Available online at JDR Website: http://journal.unublitar.ac.id/jdr
Journal of Development Research, Volume 8, Number 2, November 2024, Page 1
DOI: https://doi.org/10.28926/jdr.v8i2.398

RAD-UCD Implementation in Project Management Information System

Indra Gita Anugrah[1,*], Widyasari Puspa Permata Witra[2], Sukma Uli Nuha[3], Nugroho Setyo Wibowo[4], Denny Trias Utomo[5]

[1,2]Department of Information System, Universitas Muhammadiyah Gresik, Indonesia
[3]Department of Accounting, Universitas Muhammadiyah Gresik, Indonesia
[4,5]Department of Informatics, Politeknik Negeri Jember, Indonesia

E-mail: indragitaanugrah@umg.ac.id[1], widyasarippw@umg.ac.id[2], sukma@umg.ac.id[3], nugroho@polije.ac.id[4], denny_trias@polije.ac.id[5]

*Corresponding Author

Abstract

This study discusses the implementation of Rapid Application Development (RAD) and User-Centered Design (UCD) methods in developing project management information systems for small and medium-scale projects. Using interviews and direct trials by project managers, project team members, and clients, this study aims to understand the extent to which these two methods can improve development efficiency and user satisfaction. The results show that the RAD method accelerates the development process through flexible iterations, while UCD ensures that the resulting system is in accordance with user needs. Project managers reported increased transparency in monitoring progress, team members appreciated features that supported coordination, and clients felt more connected to project developments in real time. The combined implementation of RAD and UCD proved effective in creating a responsive, efficient, and user-friendly project management system to meet the needs of small and medium projects.

Keywords: *project management information system, rapid application development, software development life cycle, user centered design, user persona*

Introduction

IT project management plays a crucial role in ensuring that the development and implementation of information technology runs according to target, both in terms of time, cost, and quality (Doronina and Doronina, 2018; Rostova, Shirokova and Sokolitsyna, 2019; Klein, 2023). However, IT projects often face various challenges that hinder their success. One of the main problems is the lack of adequate feedback from users (Mirziianov, Kiamov and Ehlakov, 2021; Rietz, 2021). In fact, feedback from users is very important to ensure that the solutions built are relevant to their needs and expectations (Lucena *et al.*, 2017; van Oordt and Guzman, 2021; Tizard *et al.*, 2023). Without continuous input, the project team risks producing products or services that do not meet business needs.

This problem often arises due to various factors, such as limited communication between the project team and users, lack of user involvement in critical phases of the project, and the lack of formal mechanisms to collect feedback on an ongoing basis. In addition, users may feel that their involvement is only needed at the beginning and end of the project, so that there is

no consistent interaction throughout the development process (Alvertis *et al.*, 2016). This increases the chances of a gap between user expectations and the final results of the project.

The implications of a lack of user feedback can be very serious for the success of an IT project. Products that do not meet user needs have the potential to experience resistance during implementation, increase costs and time for revisions, or even result in total failure. Therefore, it is important for organizations to build effective communication mechanisms and a culture of collaboration between project teams and users. With continuous feedback, IT projects can be more adaptive and aligned with the business goals to be achieved.

The problem of lack of feedback from users in IT project management can also be overcome by understanding and implementing the concept of Software Development Life Cycle (SDLC). SDLC provides a systematic framework that covers important stages in software development, from needs analysis, design, development, to testing and maintenance. Each stage of SDLC ideally involves users to ensure that the solution built remains in line with business needs and provides optimal value. SDLC comprises the following phases: planning, analysis, design, implementation, testing, and maintenance (Britton and Doake, 2000). The SDLC models can be classified as conventional (Spiral, Incremental, and V-Shaped Models) or modern (Rapid Application Development, Agile software development). Speed and adaptability are important considerations when using modern model (Shakya and Shakya, 2020). Nonetheless, the RAD and Agile methodologies differ in how they accomplish development speed.

RAD is an effective methodology to deliver much faster development and higher quality results (Kirmani, 2017).The RAD approach offers advantages in terms of speed because it focuses on projects with a clear scope, generally small to medium scale. RAD is good for developing software that requires only minimal planning for rapid prototyping (Naz and Khan, 2015). A well-defined scope allows RAD to concentrate on creating prototypes that consumers can test and provide feedback on. This process can be repeated to create better iterations of the application while being developed (Delima, Santosa and Purwadi, 2017). User involvement in RAD can ensure IT project success (Delima, Santosa and Purwadi, 2017). In RAD, prototyping is used as a tool to get feedback from users quickly and iteratively (Delima, Santosa and Purwadi, 2017).

However, when using RAD, the developer's focus is more on technical/functional problems of a system (Daraghmi and Daraghmi, 2022). As a result, non-functional needs for a system cannot be captured effectively (Islam and Storer, 2020; Daraghmi and Daraghmi, 2022). Non-functional needs have the same level of importance as functional needs. When the non-functional needs of a system cannot be documented properly, it will result in customer dissatisfaction or additional costs (Ali *et al.*, 2022). Non-functional requirements must have a formal method used to capture non-functional user information and be able to identify its impact on other requirements during the system iteration process (Shankar *et al.*, 2020). Several attempts have been made to make the RAD approach more focused on user involvement in better requirements mining (Hassan, Qamar and Idris, 2015; Wahyuningrum *et al.*, 2021; Daraghmi and Daraghmi, 2022), but empirical evidence is still needed, especially from a IT project management perspective.

Moreover, RAD does not provide direction on how to build a IT project plan, form a team, manage the relationship between users and developers and document each step (Daraghmi and Daraghmi, 2022). The main disadvantage of RAD is the lack of documentation, making it more difficult to track changes in requirements (Islam and Storer, 2020; Daraghmi and Daraghmi, 2022). Therefore, A method is needed to improve the RAD model that can support user participation that can show the system's non-functional needs and support documentation of the changes.

There are several models that involve user participation to improve the ability to obtain non-functional system requirements, such as the Game Development Life Cycle which is usually used in game development. This GDLC model focuses on the technical interaction of humans on a system that will be evaluated in stages (Rahman and Anugrah, 2022). Good interaction between humans and systems is an interaction that successfully balances the functionality and non-functionality of the system (Naveen Bhavani *et al.*, 2022). Therefore, to perfect the research on the balance between functional and non-functional needs in the information system development process, the User Centered Design (UCD) model needs to be introduced to the software development process, especially the RAD model. The UCD model is also relevant to address the problem of lack of feedback in IT project management (Bednarik and Krohns, 2015; Landis-Lewis *et al.*, 2020). While RAD focuses on accelerating development with rapid prototyping and iteration, UCD puts users at the center of the entire design and development process (Toms, 2017; Afrianto *et al.*, 2019). In the context of IT projects, the combination of RAD and UCD allows project teams to not only accelerate the development cycle but also ensure that each iteration produces a solution that truly meets user expectations. With RAD, users can provide feedback on prototypes on a regular basis, and UCD ensures that the input is translated into relevant and easy-to-use features. The active involvement of users in both approaches helps reduce the risk of developing products that do not meet real needs. In addition, this intensive collaboration also increases user satisfaction and facilitates the adoption process of technology solutions. By implementing RAD and UCD synergistically, IT projects can be more adaptive and responsive, producing end products that are not only functional but also user-friendly and relevant to dynamic business needs.

Meanwhile, the User Centered Design model is used as one method to find out customer needs by prioritizing traceable functions on changing needs (Xiao, Su and Su, 2020). Therefore, the User Centered Design model will be integrated into activities in RAD, namely during the prototype development process. The prototype development process by integrating the UCD model will produce a prototype that will approach the results desired by the customer with various documentation that will make it easier for developers to track changes in user needs.

Research Methods

There are several stages in conducting this research, namely requirement engineering, designing solution and prototyping, constructing, and releasing.

1. Requirement Engineering (RE). RE is the process of acquiring the requirement for the information system. This process is one of the critical process from the entire research process because the output of this stages contains all of the system feature that needs to be designed on the next stages. There are several processes in requirement engineering. include:
    a. Inception. The inception stage provides an outline of the basic picture of the problems faced by an organization.
    b. Elicitation. The elicitation stage focuses on collecting needs from stakeholders. The elicitation stage uses a needs elicitation technique with a user persona approach. The user persona technique is a stage to support the approach in user characteristics during the needs extraction process (Kusuma, Rohimah and Husna, 2021). There are several activities for the elicitation process using user personas, namely (1) declaring hypotheses; (2) identifying behavioral variables; (3) interviewing behavioral variable subject maps; (4) identifying significant behavioral patterns; (5) synthesizing characteristics and goals; (6) developing descriptions of attributes and behaviors; and (7) determining the type of persona (Kusuma, Rohimah and Husna, 2021).

c. Elaboration. The main function in the elaboration stage is to carry out modeling activities and develop prototypes that describe the features and limitations of the problem.
d. Negotiation. The negotiation stage focuses on negotiations between developers and users and stakeholders regarding which needs are needed and which need to be eliminated so that the project can continue to run with limited resources. Users will be asked to prioritize needs, potential conflicts, and risk of needs.

2. Design Solution and Prototyping Stage. This stage is a stage that is carried out simultaneously where the development team will develop a solution to the problem along with a prototype that is in accordance with the Solution. This stage will produce several solutions, such as written documents, a set of models, a collection of use cases, and prototypes. At this stage, the technical team will collect all the requirements and develop a prototype. The entire stage of needs exploration, solution design, prototyping will be iterated until the user is satisfied with the overall results of the desired prototype. However, this iteration must still refer to the problems that have been described at the beginning. This iteration is carried out with the hope that the UCD model can be implemented properly. In addition, the number of changes made in each iteration is calculated. This is done with the aim of detecting the most effective iteration process.
3. Development and Release Stages. Development and release in application development is the final stage where the application that has been developed is prepared to be officially released to users. This process includes stages such as finalization of development, final testing, and preparation of the necessary infrastructure and documents. Once the application is considered ready, the launch is carried out by ensuring adequate availability of resources and technical support. In addition, promotion and marketing can also be done to increase awareness and adoption of the application by users.

Results and Discussions

Based on research that has been conducted using a combination of RAD and UCD methods, the following results were obtained:

1. Identifying requirements. Identifying application development requirements is a crucial process in software development, ensuring that the development team clearly understands the objectives and functionalities the application needs to achieve. This process, known as Requirements Engineering (RE), involves various techniques and methodologies to gather, analyze, and document the necessary requirements. Rapidly assessed and developed (RAD) approaches are useful for swiftly determining and meeting user needs, particularly for information systems (IS). These methods work well for extracting needs that are difficult to express (Carrizo, Dieste and Juristo, 2017). Different approaches each have certain qualities and are appropriate in certain situations. In practice, hybrid techniques are frequently employed, fusing components of several systems to satisfy changing needs (Molina-Ríos and Pedreira-Souto, 2020). Goal-focused requirement engineering methods can save software development expenses and effort while enhancing the quality of online apps (Chawla, Srivastava and Bedi, 2017). The inception stage provides an outline of the basic picture of the problems faced by an organization.
    a. Inception. The initial stage provides an outline of the basic picture of the problems faced by an organization. Based on the results of interviews and surveys of respondents and informants, several problems were obtained during the work on

a project in the field of information technology. Understanding the problems is shown in Table 1 below.

Table 1. Understanding The Problem

Problems	Supporting Data
The lack of transparency in monitoring progress is a significant issue in IT project management due to dispersed information and the absence of centralized project status updates, rendering traditional communication technologies like Skype and WhatsApp ineffective for project monitoring.	The project manager expressed frustration over the challenge of providing stakeholders with clear project visibility by stating, "*Stakeholders seek constant updates, but with many moving parts, it's challenging to keep everyone aligned. Weekly status reports often become outdated by the time they're reviewed.*". Noting that frequent updates are difficult due to shifting project dynamics. Miscommunication often arises from inconsistent reporting and a lack of real-time updates.
The majority of software project management systems on the market today don't provide enough means for tracking progress or providing insightful reports and visuals.	The project team shared their frustrations with the limitations of many software project tools. They emphasized, "*they don't give us a clear picture of how far along we are with each task. For example, a task can be 'in progress' for days, but there's no way to show incremental updates*".
The difficulty in coordination between teams, and inefficient resource management. Traditional IT project management processes often rely on manual or separate methods that result in data duplication, recording errors, and delays in communication and information updates.	The project manager also pointed out that the absence of centralized communication channels creates confusion by saying "*Some teams prefer email, others use instant messaging apps, and others rely on the project management platform*".
The lack of consistent standards and procedures in reporting, which can lead to different interpretations of data by stakeholders.	The client expressed concerns about the misunderstandings and misalignment among stakeholders. The client says, "*Sometimes, the reports we receive are too technical*".

b. Elicitation. The elicitation stage focuses on gathering needs from stakeholders. The elicitation stage uses the needs elicitation technique with the user persona approach. The user persona technique is a stage to support the approach in user characteristics during the needs excavation process shown in Table 2 below.

Table 2. Requirement Based on User Persona

User Persona	Description	Requirement
Project Manager (Reza)	Reza is a Project Manager with 10 years of experience. He is responsible for planning, executing, and monitoring projects. Reza needs access to real-time progress reports and tools to manage team resources.	A system that makes it easy to manage timelines and allocate resources. A dashboard feature that provides a summary of overall project progress. Automatic notifications for schedule changes or tasks that have passed their deadlines.
Developer (Budi)	Budi is a developer with 5 years of experience who focuses on feature	Integrated system with code management tools like Git.

	development and technical task completion. He needs a tool for clear task management and efficient communication with other team members.	Clear and structured workflow in task management system. Features to track and report issues or bugs directly.
Client Project (Andi)	Andi is a business owner who uses projects to develop IT solutions for his company. Andi wants full transparency in every stage of the project and regular reports on the progress of the project.	Access to an easy-to-understand dashboard to view project progress. Ability to provide feedback and revisions directly through the system. Automated reports sent periodically to monitor milestones. Features for testing and providing feedback on prototypes or beta versions of the system. Clear and easily accessible documentation of the features being developed. A flexible and customizable system to suit the specific needs of end users.

Based on the results of the user persona, the following project management system requirements were obtained Shown in Table 3 below.

Table 3. Requirement System

Requirement	Team Project	Client Project
Functional Requirement	Project teams need a system that can manage tasks and resource allocation more effectively. Real-time time tracking and work progress features are needed to monitor team performance. Integration with communication tools such as email and chat is proposed to improve coordination between team members.	Clients need access to the system to monitor project status and progress transparently. Clients want automated reporting features that provide regular summaries of project progress, including milestones achieved. The ability to provide feedback or revisions directly through the system is proposed to increase interaction and rapid response.
Non-Functional Requirement	The system should be easy to use and have an intuitive interface. Mobile access is desired so that team members can access the system anytime and anywhere. Data security is a priority, especially in managing sensitive project information.	The client emphasized the importance of a responsive and reliable system, especially in terms of system load times and availability. A simple and user-friendly interface for clients who may not have a technical background. Protection of client data from unauthorized access and confidentiality of project information must be strictly maintained.

c. Elaboration and Negotiation

Based on the elicitation process, the following requirements that is elaborated and negotiated with the stakeholders were obtained Shown on Table 4:

Table 4. Table of Elaboration and Negotiation of Requirements

Main Function	Function Name	User	Description	(√)

Project task and resource management	Add project	PM	Users can add projects to be implemented	√
	View projects	PM, PT	Users can see projects that will be, are being, and have been implemented.	√
	Edit project	PM	Users can edit projects currently underway.	√
	Delete project	PM	Users can delete projects that are currently being implemented.	√
	Add project resources	PM	Users can add project resources to be implemented.	√
	View project resources	PM, PT	Users can view project resources that will be, are being, and have been implemented.	√
	Edit project resources	PM	Users can edit the resources of the project being executed.	√
	Delete project resources	PM	Users can delete project resources that are currently being executed.	√
	Add task	PM	Users can add tasks to projects currently being implemented.	√
	View task	PM, PT	Users can view tasks on projects currently being implemented.	√
	Edit task	PM, PT	Users can edit tasks on projects that are currently being implemented.	√
	Delete task	PM	Users can delete tasks on projects that are currently being executed.	√
Tracking progress with feedback	Add progress	PM, PT	Users can add task progress to projects currently being implemented.	√
	View progress	PM, PT, PC	Users can see the progress of tasks on projects currently being implemented.	√
	Add feedback	PM, PT, PC	Users can add task progress feedback to projects currently being implemented.	√
Dashboard for periodic automated reporting	View dashboard	PM, PC	Users can see the overall progress of the project being implemented.	√
	Print report	PM, PC	Users can print reports on the overall progress of the project being implemented.	√
Prototype testing	Add prototype	PT	Users can add prototypes to projects currently under implementation.	√
	View prototype	PM, PT, PC	Users can view prototypes of projects currently under implementation.	√
	Edit prototype	PM, PT	Users can edit prototypes of projects currently underway.	√
	Add prototype feedback	PM, PT, PC	Users can add feedback to prototype projects that are being implemented.	√

2. Design Solution and Prototyping

Based on the identifying requirement process, the following is the design solutions and prototype Shown on the Figure 1:

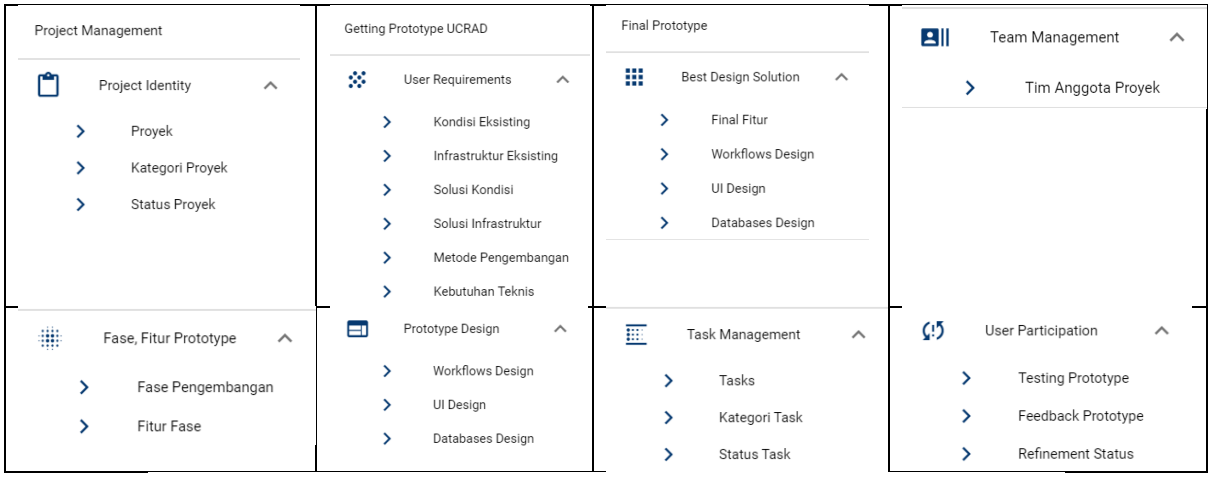

Figure 1. Design and prototype for the information architecture for the system

3. Development and Releasing
   Based on the design solution and prototyping process, the following is the development of the project management information system:

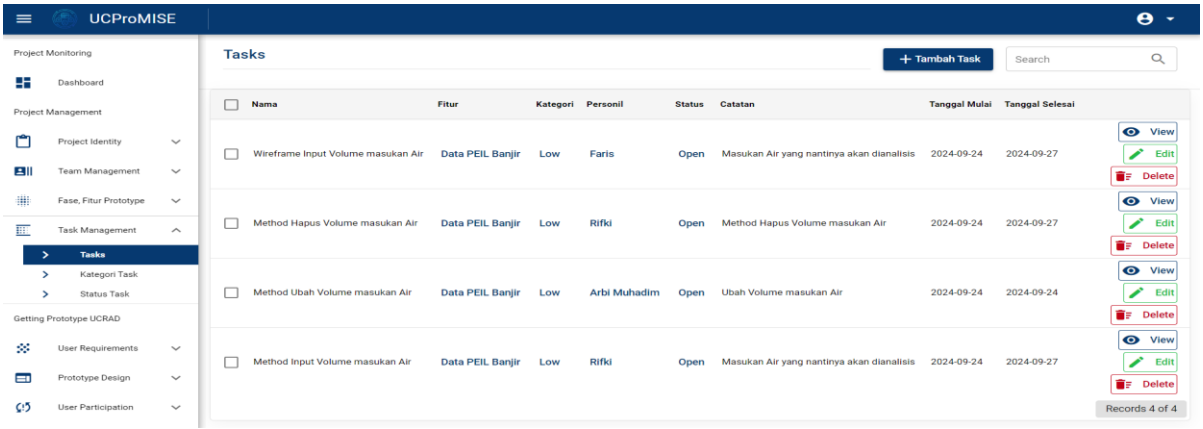

Figure 2. Developed system using The Concept of RAD and UCD

Based on the results of research that has been carried out using a combination of RAD and UCD methods, Shown on Figure 2 the following findings were obtained:

1. This study aims to understand more deeply the experiences of project managers, project teams, and clients related to the implementation of Rapid Application Development (RAD), User-Centered Design (UCD), and user persona methods in IT project management. In-depth interviews were conducted with several key roles in IT projects, namely project managers, project team members, and clients. The project manager stated that the RAD methodology provided much-needed flexibility to deal with the dynamics of IT projects by saying, "*With RAD, we can release prototypes faster and get immediate feedback from clients. These iterations help us identify errors early on, so we don't have to wait until the end of the project to make major improvements*". He appreciated the rapid iterations in RAD that allowed changes based on user requirements without significantly disrupting the project flow.
2. Project team members shared that UCD encouraged them to better understand the user perspective and design solutions that were truly relevant. They engaged in direct user interviews and used the research findings to guide feature design. This findings is further proven by the saying of one of the team project which said, *"Through UCD, we learned what users really needed, not just our own assumptions. Sometimes*

features that we thought were good turned out to be irrelevant to users. With direct feedback, we were able to adjust early on."

3. Team members also noted that RAD-UCD required a high level of commitment from users to be actively involved. Challenges arose when users were inconsistent in providing feedback at every stage, which affected the development schedule.
4. The project team acknowledged that UCD required more time and resources because each change had to go through a validation phase with users. In addition, some users did not always have the time to actively participate at every stage, which could lead to potential delays. *"Sometimes it is difficult to get commitment from users, especially when they are busy with their daily work. We have to keep reminding them to be involved at every stage, especially when testing new features."*
5. The project client stated that using RAD-UCD by using user persona helped them communicate their needs more clearly to the project team. Having detailed persona profiles helped the team better understand the client's business context and priorities. "*We found it easier to communicate our needs because the personas they created closely reflected the characteristics of us. The project team also didn't have to keep asking questions that had already been identified at the beginning, so everything was more efficient.*"
6. The client mentioned that personas helped avoid conflicts over non-essential features. However, he emphasized the importance of updating personas regularly to keep up with changing user needs over time. "*Changes in the market often occur, and user profiles can change quickly. So personas created at the beginning of the project cannot be left alone—they must be adjusted according to changing needs.*"

Based on the findings, the implementation of RAD-UCD has proven to have a positive impact on increasing user engagement and project efficiency. The RAD method facilitates rapid iteration that is responsive to changes, while UCD ensures that the developed solution is relevant and easy to use. The use of user personas helps the project team to focus on specific needs and avoid mistakes due to inaccurate assumptions.

The RAD (Rapid Application Development) and UCD (User-Centered Design) methods are very suitable for small and medium-scale project management systems. With RAD, the development process becomes faster and more flexible, allowing the project team to build iterations in stages and respond to changing needs immediately. The UCD approach ensures that user needs are always the top priority, resulting in a system that is relevant and easy to use. This second method provides advantages for small and medium-scale projects, because budget and time constraints can be overcome with efficient development cycles and solutions that are more adaptive to user feedback.

However, this study also identified several challenges. Inconsistent user commitment in providing feedback can affect the development schedule and quality. In addition, intensive coordination between the project team and users requires effective communication management.

Conclusions

Implementation of RAD and UCD methods in small and medium-scale project management information systems is effective in increasing user satisfaction and system development efficiency. Through interviews with project managers, project team members, and clients, and letting them try the developed system, this study found that the RAD method accelerates the development cycle and allows for rapid iteration based on direct feedback. The UCD approach ensures that user needs and preferences are well accommodated, resulting in a

system that is intuitive and in line with end-user expectations. The results of this study indicate that the combination of RAD and UCD provides an optimal solution for the development of small and medium-scale project management systems, with a focus on development speed and improving user experience. Project managers appreciate the increased transparency in tracking progress, team members feel helped by the integrated coordination features, and clients feel more connected to project developments in real time.

Acknowledgement

Direktorat Jenderal Pendidikan Tinggi, Riset, dan Teknologi (DRTPM) for funding this research in 2024/2025.

References

Afrianto, I. *et al.* (2019) 'Design Of E-Document System With Digital Signature Using User Centered Design Method', *Conference SENATIK STT Adisutjipto Yogyakarta*, 5. Available at: https://doi.org/10.28989/senatik.v5i0.390.

Ali, A. *et al.* (2022) 'Role of Non-functional Requirements in projects' success', in *2022 2nd International Conference on Digital Futures and Transformative Technologies (ICoDT2)*. IEEE, pp. 1–7. Available at: https://doi.org/10.1109/ICoDT255437.2022.9787463.

Alvertis, I. *et al.* (2016) 'User Involvement in Software Development Processes', *Procedia Computer Science*, 97, pp. 73–83. Available at: https://doi.org/10.1016/j.procs.2016.08.282.

Bednarik, R. and Krohns, J. (2015) 'User-centred design practice and adoption in smaller ICT companies in Eastern Finland: an interview study', *International Journal of Business Information Systems*, 18(3), p. 285. Available at: https://doi.org/10.1504/IJBIS.2015.068165.

Britton, C. and Doake, J. (2000) *Object-Oriented System Development: A Gentle Introduction*. McGraw-Hill Publishing Company.

Carrizo, D., Dieste, O. and Juristo, N. (2017) 'Contextual attributes impacting the effectiveness of requirements elicitation Techniques: Mapping theoretical and empirical research', *Information and Software Technology*, 92, pp. 194–221. Available at: https://doi.org/10.1016/j.infsof.2017.08.003.

Chawla, S., Srivastava, S. and Bedi, P. (2017) 'Improving the quality of web applications with web specific goal driven requirements engineering', *International Journal of System Assurance Engineering and Management*, 8(S1), pp. 65–77. Available at: https://doi.org/10.1007/s13198-015-0385-z.

Daraghmi, Y.-A. and Daraghmi, E.-Y. (2022) 'RAPD: Rapid and Participatory Application Development of Usable Systems During COVID19 Crisis', *IEEE Access*, 10, pp. 93601–93614. Available at: https://doi.org/10.1109/ACCESS.2022.3203582.

Delima, R., Santosa, H.B. and Purwadi, J. (2017) 'Development of Dutatani Website Using Rapid Application Development', *IJITEE (International Journal of Information Technology and Electrical Engineering)*, 1(2). Available at: https://doi.org/10.22146/ijitee.28362.

Doronina, J. and Doronina, E. (2018) 'Models of IT-Project Management', *International Journal of Computer Science and Information Technology*, 10(5), pp. 60–72. Available at: https://doi.org/10.5121/ijcsit.2018.10506.

Hassan, S., Qamar, U. and Idris, M.A. (2015) 'Purification of requirement engineering model for rapid application development', in *2015 6th IEEE International Conference on*

Software Engineering and Service Science (ICSESS). IEEE, pp. 357–362. Available at: https://doi.org/10.1109/ICSESS.2015.7339074.

Islam, G. and Storer, T. (2020) 'A case study of agile software development for safety-Critical systems projects', *Reliability Engineering & System Safety*, 200, p. 106954. Available at: https://doi.org/10.1016/j.ress.2020.106954.

Kirmani, M.M. (2017) 'Agile methods for mobile application development: A comparative analysis', in *International Journal of Advanced Research in Computer Science*, pp. 1200–1205.

Klein, G. (2023) 'The Entwinement of Project and Information Technology Research Editorial', *Project Management Journal*, 54(4), pp. 458–462. Available at: https://doi.org/10.1177/87569728231169415.

Kusuma, W.A., Rohimah, S.O. and Husna, R. (2021) 'PENGGALIAN KEBUTUHAN PENGGUNA PADA FASE ELISITASI PERANGKAT LUNAK MENGGUNAKAN USER PERSONA', *Science and Information Technology SINTECH Journal*, 4(1), pp. 22–28.

Landis-Lewis, Z. *et al.* (2020) 'Designing clinical practice feedback reports: three steps illustrated in Veterans Health Affairs long-term care facilities and programs', *Implementation Science*, 15(1), p. 7. Available at: https://doi.org/10.1186/s13012-019-0950-y.

Lucena, P. *et al.* (2017) 'IBM Design Thinking Software Development Framework', in, pp. 98–109. Available at: https://doi.org/10.1007/978-3-319-55907-0_9.

Mirziianov, R., Kiamov, A. and Ehlakov, E. V. (2021) 'System for Automatic Evaluation of Bug Report Quality', in *2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus)*. IEEE, pp. 2158–2160. Available at: https://doi.org/10.1109/ElConRus51938.2021.9396261.

Molina-Ríos, J. and Pedreira-Souto, N. (2020) 'Comparison of development methodologies in web applications', *Information and Software Technology*, 119, p. 106238. Available at: https://doi.org/10.1016/j.infsof.2019.106238.

Naveen Bhavani *et al.* (2022) 'Review on Human Computer Interface', *International Journal of Advanced Research in Science, Communication and Technology*, pp. 541–544. Available at: https://doi.org/10.48175/IJARSCT-7014.

Naz, R. and Khan, M.N.A. (2015) 'Rapid Applications Development Techniques: A Critical Review', *International Journal of Software Engineering and Its Applications*, 9(11), pp. 163–176. Available at: https://doi.org/10.14257/ijseia.2015.9.11.15.

van Oordt, S. and Guzman, E. (2021) 'On the Role of User Feedback in Software Evolution: a Practitioners' Perspective', in *2021 IEEE 29th International Requirements Engineering Conference (RE)*. IEEE, pp. 221–232. Available at: https://doi.org/10.1109/RE51729.2021.00027.

Rahman, A.S. and Anugrah, I.G. (2022) 'Pengembangan Game Edukasi Pencegahan COVID-19 Menggunakan Metode Game Development Life Cycle', *ILKOMNIKA: Journal of Computer Science and Applied Informatics*, 4(1), pp. 42–52. Available at: https://doi.org/10.28926/ilkomnika.v4i1.446.

Rietz, T. (2021) 'Designing AI-Based Systems for Qualitative Data Collection and Analysis', *Hochschulschrift* [Preprint].

Rostova, O., Shirokova, S. and Sokolitsyna, N. (2019) 'Management of project for automation of investment control at industrial enterprise', *IOP Conference Series: Materials Science and Engineering*, 497, p. 012017. Available at: https://doi.org/10.1088/1757-899X/497/1/012017.

Shakya, P. and Shakya, S. (2020) 'Critical Success Factor of Agile Methodology in Software Industry of Nepal', *Journal of Information Technology and Digital World*, 02(03), pp. 135–143. Available at: https://doi.org/10.36548/jitdw.2020.3.001.

Shankar, P. *et al.* (2020) 'Towards the formalization of non-functional requirements in conceptual design', *Research in Engineering Design*, 31(4), pp. 449–469. Available at: https://doi.org/10.1007/s00163-020-00345-6.

Tizard, J. *et al.* (2023) 'A Software Requirements Ecosystem: Linking Forum, Issue Tracker, and FAQs for Requirements Management', *IEEE Transactions on Software Engineering*, 49(4), pp. 2381–2393. Available at: https://doi.org/10.1109/TSE.2022.3219458.

Toms, E. (2017) 'User-Centered Design of Information Systems', in *Encyclopedia of Library and Information Science, Fourth Edition*. CRC Press, pp. 4803–4811. Available at: https://doi.org/10.1081/E-ELIS4-120043525.

Wahyuningrum, T. *et al.* (2021) 'Developing Suicide Risk Idea Identification for Teenager (SERIINA) Mobile Apps Prototype using Extended Rapid Application Development', in *2021 9th International Conference on Information and Communication Technology (ICoICT)*. IEEE, pp. 92–97. Available at: https://doi.org/10.1109/ICoICT52021.2021.9527508.

Xiao, B., Su, K. and Su, X. (2020) 'A Requirement Engineering Approach to User-Centered Design', *Journal of Physics: Conference Series*, 1453(1), p. 012032. Available at: https://doi.org/10.1088/1742-6596/1453/1/012032.