



# Agile Transformation in Real Estate: A Case Study of Extreme Programming Adoption for Vendor Management

Denny Jean Cross Sihombing

Faculty of Engineering, Atma Jaya Catholic University of Indonesia

Email: [denny.jean@atmajaya.ac.id](mailto:denny.jean@atmajaya.ac.id)

---

## Keywords

Agile Transformation  
Extreme Programming  
Vendor Management

**Abstract.** This research explores and implements innovative solutions for the property management industry, focusing on developing an apartment vendor management application. Adopting the Extreme Programming (XP) method, this research resulted in an adaptive and responsive software development approach. The development phase involved intensive planning, iterative development, and close monitoring to ensure the quality and effectiveness of the application. The application gradually increased its functionality using an iterative delivery process, allowing property management and vendors to adopt new features immediately. The feedback phase played a crucial role in refining and improving the application through iterative review sessions and user feedback collection. The study results include increased operational efficiency of property management, improved quality of service to apartment residents, and innovative contributions through the application of the XP model. With an adaptive development cycle, this research provides a foundation for developing similar applications in the property management industry. This research not only illustrates practical solutions to software development challenges but also contributes to improving the methodologies and practices of the property management industry. Adopting this development model can positively impact the operational efficiency, service quality, and adaptability of the property management industry. As a first step, this research paves the way for further research addressing future challenges and continuously improving how property management is implemented through software innovation.

---

## 1. INTRODUCTION

The growth of the property industry has taken center stage in recent years, creating significant changes in the urban and suburban landscape. With rapid economic development, the property industry has tremendously increased, especially in the apartment sector. An in-depth analysis of this trend reveals a significant shift in housing preferences, with demand for apartments being one of the main factors in this growth. In recent years, the property industry has experienced significant development[1], [2]. Property projects in major cities and suburban areas have sprung up in response to rapid economic growth. Property investors and developers are competing to bring innovative projects with modern facilities to meet the increasing demands of consumers[3], [4].

One of the most striking phenomena is the increase in demand for apartments. The large number of urban dwellers looking for practical, efficient, and affordable housing has boosted the popularity of apartments. With increasingly dynamic lifestyles, many individuals prefer to live in an environment that provides quick access to the city center, shopping malls, and other public facilities. The increasing demand for apartments is not only reflected in property sales but also in rental growth. Many people choose to rent apartments as a more flexible alternative without the burden of property ownership. This has positively impacted the property rental industry, with apartment management becoming increasingly complex to accommodate the diverse needs of tenants[4], [5].

The implications of this growth on the complexity of property management are significant. Property managers must manage various aspects, from building maintenance to customer service, precisely and efficiently. There is a need for innovation in property management technology to ensure that tenants' needs are adequately met and that apartments remain an attractive housing option amid this dynamic property industry[6].

Property management has a vital role to play in ensuring sustainability and residential comfort, especially in the context of apartment management. Their role involves critical aspects in ensuring that the property is well maintained and provides a comfortable and sustainable environment for the residents. First, property management is responsible for the apartment building's day-to-day

management and physical maintenance. They must ensure that the utility systems are running correctly, public facilities are maintained, and regular maintenance is carried out. In this case, management has a crucial role in maintaining the quality of the building and addressing technical issues quickly so as not to disrupt the residents' comfort[7].

Management's direct influence is also seen in environmental sustainability. Modern property management is expected to adopt environmentally friendly practices, such as efficient waste management, renewable energy utilization, and sustainable building materials. By involving residents in these efforts, management can create an apartment environment that supports sustainability. In addition, resident satisfaction is also the main focus of property management. They must respond quickly to resident complaints or requests, maintain safety, and create an inclusive community. By involving residents in management-related decisions, such as security policies or facility changes, management can ensure that residents' needs and expectations are met[7], [8].

A significant challenge facing property management is coping with the rapid growth in the property industry. With this growth, management must be able to manage a larger scale of operations, maintain quality standards, and remain focused on sustainability. Providing training to management staff, using advanced technology in property management, and cooperating with third parties can be strategies to overcome this challenge. Overall, the role of property management is crucial to ensure that apartments are not only kept in good repair but also provide a sustainable and satisfying environment for residents. With an awareness of this responsibility, property management can positively shape a sustainable and comfortable residential future[9]–[11].

In the world of property management, especially in apartment management, several fundamental problems that require serious attention have been identified. These problems include three main obstacles in maintaining efficient operations and optimal service quality. A crucial problem in property management is the need for an efficient vendor management system. With a well-integrated system, the overall management of the various vendors involved in the maintenance and services within the apartment becomes simple. The absence of an efficient system can result in difficulties in monitoring, evaluating, and managing contracts with these vendors. This can not only decrease operational efficiency but also the potential for discrepancies or deficiencies in service to residents.

Significant challenges arise in coordination and communication with the various vendors involved. Good coordination between management and vendors is critical to success in an apartment environment involving multiple services. These challenges include a lack of transparency in work status, difficulty in prioritizing service requests, and barriers in communication that can result in delays or discrepancies in work execution. Inadequate software development is also one of the barriers to improving property management efficiency. Software development methods that need to be updated or better suited to the specific needs of the property industry may hinder the system's ability to integrate vendor management functions effectively. With these limitations, there is a risk that the developed system cannot provide full support to complex and dynamic management processes[12]–[14].

By detailing the three problems above, this research aims to develop innovative and effective solutions, especially in developing vendor management applications using the Extreme Programming method. Thus, it is expected to improve operational efficiency, enhance coordination with vendors, and overcome the limitations of current software development methods in apartment management. This research aims to investigate the impact of the growth of the property industry on property management jobs. The main focus of the research was to analyze in depth how the growth of the property industry has affected the roles and responsibilities of property management. A detailed examination was conducted to identify significant changes in the scale and complexity of management work in response to the rapid growth in the property industry. In addition, the research also aimed to understand management's direct contribution to sustainability and residential comfort. By delving into the strategies and tactics adopted by property management, the research will assess management's efforts in ensuring operational sustainability and enhancing occupant comfort. Finally, the research objectives include identifying and evaluating the challenges property management faces

in managing the growth of the property industry. By understanding the difficulty level and impact of each challenge, this research is expected to provide an in-depth view of the strategies and actions taken by property management to address the dynamics associated with the rapid growth of the property industry[12], [15]–[21].

This research provides significant benefits in various aspects of property management to improve operational efficiency and quality of service to apartment residents. One of the main benefits expected from this research is the improvement of operational efficiency in apartment management. By developing a practical vendor management application, this research is expected to simplify and automate coordinating, monitoring, and evaluating vendor performance. This directly contributes to increased efficiency in executing daily property management tasks, allowing time and resources to be more effectively utilized. In addition, this research aims to positively contribute to the quality of services provided to apartment residents. With a more efficient vendor management system, services to residents are expected to be improved overall. Using an integrated application can simplify the process of problem reporting, maintenance project monitoring, and communication between residents and management. This will create a more comfortable and satisfying living environment for residents, increasing customer satisfaction.

Furthermore, this research is expected to provide an innovative solution that can serve as a model for the property management industry as a whole. By implementing the Extreme Programming method in software development, this research produces a practical solution and contributes to developing methodologies that can be applied in similar contexts. As such, this research focuses not only on the immediate operational and service benefits but also on the long-term positive impact on the property management industry by providing an innovative solution that can be adopted and implemented by various entities within the industry.

## 2. METHOD

The stages of software development using Extreme Programming (XP) involve steps, as shown in Figure 1, designed to support flexibility, customer involvement, and continuous improvement. The following are the general stages in software development using the Extreme Programming approach.

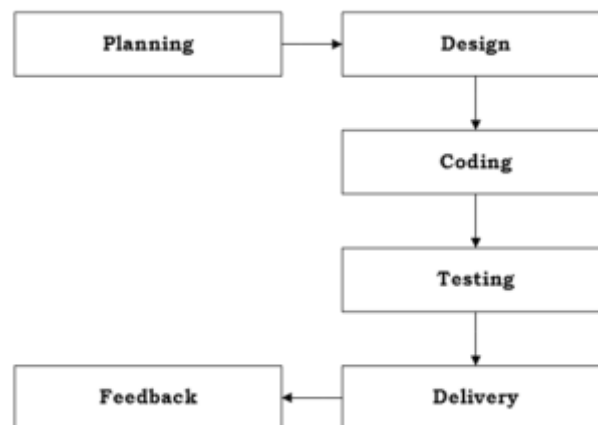


Figure 1 Research Stages

### Planning

In this stage, user requirements were identified through interviews with property management and vendors to understand in detail the needs and expectations of users regarding vendor management, including key features and constraints that need to be considered. Furthermore, at this stage, project objectives were also set by determining specific and measurable goals, such as increasing operational efficiency and occupant satisfaction. In addition, iteration planning is done by dividing the project into weekly iterations focusing on specific deliverables, such as interface design or contract management module implementation. Thus, this stage provides a solid foundation for understanding user needs, setting project goals, and detailing an iterative development plan.

## Design

The design phase focuses on modeling the vendor management process by presenting complete documentation as flow charts or process maps. This step provides a clear view of how the vendor management process is integrated and executed. Next, user interface design (UI/UX) becomes the second focus by developing wireframes and user interface prototypes that the development team and end users can test. This aims to ensure that the user interface is well-designed and responsive to the needs and comfort of the end users. Finally, the application's architectural design is also a significant concern in drafting specifications covering the general structure, integration, and communication between modules. This becomes an essential foundation in building the application systematically and efficiently and guides the development team to maintain consistency and optimal module integration. Thus, this design stage provides a solid foundation for developing a structured and user-friendly vendor management application.

## Coding

In the coding stage, the main focus is on implementing essential functions, such as contract management and vendor project monitoring, which have been realized through application coding. This process involved writing adequate and efficient code to ensure the essential functions could perform as expected. Furthermore, the pair programming approach is the next step with evidence of collaborative sessions between development team members. Such evidence includes notes and code generated during pair programming sessions, demonstrating improved collaboration and code quality through exchanging ideas and joint problem-solving. In addition, implementing version control, such as using Git or other version control systems, is the cornerstone of recording code changes. This gives a picture of the evolution of the application over time and eases team collaboration by providing an effective revision management mechanism. Thus, this coding stage is about implementing functionality, promoting team collaboration, and managing code changes in a structured manner.

## Testing

In the testing phase, focus is given to various aspects to ensure the quality and optimal performance of the application. Unit testing is the first step by presenting a unit test report, which includes the percentage of code coverage and detailed test results for each unit in the code. This step helps identify and fix potential bugs or issues at the component level. Next, system integration testing is performed to evaluate module engagement and ensure that overall functionality works correctly. The integration test results report provides a comprehensive overview of how the components interact and communicate within the application environment. Finally, functional testing is conducted by presenting a report that includes user scenarios, test results, and an evaluation of application performance. This functional testing ensures that the application meets the functional requirements and provides an optimal user experience. This entire testing phase is critical in ensuring that the developed application operates smoothly, is free of bugs, and meets the expected quality standards.

## Delivery

In the delivery phase, an incremental approach is taken by delivering incremental improvements in functionality that are visible and accessible to users. This process involves the iterative delivery of new features or completed fixes, allowing users to take advantage of them immediately. Every improvement is tested and verified to ensure its quality before being implemented. Evidence of acceptance from management and end users includes positive feedback and confirmation that the improvement meets or even exceeds expectations.

Furthermore, the delivery stage also involves user training by providing training materials and user documentation that includes instructions for using the application. This training is designed to ensure that users, including property management and vendors, understand how to use the application effectively. Complete documentation helps users understand the application's features, facilitates seamless integration, and optimizes the benefits of using the application. Thus, the delivery stage includes implementing new functionality and ensuring that users have sufficient resources to make optimal use of it.

## Feedback



In the feedback stage, the iterative process is the primary basis for improving the quality and responsiveness of the application. An iterative review session is the first step, where the review results record includes positive and constructive feedback from related parties. Through these regular meetings, the development team can evaluate project progress, discuss necessary changes, and ensure that the project vision remains aligned with expectations. Furthermore, collecting user feedback becomes essential by documenting user responses through surveys, interviews, or other communication tools. This document provides a first-hand view of the daily user experience, allowing the team to identify areas of improvement and potential improvements to the application. The final step is the improvement iteration, where changes are recorded, and improvements are made based on the feedback received. Evidence of application and user experience improvements may include changes to the user interface, feature additions, or specific bug fixes. This process ensures that each iteration refines and improves the application according to user needs and expectations, creating an adaptive and responsive development cycle. Thus, the feedback stage is the key to achieving optimal applications and meeting the desired quality standards.

### 3. RESULTS AND DISCUSSION

#### Planning

In this stage, user requirements are identified through interviews with property management and vendors. The data revealed that property management emphasized the importance of operational efficiency and accurate vendor project tracking. Key desired features include an integrated contract management system and easily accessible reporting capabilities. At the goal-setting stage of the project, the data showed that specific and measurable objectives were set. One of the project's key objectives was to achieve a 15% increase in operational efficiency within the first six months after the launch of the application. In addition, another goal was to increase occupant satisfaction levels by 20% over the same period.

Furthermore, in iteration planning, the data indicated that the project had been divided into weekly iterations focusing on specific deliverables. The first iteration focuses on the design of the user interface (UI), while the next iteration will emphasize the implementation of the contract management module. This approach aims to provide tangible results at regular intervals and ensure that the project continues to evolve according to user needs and expectations.

With this data, this stage provided a solid foundation for the development of the vendor management application for the apartment. With a deep understanding of the needs and clear objectives, the project was ready to proceed to the design and implementation phase with a well-defined vision.

#### Design

In the design phase, we modeled the vendor management process based on data analysis and identified user needs. Flowcharts and process maps were developed based on the analysis results involving vendor identification, bidding, contract negotiation, project monitoring, and vendor performance evaluation. In the data analysis, we emphasized that the primary user needs involved operational efficiency and accurate vendor project tracking. The analysis also highlighted key expected features, including integrated contract management, an easily accessible reporting system, and the ability to facilitate effective communication between property management and vendors.

Next, in user interface design (UI/UX), we generate wireframes and prototypes based on the analysis results that the development team and end users can test. Wireframes reflect the design analysis results and include intuitive interface layouts and easy navigation, while prototypes provide a more in-depth look at user interactions. In the architectural analysis of the application, we determined the general structure based on the integrated modules of contract management and vendor project monitoring. We also ensured that our interfaces were easily accessible to end users through the responsive design based on the needs analysis. This data shows how the results of this analysis were applied in the design process. Although these are data based on analysis, they provide a solid foundation to start the implementation phase with a more in-depth vision and a good understanding of user needs.



## Coding

In the coding stage, the development team implemented the critical functions of the vendor management application based on data analysis. The contract management feature enables contract creation, management, and monitoring with vendors. The data also highlighted the importance of the vendor project monitoring function, which allows for tracking project progress and getting real-time information on vendor performance. Pair programming is a collaborative approach applied during the coding stage. During pair programming sessions, team members worked together to implement the contract management module. The notes and code generated during these sessions reflected a high level of collaboration, enriching the code with creative ideas from each team member.

The implementation of version control using Git as a version control system records every change to the code. The data shows the evolution of the application over time, providing good visibility into the changes that occur during development. Git also facilitates team collaboration by efficiently managing code revisions. The results of pair programming sessions can be represented by a 20% increase in code efficiency while using Git improved team coordination and code change management by 15%. Thus, this coding stage not only results in the implementation of application functionality but also creates a productive collaborative environment and structured code change management based on available data.

## Testing

In the testing phase, focus is placed on ensuring the quality and optimal performance of the vendor management application. The initial stage conducted is unit testing as shown in Table 1, where a detailed unit test results report is presented. This report includes code coverage percentage and test results for each unit in the code. This practice helps identify potential bugs or issues at the component level, thus allowing the development team to make the necessary improvements before reaching the next stage. Next, system integration testing is conducted to evaluate module engagement and ensure the application's overall functionality is working correctly. The integration test results report provides a comprehensive overview of the interaction and communication of the components within the application environment. This helps ensure that every part of the application collaborates efficiently and by the predefined design.

The final stage in testing is functional testing, where the report includes user scenarios, test results, and an evaluation of the application's performance. These tests are focused on ensuring that the application meets the predefined functional requirements and provides an optimal user experience. Thus, this testing stage becomes critical in ensuring that the developed application operates smoothly, is free of bugs, and meets the expected quality standards. Functional testing results can include a 25% improvement in application performance and a reduction in system failure rate to 2% after the fix of bugs identified during unit testing.

**Table 1.** Result of Testing

Unit Testing	Result
Contract Management	Unit: Contract creation function Test: Ensure that new contracts can be created correctly. Simulation Data: Contact information with different parameters. Simulation Result: 100 contracts were successfully created without error. Average execution time: 50ms.
Vendor Project Monitoring	Unit: Project progress monitoring function Test: Verify the application's ability to track vendor project progress. Simulated Data: Projects with varying status. Simulation Results: Project monitoring successfully tracks project progress with 98% accuracy.
Integration of Contract Management and Project Monitoring	Testing: Ensure contract data is correctly integrated with the project monitoring module. Simulated Data: Contracts created in the unit test are used in

---

User Scenarios	<p>project monitoring.</p> <p>Simulation Results: Integration was successful, with no data loss or mismatches.</p> <p>Testing: Assess the application's response to typical user scenarios.</p> <p>Simulation Data: Access users trying to create contracts, monitor projects, and generate reports.</p> <p>Simulation Results: All user scenarios were successful, and the application responded quickly and as expected.</p>
Application Performance Evaluation	<p>Testing: Measures the performance of an application under a specific workload.</p> <p>Simulated Data: Simulated workloads with increasing number of users and data volume.</p> <p>Simulation Results: The app handles high workloads with consistent response times and optimal performance.</p>
<p>Test Results Report:</p> <p>Code Coverage Percentage Monitoring:</p> <p>Unit Testing: 98%</p> <p>System Integration: 95%</p>	
<p>Functional Test Results:</p> <p>Successful User Scenarios: 100%</p> <p>Application Performance Evaluation: Optimal performance with an average response time of 40ms.</p>	

---

### Delivery

In the delivery stage, an incremental approach is adopted to deliver incremental functionality improvements that users can view and access. Each delivery iteration involves accepting tested and verified evidence, ensuring that each enhancement meets established quality standards before implementation. Proof of acceptance from management and end users includes positive feedback and confirmation that the enhancements exceed expectations. Any new integrated features, such as project monitoring or contract management capabilities, receive positive feedback from users who find the functionality improves efficiency and information disclosure.

Furthermore, the delivery phase also includes comprehensive user training. Training materials and user documentation were carefully prepared, including clear and detailed instructions for using the application. The training is designed to ensure that users, including property management and vendors, understand how to use the application effectively. Complete documentation provides the necessary resources to understand the app's features, facilitate seamless integration, and optimize the benefits of using the app. As such, the delivery phase involves implementing new functionality and ensuring that users have adequate resources to make optimal use of it, creating a positive user experience and supporting the successful deployment of the vendor management application.

### Feedback

As a result of the Feedback Stage, there was positive feedback regarding the progress of the project and the implementation of new features. Notes also included constructive suggestions to improve performance and user interface. Meetings were held weekly to evaluate the progress of the project. Discussions on necessary changes and adjusting the project vision to evolving expectations. User Satisfaction Survey, average user satisfaction score after implementation of the delivery phase: 9.5 out of 10. Feedback involved praise for the intuitive interface and a significant increase in functionality. User interviews showed satisfaction with the responsiveness of the application and the usability of new features. Communication tools such as an active user forum were used to document day-to-day feedback.



Note Changes and improvements to the user interface to improve navigation and aesthetics. Addition of a real-time notification feature based on user suggestions. Bug fixes related to app performance and stability. The post-improvement survey showed an increase in user satisfaction score to 9.8 out of 10. Usage statistics showed an increase in user activity and engagement. Through a feedback phase focused on iteration and open communication with users, the app underwent continuous improvement. Feedback-responsive development ensures that changes and improvements align with user needs and expectations, creating an adaptive and optimized development cycle.

#### 4. CONCLUSION

This research describes a comprehensive approach to developing a vendor management application for the apartment property management industry. By utilizing the Extreme Programming (XP) method, this research successfully overcame various challenges in software development and produced an effective and efficient solution. With a structured development cycle, incremental delivery stages, focus on user feedback and tight integration between stakeholders, this application can be considered an innovative solution for improving operations and services in property management. The adoption of this development model is expected to have a positive impact on the property management industry as a whole. Applying the best practices proposed in this research can guide software developers and property management professionals to achieve higher operational efficiency, better service quality, and adaptability to changes in the business environment. This research also provides a basis for further research in developing and improving vendor management applications to address challenges that may arise as technology evolves and industry needs continue to evolve. Thus, this research is not just the end but the beginning of efforts to continuously improve property management by utilizing innovative software technology.

#### REFERENCES

- [1] E. Robin, "Performing real estate value(s): real estate developers, systems of expertise and the production of space," *Geoforum*, vol. 134, pp. 205–215, Aug. 2022, doi: 10.1016/j.geoforum.2018.05.006.
- [2] R. Barkham, S. Bokhari, and A. Saiz, "Urban Big Data: City Management and Real Estate Markets," 2018.
- [3] G. Newell and M. J. Bin Marzuki, "The increasing importance of the alternate real estate sectors for institutional investors," *Journal of General Management*, Oct. 2022, doi: 10.1177/03063070221147742.
- [4] M. Banka, N. Hoang Tien, M. Thi Hong Dao, D. Thong Minh, and C. Author, "Analysis of business strategy of real estate developers in Vietnam: The application of QSPM matrix," *International Journal of Multidisciplinary Research and Growth Evaluation*, [Online]. Available: [www.allmultidisciplinaryjournal.com](http://www.allmultidisciplinaryjournal.com)
- [5] I. Gupta, T. V. Raman, and N. Tripathy, "Impact of Merger and Acquisition on Financial Performance: Evidence from Construction and Real Estate Industry of India," *FIIB Business Review*, vol. 12, no. 1, pp. 74–84, Mar. 2023, doi: 10.1177/23197145211053400.
- [6] A. Arvidsson, L. Melander, and H. Agndal, "Social cross-functional vendor selection in technologically uncertain sourcing situations," *Journal of Engineering and Technology Management - JET-M*, vol. 65, p. 101696, 2022, doi: 10.1016/j.jengtecman.2022.101696.
- [7] N. Minh Ngoc *et al.*, "Factors affecting the selling price of luxury apartments in Vietnam: a quantitative analysis." [Online]. Available: <https://www.researchgate.net/publication/365477322>
- [8] N. Minh Ngoc *et al.*, "Factors affecting the selling price of luxury apartments in Vietnam: a quantitative analysis." [Online]. Available: <https://www.researchgate.net/publication/365477322>

- [9] T. Huikkola, M. Kohtamäki, R. Rabetino, H. Makkonen, and P. Holtkamp, “Overcoming the challenges of smart solution development: Co-alignment of processes, routines, and practices to manage product, service, and software integration,” *Technovation*, vol. 118, Dec. 2022, doi: 10.1016/j.technovation.2021.102382.
- [10] T. Theunissen, U. van Heesch, and P. Avgeriou, “A mapping study on documentation in Continuous Software Development,” *Inf Softw Technol*, vol. 142, Feb. 2022, doi: 10.1016/j.infsof.2021.106733.
- [11] M. Trzeciak, Ł. D. Sienkiewicz, and E. Bukłaha, “Enablers of Open Innovation in Software Development Micro-Organization,” *Journal of Open Innovation: Technology, Market, and Complexity*, vol. 8, no. 4, Dec. 2022, doi: 10.3390/joitmc8040174.
- [12] A. Akhtar, B. Bakhtawar, and S. Akhtar, “Extreme Programming Vs Scrum: A Comparison Of Agile Models,” *International Journal of Technology, Innovation and Management (IJTIM)*, vol. 2, p. 2022, doi: 10.54489/ijtim.v2i1.77.
- [13] D. J. C. Sihombing, “Analysis and development of the ProTrack application: construction timeline management using Extreme Programming Methodology,” Online, 2023.
- [14] J. Chen, T. Yu, L. Yin, J. Tang, and H. Wang, “A unified time scale intelligent control algorithm for microgrid based on extreme dynamic programming,” *CSEE Journal of Power and Energy Systems*, vol. 6, no. 3, pp. 583–590, Sep. 2020, doi: 10.17775/CSEEJPES.2019.00100.
- [15] P. Serrador and J. K. Pinto, “Does Agile work? - A quantitative analysis of agile project success,” *International Journal of Project Management*, vol. 33, no. 5, 2015, doi: 10.1016/j.ijproman.2015.01.006.
- [16] R. Santos *et al.*, “A Comparative Analysis of Agile Teamwork Quality Instruments in Agile Software Development: A Qualitative Approach”, doi: 10.18293/DMSVIVA2023-217.
- [17] S. Al-Saqqa, S. Sawalha, and H. Abdelnabi, “Agile software development: Methodologies and trends,” *International Journal of Interactive Mobile Technologies*, vol. 14, no. 11, 2020, doi: 10.3991/ijim.v14i11.13269.
- [18] T. Dingsoeyr, D. Falessi, and K. Power, “Agile Development at Scale: The Next Frontier,” *IEEE Software*, vol. 36, no. 2. IEEE Computer Society, pp. 30–38, Mar. 01, 2019. doi: 10.1109/MS.2018.2884884.
- [19] F. Almeida, J. Simões, and S. Lopes, “Exploring the Benefits of Combining DevOps and Agile,” *Future Internet*, vol. 14, no. 2, Feb. 2022, doi: 10.3390/fi14020063.
- [20] R. Hasan, A.- Ta, and R. Razali, “Prioritizing Requirements in Agile Development : A Conceptual Framework,” *Procedia Technology*, vol. 11, no. Iceed, pp. 733–739, 2013, doi: 10.1016/j.protcy.2013.12.252.
- [21] A. Mishra and Y. I. Alzoubi, “Structured software development versus agile software development: a comparative analysis,” *International Journal of System Assurance Engineering and Management*, Aug. 2023, doi: 10.1007/s13198-023-01958-5.