

Development of Internal Quality Audit Information System with Personal Extreme Programming (PXP) Approach

Darmansyah^{1*}, Rifqi Aulia Tanjung², Novita Lizza Anggraini³, Vincent Gravie Ebenhaezer Lumenta⁴

^{1,2,3,4}Institut Teknologi Kalimantan, Indonesia

¹darmansyah@lecturer.itk.ac.id, ²rifqi.aulia@lecturer.itk.ac.id, ³novita.anggraini@lecturer.itk.ac.id,

⁴11201091@student.itk.ac.id



*Corresponding Author

Article History:

Submitted: 25-12-2024

Accepted: 30-12-2024

Published: 10-01-2025

Keywords:

Internal Quality Audit; Personal Extreme Programming (PXP); Information Systems; Iterative Development; System Efficiency.

Brilliance: Research of Artificial Intelligence is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0).

ABSTRACT

Internal Quality Audit (IQA) is an important process to ensure the quality of implementation of activities in educational institutions in accordance with established standards. However, the implementation of IQA that is still carried out manually or semi-digitally can hamper the effectiveness, efficiency, and accuracy in data management and reporting of audit results. Therefore, an information system is needed that can support the audit process in a more systematic and integrated manner. This research examines the development of an internal quality audit system using the Personal Extreme Programming (PXP) approach that is carried out iteratively. The purpose of this study was to evaluate the effectiveness and efficiency of the system developed through four iterations, each of which focused on developing key features such as user management, document management, audit process, and assessment. The PXP methodology was applied to provide flexibility in responding to user feedback, with each iteration resulting in rapid improvements and adjustments according to emerging needs. The results showed that the developed system met the audit management objectives with high functionality, ease of use, and improved efficiency at each iteration. In addition, this iterative approach allows the system to be more responsive to changing user needs compared to conventional development methods. Overall, this research supports the use of PXP as an effective methodology in the development of internal quality audit systems, demonstrating that it can produce solutions that are adaptive, efficient, and suited to user needs.

INTRODUCTION

Internal Quality Audit (IQA) is a systematic, independent, and documented testing process to ensure the implementation of activities at the institution according to procedures and the results are in accordance with the Internal Quality Assurance System established by the institution. IQA stages consist of self-evaluation / document audit and field visitation to follow-up based on audit findings with the scope of what will be audited is education standards, research standards, service standards and supporting standards

The implementation of internal quality audits at the Kalimantan Institute of Technology is still done manually or semi-digitally. This approach has the potential to create ineffectiveness in data management, hinder the analysis process, and reduce accuracy in evaluating and reporting audit results. Based on the Regulation of the Minister of Education, Culture, Research and Technology of the Republic of Indonesia number 53 of 2023 concerning Quality Assurance of Higher Education. The need for a comprehensive and integrated Internal Quality Assurance System (IQAS) is becoming increasingly urgent. With an integrated information system, the IQA process can be managed more systematically and efficiently, from audit data documentation, collaboration between auditors, to the management of follow-up recommendations (Mendikbudristek, 2023).

The development of information systems to support IQA requires an approach that is adaptive, fast, and able to deliver products that are relevant to user needs. Software development methods that can fulfil this include agile software development (Akhtar et al., 2022). One of the methods included in agile software development is Personal Extreme Programming or PXP which is a development method from Extreme Programming that is adjusted so that it can be done by a single developer (Putra et al., 2024).

Although several studies have conducted the development of web-based internal quality audit systems, the development approach using PXP is still not widely explored. This research gap provides an opportunity to explore how the PXP method can be applied in system development, as well as how this approach can improve the efficiency and quality of the developed system.

The purpose of this research is to develop a web-based Internal Quality Audit Information System by applying the Personal Extreme Programming (PXP) method. The system is designed to digitise and automate the IQA process, thus improving the effectiveness and efficiency of its implementation. This research can contribute to the availability of



a system that can be used by the Kalimantan Institute of Technology to improve efficiency and accuracy in the implementation of IQA and provide insight into the application of PXP in the development of information systems, especially in the context of internal quality audits, which can be a reference for future research and development.

LITERATURE REVIEW

System development with Agile methodology is expected to create a system that is more flexible and responsive to user input (Djitalov et al., 2024; Trihardianingsih et al., 2023). Personal Extreme Programming (PXP) derived from the adaptation of the Extreme Programming (XP) method is an agile method designed for single developers or small teams (Semma et al., 2023). The application of PXP in the development of the Internal Quality Audit Information System refers to several related studies that have been conducted previously. Previous research will be used as a reference source in the development and creation of this information system.

(Sulthoni Akbar et al., 2024) applying the Personal Extreme Programming method in designing a meeting room booking information system at Diskominfo East Java Province was successfully implemented properly. System development remains in accordance with client needs with the estimated workmanship is appropriate based on the planning or planning stages carried out. However, this research is not specific to the development of internal quality audits.

Most research on IQA development still uses more general software development methodologies, such as waterfall. For example, (Norfifah et al., 2023) and (Hasbi et al., 2021) designed web-based quality audit information systems using the waterfall approach. This approach has proven effective in creating a stable system, but lacks flexibility in dealing with changing needs.

(Widiantoro & Yodi, 2020) produced a valid Actor Network Theory (ANT)-based framework to understand the relationship between actors in university business processes. The resulting framework is able to map the internal quality assurance workflow comprehensively. The availability of an analytical approach to designing information system architecture focused on the interaction of actors in business processes, combined with PXP provides a flexible and iterative software development methodology. This combination can address the challenge of creating an internal quality audit system that is adaptive to the dynamic needs of higher education organisations.

A research gap exists in the form of a lack of exploration of the application of PXP to IQA system development. Most studies focus on waterfall which is less adaptive. This creates an opportunity to investigate the effectiveness of PXP as a more personalised and flexible approach to meeting the dynamic needs of internal quality audit systems.

METHOD

This research uses a software development approach based on Personal Extreme Programming (PXP). The stages in this research are literature study, identification, requirements, planning, iteration initialisation, design, implementation, system testing, retrospective, deployment, and user training (Nuryasin & Dwi Wahyuni, 2023) as shown in Figure 1.

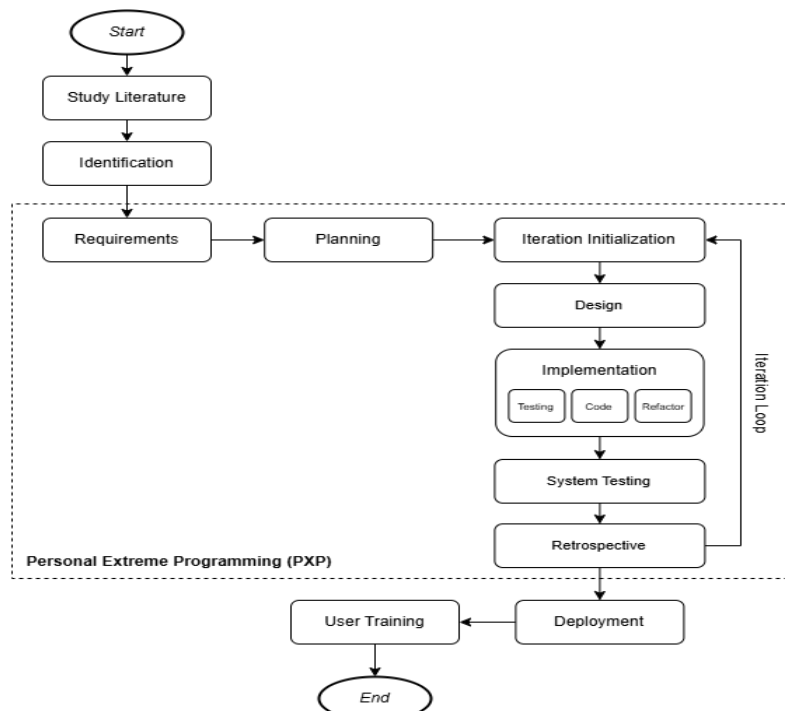


Figure 1. Personal Extreme Programming Flowchart

This research method begins with the Identification stage, which is a process of in-depth observation of the existing internal quality audit system. Data was obtained through source code analysis using reverse engineering methods, system operation, and review of relevant system documentation. The results of these observations provided insight into the system flow, data structure, and technology used, which became the basis for understanding further development needs (Saputra Kirsan et al., 2022).

In the Requirements stage, the needs of the new system were identified through interviews with the product owner to gather information on workflows and required features. The MoSCoW method was used to prioritise the requirements based on their level of urgency. After that, the Planning stage is done by dividing the requirements into units of work called user stories, each of which is given story points based on the level of complexity. Story points are scored using Fibonacci numbers to estimate the processing time with a standard five-day, eight-hour workday.

The next stage involves Iteration Initialisation, Design, and Implementation, which is done iteratively. The database structure and interface design were designed to fulfil the needs of each user story, while feature implementation included unit testing, coding, and refactoring. The system was tested using the User Acceptance Test (UAT) with the blackbox testing method. After the iteration is complete, the Retrospective stage is used to evaluate performance and determine opportunities for improvement in the next iteration. The completed system was then deployed to the production environment in the Deployment stage, followed by User Training to ensure users understand the system's operations, while providing documentation as a guide for future maintenance.

RESULT

The literature study stage was conducted to collect and analyse relevant references related to the re-engineering of ITK's internal quality assurance information system. The literature used includes articles, journals, books, institutional regulations, and previous research. Topics reviewed include Higher Education Quality Assurance System, SPMI ITK, Agile Methodology, Personal Extreme Programming (PXP) Method, Microservice, REST-API, Laravel, Black Box Testing, and other related research. This study provides a strong theoretical basis to support the development of the designed information system.

In the identification stage, the ITK internal quality assurance information system was analysed by studying documentation, operating the system, and analysing the source code using the reverse engineering method. This system has four main actors: PJM, Auditor, and Auditee. PJM acts as a super admin who manages users, determines standards, indicators, and audit schedules. The auditor is in charge of conducting audits on Kajur/Koorprodi as auditees. The current audit process is done manually by uploading and downloading documents through the system, while the technology used includes the Laravel framework for development, Bootstrap for the interface, and MySQL as a database.

The requirement stage was conducted through interviews with PJM as the product owner. The system to be developed includes three main actors: PJM, Auditor, and Auditee, with flexibility for users to have multiple roles. The quality assurance process begins with the creation of an audit document by PJM that includes categories, standards, competencies, and indicators. After that, audit forms are customised for each unit, auditors are assigned, and auditees are determined based on unit recommendations. The AMI process is divided into 6 stages, namely self-evaluation by the auditee, assessment by the auditor, giving feedback by the auditee on the auditor's assessment, verification by the auditor, auditee uploading the RTM results and the last stage Auditee uploading the Audit report.

The Planning stage begins with determining the user story that will be achieved based on the needs submitted by PJM as the product owner. Each user story is given a story point to represent the estimated processing time and level of complexity. This determination is made by considering the capacity of the developer and the technology used. Based on the results of interviews with PJM, Auditors, and auditees, the number of user stories obtained was 35 with a total estimated processing time of 148 days, examples of user stories can be seen in table 1.

Table 1. user stories

Code	Requirement	User Story	Acceptance Criteria	Priority	Story Point
US-01	PJM Adding users	As PJM, I want to add a user to the system.	PJM can add users to the system.	Must have	5
US-02	View user list	As PJM, I want to see all registered users.	PJM can see all users registered in the system.	Should have	3
US-03	Change user data	As, I want to change the user data.	PJM can change the user's role and contact to the desired.	Should have	3
US-04	Change contacts independently	As a user, I want to change my own contacts.	Users can change contacts independently.	Must have	3
US-05	Change your password	As a user, I want to change my own password.	Users can change their passwords independently.	Should have	1

US-06	Delete a user	As PJM, I want to delete the user.	PJM can delete the desired user.	Should have	1
US-07	Login	As a <i>user</i> , I want to log into the system according to my role.	Users can log in to the system with the role selected according to the assigned role.	Must have	3
US-08	Logout	As a <i>user</i> , I want to log out of the system.	The user can log out of the system.	Must have	1
US-...	...	...	...	...	...
US-35	Verifying the final audit result	As an auditor, I would like to verify the final audit results.	The auditor can fill in the form by providing the results of the field discussion and the final conclusion.	Must have	3

The iteration initialisation stage is the stage where the iteration process in the development of ITK's internal quality assurance information system is carried out. This process begins by dividing the user story into several iteration units based on the priority and needs of the user story. If a user story needs to be done first in order to work on other user stories, the user story will be included in the initial iteration. If during the development process there is a change or addition of a new feature, the developer will estimate the story points on that feature. After that, the developer will determine the priority of whether the story point can be applied in the ongoing iteration or in the next iteration, the list of iterations and estimated work can be seen in Table 2. In Iteration 1, the focus is given to the development of basic features such as user management, login, and document and form creation which are the foundation of the system. Iteration 2 continued development with user and document data management features to strengthen administrative functionality. Iteration 3 is geared towards supporting the audit process and reporting of results, while Iteration 4 contains system enhancements through additional features such as form data modification. Each iteration is complemented by a retrospective process to evaluate success and identify improvements, ensuring the system is developed incrementally and flexibly according to user needs.

Tabel 2. Iteration Plan

No	User Story	Estimated (days)
Iteration-1 : Basic System Features		
US-01	PJM Add user	5
US-07	Login	3
US-08	Logout	1
US-21	Add document	21
US-27	Add Form	21
	Velocity	51
Iteration-2 : User and Document Data Management		
US-02	View user list	3
US-03	Modify user data	3
US-05	Change password	1
US-22	Add draft document	13
US-23	View document	1
	Velocity	21
Iteration-3 : Audit and Assessment Features		
US-32	Fill in audit data	5
US-33	Assessing audit data	3
US-34	Providing audit assessment feedback	3
US-35	Verifying the final audit result	3
	Velocity	14
Iteration-4 : System Enhancements and Additional Features		
US-10	View unit list	1
US-30	Modifying form data	13
US-31	Deleting a form	2
	Velocity	16

Design and Implementation stages were conducted iteratively. In Iteration 1, the design focused on creating basic features such as user management, login/logout, and document and form creation, which were then implemented and tested as shown in Figures 2, 3 and 4.

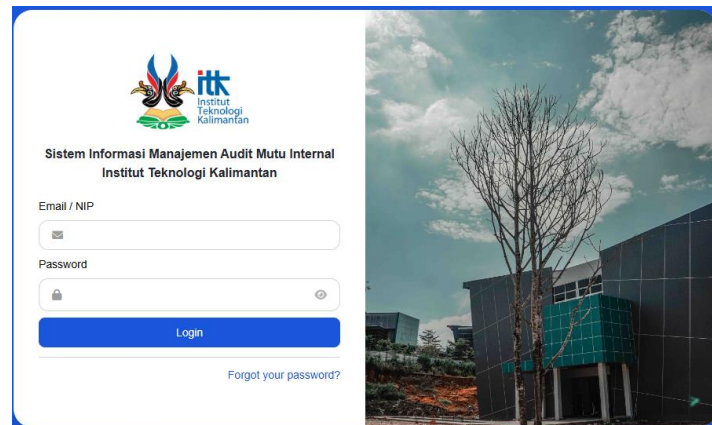


Figure 2. Login Form

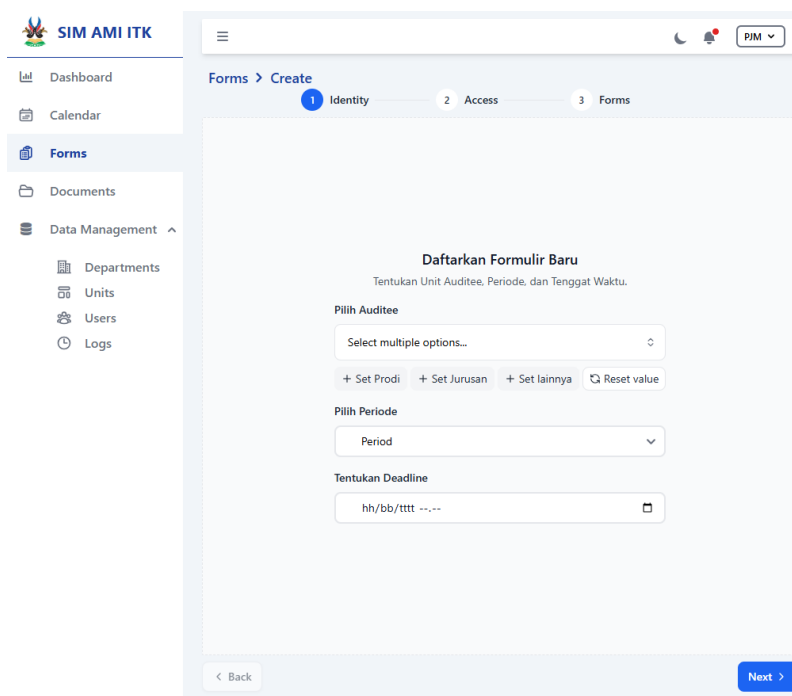


Figure 3. Form Addition

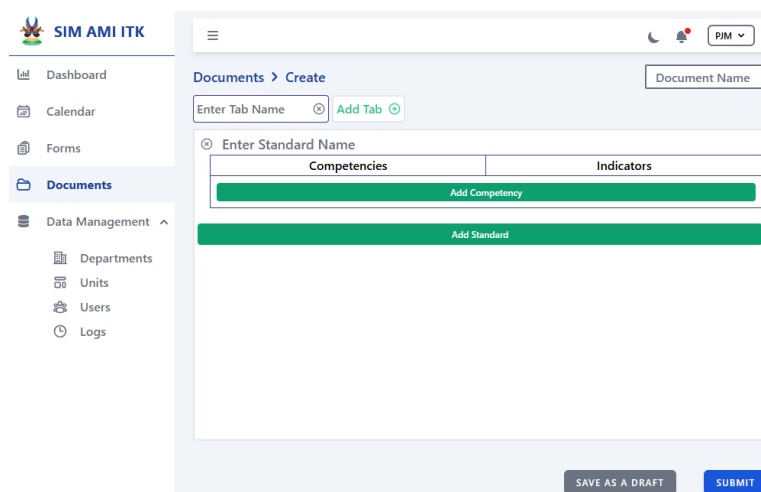


Figure 4. document addition

Iteration 2 extended the design for user and document data management as shown in Figure 5 and 6, such as the ability to view and change user data, change passwords, and add and view document lists, which were also implemented and tested to ensure functionality.

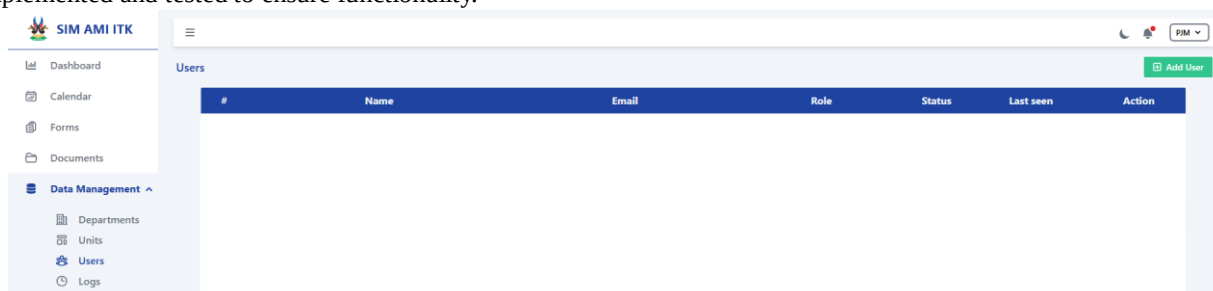


Figure 5. user list

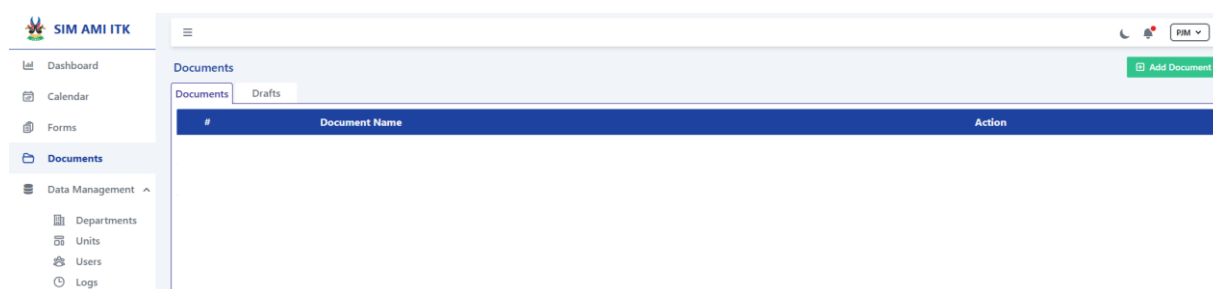


Figure 6. dokument list

Iteration 3 added designs for the audit and assessment features as shown in Figure 7, which were implemented to support the process of filling in audit data, assessment, feedback, and verification of audit results. In Iteration 4, the design focused on enhancing the system with additional features for unit management and form editing as shown in figure 8, which were implemented and tested to improve system efficiency. This iterative approach allows for continuous improvement and customisation of the system, ensuring a higher quality end result that meets user needs.

Figure 7. Audit forms

Standar Penelitian

Hasil Penelitian

Competencies	Indicators	Validation
Standar Luaran Penelitian	A.1 Jumlah Publikasi di Jurnal Internasional Q1-Q4	Option (Yes/No) ①
	A.2 Jumlah Publikasi di Jurnal Internasional	Digit (#) ①
	A.3 Jumlah Publikasi di Jurnal Nasional S1-S6	Decimal (.) ①
	A.4 Jumlah Publikasi di Jurnal Nasional	Cost (\$) ①
	A.5 Jumlah Publikasi di Seminar Internasional	Percentage (%) ①
	A.6 Jumlah Publikasi di Seminar Nasional	Rate (1/10) ①
	A.7 Jumlah Publikasi di Media Massa Internasional	Rate (1/100) ①
	A.8 Jumlah Publikasi di Media Massa Nasional	
	A.9 Jumlah artikel yang disitasi	Rekap data luaran penelitian DTPS

< Back

Submit

Figure 8. Validation forms

DISCUSSION

Analysis of the results from the development of an internal quality audit system using the Personal Extreme Programming (PXP) method iteratively. The research findings indicate that the application of PXP in the development of this system has a positive impact on its quality and effectiveness (Ch Lahama et al., 2024). The trial results show that key features such as user management, document handling, forms, and the audit and assessment processes function well and meet user needs. These features successfully support the goals of managing audits with an accessible and user-friendly system.

A comparison between the implementation results and the initial design shows a high level of alignment in terms of functionality and usability. Each iteration led to improvements in the system, with each developed feature being tested to ensure that the system functions according to user needs (Taryana et al., 2024). Users can easily manage audit-related data, documents, and forms, as well as conduct assessments and verify audits efficiently.

Additionally, the iterative approach applied in PXP allows for a more responsive system development process to user feedback. Changes and improvements can be quickly implemented in each iteration, ensuring that each feature added meets user expectations. This provides greater flexibility compared to conventional development methods, which are often less adaptive to changing user requirements (Shani et al., 2024).

Other significant findings include increased system efficiency, thanks to the additional features refined in the final iteration. New features, such as unit management and form editing, enhance the system's effectiveness in supporting the internal quality audit process more comprehensively. Overall, this research supports the use of PXP in developing internal quality audit systems, demonstrating that this method results in solutions that are more adaptive, efficient, and aligned with user needs.

CONCLUSION

The results of this study show that the application of Personal Extreme Programming (PXP) in the development of an internal quality audit system contributes significantly to the effectiveness and efficiency of the system. With an iterative development approach, the system can evolve continuously, accommodate user feedback, and fulfil emerging needs in each iteration. The implementation of key features such as user management, document management, audit process, and assessment demonstrate the system's ability to achieve the desired goals.

The iterative approach applied in PXP allows for flexible customisation and rapid adaptation, so that the system remains in line with user expectations and provides high functionality. This approach proves to be more responsive and adaptive compared to conventional development methods, which often struggle to accommodate evolving user needs. The efficiency of the system also increased with the addition of features such as unit management and form editing, which improved the overall audit process.

In conclusion, this study supports the effectiveness of using PXP as a development methodology for internal quality audit systems. The iterative approach not only resulted in a more user-focused and functional system, but also demonstrated the potential of PXP to produce adaptive and efficient solutions that fit user needs. Therefore, PXP proved to be a valuable methodology in developing systems that require continuous refinement and adaptation.

ACKNOWLEDGMENT

Thanks to the quality assurance centre and the auditors and auditees of the Kalimantan Institute of Technology's internal Quality Audit who have provided input for the development of this information system. In addition, thanks to the Kalimantan Institute of Technology through LPPM ITK which has provided strategic research grants to facilitate this service activity.

REFERENCES

- Akhtar, A., Bakhtawar, B., & Akhtar, S. (2022). Extreme Programming Vs Scrum: A Comparison Of Agile Models. *International Journal of Technology, Innovation and Management (IJTIM)*, 2(2), 80–96. <https://doi.org/10.54489/ijtim.v2i1.77>
- Ch Lahama, C. E., Ellyanne, C., Montolalu, J. C., & Tenda, E. (2024). Sistem Informasi Apotek Berbasis Web Menggunakan Metode Personal Extreme Programming (XP Programming) Dengan Arsitektur MVC. *IJIDS (Indonesian Journal of Intelligence Data Science)*, 03(01), 61–72. <https://doi.org/https://doi.org/10.35799/ijids.v3i1.50142>
- Djutalov, R., Irham, M., Putra, A. D., Elistiawan, A., & Syah, R. F. (2024). *Agile Methodology in Web and Mobile Based App of Tracer Study and Career Center*. <https://doi.org/10.47709/brilliance.v4i1.4372>
- Hasbi, M., Teknologi Informasi, F., Islam Kalimantan Muhammad Arsyad Al Banjari Banjarmasin, U., & Kalimantan Muhammad Arsyad Al Banjari Banjarmasin, I. (2021). SISTEM INFORMASI AUDIT MUTU INTERNAL (SIAMI). In *Technologia* (Vol. 12, Issue 2).
- Mendikbudristek. (2023). *PERATURAN MENTERI PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI REPUBLIK INDONESIA NOMOR 53 TAHUN 2023 TENTANG PENJAMINAN MUTU PENDIDIKAN TINGGI*.
- Norfifah, Julianto, V. J., & Yunita Prastyaningsih. (2023). Rancang Bangun Sistem Informasi Audit Mutu Internal. *Journal of Applied Computer Science and Technology*, 4(2), 108–117. <https://doi.org/10.52158/jacost.v4i2.539>
- Nuryasin, I., & Dwi Wahyuni, E. (2023). Rancang Bangun Sistem Informasi Geografis Pemetaan Penduduk menggunakan Metode Personal Extreme Programing (Studi Kasus: Desa Oro-oro Ombo). *Repositor*, 5(4), 841–848.
- Putra, W. P., Saputra, H. K., Hadi, A., & Mursyida, L. (2024). Penerapan Teknologi Kompresi Data untuk Optimalisasi Manajemen Bahan Ajar Digital Pada Aplikasi Respository Menggunakan Personal Extreme Programming (PXP). *J-HyTEL: Journal of Hypermedia & Technology-Enhanced Learning*, 2(1), 15–29. <https://doi.org/10.58536/j-hytel.v2i1.108>
- Saputra Kirsan, A., Novi Arisa, N., & Fitratunnany Insanittaqwa, V. (2022). Development of SIAKAD Applications in Balikpapan Schools using APXP: Advanced Personal Extreme Programming. *Jurnal Inovtek Polbeng - Seri Informatika*, 7(1), 97–109. <https://doi.org/https://doi.org/10.35314/isi.v7i1.2196>
- Semma, A. B., Saerozi, M., Kusri, K., Syukur, A., & Maimun, A. (2023). An Extreme Programming Approach to Streamlining Thesis Writing. *International Journal on Advanced Science, Engineering and Information Technology*, 13(6), 2308–2313. <https://doi.org/10.18517/ijaseit.13.6.18701>
- Shani, T. R., Mintoro, S., & Ngajiyanto, N. (2024). Leveraging Personal Extreme Programming for Developing an Online Registration System. *PIKSEL : Penelitian Ilmu Komputer Sistem Embedded and Logic*, 12(2), 267–274. <https://doi.org/10.33558/piksel.v12i2.9638>
- Sulthoni Akbar, M., Nuryasin, I., & Chandranegara, R. (2024). Implementation Of the Personal Extreme Programming Method in Designing an Android-Based Meeting Room Reservation Application at Diskominfo East Java. *Jurnal Teknologi Infomrasi, Komputer, Dan Aplikasinya*, 6(2), 308–315. <https://doi.org/https://doi.org/10.29303/jtika.v6i1.348>
- Taryana, Y., Darsanto, S. T., & Kom, M. (2024). Rekayasa Sistem Buku Tamu Berbasis Perangkat Rfid Menggunakan Metode PXP pada Dinas Perpustakaan dan Arsip Kabupaten Indramayu. *Jurnal Informatika Software Dan Network*, 05(01), 21. <https://jurnal.dccpringsewu.ac.id/index.php/ji/article/view/61>
- Trihardianingsih, L., Istighosah, M., Alin, A. Y., & Ghonim Asgar, M. R. (2023). Systematic Literature Review of Trend and Characteristic Agile Model. *Jurnal Teknik Informatika*, 16(1), 45–57. <https://doi.org/10.15408/jti.v16i1.28995>
- Widiantoro, S., & Yodi, Y. (2020). KERANGKA KERJA TEKNOLOGI INFORMASI UNTUK AUDIT MUTU INTERNAL BERBASIS IAPS 4.0 DENGAN ACTOR NETWORK THEORY. *JURTEKSI (Jurnal Teknologi Dan Sistem Informasi)*, 7(1), 89–94. <https://doi.org/10.33330/jurteksi.v7i1.927>