Johannesburg, 2095, 082 533 3484 & Ts330ci@gmail.com

ABSTRACT: Developing software is a huge job, which is why digital product teams rely on the software development life cycle (SDLC). SDLC is a critical framework for digital product teams, and software testing is its most vital component. Testing evaluates software components to identify properties of interest, detect defects, and ensure alignment with requirements. If not optimized, testing can be costly, and its omission or inadequate execution can lead to software failures, compromising business operations and reputation. This study explores the challenges of software testing, adopting an interpretivist approach with semi-structured data collection and analysis guided by Structuration theory's duality of structure. The key findings are: (1) Software testing is crucial for delivering quality products and services, ensuring that software meets client requirements and is free from defects. (2) Effective communication and collaboration among agents, including software testers, developers, and project managers, are vital for successful software testing outcomes. (3) Power dynamics and decision-making processes significantly impact software testing outcomes, with project managers' decisions often dominating software testers' work. (4) Adhering to organizational processes and standards is essential for ensuring quality software delivery, preventing software testing from being bypassed or done hastily. (5) Legitimization of software testing practices is necessary for instilling social attachment and control among software testers, recognizing the importance of their role in delivering quality software. These findings highlight the significance of software testing in ensuring software quality and business continuity, emphasizing the need for effective communication, collaboration, and organizational processes to support software testers in their critical role.

KEYWORDS: Structuration Theory, Software testing, Software development, Software implementation, Software Development Life Cycle (SDLC), Information Systems

1. Introduction

In recent years, software testing has gained prominence in the software development industry [1]. Organisations of all sizes rely on software to deliver services and enhance productivity [2]. SDLC is a widely used methodology that outlines the stages of software development, from initiation to implementation [3]. Within SDLC, software testing is a critical phase that ensures software reliability and adds value to organisations [4]. A significant portion of software development budgets is allocated to testing [2]. Software testing encompasses various technical and non-technical sections, including specification, design, implementation, maintenance, and management issues [2]. Testing verifies that software meets organizational objectives and identifies errors or failures [5].

However, challenges like time constraints and regression testing hinder effective software testing [6]. This qualitative study employs structuration theory to analyze data and identify factors affecting software testing in organisations. The dynamics between social structures and human agency play an important role in shaping the adoption, use, and impact of information systems within organisations [7] [8]. Structuration Theory (ST), developed by Anthony Giddens, offers a valuable lens for examining these dynamics [7]. Despite its potential, ST has been underutilized in IS research [9], [10]. This study aims to address this gap by applying ST to explore the interplay between social structures and human agency in the context of IS implementation [11]. Specifically, this research seeks to understand how social structures influence human agency and vice versa [12], and how this interplay affects IS outcomes [13]. By examining these dynamics, this study contributes to a deeper understanding of the complex factors that shape IS success and failure [14]. The paper is organized into five sections: literature review, research methodology, data analysis, findings, and conclusion.

2. Literature Review

This section covers existing literature in the following key areas of the study: (i) software development, (ii) Software testing, (iii) Software implementation, and (iv) Structuration theory.

2.1. The role of Software Testing

Software testing plays an essential role in ensuring the quality, consistency, and security of software products [4]. As software becomes increasingly pervasive in society, the significance of software testing cannot be overstated [15]. According to [16], software testing is a critical process that detects defects and ensures software meets user requirements. Effective software testing strategies enable organisations to identify and mitigate potential risks, reducing the likelihood of software failures and minimising their impact [17].

Any product that is created must be tested before it can be released to the general public for use or consumption. Same applies to any software that is developed by organisations to carry out their day-to-day duties. Software testing is the approach that guarantees that quality products are distributed to consumers, which in turn uplifts customer satisfaction and trust. The aim of software testing is to identify defects and issues in the software development process so that they can be fixed prior to its release. According to [18], it is vital to test software as it helps to verify its quality and reliability particularly in modern software development processes, where very sophisticated software is continuously released faster and quicker.

Even though software testing is important, its activities are usually ignored even by big organisations when executing significant software projects as they are often regarded unlikeable, time wasting as well as tedious when compared to more innovative and fulfilling activities such as software design or coding [19]. In [18], the authors defines software testing as "the process of evaluating software to ensure that it meets its originally specified requirements and revealing faults and defects that may affect the code". It verifies that the software meets the functional, performance, design as well as the implementation requirements identified in the functional requirement specification.

The primary intent of software testing is to guarantee that software functions as expected, meets user requirements, and is reliable, maintainable, and secure [20]. Software testing involves various activities, including test planning, test case development, test execution, and test reporting [21]. These activities guarantees that software is thoroughly vetted and meet the required standards before deployment [22].

Moreover, software testing is the fundamental component of the software development lifecycle, complementing activities such as system analysis, design, coding, and implementation [23]. Integrating software testing into the software development process, organisations can identify and report defects early, minimizing the overall cost and time needed for software development [24].

In conclusion, software testing is the important part of software development, guaranteeing that software meets the user requirements, are reliable, maintainable, and secure. By adopting effective software testing strategies, organisations can mitigate potential risks, reduce software failures, and release quality software products that meet the growing needs of society [25]. Lately, organisations, are focusing on software quality, and they are identifying broad requirements, such as more software functions, quicker response speed, as well as reliable and safe operation [26]. Software testing can be conducted in two main forms, either manually or automated. According to [27], manual testing can be time consuming, resource intensive and make testers not to discover some defects hence there are automation tools in place to enable both automation and performance engineers to record and rerun the test cases which could be tested manually by a software tester. Automated testing reduces the cost and time for testing software, it increases testing coverage by executing more test cases faster and eliminates human errors (humans gets tired when doing repetitive tasks and make errors) increased software reliability, user satisfaction and reduces the amount manual work that needs to be conducted by software testers. According to [27], tools for automating software testing enable software testers to consistently perform testing in less time and can frequently reuse them to retest the software. Automated testing decreases the volume of manual work, increases high coverage by executing additional test cases and reducing human errors remarkably when humans are tired after several repeats [28]. In [29] further alluded that software testing is not performed only to detect defects but to assist software developers to notice the mistakes they made, provide tips on how to resolve those mistakes and also to ensure that the software performs as specified in the requirement specification.

2.2. System Development Life Cycle

Software development is a systematic process used to create software products that meet specific requirements and enable agents in a social system to achieve particular goals [25]. Organisations adopt software development strategies to manage software activities effectively and ensure alignment with business objectives [30]. According to [24] software development strategy refers to the approach organisations employ to develop the software. Software development encompasses various activities, including system analysis, design, coding, and testing [31]. Organisations consider the development of software as crucial for achieving business objectives and goals [25]. In [16], the author emphasizes the vital role of software in social systems, highlighting the need for extensive research, to understand, enhance, and support in software development.

2.3. Software Testing

Software testing is a crucial method that ensures software functions as expected, without defects or issues [4]. According to [16], software testing evaluates software quality and identifies areas for improvement. The primary goal of testing software is to discover defects and guarantee it meets user requirements [17]. However, it is essential to take into cognizance both functional and non-functional requirements during testing. Failure to do so may negative impact the quality of software [21]. For instance, the Gauteng online registration system failed to handle user load, despite functional testing [15]. Software testing tools enable testers to conduct both functional and non-functional testing, including performance testing, which determines software behavior under various conditions [20]. Automation tools enhance testing efficiency, reliability, and repeatability, reducing human error [20]. The ultimate goal of software testing is to deliver high-quality software, ensuring business confidence in the tested product [25]. Following testing, software implementation ensues, aiming to deploy the software for use.

The importance of software testing in the software development process cannot be overstated [4]. Software testing is crucial for ensuring the delivery of high-quality software products that meet user requirements and are reliable [17]. According to [16], software testing plays a vital role in identifying and fixing defects, errors, and bugs in the software, thereby reducing the likelihood of software failures and minimizing their impact.

Moreover, software testing helps save time and money by detecting defects early in the development process [24]. This is supported by [24], who argues that testing is an essential component of the software development lifecycle, complementing activities such as system analysis, design, coding, and implementation. Furthermore, software testing improves user experience by ensuring that the software meets user requirements [25]. It also enhances security by identifying security vulnerabilities and ensuring that the software is secure and protected against threats [15].

There are two main common types of software testing namely, black box and white box. The main of intent of black box testing is to test the behaviour of the software whereas white box testing focuses on testing the internal operation of the software. Black box testing also referred to as functional testing is a process whereby the software is tested without the knowledge of the internal workings of the software [32]. It is a method that enables the test engineer to design the test cases based on the information from the specification and does not allow the test engineer access to source code of the software [33] With black box, the test engineer is not required to have programming knowledge.

On the other hand, white box testing also referred to glass box testing/structural testing is a method that enables the test engineer/tester to design the test cases based on the information derived from source code [34]. The tester is required to have programming background with this type of testing as they are granted access to the source code. Grey box testing is a third method whereby the tester has limited knowledge about the internal workings of the software and has the knowledge of fundamental aspects of the software [35].

In addition, software testing supports continuous improvement by providing feedback for refinement and enhancement of the software [20]. This is critical for building trust with customers, stakeholders, and users, enhancing the organization's reputation [22]. In conclusion, software testing is essential for delivering high-quality software products that meet user requirements, are reliable, and provide a positive user experience [21]. By prioritizing software testing, organisations can reduce risks, save time and money, and build trust with their customers.

2.4. Software Implementation

All software follows a particular lifecycle prior to its completion, from development to deployment, irrespective of the methodology employed (agile or traditional methodology), depending on the requirement [3]. Software implementation refers to the process of making software available for operation [1]. The term implementation is used interchangeably, but in the context of this study, it means making software operational. In the SDLC, implementation refers to applying system requirements, or actual coding [33]. Others describe it as constructing or building software [21], or making it available for use after development [7]. Software implementation occurs after quality assurance accompanied by various tests, including user acceptance testing [22]. Quality assurance is a challenging factor during implementation [1]. Top management approval is also crucial for successful implementation, as they must approve software before it is implemented or changed [7].

Software implementation poses several challenges that can hinder its success. One of the primary challenges is ensuring quality assurance, as inadequate testing can lead to software failures and errors [1]. Additionally, resistance to change from end-users can also pose a significant challenge, as they may be reluctant to adopt new software and processes [7]. Furthermore, the implementation of software needs substantial resources, including time, money, and personnel, which can be a challenge for organisations with limited budgets [33]. Moreover, integrating new software with existing systems and infrastructure can also be a complex challenge [23]. In [33], finally, top management approval and support are crucial for successful implementation, and lack of commitment from leadership can lead to implementation failure [22].

In conclusion, software testing and implementation are critical components of the SDLC. Effective software testing ensures that software meets user requirements, is reliable, and provides a positive user experience. However, software implementation poses several challenges, including quality assurance, resistance to change, resource constraints, integration with existing systems, and top management approval. To overcome these challenges, organisations must prioritize software testing and implementation, adopt effective software development methodologies, and ensure stakeholder commitment. By doing so, organisations can deliver high-quality software products that meet user needs and drive business success.

2.5. Structuration Theory

The selection of an appropriate theory to underpin a study is critically important because it assists in determining the outcomes of the study [11]. Structuration theory (ST) was developed by [8], it is a sociology theory but has also gained popularity in the information system (IS) field where it has been borrowed to analyse data [36]. The theory takes a stance that social action cannot be explained in detail through structure or agency alone, but it appreciates the actors operating within the context of rules shaped by social structures but act in a biddable manner that these structures reinforced. In [37], the authors defines ST "as the reproduction of social structures through human actions. Social structures and human actions are viewed as two aspects of the same whole, instead of seeing human actions happening outside of the constraints of social structure". According to [38], ST puts an emphasis on agency and structure, their duality within a social system which implies that the agent/agency entails technical such as technology and non-technical such as human entities. The structure is the rules and resource in structuration. In [39], the authors further states that ST focuses on how events and social systems are produced and reproduced over a period of time and space. In [40], the researchers alluded that human agency as well as social structure cannot be treated as separate ideas but are two ways of regarding social action and is termed as duality of structure. Structure is the recurrent patterned arrangements which influence or limit the choices and opportunities available. Whilst agency is the capacity of individuals to act independently and to make their own free choices [36]. ST is divided into structure, modality and interaction whereby the modality provides interaction between structure and interaction [37].

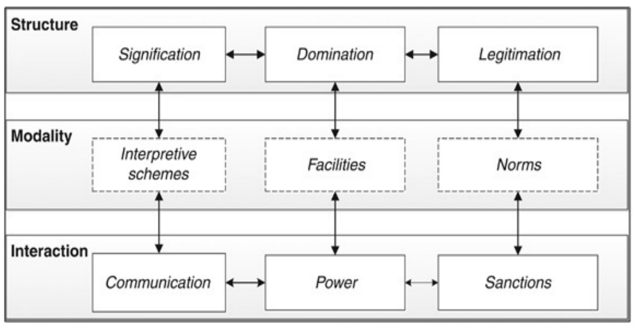


Figure 1: Dimensions of the duality of structure [26]

Both the social structures as well as the human actions are treated as two aspects of the same whole within the duality of structure [8]. In [40] and [41], the authors emphasizes Giddens' claim that social phenomena emantes from both social structure and agency, not just one or the other. Figure 1 illustrates the dimensions of the duality of structure [8]. In IS research, various theories are used, including social theories [11]. IS researchers have borrowed b social theories, such as Actor-Network Theory (ANT), Activity Theory (AT), and Structuration Theory (ST) to underpin their studies [8].

This underscores the importance of software testing in ensuring that developed software meets requirements and is reliable, maintainable, and secure [4]. The SDLC comprises a sequence of connected methods from planning to system testing, ensures consistency and produces well-developed software [20]. As software becomes increasingly prevalent in society, the call for reliable, maintainable, and secure software requirements continues to grow [15].

3. Research Methodology

Qualitative approach was adopted for this study to gain an in-depth understanding of software testing in organisations from the participants' perspective [42]. Qualitative methods, as noted by [43], are subjective in nature, focusing on beliefs and encounters rather than statistical figures. A case study was used, which is usually employed in qualitative research [44]. This design allows for an in-depth examination of a phenomenon within its real-life setting [44]. The case study organization, Setlamo Technologies (a pseudonym), is a public sector organization operating in South Africa, with a dedicated software testing team. Semi-structured interviews were used to collect data, offering flexibility and allowing for clarifications during the interviews [45]. Therefore, this approach allows for in-depth examination without deviating from the research's core focus [46]. Fourteen participants were interviewed until saturation was reached [47].

The data was transcribed and analyzed using Structuration theory as a lens, focusing on the duality of structure [8]. The vertical approach was adopted, and an interpretivist approach was employed to analyze the findings [47]. The interpretive paradigm was used to subjectively interpret the findings [11].

This study adopted a qualitative case study approach, utilizing semi-structured interviews, to explore software testing practices in-depth [44]. This methodology is best suited for examining complex phenomena, such as software testing, in real-life settings [42]. Semi-structured interviews provide flexibility to explore topics in-depth, allowing participants to share experiences and opinions in their own words [45]. The case study method enables a comprehensive understanding of the organization, including social structures, human agency, and technology interactions [8]. This approach allows for a detailed examination of software testing practices, processes, and contextual factors, making it the most appropriate methodology for this study [48]. Moreover, identifying influential factors and potential adopters, as well as understanding their decision-making processes, is crucial for a thorough understanding of the subject [49].

4. Data Analysis

For the purpose of data analysis, participants and organisations were labeled. Fourteen employees from Setlamo Technologies participated in the research. The referencing standard is exemplified as ST01, 7:17-20, indicating organization ST, participant 01, page number 7, and line numbers 17 to 20.

i. Setlamo Technologies: Participants ST01 to ST14

At Setlamo Technologies, agents and structures were involved in software testing. These agents and structures were from the IT department. The agents comprised both technical and non-technical individuals. The structure entailed the rules and resources utilized in software testing. All participants shared similar interest for realizing organisational objectives through software testing and utilization.

4.1. Structuration Theory

According to [33], Structuration Theory (ST) encompasses agents (agency) and structures. Agents can be technical or non-technical, while structures consist of rules and resources involved in software testing activities within the organisation. The researchers identified agents followed by structures.

4.2. Agents (Agency)

Agency refers to an organisation comprising technical and non-technical agents, where some apply knowledge/conscious (human) and others lack knowledge capability [50]. Agency can also be associated to individuals or group of abilities within a particular environment. In the case of this study individuals are those who are involved in the SDLC such as software developers, software testers and others. They have particular skills to enable them to perform their duties. Therefore, these individuals can collaborate and apply their various skills to deliver a working solution within the organization (environment). An agent can be understood as anything with the potential to make a difference in a social structure [51]. The research identified technical agents, including software testing tools such as Rational Quality Manager and Meter, as well as SAP, Oracle, and the environment comprising development, testing, and production. Additionally, technical agents included the Integrated Financial Management System (IFMS) and International Software Testing Qualifications Board (ISTQB) training.

Non-technical agents included software test analysts, project managers, business analysts, functional support personnel, software developers, and individuals involved in software development methodology, either Agile or traditional. Furthermore, non-technical agents comprised individuals responsible for software testing standards, implementation policies, and the change management committee. Documentation, including functional requirements, test cases, and test results, as well as the test lab, were also identified as non-technical agents.

4.3. Structure

The structure refers to the protocol followed to accomplish the tasks assigned to individuals within the team. Every individual has to take responsibility to delivering their tasks on scheduled time. Continuous reporting is necessary especial when an individual is struggling to perform what is assigned to them. These enables others to chip in and assist so that the project can be delivered on a promised time. Agents utilize structures to create and recreate social activities [34]. In the context of Structuration Theory, structure refers to rules and resources. Rules comprise regulations and policies guiding software testing activities, including the change management process and implementation policy. Resources encompass material and non-material objects used to carry out actions In this research, resources included software test analysts, project managers, business analysts, functional support personnel, software developers, software development methodology, software testing standards, implementation policies, change management committee, documentation (functional requirement, test cases, test results, test plan), and test lab [52].

4.4. Duality of Structure

4.4.1. Signification

Organisations develop or enhance existing software to render services, sell products, and conduct day-to-day activities. This software must be rigorously tested prior to implementation to eliminate defects. Failure to do so may result in losing existing clients and failing to attract potential clients. One participant emphasized that *"Testers uncover and eliminate things that we designers and analysts have overlooked when we were planning"* (ST_06, 17:647-648). Software testing ensures business continuity and quality, as highlighted by a functional support personnel: *"We want to deliver a quality and functional software that meets the user requirements"* (ST_11, 27:1041). Interpretive Scheme (Stock of Knowledge)

It was vital for other employees within the organization to understand the significance of software testing. However, employees from other departments perceived software testing as a waste of time and a delay in implementation. This lack of understanding developed a negative perception towards software testing. One participant noted that *"People who are not involved in software testing think that software testing is not important"* (ST_01, 06:215). Consequently, only the software testing team comprehended the value of software testing, as stated by a software tester: *"Maybe do some roadshow and explain to them what exactly testing entails in order for them to get a broader picture of what testing is"* (ST_03, 08:312-313).

Even a software developer expressed uncertainty about the role of software testing: *"I do have an idea but I don't really know what they do...let's say after unit testing I know that we hand over the software to them...they already have some test cases, test scripts or whatever...mostly they will be verifying characters and send the results to me"* (ST_13, 29:1142-1144). This lack of understanding created animosity between non-technical agents (project team members).

4.4.2. Communication

Effective communication was crucial among project team members to deliver quality software. Both functional and non-functional requirements were communicated through the functional requirements specifications. Verbal communication among team members and updates to the functional requirement specification ensured clarity. A software tester emphasized that *"Others felt that by verifying software we are judging that they did not do their work properly...especially when a tester logs a defect against the developer...they will argue with you and even fight with you verbally"* (ST_03, 08-09:316-319).

Software testers relied on the functional requirement specifications to verify the validity of the developed software. The developed software needed to correspond with the specified documents to deliver quality software. Therefore, within this organization, requirements were communicated through documentation, such as business requirements, functional specifications, and technical requirement specifications: *"All the user requirements are documented, and the user requirement specification is signed off prior to development"* (ST_05, 12:472-473).

4.4.3. Power

Solid decisions needed to be made regarding software development, testing, and implementation. The change management committee held the power to decide whether software development was necessary. They relied on testing results from the software testing team to decide whether software was ready for implementation. A functional support personnel stated that *"We have a change management department where they actually decide whether the change is needed or not"* (ST_07, 18:674-675).

Other employees within the organization exercised their power to influence project prioritization. This power was influenced by the position (facility) individuals occupied within the organization, such as the CIO. When the CIO committed to a project, it dominated other projects and received high priority. A test analyst noted that *"The CIO committed to that project, and they drilled down to the testing team, and the project was really regarded as important, and it had to undergo testing"* (ST_08, 21:797-799).

4.4.4. Facility

The change management committee exercised power over the agency (software development team) based on the facility (authority) granted to them by the organization. They made decisions about what needed to be developed, tested, and implemented. Consequently, they relied on software testing results to decide whether software was implemented or not: *"We have a change management department where they actually decide whether the change is needed or not"* (ST_07, 18:674-675).

On the other hand, project managers had the tendency to decide on behalf of the software testing team. As a result, project managers dominated other teams, such as the software testing team, by imposing timelines without consulting the team. A software tester noted that *"Unrealistic schedules from the project team...because if the project is not scheduled properly, it puts testing under pressure...for example, testing could be allocated three months for conducting all the testing, which makes it difficult for the testing team to meet timelines"* (ST_08, 20:781-783).

4.4.5. Sanction

The organisation had processes in place that needed to be followed to implement software. However, instances occurred where these processes were bypassed by some employees, becoming a norm for many projects. Consequently, many software failures occurred in production. A participant stated that *"Processes are not followed at all, and management is doing nothing about the situation"* (ST_05, 14:545).

According to the organisation's regulations, software should be developed, tested, and then implemented. This regulation should be the norm for the agency (software development team). However, the change management committee needed testing results to decide whether to implement software.

4.4.6. Norm

Adhering to standards, procedures, and policies became a norm for employees within the organization. However, instances occurred where project managers prepared project schedules and made estimations without consulting the testing team. Also, project managers promised to deliver software to business within a particular duration that was not agreed upon with the software testing team. As a result, the software testing team was pressed to complete testing within a short period, leading to working overtime and even coming in on weekends to finish their work. Such working conditions negatively impacted software quality, as a tired software tester was more likely to make mistakes.

The relationship between software testers and software developers was strained due to logged defects. Software developers felt that software testers were not recognizing their hard work. Some software was implemented without being tested and was tested in production because scheduled timelines were not met. The test manager asserted that *"Quality will be impacted, and as a result, you are likely to produce production issues when you deploy any application or system that didn't follow a proper process"* (ST_14, 34:1317-1318).

As a result, it became a norm not to follow proper processes when testing software within the organization. Therefore, it was management's responsibility to enforce standards, procedures, and policies. The software testing team complained to their test manager, who alerted management about processes not being followed by other members of the software development team. However, nothing seemed to be changing. Thus, one software test analyst angrily stated that *"Processes are not followed at all, and management is doing nothing about the situation"* (ST_05, 14:545).

It was vital for employees to follow organizational processes to deliver quality software. Consequently, Setlamo Technologies' clients would be satisfied with what was delivered to them. One participant stressed that *"It is important to make sure that processes are followed because, in that case, testing will not be bypassed"* (ST_04, 11:427-428).

4.4.7. Legitimation

Irrespective of either right or wrong, a norm is legitimized. Thus, it is important to do things properly within the organization. If nothing is done about it, then the organization would fail to achieve its goals, and the client would be unhappy with the quality of software delivered. Therefore, some agents in the organization legitimize software testing practices by accepting it as the norm. These agents understood that once software is developed, it needs to be tested to ensure it meets client requirements. A software tester noted that *"Testing is a crucial part of the software development life cycle...you cannot just develop and implement without testing"* (ST_02, 07:263-264). This legitimation of software testing practices is essential for the organization to deliver quality software to its clients. However, some agents within the organization did not legitimize software testing practices, leading to software testing being bypassed or not being done properly.

4.4.8. Domination

The power dynamics within the organization led to domination by some agents over others. Project managers dominated software testers by imposing timelines without consulting them. This domination led to software testing being done hastily, resulting in poor quality software being delivered to clients. A software tester stated that *"Project managers promise to deliver software to the business within a particular duration that was not agreed upon with the testing team"* (ST_08, 20:781-783).

4.4.9. Signification

Software testing signified quality software delivery to clients. It ensured that software met client requirements and was free from defects [28]. A software tester emphasized that *"Testing ensures that the software meets the requirements...it ensures that the software is functional and works as expected*" (ST_06, 17:647-648). However, some agents within the organization did not signify software testing, leading to poor quality software being delivered to clients.

4.5. Software Testing in the SDLC: A Structuration Theory Perspective

At Setlamo Technologies, software testing is a critical component of the Software Development Life Cycle (SDLC). Our research identified agents (technical and non-technical) and structures (rules and resources) involved in software testing [29]. Technical agents included testing tools and methodologies, while non-technical agents comprised project managers, business analysts, and software developers.

4.6. Current SDLC Market Trends

- Agile methodologies emphasize collaboration and communication among agents.
- DevOps practices integrate testing into the development process.
- Continuous Testing and Continuous Integration/Continuous Deployment (CI/CD) pipelines automate testing processes.
- Artificial Intelligence (AI) and Machine Learning (ML) enhance testing efficiency and effectiveness.

Agile methods vary in their ways, but they share a common aim which is to enable their teams swiftly respond to change [53]. In [54] stated that when modifications are expensive to adjust to later in the project, the capability to respond quicker to modification minimizes the project risks and their budgets [55]. In [56] alluded that while agile methods are efficient, huge, and complex software products often needs methodical discipline with the obligatory process to guarantee success. On the other hand, DevOps was introduced around 2007 and 2008 after software development communities realized the fatal dysfunction within the software development landscape. There was a disconnect between those who develop the software and those who implement and maintain the software. According to [62] often in the software deployment, employees who are involved in the development of software are not necessarily the ones who are involved in the implementation, hence the disconnect is encountered. DevOps approach assists in delivering value faster and uninterruptedly, minimizing challenges because of miscommunication between team members as well as fast-tracking problem resolution [57]. In [58], alluded that DevOps is an organisational shift which substitute distributed siloed groups executing tasks separately with cross-functional teams which work on continuous operational feature deliveries. In simple terms DevOps is a culture shift which provides collaboration amongst development, quality assurance and operations. In [59] the authors highlighted that while continuous integration (CI) combines work-in-progress numerous times a day, continuous deployment (CD) focuses on to possible release values to consumers faster and capably by employing automation as much as possible. Artificial

Intelligence (AI) and Machine Learning (ML) are not just a buzzword in the digital era but the best way of doing things. Gone are those times of doing things in a traditional manner. AI makes provision for countless improved results on a bigger scale and more complex neural networks, packed with many layers deep learning and much progress can be ascribed to bigger data sets and large-scale learning/training on graphic processing unit [60]. Computers are taught to emulate humans through performing complex tasks which used to be historically performed by humans such as reasoning, making decisions or solving problems. Whilst ML which is a subset of AI which is used to learn from large data sets. It enables computers to learn from data without being explicitly programmed [61]. DL on the other hand enables computers to learn complex concepts through creating them out of simpler ones [62]. It uses neural networks to process data like humans.

Our research highlights the importance of software testing in the SDLC, emphasizing the need for effective communication, collaboration, and process adherence. As the SDLC market continues to evolve, organisations must prioritize software testing to deliver high-quality products and services. By embracing current trends and best practices, organisations can optimize their software testing processes and stay competitive in the market.

5. Findings

Poor management and lack of process compliance were significant factors contributing to poor quality software at Setlamo Technologies. The organisation's failure to enforce processes, policies, standards, and procedures led to software testing being treated as an afterthought, resulting in poor quality software. This lack of emphasis on software testing also led to a culture of neglect, where software testing was seen as a mere formality rather than a critical aspect of software development.

The software testing team faced unrealistic timelines, and their concerns were ignored by management, leading to frustration and high turnover rates. This highlights the need for management to prioritize software testing and provide adequate resources and support to the testing team. Non-compliance to processes was a norm, with some employees bypassing the change management committee and implementing software without testing. This lack of compliance led to poor quality software, reputational damage, and loss of customers. It also suggests a lack of accountability and a culture of siloed work, where individuals prioritize their own goals over the organization's overall objectives.

Furthermore, a lack of software testing knowledge among employees contributed to the undervaluing of software testing, leading to frustration among software testers and a high turnover rate. This highlights the need for training and education programs to ensure that all employees understand the importance and benefits of software testing. Furthermore, a lack of software testing knowledge among employees contributed to the undervaluing of software testing, leading to frustration among software testers and a high turnover rate. This highlights the need for training and education programs to ensure that all employees understand the importance and benefits of software testing.

The disconnect between project stakeholders, including project managers, software developers, and testers, also hindered effective software development and testing. This suggests a need for improved communication, collaboration, and integration among stakeholders to ensure that software development and testing are aligned with organizational goals.

To address these issues, management must enforce processes, policies, standards, and procedures, and ensure that all employees understand the value of software testing. Additionally, stakeholders must work collaboratively to deliver good quality software, and management must address non-compliance and knowledge gaps to prevent poor quality software. This may involve implementing quality control measures, providing training and education programs, and fostering a culture of collaboration and accountability.

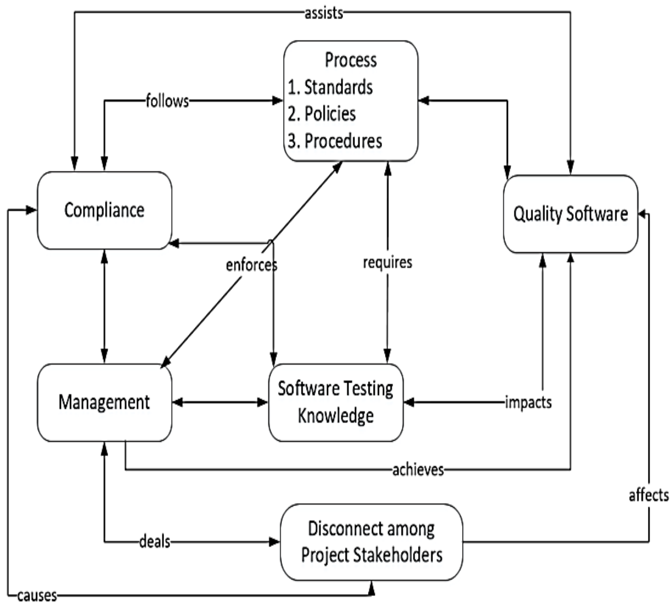


Figure 2: Factors affecting the quality of software at Setlamo Technologies

6. Summary

In summary, the study reveals significant challenges in software testing practices at Setlamo Technologies, including:

6.1. Poor Management and Lack of Adherence to Processes, Policies, and Standards

Poor management and lack of adherence to established processes, policies, and standards were significant contributors to the software quality issues at Mmuso Technologies. The absence of effective leadership and clear goals led to a lack of direction and focus among team members. Furthermore, the failure to enforce processes, policies, and standards resulted in a culture of non-compliance, where employees bypassed established procedures, leading to inconsistent and poor-quality software development.

6.2. Non-Compliance to Processes, Leading to Untested Software Implementation

Non-compliance to processes led to untested software implementation, which significantly impacted software quality. The lack of adherence to established processes resulted in software being implemented without proper testing, leading to errors, bugs, and defects. This not only affected the software's performance but also compromised its reliability and security. The failure to follow established testing processes led to a lack of confidence in the software's quality, ultimately affecting customer satisfaction.

6.3. Poor Quality Software, Resulting from Inadequate Testing and Lack of Cooperation among Stakeholders

Poor quality software was a direct result of inadequate testing and lack of cooperation among stakeholders. Insufficient testing led to undetected errors, bugs, and defects, while the lack of cooperation among stakeholders hindered effective communication, collaboration, and coordination. This resulted in software that failed to meet customer requirements, was unreliable, and lacked security. The absence of a collaborative environment led to a lack of accountability, further exacerbating software quality issues.

6.4. Lack of Software Testing Knowledge among Employees, Leading to Frustration and Undervaluation of Testing

The lack of software testing knowledge among employees led to frustration and undervaluation of testing. Employees without proper training and understanding of testing principles and methodologies struggled to effectively test software, leading to inadequate testing and poor software quality. The undervaluation of testing resulted in a lack of resources, support, and recognition for testing efforts, further demotivating employees and perpetuating software quality issues.

6.5. Disconnect Between Project Stakeholders, Causing Process Non-Compliance and Poor Software Quality

The disconnect between project stakeholders led to process non-compliance and poor software quality. Poor communication, collaboration, and coordination among stakeholders resulted in a lack of understanding of project requirements, leading to non-compliance with established processes. This, in turn, led to poor software quality, as stakeholders worked in silos, prioritizing individual goals over project objectives. The absence of a unified approach led to a lack of accountability, further exacerbating software quality issues. These challenges lead to reputational damage, customer loss, and decreased trust in the organization. To address these issues, the organization must:

1. Enforce processes, policies, and standards.
2. Educate employees on software testing's value.
3. Foster cooperation and communication among stakeholders.
4. Address knowledge gaps and provide training.
5. Encourage a culture of quality and testing.

By addressing these challenges, Setlamo Technologies can improve software quality, increase customer satisfaction, and maintain a competitive edge in the industry.

6.6. Key Findings:

Software testing is crucial for delivering quality products and services, ensuring that software meets client requirements and is free from defects.

1. Effective communication and collaboration among agents, including software testers, developers, and project managers, are vital for successful software testing outcomes.
2. Power dynamics and decision-making processes significantly impact software testing outcomes, with project managers' decisions often dominating software testers' work.
3. Adhering to organizational processes and standards is essential for ensuring quality software delivery, preventing software testing from being bypassed or done hastily.
4. Legitimization of software testing practices is necessary for instilling a sense of belonging and

control among software testers, recognizing the importance of their role in delivering quality software.

7. Recommendations

1. Establish clear communication channels and collaboration frameworks to facilitate effective interaction among agents involved in software testing.
2. Empower software testers by involving them in decision-making processes and providing autonomy in their work to ensure quality software delivery.
3. Develop and enforce organizational processes and standards that prioritize software testing, preventing domination by project managers' decisions.
4. Provide training and resources to software testers to enhance their skills and knowledge, legitimizing their role in delivering quality software.
5. Conduct regular assessments and evaluations to identify areas for improvement in software testing practices, ensuring continuous quality improvement.

8. Conclusion

In conclusion, this study highlights the critical role of software testing in delivering quality products and services, emphasizing the need for effective communication, collaboration, and empowerment of software testers. The findings underscore the impact of power dynamics and decision-making processes on software testing outcomes, stressing the importance of adhering to organizational processes and standards. By implementing the recommended measures, organisations can legitimize software testing practices, foster a sense of belonging and control among software testers, and ultimately ensure the delivery of high-quality software that meets client requirements. Moreover, this study demonstrates that addressing the challenges in software testing practices can have far-reaching benefits, including:

- Enhanced software development life cycle
- Improved customer satisfaction
- Increased trust and reputation
- Better decision-making processes
- Empowered software testers
- Competitive edge in the industry

By prioritizing software testing and addressing the identified challenges, organisations can unlock these benefits and deliver high-quality software products and services that meet the evolving needs of their clients. Ultimately, this study contributes to the growing body of knowledge on software testing practices, emphasizing the need for a collaborative, empowered, and process-driven approach to software testing.

Acknowledgement

We extend our heartfelt appreciation to ESKOM South Africa, Department of Research, for their generous sponsorship, which has enabled us to publish this paper. We are deeply grateful for their recognition of the significance of research and its contribution to the existing body of knowledge. Their support and understanding are truly valued, and we express our sincere thanks.

References

[1] T. Bryant, Software Development: A Practitioner's Approach, Routledge, 2017.

[2] R. Tuteja and S. K. Dubey, Software Testing: Concepts and Operations, PHI Learning, 2012.

[3] J. P. Kotter, "Leading Change," *Harvard Business Review Press*, 2012.

[4] G. J. Myers, "The Art of Software Testing," *John Wiley & Sons*, 2011.

[5] M. Oluigbo, L. Erasmus, and R. Snyman, "An Exploratory Study of Software Testing Practices in South Africa," *South African Computer Journal*,, vol. 29, no. 1, 1-15, 2017.

[6] J. Cameron and P. Green, Software Testing: A Guide to the TMap Approach, Pearson Education,, 2015.

[7] A. Giddens, Central Problems in Social Theory: Action, Structure, and Contradiction in Social Analysis, University of California Press, 1979.

[8] W. J. Orlikowski, "The Duality of Technology: Rethinking the Concept of Technology in Organizations", Organization Science," vol. 3, 398-427, 1992.

[9] C. Jones, "Software Project Management Practices: Failure to Apply Project Management Principles," 2011.

[10] M. Pozzebon, "The Influence of a Quality Management System on the Software Development Process," *Journal of Systems and Software*, 2004.

[11] H. K. Klein, M. D. Myers, "A Set of Principles for Conducting and Evaluating Interpretive Field Studies in Information Systems," *MIS Quarterly*, 2011.

[12] W. H. Sewell, "A Theory of Structure: Duality, Agency, and Transformation," *American Journal of Sociology*, vol. 98, no. 1, pp. 1-29 , 1992.

[13] W. J. Orlikowski, "Using Technology and Constituting Structures: A Practice Lens for Studying Technology in Organizations. Organization Science”," *Organization Science*, vol. 11, 404-428, 2000.

[14] G. Walsham, "Interpreting Information Systems in Organizations," *John Wiley & Sons*, 1993.

[15] I. I. IEEE29119-1:2018, "Software and Systems Engineering — Software Testing — Part 1: Concepts and Definitions," *International Organization for Standardization*, 2018.

[16] E. Dustin., "Automated Software Testing: A Guide for Software Project Managers," *Charles River Media*, 2017.

[17] C. Kaner, Lessons Learned in Software Testing: A Context-Driven Approach, John Wiley & Sons, 2013.

[18] T. Fulcini et al., "A review on tools, mechanics, benefits, and challenges of gamified software testing," *ACM Computing Surveys*, vol. 55, no. 14s, 1-37, 2023.

[19] D. Deak et al., "The Impact of Agile Methods on Software Project Management," *International Journal of Information Technology Project Management*, 2016.

[20] IEEE, "IEEE Standard for Software and System Test Documentation (IEEE Std 829-2019)," *IEEE Computer Society*, 2019.

[21] P. E. Black, "Managing Software Projects. In Encyclopedia of Software Engineering," *CRC Press* , 359-373, 2008.

[22] ITIL, ITIL Foundation: ITIL 4 Edition, AXELOS, 2019.

[23] I. Sommerville, Software Engineering, Pearson, 2016.

[24] R. S. Pressman, "Software Engineering: A Practitioner's Approach," *McGraw-Hill*, vol. 2, 41-42., 2010.

[25] K. Laudon, J. P. Laudon, Management Information Systems: Managing the Digital Firm, Pearson, 2015.

[26] Y. Zhao et al., "Software Quality Requirements in the Context of Digital Transformation," *International Journal of Software Engineering and Knowledge Engineering*, 2021.

[27] T. Sekgweleo, T. Iyamu, "Software testing: some influencing factors in a South African organisation," *Journal of Contemporary Management*, vol. 17, no. 1, 86-107, 2020.

[28] O. Ibitomi et al., "Automation of Software Testing: A Systematic Review," *Journal of Software Engineering and Applications*, vol. 17, no. 1, 1-22, 2021.

[29] T. Sekgweleo, "Disjoint between development and deployment of software," (Masters dissertation, Tshwane University of Technology, 2011).

[30] G. Bansal, "Software Development Strategy, In Encyclopedia of Software Engineering," *Taylor & Francis*, 1-10, 2008.

[31] G. Ghosh, "Software Development: Principles, Methodologies, Tools, and Techniques," *CRC Press*, 2017.

[32] T. G. Sekgweleo, "A decision support system framework for testing and evaluating software in organisations," (Doctoral dissertation, Cape Peninsula University of Technology, 2018).

[33] K. Avison, G. Fitzgerald, Information Systems Development: Methodologies, Techniques and Tools, Pearson, 2015.

[34] S. Nidhra, J. Dondeti, P. Katikar and S. Tekkali, "Implementing the concept of refactoring in software development," *In 2012 CSI Sixth International Conference on Software Engineering (CONSEG)*, 1-8, 2012.

[35] M.E. Khan, F. Khan, "A comparative study of white box, black box and grey box testing techniques," *International Journal of Advanced Computer Science and Applications*, vol. 3, no. 6, 1-141, 2012.

[36] T. Sekgweleo et al., "Structuration Theory: A Review of the Literature," *Journal of Sociology and Social Anthropology*, vol. 8, no. 2, 147-164, 2017.

[37] T.G. Sekgweleo, M. Makovhololo, "Structuration Theory: A Framework for Understanding Software Testing," *Journal of Software Engineering and Applications,*, vol. 16, no. 1, 1-15, 2023.

[38] T. Iyamu, D. Roode, "The use of structuration theory and actor network theory for analysis: Case study of a financial institution in South Africa," *Social influences on information and communication technology innovations, IGI Global*, 1-19, 2012.

[39] L. Ma, Knowing and teaching elementary mathematics: Teachers' understanding of fundamental mathematics in China and the United States, Routledge, 2010.

[40] B. P. Lamsal, "Production, health aspects and potential food uses of dairy prebiotic galactooligosaccharides," *Journal of the Science of Food and Agriculture* , vol. 9, no. 10, 2020-2028, 2012.

[41] W. H. Sewell. Jr, "A theory of structure: Duality, agency, and transformation," *American journal of sociology*, vol. 98, no. 1, 1-29, 1992.

[42] J. W. Creswell, "Research Design: Qualitative, Quantitative, and Mixed Methods Approaches," *Sage Publications*, 2014.

[43] M. Q. Patton, "Qualitative Research and Evaluation Methods," *Sage Publications*, 2002.

[44] L. Rademaker, "Qualitative Research from Start to Finish: A Book Review," *Qualitative Research*, vol. 16, no. 5, 1425-1428, 2011.

[45] S. Kvale, "Interviews: Learning the craft of qualitative research interviewing," *Sage*, 2009.

[46] P. Nemutanzhela, T. Iyamu, "A framework for enhancing the information systems innovation: using competitive intelligence," *Electronic Journal of Information Systems Evaluation*, vol. 14, no. 2, 242-253, 2011.

[47] J. Low, "A pragmatic definition of the concept of theoretical saturation," *Sociological focus*, vol. 52, no. 2, pp. 131-139, 2019.

[48] G. Walsham, "Decentralization of IS in developing countries: power to the people?," *Journal of Information Technology*, vol. 8, no. 2, 74-81, 1993.

[49] P. Makovholo et al., "Diffusion of innovation theory for information technology decision making in organisational strategy," *Journal of Contemporary Management*, vol. 14, no. 1, 461-481, 2017.

[50] Y. Sarason et al.,, "Entrepreneurship as the nexus of individual and opportunity: A structuration view," *Journal of business venturing*, vol. 21, no. 3, 286-305., 2006.

[51] M. Peillon, "The Constitution of Society, Outline of the Theory of Structuration," *Oxford University Press*, vol. 1, no. 3, 261-263, 1985.

[52] N. Barqawi, "Software service innovation: an action research into release cycle management," 2014.

[53] M. Coram, S. Bohner, "The impact of agile methods on software project management," *In 12th IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS'05)*, 363-370, 2005.

[54] F. Paetsch et al., "Requirements engineering and agile software development," 2003.

[55] K. Beck, eXtreme Programming Explained, Addison-Wesley, 2000.

[56] T. Sekgweleo, T. Iyamu, "Empirically Examined the Disjoint in Software Deployment: A Case of Telecommunication," *International Journal of Actor-Network Theory and Technological Innovation*, vol. 4, no. 3, 36-50, 2012.

[57] M. Virmani, "Understanding DevOps & bridging the gap from continuous integration to continuous delivery," 2015.

[58] M. Standar, "Continuous architecture in a large distributed agile organization: A case study at Ericsson," *IEEE Explore*, vol. 33, no. 3, 1-104, 2017.

[59] R. T. Yarlagadda, "Understanding DevOps & bridging the gap from continuous integration to continuous delivery," *International Journal of Emerging Technologies and Innovative Research*, 2349-5162, 2018.

[60] R. Feldt et al., "Ways of applying artificial intelligence in software engineering," 2018.

[61] B. Mahesh, "Machine learning algorithms-a review," *International Journal of Science and Research (IJSR*, vol. 9, no. 1, 381-386, 2020.

[62] M. R. Minar, J. Naher, "Recent advances in deep learning: An overview," *arXiv preprint arXiv:1807.08169*, 1-31, 2018.

Tefo Gordon Sekgweleo has done his master's degree from Tshwane University of Technology in 2012. He completed his PhD degree from Cape Peninsula University of Technology in 2018. He started his career as a software developer, he moved to software testing as a software automation engineer. He became a software testing manager, I have 36 publications, and currently working as a research manager for digitalization.

Phathutshedzo Makovhololo is a distinguished IT professional and scholar with 18 years of experience in leadership and senior management roles. Holding a PhD in Informatics, she possesses expertise in IT governance, policy management, business analysis, and project management. With a strong ability to bridge the gap between technology and business strategy, Dr. Makovhololo has a proven track record of effective leadership, lecturing, and research. Her notable strengths include visionary leadership, excellent communication, and strategic thinking, complemented by a strong research and analytical skillset.

Received: 15 September, 2024 July, 2024, Revised: 30 October, 2024, Accepted: 23 November, 2024, Online: 22 December, 2024

DOI: https://doi.org/10.55708/js0312002

Advanced Digital Twin of a Industrial Robotic System for Measuring Pipe Wall Thickness in Nuclear Power Plants

Rogério Adas Pereira Vitalli *, João Manoel Losada Moreira
Federal University of ABC (UFABC), Interdisciplinary Laboratory of Nuclear Energy (NUC-LAB), Santo André, São Paulo, 09210-580, Brazil
*Corresponding author: Rogério Adas Pereira Vitalli, rogerio.vitalli@ufabc.edu.br

ABSTRACT: This paper presents the development of the digital twin of an advanced industrial robotic system for pipe wall thickness inspection in the turbine building of nuclear power plants. The robotic inspection system consists of 3 units, the first being the mobile unit, the second a robotic unit for automatic pipe wall thickness measurement using the ultrasound technique, and the third unit for power supply, communication and auxiliary services. To develop the advanced robotic inspection system, a prototype was built in the laboratory to study different situations and geometric configurations that may arise in the field (turbine building). With Process Simulate software version 16.1.2, the digital twin of the prototype was developed including industrial robot, metal platform and pipe sections. This paper presents the results for the virtual commissioning of the wall thickness measurement of a vertical pipe section and the novel design of the End-Effector of the industrial robot. Therefore, technical discussions are made on the requirements to deal from the design to the functional requirements of developing the end-effector, digital twin and the creation of a commissioning method for industrial robots. The analyses and insights from virtual commissioning made it possible to verify that the robot could access any part of the pipe surface through interpolated movements with spatial coordinates of the robotic arm along its sides.

KEYWORDS: Digital Twin, Ultrasound Measurement, Pipe Wall Thickness, Virtual Commissioning, Process Simulate

1. Introduction

Complex industrial facilities such as nuclear power plants require pipe wall thickness inspection for safe and reliable operation. These pipes, submitted to high pressure and temperature, connect the steam generators, turbine, condenser, and may undergo thickness reduction due to corrosion and erosion [1]-[3]. The inspection environment is hot and humid (50ºC and 100 % relative humidity). The field space has complicated geometry and access to inspection locations, the pipes have diameters varying from 50 cm to 65 cm, and are positioned in horizontal, vertical, or inclined directions. During plant panned shutdown, the inspections may include tens of thousands measurement points of wall thickness [4], [5].

Traditional inspection systems are usually rigid and require intense human action to circumvent field unforeseen scenarios to make wall thickness measurement. Given the amount of work involved during inspections, it is desirable to automate these activities, and foresee access difficulties and variability of inspection conditions and pipes geometric configurations. To increase the efficiency to handle complex activities in changing scenarios, several authors propose the use of robotic automation systems with and without user assistive actions [6]-[8], and the use of digital twins and other industry 4.0 approaches to streamline their design and commissioning phases [9].

This work presents part of the development of a robotic system for measuring the wall thickness of the secondary system pipes of the Angra 1 nuclear power plant, more specifically the development of its digital twin. This method allows dividing the development process into two stages, the first being development via simulation (virtual project) and the second stage in which the physical project is built from the virtual project data, including the robot programming that is exported from the virtual project to the field robot [10].

Digital twins are very realistic virtual models of a process and its interaction with the actual world environment [11]. They include the equipment and all the steps to carry out a certain production process [12]. Using the benefits of the digital twin and virtual commissioning tools, it is possible to increase communication and coordination between the parts of the process, allowing for smarter decision-making [12]. This technology makes it possible to quickly and flexibly develop different situations aimed at automating and integrating operator agents and robes to perform complex tasks [13]. It allows integrating sensory data from physical assets and simulating a variety of robotic conditions in which an agent can interact. As a result, the agent can learn an adaptive task allocation strategy that increases project performance. In [13], the authors tested this method for the robot-automated construction problem. Results indicated that model task allocation reduced build time by 36% in three dynamic test environments when compared to the conventional method.

The biggest benefit of virtual commissioning is the savings. It is possible to test an investment in advance, which could be hundreds of thousands or even millions of dollars in equipment, since a single robot could cost you around $50 to $70,000. [14]. So, five robots, equipment, clamp systems, accessories, different products, and all that together in one manufacturing cell is a significant investment. Companies do not intend to develop projects without knowing if it can satisfactorily perform at the expected levels and results. Process reduction Standard operating. Possibility of carry out part of this process in a more convenient environment (no necessarily on-site) combined with the opportunity to use the emulation model for training workers. Parallel development and optimization of mechanical parts, especially mechanisms mechatronics. [15]. Programming and debugging simultaneously from the control software. The Companies aim to implement sustainable manufacturing of robotic cells, in order to improve profitability, reducing resource consumption, global expenses, as well as satisfying the regulatory input of ecological impacts. In addition, the widespread adoption of industrial robots, necessary to satisfy the ever-increasing requirements in terms of manufacturing quality, customization and flexibility, further increased the need to improve the energy efficiency of robotic cells. [16].

This article is organized as follows: section 2 presents a State of Art, section 3 presents a brief description of the robotic system for pipe wall thickness measurement and section 4 presents the method used in this research, data and details of the equipment used. In section 5 we present the results and discussions and in section 6, the final considerations.

2. State of Art

This paper seeks to present, through emerging projects, the concept of digital twins was named one of Gartner's Top 10 Strategic Technologies Trends for 2022 [17]. Thomas Kaiser, SAP Senior Vice President, IoT, put it this way: “Digital twins are becoming a business imperative, covering the entire lifecycle of an asset or process and forming the basis for connected products and services. Companies that fail to respond will be left behind.” It is estimated that within three to five years, billions of things will be represented by digital twins [18]. Using physical data about how a thing's components operate and respond to the environment, as well as data provided by sensors in the physical world, a digital twin can be used to analyze and simulate real-world conditions, respond to changes, improve operations, and add value. Industry 4.0 (i4.0), which has been known as the “Fourth Industrial Revolution”, emerges with the aim of meeting this new global demand with customized products and services [19]. One of the concepts that will be widely used in i4.0 is the Digital Twin, which from a production perspective, incorporates the virtual context into the real context of a production system. In [20], Digital Twins are very realistic virtual models of the current state of the process and its own behavior in interaction with the real-world environment, including the equipment and all the steps to carry out a given production process [21], [22]. Using the benefits of the digital twin (DT) and virtual commissioning (VC) tools, it is possible to increase the advantage of communication and coordination between manufacturing areas, allowing for smarter decision-making. This allows automated systems and robotic cells to start operating (running) much faster and with fewer errors [23].

According to [24], the growing use of Virtual Commissioning during the development process of automated factories, combined with the growing demand for better quality control, leads to the need for improved virtual plants that systematize the configuration procedures required to perform their processes. Common plant simulation techniques based on the concept of virtual commissioning of robotic cells go beyond the need to validate control algorithms. In other words, new approaches need to be developed to meet the demand for reconfiguring their operational resources in a systematic way, compatible with the flexibility that these autonomous resources currently have. The need for new solutions in Robotics Engineering® for the design of complex projects involving physical systems and the virtual part associated with them has never been so present. Virtual commissioning technology can be considered as one of the established trends in automotive assembly. Among other benefits, it promotes more efficient treatment of the complexity associated with assembly systems, capable of causing a reduction in the acceleration time of the system itself and a reduction in the development time of the product capable of meeting market competitiveness.

According to [25], the digital twin synthesizes sensory data from physical assets and is used to simulate a variety of robotic construction site conditions in which an agent may interact. As a result, the agent can learn an adaptive task allocation strategy that increases project performance. They tested this method with a case project in which a virtual robotic construction project (i.e., interlocking concrete bricks are delivered and assembled by robots) was digitally twinned for training. The results indicated that the model task allocation reduced construction time by 36% in three dynamic test environments when compared to an imperative model rule-based method.

Along with the trends of mass customization, flexible robotic applications are becoming increasingly popular. Although conventional robotic automation of workpiece handling seems to have been solved, advanced tasks still require a lot of effort to achieve. In most cases, on-site robot programming methods, which are intuitive and easy to use, are not applicable in flexible scenarios. On the other hand, the application of offline programming methods requires careful modeling and planning. Consequently, [26] proposes a generalized approach to the development methodology for flexible pick-and-place robotic work cells in order to provide guidance and thus facilitate the development process. The methodology is based on the Digital Twin (DT) concept, which allows iterative refinement of the work cell in both the digital and physical space. The goal is to streamline the overall commissioning (or reconfiguration) process and reduce the amount of work in the physical work cell. This can be achieved by digitizing and automating the development and maintaining sufficient proximity. With this, the operation of the digital model can be performed accurately in the physical work cell. The methodology is presented through a semi-structured pick-and-place task, performed in an experimental robotic work cell in conjunction with a reconfiguration scenario [27]-[28].

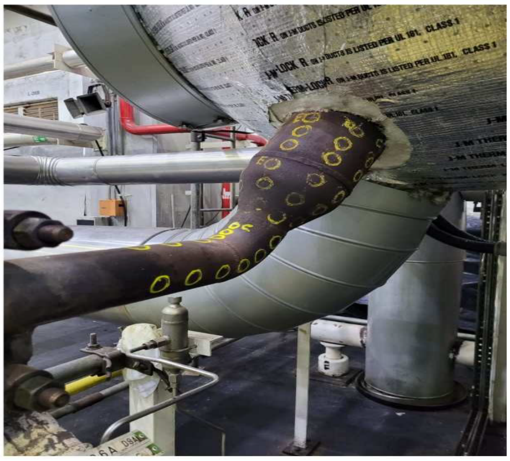
3. Brief Description on of the Advanced Robotics System

The Robotic System for pipe wall Thickness Measurement (RSTM) aims to meet the licensing needs of the Angra 1 Nuclear Power Plant for long-term operation in fast and efficient manner [29]. This RSTM has 3 units, the first, named Unit of Remotely Operated Vehicle (UROV), being a vehicle that can be operated remotely to allow displacement of the whole RSTM to the pipe area under inspection. The second unit, named Unit of Measuring Robotic Cell (UMRC), is a robotic work cell indexed on the UROV and equipped with a robotic arm that performs the movements and inspection actions and an effector with sensors for spatial orientation and for measuring pipe wall thickness through the ultrasound technique. The third unit contains all other RSTM systems such as power supply cables, data transmission cables, real-time information, computer, and ancillary equipment. This unit is named Unit of Communication, Power supply and Ancillary services (UCPA) [29].

The RSTM must be able to approach the defined inspection location and measure the thickness of the pipe wall. Figure 1 shows the complexity and geometric variability of the piping system in the field (turbine building). The solution provided for the inspection problem is a hybrid, contemplating the assistance action of the user to approach the inspection region, and robotic automation to perform the final approach to the inspection locations on the pipe surface and wall thickness measurements. An example of actual inspection locations is the grid of yellow circles shown in Fig. 1. The robot end-effector has an ultrasound-based approximation sensor to provide spatial orientation and an ultrasound-based sensor for pipe wall thickness measurement [30].

Figure 1: Piping section showing the environment and situation of pipe inspection at Angra nuclear power plant 1. Circles show positions indicated by the engineering group for pipe thickness inspection.

The user operates the UROV to approach the RSTM to



the measuring locations on the pipe external surface. The first grid location must be manually identified and recorded by the user, that is, with the help of the "teach pendant" he or she moves the industrial robot end-effector through the piping environment to the first measurement position marked on the pipe wall and register its coordinates. From there, the robot present in the UMRC, with its spatial orientation system, automatically scans all the measuring points on the grid and automatically performs the wall thickness measurements. After carrying out the measurement, the result must be transmitted to the UCPA through the communication interface with the plant.

While conventional robotic automation for handling workpieces and performing specific tasks seems to have been resolved, more complex tasks in less controlled environments, such as in the field, require more flexible solutions. In most cases, robot programming methods in a controlled environment are not applicable in variable scenarios found in the real world [31].

4. Methods, Data and Equipment

The use of the Digital Twin concept in an iterative way, both in the digital space and in the physical space, to refine the work cell development process is the solution adopted in this work. The objective is to streamline the overall commissioning process with different configurations or reconfigurations according to the work situation [31]-[32].

Figure 2 presents the main steps of the approach based on the digital twin technique for the UMRC development. The software adopted to develop the digital twin is the Tecnomatix Process Simulate, version 16.1.2 and the robot used is the ABB IRB 1600-6/1.45 model.

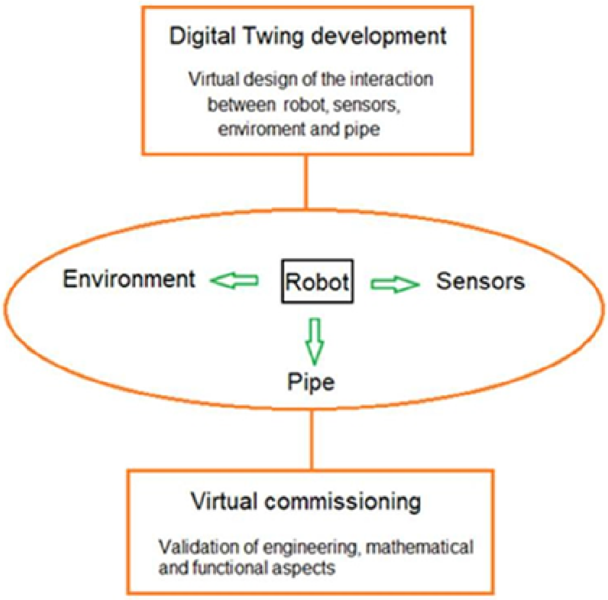


Figure 2: Scheme of the method for developing robotic systems using the digital twin technique. All the capabilities of the robot are developed via programming (software) and virtual commissioning is carried out via simulation of the various tasks that the robot must perform until its validation.

Figure 3 shows the robotic cell prototype installed at the UFABC Robotics Laboratory to develop the UMRC that contains the intelligent part of the robotic measurement system. The robotic cell includes ABB IRB 1600-6/1.45 robot, the metal platform and two pipe sections taken from the Angra 1 nuclear power plant. The pipe section is assembled on the platform in different geometric configurations and situations emulating actual measurement conditions. These configurations are obtained from field situations like the one shown in Figure 3 or others from field situations as shown in Figure 1.

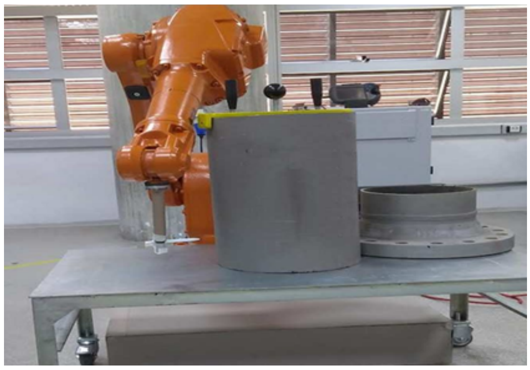


Figure 3: Piping Robotic cell in the Laboratory emulating the UMRC. One sees the ABB IRB 1600-6/1.45 robot, the metallic platform and two pipe sections taken from the Angra 1 power plant

Automatic measurement was performed numerous times using various measurement techniques to ensure repeatability, accuracy and robustness of the measured data. The ultrasonic thickness gauge from the manufacturer METROTOKYO used was the MTK-1301 PRIME model. The electronic transducer was the standard 5MHz/Ø10mm model. The longitudinal wave speed for the "glycerin" material was 1920 m/s and for the 330-steel material it was 5600 m/s, respectively, the speed of sound. Figure 4 shows the specification of the system's measuring sensors.

Figure 4: Meters used: A-) Model Prime 1300. B-) Model Prime 1301. C-) Transducer Model.

The older model did not allow data extraction, as it did not have a USB connection. Therefore, the MTK-1301 PRIME meter model was used, with the following features:

- Manual or automatic shutdown after a period of non-use;
- Table of 9 pre-established ultrasound speeds;
- Calibration function with 3mm standard attached to the meter;
- Calibration of the speed of sound with 1 or 2 points;
- Simple or sweep measurement;
- Individual measurement or with average calculation;
- Measurement with audible alarm for minimum/maximum tolerance;
- Measurement with differential calculation with percentage;
- Measurement on a surface with high temperature (optional sensor);

- Improved measurement for pipe walls. measurement

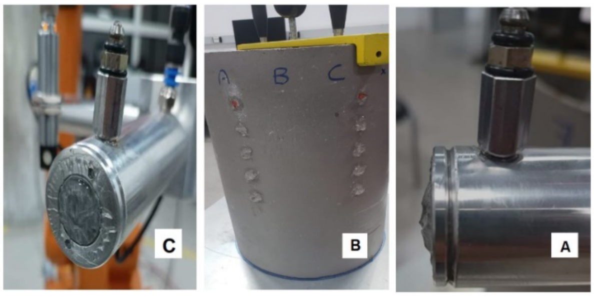


Figure 5: Ideal amount of glycerin: A-) Approximately 50g of glycerin controlled by the time the regulating valve is activated. B-) Spraying at 40 different points in the piping. C-) Visual inspection of the "micro-mesh" to prevent drying.

Figure 5 shows evidence of the unprecedented procedure developed in the laboratory for applying the coupling liquid. It was observed through numerous tests that a small amount of coupling liquid (glycerin) interferes with the quality of the measurement, that is, there are chemical reactions between the electronic transducer nozzle, the type of pipe material and the amount of glycerin. The measurements remained stable with two sprays per point and the robotic manipulator program was adjusted for this function through the "pulse" command.

The Communication, Power Supply and Ancillary Services Unit (UCPA) fulfilled the scope of its title. For the communication part, the principles of Industry 4.0 were adopted, which seeks to integrate the "work field", the engineering group and other sectors that Eletronuclear may require. After the thickness measurement was performed, the result was transmitted to the UCPA through a communication interface. These enabling technologies of Industry 4.0 follow data security and interface communication protocols based on APIs (Application Programming Interface) that allow applications to be customized according to Eletronuclear's needs. The UCPA was created and developed based on a new, more modern thickness gauge model that allowed access and transfer of the measured data that was stored inside the gauge to an external computer. Figure 7 and Figure 7 shows both systems.

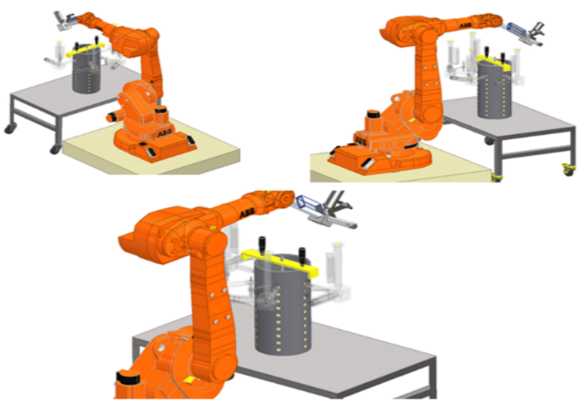


Figure 6: Different views and settings of automatic measurement.

Figure 7: UCPA Communication Interface.

5. Results and Discussions

The workstation geometric data representing the UMRC prototype (Figure 3), containing information about the robot, workstation platform, piping section and sensors, were transferred to the Process Simulate software. After modeling the UMRC prototype digital twin with all its physical characteristics (ABB IRB 1600-6/1.45 robot and end- effector), mathematics, kinematics, and information from the environment, virtual commissioning began in order to test and validate the prototype solution for building the robotic cell. Figure 4 presents the UMRC prototype digital twin developed with Process Simulate software for the configuration of a pipe section with vertical orientation. The two boxes connected at the end of the robotic arm represent volumes occupied by the end-effector. Table 1 presents the positioning of the robotic arm for a thickness measurement activity performed during virtual commissioning. It shows the positioning in terms of the angles from the 6 joints of the robot's digital twin at the measurement location indicated in Figure 8.

Virtual commissioning was performed for various locations of wall thickness measurement in this vertical pipe section ensuring that the robot could accesses any part of the pipe external surface and perform measurements. Accesses were verified by moving the robotic arm along the sides of the pipe. The result in Table I exemplifies a measurement location accessed by the robot's digital twin.

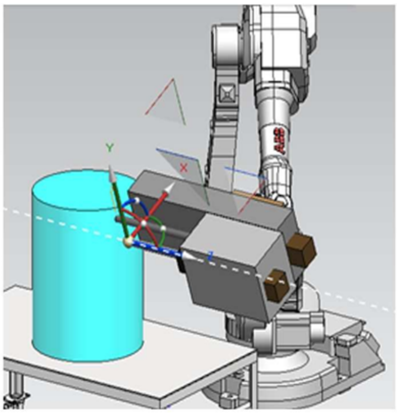


Figure 8: Digital twin of ABB model IRB 1600-6/1.45 robotic manipulator on workstation representing UMRC unit. The digital twin shows the robot, pipe section, and end-effector frame. Digital validation was performed for various measurement configurations of pipe wall thickness.

Table 1: Angles for the joints furnished by the digital twin of robot ABB IRB 1600-6/1.45. The joint angles represent a measuring position on the pipe section shown in Figure 8.

Joint	Angle (degree)
j1	37.45
j2	2.98
j3	37.70
j4	71.51
j5	36.32
j6	141.12

Figure 8a presents the OLP program generated by Process Simulate for export to the real robot ABB IRB 1600-6/1.45.

```
VERSION:1
LANGUAGE: ENGLISH
MODULE irb120_3_0_58_PNP_Angra1
!# Process Simulate Disconnected 15.1.2 to ABB Rapid download
!# ------------------------------------
!# CREATED : 22/04/2023 16:31:22 by VITALLI
!# CELL : UFABC_ABB
!# ROB    irb1600_6_145
!# ------
!# ------ ROBTARGET
!# ---------------------------------------------
 LOCAL CONST robtarget Ent:=[[623.32,231.38,364.23],[0.659024,-
0.256295,0.659024,0.256295],[0,-1,-
4,1],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
 LOCAL CONST robtarget pick1:=[[415.39,131.33,19.98],[0.659024,-
0.256295,0.659024,0.256295],[-1,-1,-1,-
1],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
 LOCAL CONST robtarget Out:=[[415.39,131.33,104.73],[0.659024,-
0.256295,0.659024,0.256295],[0,-1,-
4,1],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
 LOCAL CONST robtarget place1:=[[311.32,302.111,46.01],[0.659024,-
0.256295,0.659024,0.256295],[-1,-1,-1,-
1],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
 LOCAL CONST robtarget
In_Place:=[[325.32,337.69,137.73],[0.659024,-
0.256295,0.659024,0.256295],[0,0,-
1,1],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
 LOCAL CONST robtarget
Out_Place:=[[378.46,324.44,192.45],[0.659024,-
0.256295,0.659024,0.256295],[0,0,-
1,1],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
ENDMODULE
```

Figure 8: *Offline* (OLP) robotic program generated by Process Simulate for export to the real ABB IRB 1600-6/1.45 robot. The program was generated in the ABB RAPID language for ABB industrial robots.

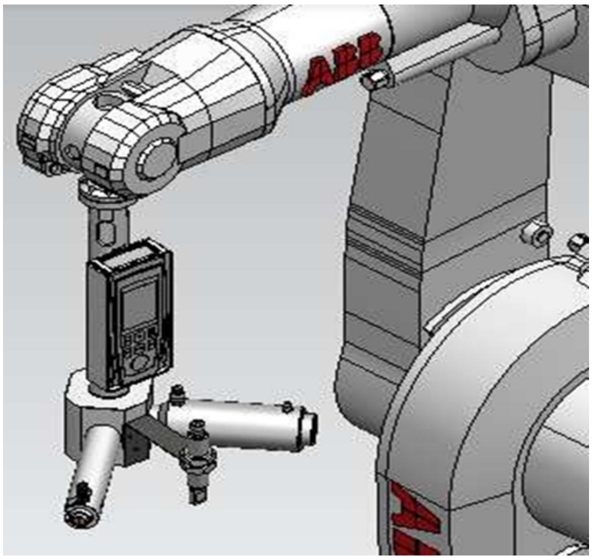


Figure 9: Final innovation design of the End-Effector.

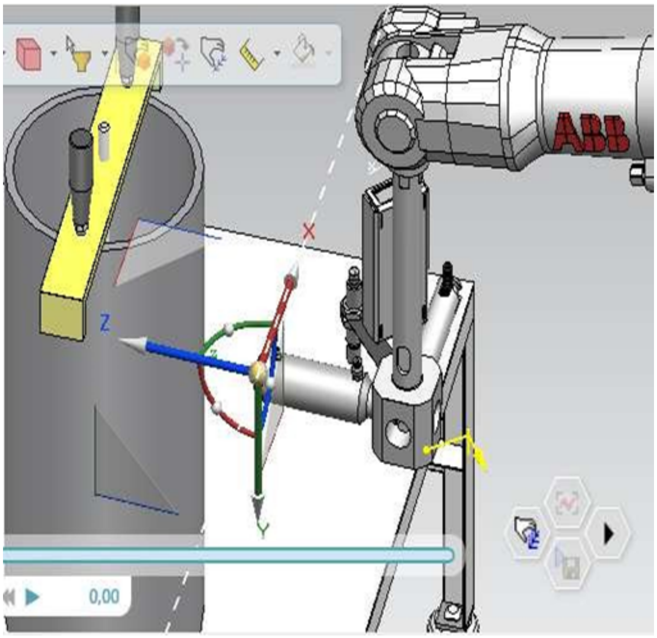


Figure 10: Projection of point vectors and spatial coordinates. Defined type of movement for the robot.

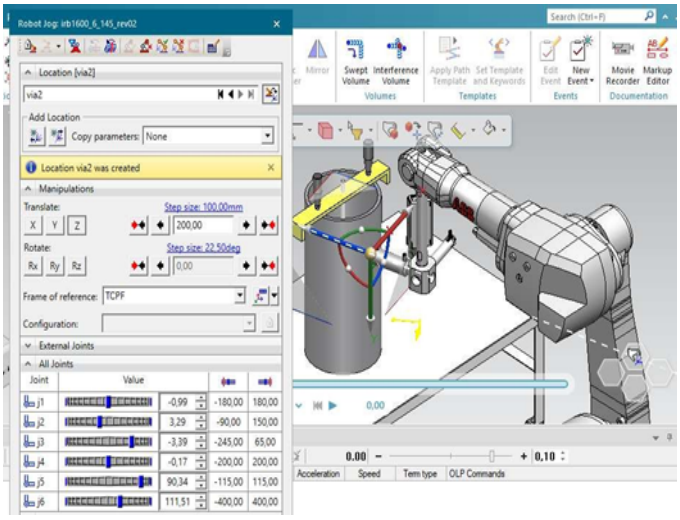


Figure 11: Recording of the first points of the trajectory of the ABB robotic manipulator with the Digital Twin scenario completed during automatic thickness measurement.

Figure 12: The ABC mesh on one side (right) of the pipe and the DEF mesh on the other side (left) of the pipe, together total 180 degrees of automatic measurement with the robot, divided into 10 layers with 40 distinct points.

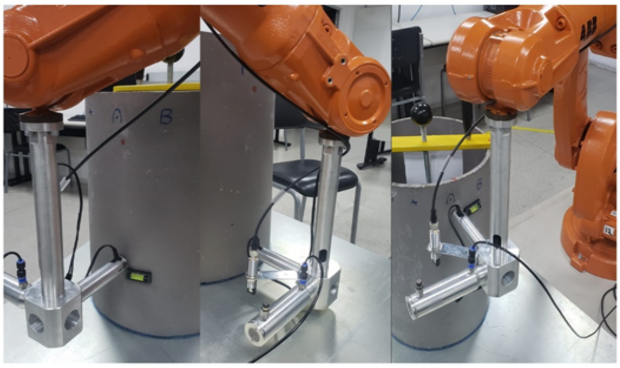


Figure 13: Automatic measurements on the last layer performed by the robotic system.

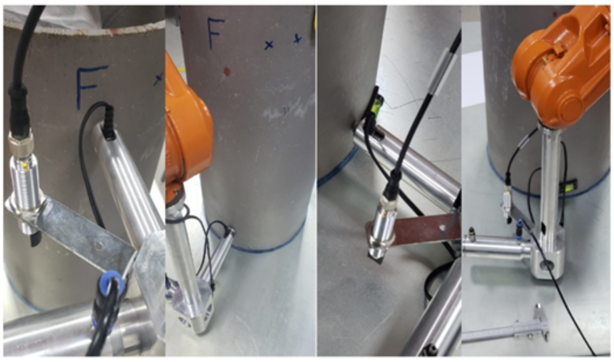


Figure 14: Automatic measurements on the first layer performed by the robotic system.

Figure 15: Example of automatic measurement performed by the robot in an unprecedented way with the value shown on the display.

Upload time:2024-6-21 23:15:52

Group 0001	No.	Thickness(mm)
	01	11,08
	02	10,43
	03	10,46
	04	10,55
	05	11,22
	06	10,55
	07	10,38
	08	10,62
	09	10,66
	10	10,79

Figure 16: Thickness measurement data from the first ten points measured on the pipe are automatically exported to any device via USB cable in a secure manner to avoid manual typing errors.

The success of automatic measurements with robotic technology was guaranteed by the innovation in the development of the automatic "spraying" process of the coupling liquid (glycerin). This accessory present in the robot's End-Effector was essential to spray the ideal amount of glycerin before the electronic transducer accurately measured the thickness at the point. It was configured and activated by the robot's control logic through a solenoid valve in conjunction with a unidirectional flow regulator to automatically control the output pressure within a range of 2.4 bar (Kgf/cm2). The ultrasonic sensor that accurately measures the distance of proximity of objects performed the measurement through a high-frequency sound wave, which is reflected by the object to be detected, thus avoiding collision between the robot and the pipe wall. The performance of the coupling liquid reservoir was extensively tested after corrections and improvements, guaranteeing expressive and unprecedented results. The End-Effector technology was 100% innovative, from its conception, design, improvements, manufacturing and implementation, validated in a real laboratory environment. The results of the digital twins revealed the importance and support they offer in the development of scenarios for the study of robotic applications. It should be noted that they were not used for real-time monitoring of the problem. The aerial trajectory positions for the robot and their respective corrections and adjustments were defined. The complete version of the digital twin was validated by comparing the virtual scenario with the real scenario. The robot's TCP was accurate and stable during the recording of points in a virtual environment. The first point was successfully recorded following the sequence of defined operations and without any type of human intervention.

The first complete thickness measurement operation was created in a customized virtual environment and as a result the OLP (offline program) code for the real industrial robot was generated. The entire laboratory layout was calibrated with a laser system to avoid variation of the piping on the bench in relation to the floor and the base of the robot. Numerous tests were performed on the robotic system before the measurements were taken, which were extensively tested.

The counter-tests of the measurements at the first points of the pipeline were performed numerous times using a caliper to ensure a standard for comparison. The discussion of physical contact in the pipeline for automatic thickness measurement between the electronic measuring transducer and the pipeline took place after spraying the coupling liquid; a fundamental factor for the measurement to occur. When the robot's End-Effector physically "touches" the pipeline, the sensor detects its presence and it enters "Hold" (a temporary stop). After this, the manipulator still exerts a small axial force to completely touch the face of the measuring transducer and ensure the quality of the procedure. Finally, the digital LCD display shows the measurement data and automatically records the value in its internal memory. Regarding the MTK 1301 Prime meter, the greatest difficulty was to equate the following variables: pre-established ultrasound speeds, calibration function with 3mm standard attached to the meter, sound speed calibration with 1 or 2 points, simple measurement or with sweep, individual measurement or with average calculation, measurement with audible alarm of minimum/maximum tolerance, measurement with differential calculation with percentage and improved measurement for pipe walls. The robotic manipulator operated at a safety speed of 10% (low inertia) in relation to the final nominal speed because the base of the robot was not "anchored" to the floor. The complete cycle time for spraying the glycerin and measuring each individual point took approximately 30 seconds.

6. Final Considerations

In this paper, we present the development of an advanced industrial robotic system for automatic inspection of tube wall thickness in the turbine building of nuclear power plants. A prototype of the system was built in the laboratory to study different situations and geometric configurations for measuring the external tube wall thickness. The digital twin of this prototype was developed using Process Simulate software version 16.1.2., including a 6-degree-of-freedom industrial robot, a metal platform, and nuclear tube sections. Digital validation was performed for various configurations of measuring the external tube wall thickness through virtual commissioning. The results of the virtual commissioning for a vertical pipe section were presented.

The virtual commissioning allowed to verify that any part of the external surface of the tube could be accessed by means of movements of the robotic arm along the sides of the tube. It was demonstrated that it is possible to measure the external tube thickness anywhere on the external surface of the nuclear tube.

The improvements in the End-Effector design were significant and the results in the design were unprecedented with numerous innovations, among them: automatic application of the coupling liquid.

The technology was digitally validated in the laboratory as shown in figures 5, 6, 7. It is currently in the final phase of experimental validation and will be presented to the plant directors. In the future, if it is of interest, it may be applied within the plant itself.

Conflict of Interest

The authors declare no conflict of interest.

Acknowledgment

The authors are grateful for the financial support of Eletrobras Termonuclear – Eletronuclear, Amazônia Azul Tecnologias de Defesa, Fundação Parque de Alta Tecnologia da Região de Iperó, and Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – CAPES.

References

[1] USNRC-NUREG–2191. *"Generic Aging Lessons Learned for Subsequent License Renewal (GALL-SLR) Report"*, Final Report Vol. 2. US Nuclear Regulatory Commission, 2017.

[2] IAEA-SRS 82. Ageing management for nuclear power plants: International generic ageing lessons learned (IGALL), Rev. 1. *International Atomic Energy Agency, 2020.*

[3] S. A. Cancemi; R. L. Frano; Preliminary study of the effects of ageing on the long-term performance of NPP pipe. *Progress in Nuclear Energy*, 131, 103573, 2021.

[4] Q. Zhang; Y. Li; E. Lim; J. Sun. Real Time Object Detection in Digital Twin with Point-Cloud Perception for a Robotic Manufacturing Station. *Proceedings of the 27th International Conference on Automation & Computing*, University of the West of England, Bristol, UK, 1-3 September 2022.

[5] H. Zhou; S. Zhang; J. Zhang; C. Zhang; S. Wang; Y. Zhai; W. Li. Design, development, and field evaluation of a rubber tapping robot. *Journal of Field Robotics*, 39, 28–54, 2021.

[6] Tugal, H.; Cetin, K.; Petillot, Y.; Dunnigan, M.; Erden, M. S. Contact-based object inspection with mobile manipulators at near- optimal base locations. *Robotics and Autonomous Systems*, 161, 104345, 2023.

[7] J. Barbosa, P. Leitão, E. Adam and D. Trentesaux, "Dynamic self-organization in holonic multi-agent manufacturing systems: The ADACOR evolution," *Computers in industry*, v. 66, p. 99-111, 2015.

[8] J. Barata, L. Camarinha-Matos and M. Onori, "A multi-agent-based control approach for evolvable assembly systems,"

INDIN'05. 2005 3rd *IEEE International Conference on Industrial Informatics*, pp. 478-483, August 2005.

[9] E. Trunzer, A. Calà, P. Leitão, M. Gepp, J. Kinghorst, A. Lüder et. al., "System architectures for industrie 4.0 applications," *Production Engineering*, v. 13, n. 3, p. 247-257, 2019.

[10] Boschert, S.; Rosen, R. *Digital Twin—The Simulation Aspect*. In: Hehenberger, P.; Bradley, D. (Editors), Mechatronic Futures - Challenges and Solutions for Mechatronic Systems and their Designers. pag. 59-74, Spring, 2016.

[11] Bratchikov,S.; Abdullin, A.; Demidova, G. L.; Lukichev, D. V. Development of Digital Twin for Robotic Arm. 2021 *IEEE 19th International Power Electronics and Motion Control Conference (PEMC)* DOI: 10.1109/PEMC48073.2021.9432535, 2021.

[12] Nekoo, S. R.; Acosta, J. A.; Heredia, G.; Ollero, A. A benchmark mechatronics platform to assess the inspection around pipes with variable pitch quadrotor for industrial sites. *Mechatronics* 79, 102641, 2021.

[13] Lee, D. et al. Digital twin-driven deep reinforcement learning for adaptive task allocation in robotic construction. *Advanced Engineering Informatics*, v. 53, 1 ago. 2022.

[14] *Gartner Top 10 Strategic Technology Trends For 2018*. Disponível em: <https://www.gartner.com/smarterwithgartner/gartner-top-10-strategic-technology-trends-for-2018>. Acesso em: 20 fev . 2023.

[15] *What Is Digital Twin Technology - And Why Is It So Important?* Disponívelem: <https://www.forbes.com/sites/bernardmarr/2017/03/06/what-is-digital-twin-technology-and-why-is-it-so-important/?sh=1365c81c2e2a>. Acesso em: 20 fev. 2023.

[16] Tugal, H.; Cetin, K.; Petillot, Y.; Dunnigan, M.; Erden, M. S. Contact-based object inspection with mobile manipulators at near-optimal base locations. *Robotics and Autonomous Systems*, 161, 104345, 2023.

[17] Bratchikov,S.; Abdullin, A.; Demidova, G. L.; Lukichev, D. V. Development of Digital Twin for Robotic Arm. 2021 *IEEE 19th International Power Electronics and Motion Control Conference* (PEMC) DOI: 10.1109/PEMC48073.2021.9432535, 2021.

[18] Nekoo, S. R.; Acosta, J. A.; Heredia, G.; Ollero, A. A benchmark mechatronics platform to assess the inspection around pipes with variable pitch quadrotor for industrial sites. *Mechatronics* 79, 102641, 2021.

[19] Qiu, B. et al. A Feasible Method for Evaluating Energy Consumption of Industrial Robots. Proceedings of the 16th *IEEE Conference on Industrial Electronics and Applications, ICIEA* 2021. Anais...Institute of Electrical and Electronics Engineers Inc., 1 ago. 2021a.

[20] Lee, D. et al. Digital twin-driven deep reinforcement learning for adaptive task allocation in robotic construction. Advanced *Engineering Informatics*, v. 53, 1 ago. 2022.

[21] Partiksha; K., A. Robotic Tele-operation Performance Analysis via Digital Twin Simulations. 2022 14*th International Conference on COMmunication Systems and Networks, COMSNETS* 2022. Anais...Institute of Electrical and Electronics Engineers Inc., 2022.

[22] Fan, S. et al. A new approach to enhance the stiffness of heavy-load parallel robots by means of the component selection. *Robotics and Computer-Integrated Manufacturing*, v. 61, 1 fev. 2020.

[23] Huynh, H. N. et al. Modelling the dynamics of industrial robots for milling operations. *Robotics and Computer-Integrated Manufacturing*, v. 61, 1 fev. 2020.

[24] Vitalli, R. *Método de abordagem e estrutura geral do projeto de pesquisa desenvolvimento da RSTM.* Documento Número: UFABC- ROBOT-001-Rev00, 2023.

[25] Partiksha; K., A. Robotic Tele-operation Performance Analysis via Digital Twin Simulations. 2022 *14th International Conference on Communication Systems and Networks, COMSNETS* 2022. Anais...Institute of Electrical and Electronics Engineers Inc., 2022.

[26] Fan, S. et al. A new approach to enhance the stiffness of heavy-load parallel robots by means of the component selection. *Robotics and Computer-Integrated Manufacturing*, v. 61, 1 fev. 2020.

[27] Huynh, H. N. et al. Modelling the dynamics of industrial robots for milling operations. *Robotics and Computer-Integrated Manufacturing*,v. 61, 1 fev. 2020.

[28] SIEMENS PLM, *Process Simulate*. Germany. Siemens Product Lifecycle Management Software. 2018. Document ID: MT45215. Version: 15.1.2.

[29] ABB, *Product manual IRB 1600/1660*. Sweden. ABB Robotics. 06/01/2022. Document ID: 3HAC026660-001. Revision: AH.

[30] Wildgrube F, Perzylo A, Rickert M, et al. Semantic mates: Intuitive geometric constraints for efficient assembly specifications. 2023 *IEEE/ RSJ International Conference on Intelligent Robots and Systems (IROS)*; 2023.

[31] Rauen H. *Industrie 4.0 in practice* - Solutions for industrial applications. Frankfurt: VDMA Industrie 4.0 Forum; 2023.

[32] Karaman S, Frazzoli E. Sampling-based algorithms for optimal motion planning. *The International Journal of Robotics Research*.2024;30(7):846–894.

ROGÉRIO VITALLI has PhD (2023) in Mechanical Engineering (Area: Mechatronics) from UNICAMP-FEM with a sandwich period in Israel and at the German Aerospace Center (Deutsches Zentrum für Luft und Raumfahrt, DLR, Robotik und Mechatronik Zentrum, Oberpfaffenhofen, Robotics and Mechatronics Center, RMC, Institute of Robotics and Mechatronics). He has experience in the area of Digital Twins, Advanced Manufacturing, Industry 4.0 and Mechatronics, working mainly on the following themes: Advanced Mechatronic Systems, Industrial Robotics, Modeling and Simulation of Robotic Manipulators, Forward Kinematics, Inverse Kinematics, Advanced Dynamics and Jacobians of Industrial Robots. He has published 14 articles in journals, 1 book, 4 book chapters, 6 articles in conference proceedings and 8 articles in newspapers. He has trained more than 200 roboticists, 9 specialists in robotics engineering. He is currently a Nuclear Engineering researcher at UFABC and a post-doctoral fellow at the University of São Paulo - USP.

JOÃO MOREIRA obtained a master's degree in nuclear science (1981) and a PhD in nuclear engineering (1984) from the University of Michigan, USA. He worked at the Institute for Energy and Nuclear Research (IPEN) between 1984 and 1989 and at the Navy Technological Center in São Paulo (CTMSP) between 1990 and 2006. In 2006 he moved to the academic field, joining the Federal University of ABC, where he is a Full Professor of Nuclear Energy. He has published 41 articles in journals, 3 books, 10 book chapters, 84 articles in conference proceedings and 4 articles in journals. He has trained 26 masters and 7 doctors and coordinated 2 research projects on the decommissioning of CNAAA (nuclear plants Angra 1, 2 and 3) for Eletronuclear.

Received: 24 September, 2024, Revised: 22 November, 2024, Accepted: 23 November, 2024, Online: 18 December, 2024

DOI: https://doi.org/10.55708/js0312003

Advanced Cloud-Based Solutions for Peripheral Artery Disease: Diagnosis, Analysis, and Visualization

Mohammed A. AboArab[1,2], Vassiliki T. Potsika[1], Dimitrios I. Fotiadis*[1,3]

[1]Unit of Medical Technology and Intelligent Information Systems, Dept. of Materials Science and Engineering, University of Ioannina, Ioannina, GR45110, Greece

[2]Electronics and Electrical Communication Engineering Dept., Faculty of Engineering, Tanta University, Tanta, Egypt

[3]Biomedical Research Institute, Foundation for Research and Technology-Hellas, University Campus of Ioannina, Ioannina, GR45110, Greece

E-mail: m.aboarab@uoi.gr / vpotsika@uoi.gr / fotiadis@uoi.gr

*Corresponding author: Dimitrios I. Fotiadis, Ioannina, GR45110, Greece, +302651005580 & fotiadis@uoi.gr

ABSTRACT: Peripheral artery disease (PAD) affects 237 million people globally, leading to significant morbidity and mortality. Traditional diagnostic methods are invasive, costly, and require specialized expertise, emphasizing the need for more accessible, and accurate alternatives. This paper introduces the DECODE cloud platform, an advanced tool that leverages cloud computing, machine learning, and high-performance data visualization to enhance PAD diagnosis and treatment. The platform integrates modules for peripheral artery segmentation, reconstruction, and comprehensive data warehousing, supporting 2D and three-dimensional (3D) rendering visualization. It enables the simulation and optimization of drug-coated balloons, enhancing clinical decision-making through robust data analytics. The evaluation metrics demonstrate the platform's efficacy: the multiplanar visualization module achieved a performance score 94%, and the 3D rendering module scored 89%, with both modules attaining perfect scores in best practices and search engine optimization. These results highlight the DECODE platform's capacity to provide scalable, noninvasive diagnostic solutions, setting a new standard in digital health technologies for PAD. This study underscores the transformative potential of integrating advanced visualization and computing techniques in medical diagnostics.

KEYWORDS: Digital health technologies, peripheral artery disease, cloud computing, medical imaging visualization, noninvasive diagnostics

1. Introduction

Peripheral artery disease (PAD) is a globally prevalent circulatory disorder characterized by the narrowing of peripheral arteries, which leads to reduced blood flow to the limbs [1]. This reduction in blood flow can cause symptoms such as leg pain or claudication during physical activity, and slow or non-healing wounds on the feet or toes. In more advanced stages, PAD can result in critical limb ischemia, leading to severe pain at rest, and in some cases, the need for amputation. This condition significantly impacts mobility and quality of life and poses serious health risks. Worldwide, PAD is estimated to affect approximately 7.4% of individuals in high-income countries and 5.1% of those in low-income countries. In Europe, particularly in high-income regions, the prevalence is higher than 10% among people aged 65 and older, emphasizing the disease's burden on elderly populations [2]. In the United States, PAD affects 8–10 million individuals, with projections suggesting that this number could rise to 19 million by 2050 [3]. The global PAD market, valued at USD 4.45 billion in 2022, is expected to reach USD 9.15 billion by 2031, growing at a compound annual growth rate (CAGR) 8.35% [4, 5]. This economic growth underscores the increasing demand for effective diagnostic and therapeutic solutions for PAD. Early and accurate diagnosis of PAD is essential to mitigate its high morbidity and mortality rates.

Traditional diagnostic methods, such as digital subtraction angiography (DSA) and magnetic resonance angiography (MRA), provide high-resolution imaging of arterial structures and are considered the gold standard for PAD diagnosis. However, DSA is invasive and involves catheterization and contrast injection, which poses risks such as bleeding, infection, and radiation exposure. While MRA is noninvasive and avoids ionizing radiation, it is expensive. Both methods are resource intensive, rely on advanced infrastructure and expertise, underscoring the need for cost-effective, scalable, and noninvasive diagnostic alternatives [6].

Recent advancements in cloud computing and machine learning are revolutionizing healthcare by offering innovative solutions to the challenges posed by traditional diagnostic methods for PAD [7, 8]. Cloud-based platforms are transforming the way health data are managed, enabling secure storage, rapid processing, and sophisticated analysis of vast datasets [9]. These platforms facilitate real-time diagnostic insights, which are crucial for timely and accurate medical decision-making. Moreover, the integration of advanced visualization techniques, such as multiplanar and three-dimensional (3D) rendering, within cloud platforms enhances the ability to examine complex anatomical structures in detail. This synergy between cloud technology and machine learning in medicine improves diagnostic accuracy and supports personalized patient care by allowing the continuous and scalable application of advanced analytics and visualization across diverse clinical settings.

In this work, the DECODE cloud platform is presented, which addresses the complexities of PAD diagnosis and treatment by integrating cutting-edge data processing, analysis, and visualization capabilities. The platform supports the simulation and optimization of drug-coated balloons (DCBs), which are increasingly used in PAD treatment. Through robust data analytics, the DECODE platform enhances clinical decision-making, allowing users to upload, process, and analyze extensive datasets, perform image reconstruction, and utilize computational modeling and machine learning techniques to accurately predict disease progression, assess treatment outcomes, and tailor personalized therapeutic strategies for PAD patients. The platform's ability to integrate these advanced technologies positions it as a pivotal tool in improving the diagnosis and management of PAD, aligning with the projected growth and evolving needs of the global PAD market.

2. Related Work

The *YORwalK* app was developed to promote exercise and track walking ability in PAD patients; however, its effectiveness requires further validation through direct patient feedback and clinical trials [10]. *TrackPAD*, a mobile intervention for supervised exercise therapy (SET), demonstrated promising initial results; however, it faced limitations due to a small sample size and short follow-up period, necessitating larger, long-term studies [11]. A smartphone-enabled exercise program introduced in the smart step trial encountered challenges in maintaining participant engagement and measuring adherence, particularly in low-resource settings [12]. Another study highlighted the need for better disease literacy and SET support in mobile interventions; however, it was constrained by a small sample size [13]. A mobile app designed for supervised home-based exercise therapy aimed to facilitate remote interventions; however, it involved minimal patient interaction, compromising data quality and requiring further development for effective supervision [14]. Research exploring machine learning (ML) and artificial intelligence (AI) applications to improve PAD outcomes emphasized critical issues such as data interoperability, algorithm bias, and the need for extensive validation [15]. An intelligent oscillometric system for PAD detection achieved high accuracy; however, it did not account for the effects of age, necessitating further validation across diverse age groups [16]. A smartphone app compared with motivational interviewing (MI) for increasing walking distance and weight loss in PAD patients found MI to be more effective; however, the study faced limitations, including a small sample size, app usage variability, and lack of an iOS version [17]. The *HOBBIT-PAD* platform was introduced to promote exercise among PAD patients; however, it is currently available only on Android and requires further evaluation by healthcare providers [18]. Moreover, a comprehensive review of SET's role in managing PAD emphasized the potential of digital health technologies for home-based interventions. However, despite their promise, these technologies suffer from low participation rates and require optimization and evaluation of long-term outcomes [19].

These studies underscore the potential of digital health technologies for PAD management but emphasize the need for larger, more comprehensive studies and further validation. Our research seeks to address these limitations by developing the DECODE cloud platform, which includes multiple modules: a PAD segmentation and reconstruction module, a data warehouse module, and a multiplanner reconstruction module featuring 2D and 3D WebGL visualization. On the basis of RESTful APIs [20], the DECODE cloud platform enables scalability and modular integration, offering a comprehensive approach to enhance diagnosis, data management, and visualization in PAD treatment, ultimately improving patient outcomes with robust and scalable solutions.

3. Materials and Methods

3.1. Data Provider

The DECODE cloud platform relies on a data provider to ensure the availability and accessibility of medical data essential for simulations, optimizations, and treatment planning [21].

Figure 1, the data provider functions as a critical bridge between the platform's core modules and various analytical tools. It integrates data from diverse sources, including medical databases, patient records, imaging systems, and research studies, harmonizing these into a unified repository for analysis and visualization.

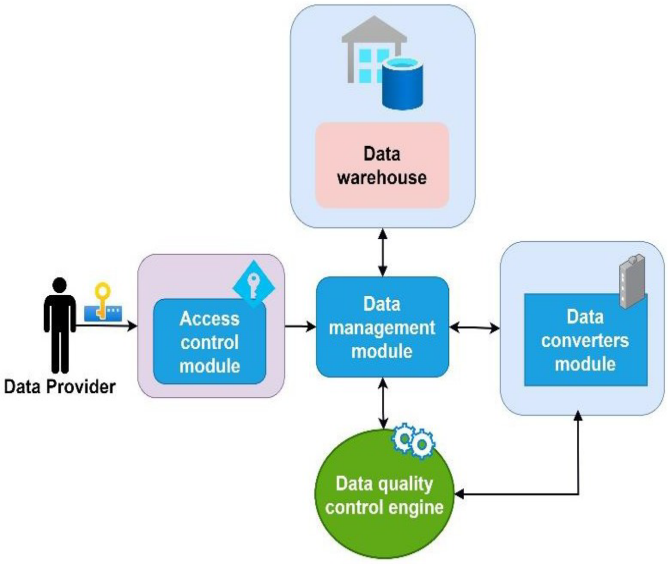


Figure 1: Data Provider Architecture within the DECODE Cloud Platform.

To maintain accuracy, the data provider continuously updates the platform with real-time data, enabling informed decision-making on the basis of the latest medical insights. The process includes cleaning, normalizing, and structuring raw medical data to ensure consistency and reliability. Quality control measures are implemented to identify and rectify any inconsistencies, enhancing the platform's reliability [22].

Security and privacy are prioritized through strict authentication and authorization protocols, allowing only authorized users to access sensitive data. The data provider supports advanced analytics, enabling the extraction of insights from complex datasets, which informs clinical decisions. As the platform scales, the data provider ensures optimal performance through efficient data retrieval and caching strategies, maintaining responsiveness even with large datasets.

3.2. *DECODE Cloud Platform Layers*

The DECODE cloud platform is a transformative solution designed to revolutionize the treatment of PAD via DCBs. By enhancing DCB simulation and optimization, the platform integrates advanced technologies and innovative research methodologies to improve treatment outcomes and patient care [23]. The conceptual multilayer hierarchical framework of the DECODE cloud platform, which provides a comprehensive overview of its architecture, is illustrated in Figure 2.

Figure 2: The DECODE Cloud Platform's Multi-Layered Hierarchical Framework.

3.2.1. *Front-End Layer: User-Centric Interactions*

The front–end layer provides clinicians with intuitive, user-friendly tools for interacting with complex medical data. These components enable seamless access to advanced visualization and analysis functionalities, enhancing clinical decision-making and improving the overall user experience.

3.2.2. *Back-End Layer: Computational Core*

The back-end layer forms the computational backbone of the platform. It incorporates advanced machine learning algorithms for automated diagnostics, computational modeling tools for simulating and optimizing PAD-specific interventions, and scalable data management modules for handling large, multi-modal datasets. Delivered as SaaS, this layer ensures high-performance processing and accurate, data-driven therapeutic strategies for PAD management.

3.2.3. *Workflow Layer: Seamless Integration and Management*

The workflow layer acts as a central hub for coordinating platform operations. Functioning as Platform-as-a-Service (PaaS), it includes a workflow management module to streamline data processing, a data quality control engine to ensure dataset accuracy, and a Docker engine and REST API manager [25] to facilitate efficient containerized application management and secure API interactions. This layer ensures seamless integration and operational consistency across the platform's processes.

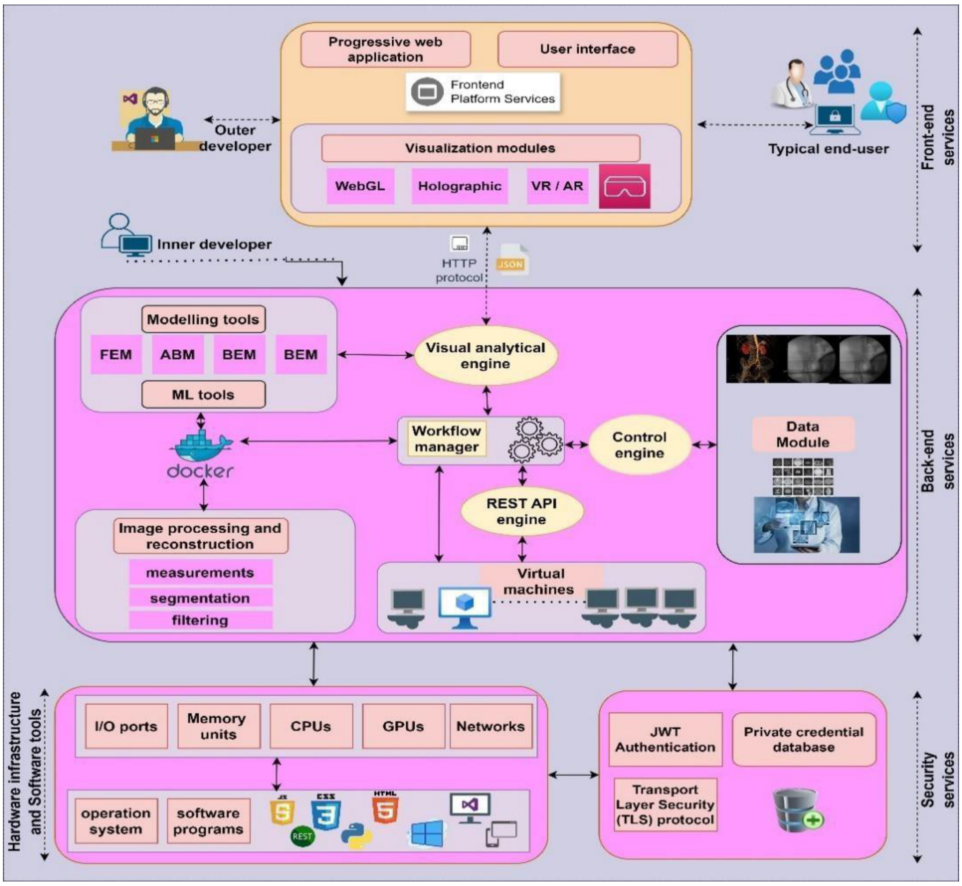


Figure 3: Comprehensive Architecture and Service Distribution of the DECODE Cloud Platform.

3.2.4. *Security Layer: Robust Data Protection*

The security layer implements advanced data protection measures, including secure socket layer (SSL) and transport layer security (TLS) protocols for encrypting data in transit, ensuring that sensitive patient information remains confidential during transfer between the platform's components. In addition, the layer employs the JSON Web Token (JWT) API framework for robust user authentication and authorization, allowing granular access control on the basis of user roles and ensuring that only authorized personnel can access specific data and functionalities within the platform.

3.2.5. *Hardware Layer: High-Performance Infrastructure*

The hardware layer, categorized as infrastructure as a service (IaaS), consists of high-performance components designed for optimal efficiency and reliability. It includes multicore dedicated servers for parallel processing, and the storage infrastructure combines SSDs for rapid data access and HDDs for cost-effective long-term storage, while GPUs are employed to expedite computationally intensive tasks such as image rendering and machine learning model training. By classifying the DECODE cloud platform's layers according to cloud computing service models, this study elucidates how the platform leverages cloud technology to provide scalable, accessible, and secure services for stakeholders, thereby opening new possibilities in PAD treatment and fostering collaboration among healthcare professionals.

3.3. *Distribution of the DECODE Cloud Computing Services to Stakeholders*

The DECODE cloud platform's architecture, as illustrated in Figure 3, is strategically designed to distribute its computing services efficiently across a diverse range of stakeholders, including clinicians, researchers, and developers. This distribution model ensures that each stakeholder can fully utilize the platform's capabilities to achieve their specific objectives in the management and treatment of PAD.

At the forefront of this distribution strategy are end-user services, which are tailored to meet the needs of clinicians and medical professionals. These services are delivered through user interfaces and visual representation modules, such as WebGL for 3D rendering, multiplanar visualization (MPV), holographic, and VR/AR technologies. These tools enable users to interact directly with patient data, provide detailed analyses and facilitate informed decision-making in real time. Front-end services, accessible via progressive web applications (PWAs) [26], form the primary touchpoint for end-users, offering an intuitive and seamless experience.

Research and development services are critical aspects of the platform's distribution strategy. These services support researchers and developers by providing access to advanced modeling tools, ML algorithms, and robust data processing capabilities. These tools are integrated within backend services, allowing researchers to conduct complex simulations, analyze clinical data, and develop new treatment methodologies.

The workflow and integration services within the DECODE platform are designed to increase operational efficiency and promote seamless collaboration among internal teams [25]. These services, which are managed by components such as the workflow manager and the REST API engine, orchestrate the flow of tasks and data across the platform modules, ensuring seamless integration of the components. By automating critical processes such as data quality control and enabling smooth integration between various analytical tools, the platform ensures a streamlined approach to patient care.

Finally, the infrastructure services layer provides the necessary computational resources that underpin the platform's operation [27]. This layer includes essential hardware components such as I/O ports, memory units, CPUs, GPUs, and networking equipment, as well as software components and operating systems. These resources are crucial for handling the platform's computational demands, ensuring that it remains responsive and capable of scaling to meet the growing needs of its users.

4. Visual Representation Modules

The implementation architectures for the MPV and 3D rendering modules within the DECODE cloud platform are illustrated in Figure 4. These modules are integral for providing detailed and interactive visualization of medical imaging data, which are specifically tailored for diagnosing and treating PAD.

4.1. Multiplanar Visualization

4.1.1. Design and Implementation

The MPV module enables the reconstruction and visualization of 3D anatomical structures from 2D image slices, which are typically obtained from modalities such as computed tomography (CT) and magnetic resonance imaging (MRI) [28]. The implementation of the MPV module is centered around PWAs, ensuring seamless access and interaction across various devices and platforms. The MPV module is implemented via modern web technologies, including React [29] and Cornerstone.js [30], ensuring a responsive and user-friendly interface.

These services are accessible via a PWA, which ensures seamless functionality across various devices without requiring installation. A service worker is employed to manage background tasks, such as caching and retrieving data. This enhances the performance and reliability of the PWA, allowing for offline access and faster loading times [26]. Cached data are stored in Cache Storage, ensuring that frequently accessed information is readily available.

The module uses IndexDB for local storage, allowing the browser to store large amounts of structured data [31]. This is essential for managing the vast datasets involved in medical imaging. The data are retrieved via GET requests and processed in JSON format, ensuring compatibility and ease of manipulation. The frontend platform services interact with the visualization components to render multiplanar views of the DICOM images. Users can navigate through different planes (axial,

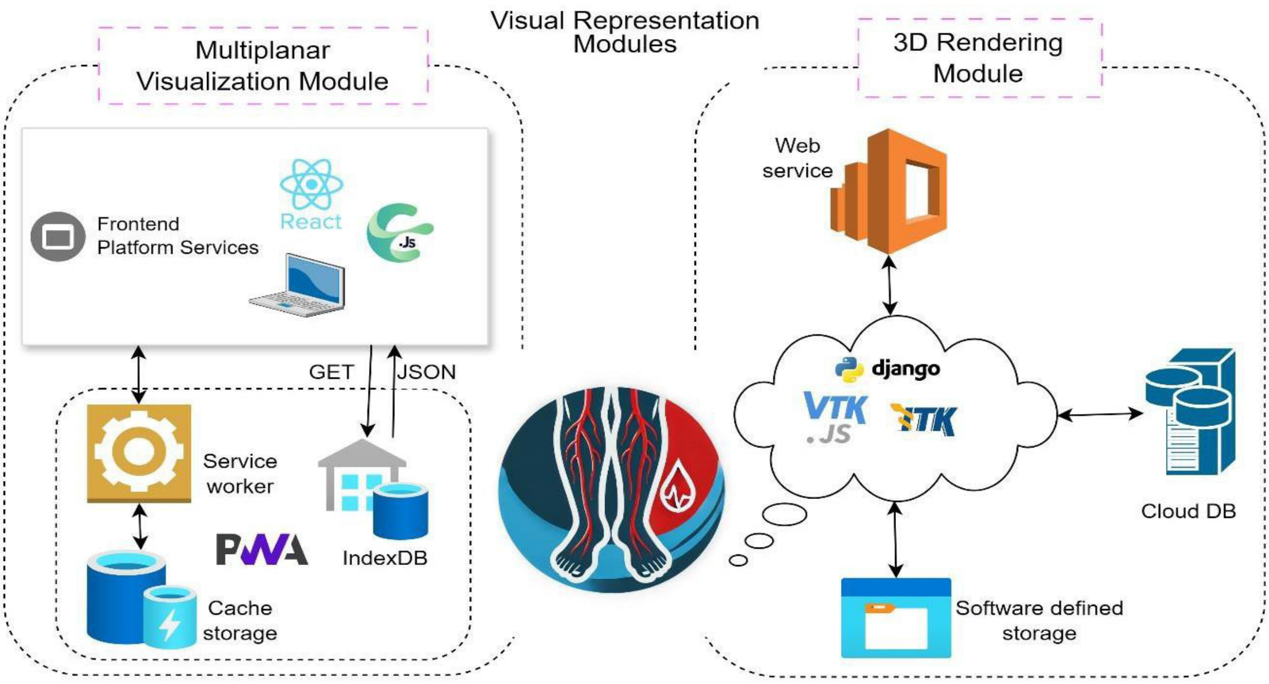


Figure 4: Implementation Architecture for Multiplanar Visualization and 3D Rendering Modules within the DECODE Cloud Platform.

sagittal, and coronal) to examine specific regions of interest. This functionality is crucial for accurate diagnosis and treatment planning.

The MPV module is built via a robust architectural framework designed to facilitate a smooth transition from data upload to advanced visualization. The key components of this framework include metadata and image information retrieval, DICOM tool activation, and DICOM image loading functionalities, which are crucial for efficient user experience and functionality.

Enhanced DICOM tools, including tools for magnification, angle measurement, and windowing, are integrated into the MPV module to complement the PWA paradigm. These tools leverage the capabilities of the Cornerstone library, which provides expanded functionalities and compatibility with various image sources. DICOM image loading is managed through the Cornerstone setup, incorporating robust success and failure handling mechanisms.

4.1.2. User Interaction and Visualization

The PWA approach enables seamless DICOM store connectivity, allowing users to search for and load specific studies for detailed analysis. Interactive viewers control support zooming, panning, and resetting, improving user engagement and interaction. The functionality of reference lines and planes is tailored to create spatial relationships within the MPR views, facilitating the alignment and comparison of DICOM images. This integration of reference lines involves constructing 3D lines and planes and coordinating transformations to convert 3D patient points to 2D pixel coordinates [32].

By leveraging these advanced technologies and methodologies, the MPV module within the DECODE platform offers a comprehensive and robust solution for the visualization and analysis of complex medical data.

4.1.3. 3D Rendering Module

The 3D rendering module of the DECODE cloud platform is designed to provide high-fidelity, interactive visualizations of complex medical imaging data. This module leverages advanced WebGL technology to transform 2D CT image slices into detailed 3D representations, enhancing the ability of clinicians to assess and plan treatments with precision [33].

4.1.4. Design and Implementation

The 3D rendering module is crucial for enhancing the diagnostic capabilities for PAD by providing a comprehensive view of the anatomical structures. The web service, represented by a cloud-based architecture, handles the computational demands of rendering and data processing. This setup ensures scalability and efficient resource utilization, which are critical for handling large medical datasets.

The backend services of the 3D rendering module are powered by the Django framework [34], which is known for its robustness in managing heavy computational tasks. Django coordinates the processing and storage of medical imaging data, ensuring seamless integration with the front-end components. This integration is essential for providing a responsive and interactive user experience.

The core rendering functionalities are implemented using vtk.js [35] and itk.js [36], JavaScript libraries specifically designed for medical image processing and visualization. These toolkits provide extensive capabilities for 3D graphics, enabling the creation of interactive and detailed visualizations. The use of these libraries ensures that the rendered images are accurate and high-quality, which is crucial for effective diagnosis and treatment planning.

The processed data are stored in a cloud database, ensuring secure and scalable data management. In addition, software-defined storage solutions are utilized to handle dynamic storage requirements, providing flexibility and efficiency. This approach ensures that the platform can manage large volumes of data efficiently without compromising performance or security.

The 3D rendering module follows a systematic workflow, beginning with the input of volumetric data composed of multiple CT slices. These data undergo a hardware estimation process to ensure the optimal allocation of computational resources, promoting efficient data handling and processing.

4.1.5. User Interface and Controls

The module's user interface is designed for seamless interaction, offering controls for panning, zooming, and rotating the 3D models. These controls are essential for exploring volumetric data from various angles and perspectives, ensuring that clinicians can thoroughly inspect anatomical structures. In addition, interactive widgets enable users to adjust visualization parameters such as color mapping and opacity, allowing customized views that highlight different tissues and structures as needed [37].

By integrating these advanced technologies and methodologies, the 3D rendering module within the DECODE cloud platform significantly enhances the visualization and analysis of peripheral artery CT images. This module supports clinicians in making informed decisions, ultimately improving patient outcomes and advancing the use of CT and angiography in the diagnosis and treatment of PAD.

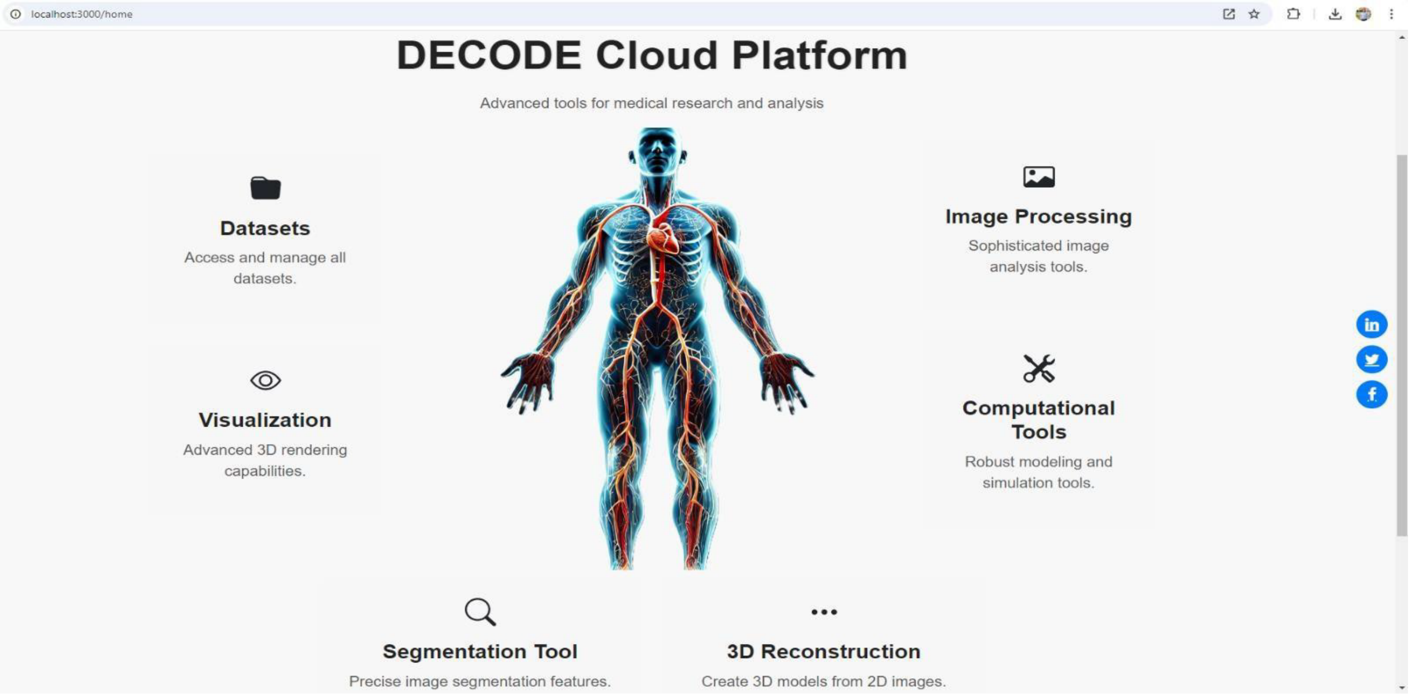


Figure 5: DECODE Cloud Platform Home Interface.

4.2. Evaluation Metrics

To assess the effectiveness and quality of the visual representation modules of the DECODE cloud platform, an evaluation was conducted via Google's Lighthouse tool [38]. Lighthouse is an open-source tool designed to evaluate the performance, accessibility, and best practices of web applications. These metrics were selected to ensure that the platform meets the critical demands of healthcare applications, which require efficiency, inclusivity, and adherence to industry standards.

The evaluation focused on two core modules of the DECODE platform: the MPV module and the 3D rendering module. Each module was tested to determine its performance in several key areas:

- Performance: This metric evaluates the speed and responsiveness of the platform, which is crucial for real-time medical imaging applications where delays can impact clinical decision-making. For example, high performance ensures seamless interaction with large datasets, enabling rapid diagnosis and analysis.
- Accessibility: This parameter evaluates whether the modules are usable by individuals with different types of disabilities, including visual, auditory, motor, or cognitive impairments. The evaluation ensures that the modules comply with international accessibility standards, such as the web content accessibility guidelines (WCAGs), which are designed to make web applications more accessible to all users.
- Best Practices: This metric assesses the platform's compliance with modern web development standards, including security and resource optimization. In healthcare, where sensitive patient data are involved, adherence to these best practices ensures robust and reliable performance.
- Search Engine Optimization (SEO): SEO is used to ensure that the platform's content is well structured and easily indexable, contributing to the user experience, which aids in the dissemination of research findings and platform awareness among broader medical communities.

These evaluation criteria were chosen to comprehensively assess the visual representation capabilities of the DECODE platform, ensuring that it provides a reliable, accessible, and high-performing tool for clinicians and researchers involved in the diagnosis and treatment of PAD.

5. Results

The DECODE cloud platform was developed and evaluated to enhance the diagnosis, analysis, and treatment planning for PAD through its sophisticated visualization and data management capabilities. The platform integrates multiple modules, such as datasets, image processing, computational tools, segmentation tools, 3D reconstructions, and visual representations, each designed to handle specific aspects of medical imaging and data processing, as detailed in Figure 5.

The MPV module, depicted in Figure 6, was instrumental in facilitating the detailed exploration of 2D DICOM images across multiple planes. This module's advanced capabilities enable clinicians to obtain comprehensive views of the anatomical structures of the PAD. The implementation of a PWA provided seamless access across various devices, significantly enhancing the platform's usability.

Figure 6: Multiplanar Visualization of Medical Imaging Data in the DECODE Cloud Platform

Figure 7: User Interface for 3D Rendering of Medical Imaging Data in the DECODE Cloud Platform.

The MPV module is accessible via the RESTful API endpoint*/api/visualization/mpv/*, which allows for efficient integration and data retrieval. This API endpoint supports various functionalities, including the collection of DICOM images, metadata retrieval, and real-time updates. The interactive tools for zooming, panning, and navigating through different planes (axial, sagittal, and coronal) are seamlessly integrated through this API, enabling the precise examination of specific regions of interest. These capabilities ensure that clinicians can interact with the visualization data dynamically, facilitating real-time adjustments and increasing diagnostic accuracy. The robust architecture of the MPV module, combined with its API-driven approach, ensures that the visualization tools are both scalable and adaptable to different clinical requirements. This integration not only improves the efficiency of data handling and processing but also ensures that the visualization remains consistent and reliable across various use cases and devices. Thus, the MPV module, with its advanced interface and RESTful API support, significantly enhances the diagnostic process and treatment planning for PAD.

The 3D rendering module, accessible via the RESTful API endpoint*/api/visualization/3d-rendering/*, is a critical component of the DECODE cloud platform. This endpoint facilitates efficient data retrieval and interaction, enabling real-time updates and dynamic adjustments. Figure 7 shows the user interface for this module, highlighting its robust functionality and interactive features that significantly enhance the analysis and treatment planning

Table 1: Performance Metrics for Multiplanar Visualization Module and 3D Rendering Module in the DECODE Cloud Platform obtained by Google's Lighthouse tool

Metrics (%)	Multiplanar Visualization Module	3D Rendering Module
Performance	94	89
Accessibility	90	90
Best Practices	100	100
Search Engine Optimization	100	90

for PAD. The user interface allows users to upload and render volumetric medical imaging data seamlessly. It includes a variety of controls for manipulating 3D visualization, such as gradient adjustment, scalar unit specification, resolution settings, and sample distance control. These controls enable users to tailor the rendering process to highlight specific anatomical features, facilitating a more precise examination of the imaging data.

Interactive features such as volume clipping, rotation, zooming, and panning are integrated into the interface, allowing users to explore the 3D models from various perspectives. The presence of checkboxes for enabling or disabling visibility, context visibility, handle visibility, and various handle controls (face, edge, corner) further enhances the flexibility and customization available to the user.

5.1. Evaluation of Visual Representation Modules

The performance metrics for the DECODE cloud platform's MPV module and 3D rendering module are summarized in Table 1. These metrics provide a quantitative assessment of the platform's capabilities across several key areas.

The MPV module demonstrated exceptional results, with a performance score 94%, indicating a highly responsive user experience. The module also scored 90% in accessibility, underscoring its commitment to inclusivity by meeting essential accessibility standards. In addition, the module achieved perfect scores (100%) in both best practices and SEO, reflecting its adherence to modern web development standards and its content's discoverability via search engines.

The 3D rendering module also performed strongly, with a performance score 89%. While slightly lower than the MPV module, this score still indicates a high level of efficiency, particularly given the complex computations required for 3D rendering. The module matched the MPV module with an accessibility score 90% and perfect scores 100% in both best practices and SEO, confirming its robust design and compliance with industry standards.

6. Discussion

The DECODE cloud platform represents a significant breakthrough in the diagnosis, analysis, and treatment planning for PAD. Leveraging cutting-edge cloud computing and advanced data visualization technologies, the DECODE platform surpasses the capabilities of current state-of-the-art (SoTA) solutions. It is the first platform specifically designed to integrate high-performance data processing, 3D medical imaging, and computational modeling into a unified system tailored for PAD. This integration allows for precise simulation of treatment interventions, such as drug-coated balloons, and provides clinicians with real-time, data-driven insights to optimize therapeutic strategies. Compared with other cloud-based platforms, such as teleophthalmology systems [39] and the BioData Catalyst (BDC) [40], DECODE has demonstrated significant advancements in terms of functionality and adaptability. While tele-ophthalmology systems excel in scalable image storage and annotation and BDC specializes in managing large-scale genomic datasets, DECODE uniquely integrates segmentation, 3D visualization, and real-time analysis to address PAD-specific clinical workflows. Furthermore, it mitigates common challenges observed in cloud-based solutions, such as inefficient handling of large datasets, access control, and vendor conformity, by employing tailored optimization techniques and ensuring robust security, scalability, and compliance with healthcare standards.

The DECODE platform overcomes the limitations of the SoTA by integrating comprehensive data management and advanced visualization tools. The MPV module's robust design and high performance underscore its effectiveness in medicine. The module's ability to provide detailed 2D and 3D views of anatomical structures significantly enhances diagnostic and treatment planning. The 3D rendering module's performance metrics also highlight its strengths in providing high-fidelity visualizations crucial for PAD treatment. The module's slightly lower performance score than the MPV module can be attributed to the complex computations required for rendering detailed 3D models. However, its accessibility score and perfect scores in best practices and SEOs indicate that it remains an invaluable tool for clinicians.

While the performance metrics highlight the platform’s robustness, several limitations must be acknowledged. One notable challenge is scalability, as the platform's current architecture may encounter bottlenecks when handling significantly larger datasets or accommodating simultaneous users over extended periods. Such scenarios could impact real-time processing and responsiveness, particularly in high-demand clinical settings. In addition, the reliance on high-performance GPUs and advanced cloud infrastructure poses a limitation, potentially restricting platform adoption in resource-constrained environments where access to such resources may be limited.

Future work will prioritize optimizing the platform architecture to increase scalability and explore cost-effective hardware alternatives, thereby ensuring broader accessibility and adoption in diverse healthcare settings. In addition, the planned development of a 3D segmentation and reconstruction module will focus on the precise delineation of anatomical structures, specifically targeting the segmentation of the peripheral artery with intima and adventitia thickness. This enhancement will enable detailed 3D reconstructions, improve the analysis of vascular health and support the personalization of treatment strategies. By accurately modeling these critical arterial layers, the module will provide clinicians with deeper insight into disease progression, plaque buildup, and potential treatment outcomes, ultimately contributing to more precise and effective clinical interventions. The integration of a virtual reality (VR) module will further enhance the platform by offering immersive visualization tools to improve spatial understanding of complex anatomical structures, supporting preoperative planning, educational applications, and patient engagement. Challenges such as ensuring hardware compatibility, optimizing real-time performance, and maintaining compliance with healthcare data standards will be actively addressed to maximize the impact of these innovations. These advancements position the DECODE platform as a versatile and comprehensive solution that not only surpasses existing SoTA solutions but also sets a new benchmark for cloud-based technologies in PAD diagnosis and management.

7. Conclusion

The DECODE cloud platform represents a transformative advancement in the diagnosis, analysis, and treatment of peripheral artery disease (PAD). By leveraging the capabilities of cloud computing and high-performance data visualization, the platform addresses the significant limitations of traditional diagnostic methods, offering a noninvasive, cost-effective, and scalable solution. The DECODE platform's comprehensive modules, including a data warehouse and multiplanar reconstruction with 2D and 3D rendering visualization, enable detailed and interactive processing and presentation of medical imaging data. The evaluation metrics underscore the platform's high performance and user accessibility. The multiplanar visualization module achieves a performance score 94, an accessibility score 90, and perfect scores 100 in best practices and search engine optimization (SEO). Similarly, the 3D rendering module scored 89 in performance, 90 in accessibility, and 100 in best practices and SEO, demonstrating the platform's robust design and compliance with industry standards. Our work highlights the transformative potential of integrating cloud computing in medical diagnostics, paving the way for further research and development in this critical area of healthcare.

Conflict of Interest

The authors declare that they have no conflicts of interest.

Acknowledgment

This work has received funding from the European Union’s Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No 956470, as part of the DECODE project.

References

[1] M. M. S. Radwan, Sini Thomas, Sithara, "Environmental Factors and Peripheral Artery Disease," in Environmental Factors in the Pathogenesis of Cardiovascular Diseases: Springer, 2024, 193-208.

[2] M. N. Søgaard, Peter Brønnum Eldrup, Nikolaj Behrendt, Christian-Alexander Nicolajsen, Chalotte W Lip, Gregory YH Skjøth, Flemming, European Journal of Vascular Endovascular Surgery, "Epidemiological trends and projections of incidence, prevalence, and disease related mortality associated with peripheral arterial disease: observations using nationwide Danish data," vol. 66, no. 5, 662-669, 2023, doi: 10.1016/j.ejvs.2023.08.005.

[3] S. P. S. Cartland, Christopher P Bursill, Christina Passam, Freda Figtree, Gemma A Patel, Sanjay Loa, Jacky Golledge, Jonathan Robinson, David A Aitken, Sarah, International Journal of Molecular Sciences, "Sex, Endothelial Cell Functions, and Peripheral Artery Disease," vol. 24, no. 24, 17439, 2023.

[4] S. Research, "Peripheral Artery Disease Market Growth, Trends and Forecast to 2031 " 2024, doi: 10.3390/ijms242417439.

[5] R. U. M. Center, "Peripheral Vascular Disease (PVD), https://www.rush.edu/conditions/peripheral-vascular- disease-pvd# " 2024

[6] J. D. Csore, Madeline Roy, Trisha L, Journal of Vascular Surgery Cases, Innovations Techniques, "Peripheral arterial disease treatment planning using noninvasive and invasive imaging methods," vol. 9, no. 4, 101263, 2023, doi: 10.1016/j.jvscit.2023.101263.

[7] W. R. Abbaoui, Sara El Bhiri, Brahim Kharmoum, Nassim Ziti, Soumia, Informatics in Medicine Unlocked, "Towards

revolutionizing precision healthcare: A systematic literature review of artificial intelligence methods in precision medicine," 101475, 2024, doi: 10.1016/j.imu.2024.101475.

[8] G. G. Gabrani, Sunil Vyas, Sonali Arya, Pradeep, "Revolutionizing Healthcare: Impact of Artificial Intelligence in Disease Diagnosis, Treatment, and Patient Care," in Handbook on Augmenting Telehealth Services: CRC Press, 2024, 17-31.

[9] K. K. Kavitha, C, "Cloud-Based Data Analytics for Healthcare 5.0," in Pioneering Smart Healthcare 5.0 with IoT, Federated Learning, and Cloud Security: IGI Global, 2024, 44-56, doi: 10.4018/979-8-3693-2639-8.ch004.

[10] A. A. Shalan, Abubakar Habli, Ibrahim Tew, Garry Thompson, Andrew, "YORwalK: desiging a smartphone exercise application for people with intermittent claudication," in Building Continents of Knowledge in Oceans of Data: The Future of Co-Created eHealth: IOS Press, 2018, 311-315, doi: 10.3233/978-1-61499-852-5-311.

[11] K. S. Paldán, Jan Ullrich, Greta Steinmetz, Martin Rammos, Christos Jánosi, Rolf Alexander Moebus, Susanne Rassaf, Tienush Lortz, Julia, JMIR research protocols, "Feasibility and clinical relevance of a mobile intervention using TrackPAD to support supervised exercise therapy in patients with peripheral arterial disease: study protocol for a randomized controlled pilot trial," vol. 8, no. 6, e13651, 2019.

[12] A. V. Harzand, Alexander A Alrohaibani, Alaaeddin Abdelhamid, Smah M Gordon, Neil F Thiel, John Benarroch‐Gampel, Jaime Teodorescu, Victoria J Minton, Keri Wenger, Nanette K, Clinical Cardiology, "Rationale and design of a smartphone‐enabled, home‐based exercise program in patients with symptomatic peripheral arterial disease: the smart step randomized trial," vol. 43, no. 6, 37-545, 2020, doi: 10.1002/clc.23362.

[13] J. S. Lortz, Jan Kuether, Tabea Kreitschmann-Andermahr, Ilonka Ullrich, Greta Steinmetz, Martin Rammos, Christos Jánosi, Rolf Alexander Moebus, Susanne Rassaf, Tienush, JMIR Formative Research, "Needs and requirements in the designing of mobile interventions for patients with peripheral arterial disease: questionnaire study," vol. 4, no. 8, e15669, 2020.

[14] H. P. Paredes, Dennis Barroso, João Abrantes, Catarina Machado, Isabel Silva, Ivone, "Supervised physical exercise therapy of peripheral artery disease patients: M-health challenges and opportunities," 2021, http://hdl.handle.net/10125/71086.

[15] A. M. D. Flores, Falen Leeper, Nicholas J Ross, Elsie Gyang, Circulation research, "Leveraging machine learning and artificial intelligence to improve peripheral artery disease detection, treatment, and outcomes," vol. 128, no. 12, 1833-1850, 2021, doi: 10.1161/CIRCRESAHA.121.31822.

[16] N. M. Forghani, Keivan Dabanloo, Nader Jafarnia Farahani, Ali Vasheghani Forouzanfar, Mohamad, IEEE Journal of Biomedical Health Informatics, "Intelligent oscillometric system for automatic detection of peripheral arterial disease," vol. 25, no. 8, 3209-3218, 2021, doi: 10.1109/JBHI.2021.3065379.

[17] T. G. Collins, Mugur Overton, Kathryn Benton, Mary Lu, Liuqiang Khan, Faarina Rohleder, Mason Ahluwalia, Jasjit Resnicow, Ken Zhu, Yiliang, JMIR Formative Research, "Use of a smartphone app versus motivational interviewing to increase walking distance and weight loss in overweight/obese adults with peripheral artery disease: pilot randomized trial," vol. 6, no. 2, e30295, 2022.

[18] M. K. Kim, Yesol Choi, Mona, BMC Medical Informatics Decision Making, "Mobile health platform based on user-centered design to promote exercise for patients with peripheral artery disease," vol. 22, no. 1, 206, 2022, doi: 10.1186/s12911-022-01945-z.

[19] F.-Q. D. Wu, Qian-Wan Wang, Ji-Guang Li, Wen-Zhu, Current Treatment Options in Cardiovascular Medicine, "The Role of Supervised Exercise Therapy in the Management of Symptomatic Peripheral Artery Disease with Intermittent Claudication," vol. 25, no. 10, 501-513, 2023, doi: 10.1007/s11936-023-01001-7.

[20] J. I. Zaki, SM Riazul Alghamdi, Norah Saleh Abdullah-Al-Wadud, Mohammad Kwak, Kyung-Sup, IEEE Access, "Introducing cloud-assisted micro-service-based software development framework for healthcare systems," vol. 10, 33332-33348, 2022, doi: 10.1109/ACCESS.2022.3161455.

[21] D. O. Alekseeva, Aleksandr Arponen, Otso Lohan, Elena Simona, Computer Science Review, "The future of computing paradigms for medical and emergency applications," vol. 45, p. 100494, 2022, doi: 10.1016/j.cosrev.2022.100494.

[22] M.-A. B. Filz, Jan Philipp Herrmann, Christoph, Journal of Intelligent Manufacturing, "Digitalization platform for data-driven quality management in multi-stage manufacturing systems," vol. 35, no. 6, 2699-2718, 2024, doi: 10.1007/s10845-023-02162-9.

[23] AboArab, Mohammed A., Vassiliki T. Potsika, Nikola Petrović, and Dimitrios I. Fotiadis, "DECODE cloud platform: A new cloud platform to combat the burden of peripheral artery disease," in 2022 Panhellenic Conference on Electronics & Telecommunications (PACET), 2022, 1-6: IEEE, doi: 10.1109/PACET56979.2022.9976356.

[24] Z. A. K. Mughal, Adnan Ahmed, Syed Sohail Qazi, Salman, Journal of Software Engineering, "Key factors and features analysis of popular SaaS ERP Systems for Adoptability," vol. 1, no. 1, 11-21, 2022.

[25] D. C. L. De Oliveira, Ji Pacitti, Esther, Data-Intensive Workflow Management. Springer Nature, 2022.

[26] R. K. Fauzan, Ice Nurwibowo, Bima Dinda Wibowo, Della Aulia, IPTEK The Journal for Technology Science, "A systematic literature review on progressive web application practice and challenges," vol. 33, no. 1, 43-58, 2022, doi: 10.12962/j20882033.v33i1.13904.

[27] D. V. Hesmondhalgh, Raquel Campos Kaye, D Bondy Valdovinos Li, Zhongwei, Media Communication, "Digital platforms and infrastructure in the realm of culture," vol. 11, no. 2, 296-306, 2023, doi: 10.17645/mac.v11i2.6422.

[28] A. G. Blum, R Rauch, A Urbaneja, A Biouichi, H Dodin, G Germain, E Lombard, C Jaquet, P Louis, M, Diagnostic interventional imaging, "3D reconstructions, 4D imaging and postprocessing with CT in musculoskeletal disorders: past, present and future," vol. 101, no. 11, 693-705, 2020, doi: 10.1016/j.diii.2020.09.008.

[29] M. T. Thakkar, Mohit, Building React Apps with Server-Side Rendering: Use React, Redux, Next to Build Full Server-Side Rendering Applications, "Introducing react. js," 41-91, 2020.

[30] E. U. Ziegler, Trinity Brown, Danny Petts, James Pieper, Steve D Lewis, Rob Hafey, Chris Harris, Gordon J, JCO clinical cancer informatics, "Open health imaging foundation viewer: an extensible open-source framework for building web-based imaging applications to support cancer research," vol. 4, 336-345, 2020, doi: 10.1200/CCI.19.00131.

[31] F. V. Paligu, Cihan, Future Internet, "Browser Forensic Investigations of Instagram Utilizing IndexedDB Persistent Storage," vol. 14, no. 6, p. 188, 2022, doi: 10.3390/fi14060188.

[32] J. K. Udupa, "3D imaging: principles and approaches," in 3D Imaging in Medicine, Second Edition: CRC Press, 2023, 1-73.

[33] R. L. T. Cieri, Morgan L Carney, Ryan M Falkingham, Peter L Kirk, Alexander M Wang, Tobias Jensen, Bjarke Novotny, Johannes Tveite, Joshua Gatesy, Stephen M, Journal of morphology, "Virtual and augmented reality: New tools for visualizing, analyzing, and communicating complex morphology," vol. 282, no. 12, 1785-1800, 2021, doi: 10.1002/jmor.21421.

[34] P. J. Thakur, Prashant, "Django: Developing web using Python," in 2023 3rd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), 2023, pp. 303-306: IEEE, doi: 10.1109/ICACITE57410.2023.10183246.

[35] Kitware, Inc. (2024). VTK.js. Accessed July 2024. https://kitware.github.io/vtk-js, doi: 10.5281/zenodo.4552412, 2021.

[36] S. J. Matt McCormick, Forrest Li, Alexis Girault, David Thompson, Juan Carlos Prieto, Scott Wittenburg, & HastingsGreer. (2021). InsightSoftwareConsortium/itk-js: v14.1.1 (v14.1.1), doi: 10.5281/zenodo.4957207.

[37] R. JohnsonChris, Health Data Science, "A review of three-dimensional medical image visualization," 2022, doi: 10.34133/2022/9840519.

[38] M. P. Manca, Vanessa Paternò, Fabio Santoro, Carmen, ACM Transactions on Accessible Computing, "The transparency of automatic web accessibility evaluation tools: design criteria, state of the art, and user perception," vol. 16, no. 1, 1-36, 2023, doi: 10.1145/3556979.

[39] Schweitzer, M., Ostheimer, P., Lins, A., Romano, V., Steger, B., Baumgarten, D., & Augustin, M. (2024). Transforming Tele-Ophthalmology: Utilizing Cloud Computing for Remote Eye Care. In dHealth 2024 (pp. 215-220). IOS Press, doi: 10.3233/SHTI240040.

[40] Ahalt, S., Avillach, P., Boyles, R., Bradford, K., Cox, S., Davis-Dusenbery, B., ... & Asare, J. (2023). Building a collaborative cloud platform to accelerate heart, lung, blood, and sleep research. Journal of the American Medical Informatics Association, 30(7), 1293-1300, doi: 10.1093/jamia/ocad048.