Defining Integrated Agriculture Information System Non-Functional Requirement and Re-engineering the Metadata

Argo Wibowo[1], Antonius Rachmat Chrismanto[2], Gabriel Indra Widi Tamtama[3], Rosa Delima[4]
Information System Department, Universitas Kristen Duta Wacana, Yogyakarta, Indonesia[1, 3]
Informatics Department, Universitas Kristen Duta Wacana, Yogyakarta, Indonesia[2, 4]

***Abstract*—Developing a well-functioning information system like integrated agriculture information system (IAIS) requires a list of task requirements that will be transformed into system features. Feature Driven Development (FDD) model is suitable for this situation. The requirements for building an information system are not solely based on functional needs but also non-functional requirements (NFR). Non-functional requirements also play a crucial role in system development as they affect business process management. A well-defined business process will ultimately result in robust system features. It is essential to map non-functional requirements to the business process to clearly identify the information system requirements that will become new features. Not only can NFR enrich system metadata and databases, but they also serve as the initial foundation for the system coding process, leading to the final information system output. This study creates a flow diagram mapping NFR to the business process using Business Process Management Notation (BPMN). Several identified NFR categories are then transformed into metadata and use case diagrams. The formation of this NFR mapping flow diagram is expected to facilitate information system development by visualizing system requirements in a forward and backward flow according to the sequence of processes. Feature development can be streamlined in the event of NFR changes by tracing NFR and related features.**

Keywords*—Information system; non-functional requirements; BPMN; metadata; feature-driven development

I. Introduction

In the rapidly evolving world of information system development, delivering high-quality software that meets user expectations is crucial for the success of the software so that it can be effectively utilized. While functional requirements determine what the system should do, non-functional requirements (NFR), business process management (BPM), and requirements reengineering play a significant role in shaping software performance [1], aligning with business objectives, and adapting to changing needs. This article is written to explore the integration of these three components within the context of Feature-Driven Development (FDD), a popular software development methodology.

In the case described in this article, the Dutatani research team has already developed IAIS with a web-based portal application [2] and a farmer registration application [3]. Both studies have successfully described the development of the portal application, farmer registration, and their testing. In this paper, the research will be further developed by adding land and ownership mapping. This feature has previously existed in studies [4] [5], but it will be adjusted to accommodate new needs in the new portal and new business processes. The existing land mapping system could only accommodate one plot of land per farmer. However, the latest requirements state that farmers should be able to own multiple plots of land, and each plot can be owned by multiple farmers. Land ownership features will also be differentiated based on their status, reflecting common agricultural business processes that require distinguishing land ownership patterns, such as individual ownership, lease, and profit-sharing [6]. The aim of this reengineering effort in this research is to align the Dutatani web system's business processes with general agricultural practices in Indonesia. If the system can accommodate common business processes, it is expected to have widespread usability. This is inline with IAIS, namely integrating various user needs, especially in the agricultural sector.

The main problem in this research is the need to align new feature requirements with the existing system. To achieve alignment between new feature requirements and the existing system, a thorough analysis is required, involving functional and non-functional requirements, business process management, and ultimately leading to application reengineering [7]. A well-conducted analysis up to the Business Process Management diagram formation phase is expected to minimize the resource consumption during the application re-engineering. Features Driven Development is also used because the application is initially developed based on per-feature needs. Non-functional requirements determine the quality and constraints of software systems [8], such as performance, security, scalability, usability, and reliability. These requirements are crucial to ensure that the software meets user expectations and operates efficiently. In FDD, NFR is given equal importance alongside functional requirements in preparing features as a comprehensive software solution [9].

By integrating non-functional requirements, business process management, and reengineering within the FDD framework, the information system development team can enhance the quality, performance, and adaptability of their system solutions. The combined benefits of a holistic approach, NFR identification, business process alignment, and iterative requirement reengineering empower organizations to build software that not only meets functional needs but also provides optimal business value. This integration can help software development teams simplify the complexity of modern software development and achieve successful outcomes. This software development case study is different from previous research in the context of a farmer group in the Minggir, an area in Yogyakarta, with the hope of expanding the application's usage and accommodating more agricultural business processes.

This paper is organized as follows: 1) the first part presents the introduction, which includes the problem background, the objectives to be addressed, and the general method of resolution, 2) the second part is a literature review, containing references to previous studies, their relations, differences from this research, and the theoretical foundations used, 3) the third part is the research methodology, 4) the fourth part contains the results and discussion, which comprehensively presents the research findings and analysis, and finally 4) the conclusion and suggestions for further research development.

II. LITERATURE REVIEW

A. Dutatani System

This research is a continuation of previous Dutatani research. In the previous study, the Dutatani research team conducted the development of an integrated agricultural system, which consisted of a web-based portal system and farmer registration application. Land mapping was also created using native programming. Additionally, a study on the migration testing of the system from the old portal to the new portal has demonstrated significant improvements in the agricultural information. The study successfully identified and bridged the gap between the business process side and the information system side. The aim of this study is therefore to fill this gap through the proposed NFR classification and its translation into metadata, use case diagrams and functional design. NFRs can enrich the use case diagrams, metadata, and system features by integrating the successfully identified NFRs.

The conclusions of this research are as follows:

Integrating NFRs into use case diagrams and BPMN workflows. The proposed NFR classification has enriched the relationship between BPMN in the business process and its appropriate usage in the system metadata model.

Achieving forward and backward traceability from the business process to the information system model. The proposed classification has facilitated the mapping of NFRs from the business process into the information system. This research contributes to tracking NFRs and system features. If NFRs can be tracked, it enables the development of features for future development in case of any NFR changes. The proposed classification allows it to be used with other diagrams such as activity diagrams, flowcarts and even Data Flow Diagram (DFD). However, further study is needed to be able to develop other diagrams.

For future work, the research team plans to develop a matrix to connect NFRs to information system testing. Additionally, the matrix is expected to automatically assist in generating testing tables from BPMN to use case diagrams. Furthermore, the research team plans to investigate the effectiveness of NFRs on other system requirements such as load testing and system usability. Further details on the four research outcomes can be observed in Table I.

TABLE I. SUMMARY RELATED WORK

Topic	Summary, Reference and Comparison
Implementation of Feature-Driven Development to Facilitate Feature Equity and Adaptation in the Development of Dutatani Web and Mobile Portal [2]	This article describes the development of an information system based on features using the Feature-Driven Development (FDD) approach. The results obtained indicate that a feature-based approach can streamline the system development according to the features expected by users. In proposed method, we continue that list of feature from previous research and conduct with NFR and BPMN.
Blackbox Testing on the ReVAMP Results of The DutaTani Agricultural Information System [3]	In this article, the testing of the new agricultural data system is explained. The results show that the system can perform better and more efficiently compared to the old agricultural information system. Different with proposed method, we are using NFR to fullfill the gap between user and system feature. It also aims to increase the efficiency of using the system in different ways.
Feasibility Study of Web Mapping System Implementation Using the TELOS Method: A Case Study of Harjo and Rahayu Farmer Groups [4]	The readiness of users of the agricultural information system is elaborated in this article, and the resulting outcome is a score of 8.4, indicating the preparedness of users and the satisfactory performance of the agricultural information system. The proposed method also aims to increase the satisfactory of using the system in different ways, that use NFR and BPMN mapping.
Developing Agriculture Land Mapping using Rapid Application Development (RAD): A Case Study from Indonesia [5]	This article explains the stages of creating an agricultural land registration application on the old information system portal using the Rapid Application Development (RAD) approach. The study will extract use cases and existing business processes, which will then be applied to the new information system portal based on FDD. In this proposed method, data extraction was also carried out, but what was extracted was different, namely NFR based on the FDD that had been carried out.
The Effects of Land Ownership on Production, Labor Allocation, and Rice Farming Efficiency [6]	The article delves into the status of agricultural land ownership, which has been prevalent in the world, particularly in Asia, including individual ownership, lease, and profit-sharing. One plot of land can be owned by several individuals with different ownership statuses simultaneously. In the proposed method we try to use feature in this article so that existing features can be compared with community needs.

Based on several articles that serve as the main references in this study, it can be seen that this research is mutually interconnected and continues the best practices from previous research outcomes. In this current study, non-functional requirements will also be involved to add metadata information for land mapping. The agricultural land mapping business process that has been previously implemented will be continued in the new information system portal with minor modifications to land ownership.

The limitation of this research is that it is only limited to NFR analysis and metadata formation. Fixed features are also limited to features that are deemed to need to be improved based on NFR findings. The output is also a framework for mapping the NFR into metadata so that it is easy to trace when there are changes to the NFR.

B. Non-Functional Requirements

Non-functional requirements are requirements that state limitations on the services or functions offered by the system. These include time constraints, limitations on the development process, and limits set according to existing standards. Non-functional requirements are usually applied to the entire system. Non-functional requirements include speed, size, ease of use, reliability, robustness, and portability [10]. NFR also discusses issues related to product availability, maintenance, modifiability, timeliness, throughput, responsiveness, security, and scalability [11].

Defining NFR is a crucial element in producing a quality system. In cloud applications, determining NFR, workload, and Quality of Service (QoS) must be considered in deciding technology infrastructure [12]. Sumesh et al.'s study sought to achieve multi-objective optimisation of interdependent stakeholders and developed a framework to capture the competing NFR goals [13]. Soter [14], a method for modeling and translating NFR models, was introduced by DeVries and Cheng. Soter converts non-functional models into non-functional goal model fragments to be analyzed using the system-to-be goal model.

In research [15] [16], they classify NFR using a neural network approach. Other studies have attempted to optimize and balance the fulfillment of functional and non-functional requirements using the goal model approach [17][18]. A goal-oriented approach is also used to evaluate NFR compliance through the i* framework and Architecture modeling language (ArchiMate) [19].

III. Research Methodology

A. Gathering NFR (Non-Functional Requirements)

In FDD, NFRs must be identified and documented during the initial stages, along with the functional requirements. This proactive approach ensures that NFRs are considered from the outset, preventing time-consuming and costly rework. Additionally, NFRs help prevent delays in the later stages of information system development.

B. Mapping NFRs to Business Processes

By aligning NFRs with related business processes, the FDD team can identify critical NFRs that directly impact the success of those processes. This mapping helps prioritize NFRs and make informed decisions during feature selection and implementation.

C. Iterative Reengineering

The iterative nature of FDD allows for continuous evaluation and improvement of information system requirements. Through periodic reviews, the team can identify areas where NFRs need to be reengineered to enhance the performance, security, or usability of the information system. The outcome of this stage is the creation of new metadata an land mapping features in the new agricultural information system portal.

D. Collaboration and Communication

Effective collaboration among stakeholders, including business analysts, developers, and quality assurance teams, is crucial to ensure the integration of NFRs, BPM, and reengineered requirements in FDD. Ongoing communication facilitates shared understanding and minimizes the risk of misalignment.

IV. Result and Discussion

The results obtained are based on the sequence of steps described in Section III.

A. Identified NFRs

Four categories of NFRs were successfully identified, as shown in Fig. 1. These categories were grouped based on data collection and in accordance with NFR type identification standards [20]. The explanations for the four NFR categories used in this research are as follows:

System express desired quality characteristics associated with software as constraints associated with product and organizational aspects. The required system constraints include:

1) The system requires a farmer group.
2) The system requires autocomplete in selecting farmer groups.
3) The system requires the addition of farmer land ownership.
4) The system requires land ownership status.

Actor shows the desired quality attributes or constraints that related to resource users, departments. Other parties or companies that interacting with system constraints also represent actors. The obtained requirements are:

1) Up-to-date data display showing land ownership data.
2) Related display with the previous add land menu.

Data keeps the desired data quality attributes that used and present in information system. Data represents the information objects used and displayed in the information system. The following are the new data attributes obtained by the research team:

1) Land can be owned by multiple farmers.
2) Land ownership status can differ for each farmer, but one land must be owned by at least one farmer.
3) When deleting land data, it will also delete farmer ownership data, but deleting ownership data will not delete the land data itself.

External presents information system limits and describes policies, standards, and regulations identified by business category. External factors that influence system requirements are land status rules, such as private ownership, leasing and profit sharing, which are prevalent in most countries, especially in Asia.

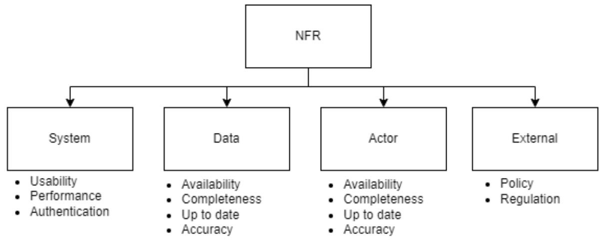

Fig. 1. Identified NFR categories.

After knowing what NFRs will be transformed into a new system, the next step is mapping the new business processes. To simplify the mapping and validation process, in the NFR mapping process an NFR map is first created as shown in Fig. 2.

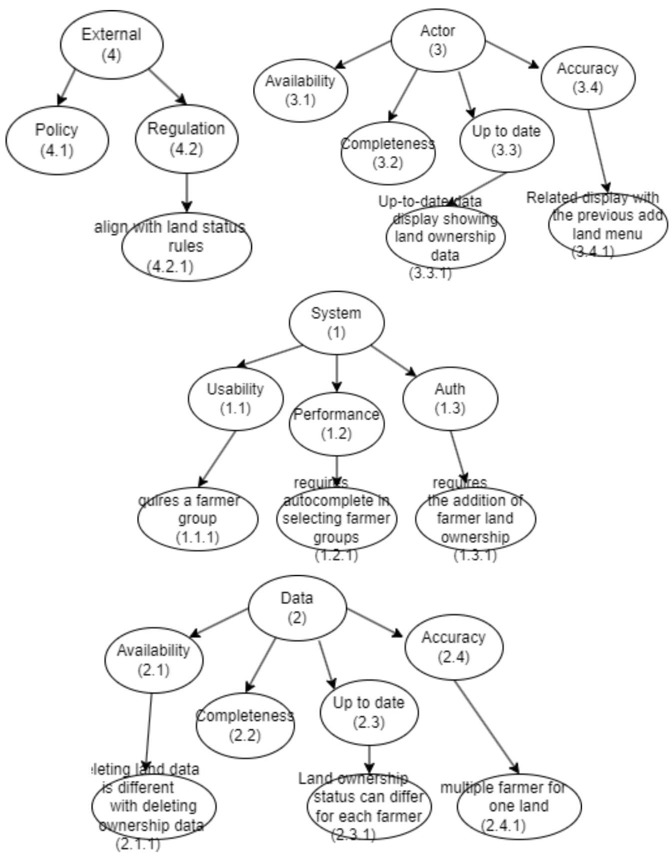

Fig. 2. NFRs mapping.

B. Business Process Mapping

Based on the identification of NFRs, a new business process for the land ownership management feature is obtained. The important point at this stage is to accommodate every need for NFRs Mapping into BPMN. So to make this process can be done easily, NFRs are represented by numbers that have been previously mapped, connected to tasks and BPMN. The process of mapping NFR into tasks and BPMN can be seen in Table II. It can be seen that tasks with brackets are new tasks adapted to NFRs. The new business process that related to the NFRs Mapping, as shown in Fig. 3. This task will later turn into a use case diagram in next step.

TABLE II. BUSINESS PROCESS TASK MAPPING

No	Tasks	Related NFRs
T1	Input land and farmer data (with group and ownership)	1.1.1, 1.3.1, 2.3.1, 2.4.1
T2	Validate land and farmer data (with status of ownership and multiple farmer)	2.1.1, 2.3.1, 3.4.1
T3	Manage land and farmer data (including deleting land that takes into land ownership status)	2.1.1, 4.2.1
T4	Display land data	2.3.1, 3.3.1, 3.4.1, 4.2.1,
T5	Search with autocomplete (new)	1.2.1

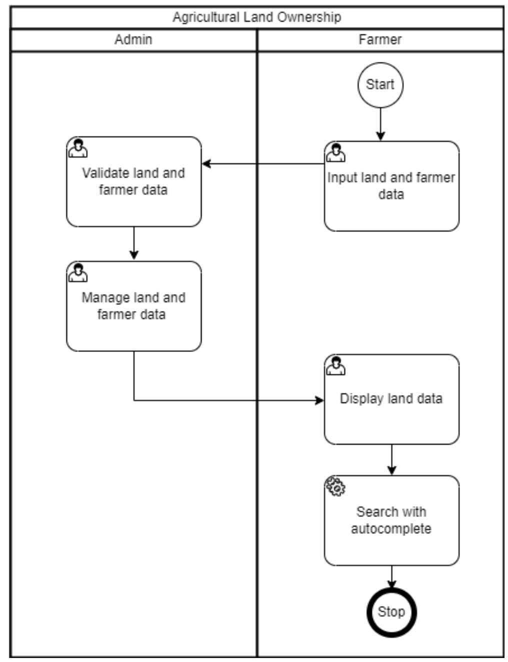

Fig. 3. BPMN of feature agricultural land ownership management.

The process flow begins when farmers input their data and land data independently. This is mandatory as land ownership status must be owned by at least one farmer. Once the farmer and land data are entered, the land ownership can be managed. The management process includes features for adding, modifying, and deleting land data. After the admin completes the data management, up-to-date data can be viewed by the respective farmers.

With the addition of a new business process, the use case is expanded due to the inclusion of the land management feature. The new use case can be seen in Fig. 4. Only one new use case is added, which is the Create Read Update Delete (CRUD) Agricultural Land, and this aligns with the Use Case diagram in the previous research [3], which had a total of 8 Use Cases.

C. Information System Reengineering Process

The next step involves designing new metadata based on the existing metadata. Considering the NFRs related to data, new metadata is required to accommodate land ownership status. Not only metadata, but the information system database also needs to undergo changes to accommodate the transaction table to handle land ownership dynamics.

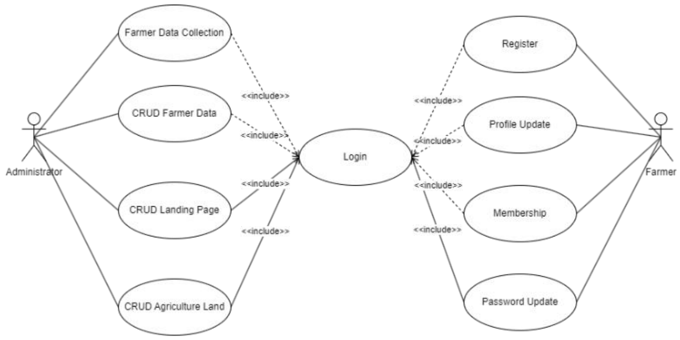

Fig. 4. New use case.

The farmer metadata remains unchanged, while the land data has new metadata due to the addition of the status attribute. The new metadata for agricultural land includes name, type, organic status, farmer group, central point, boundary point of the land, and many farmer object entries. Land ownership status is associated with farmer objects. Since one land data can be owned by multiple farmers, and one farmer data can have multiple land data, the database requires a new transaction table to accommodate the n:n relationship. Therefore, the latest database is depicted in Fig. 5. This paper only illustrates the relationships between farmers, land, and the land ownership transaction table.

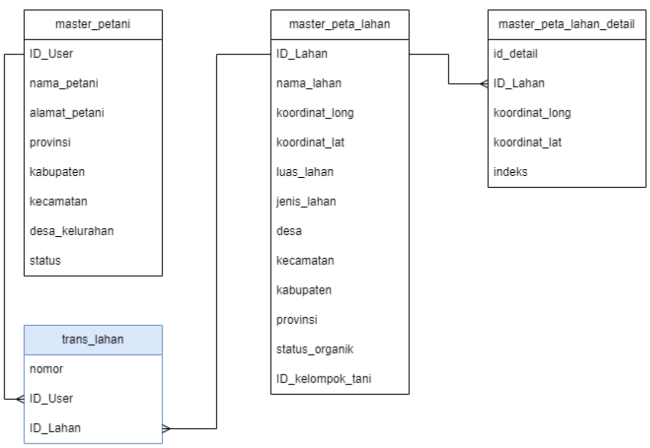

Fig. 5. The new model of physical data.

Based on physical data model above, the new metadata can be produced. New metadata can be seen in Fig. 6 at right side. The left side is former metadata from previous research in 2023 [21] that not contain multiple farmer land. Based on NFR that metadata need to be updated with multiple land ownership, new metadata in Fig. 6 already accomodated that needs.

FDD (Feature-Driven Development) is the next stage as the metadata and database are ready for use. In FDD, a complete list of features in land management will be registered. Additionally, the desired feature identified during NFR identification, which is the farmer group autocomplete, will also be included. The following is the complete list of features that will be added to the Dutatani agricultural information system. This feature already accommodate BPMN mapping from the previous stage which is marked with a task number in bracket

1) Farmer's land registry (T1)
2) Land search (T5)
3) Detailed land information (T4)
4) Land map display (T4)
5) Add owner/farmer (T1)
6) Detailed land ownership for each farmer (integrated with existing farmer and land registration features) (T1, T2)
7) Ownership modification feature (T3)
8) Ownership deletion feature (T3)

A total of 8 detailed features will be incorporated into agricultural land management, along with 1 autocomplete feature for farmer group search. The autocomplete feature is designed as shown in Fig. 7. Data will be retrieved in real-time and up-to-date from the database, in accordance with NFR criteria. Similarly, the design of the land registry and land search features can be observed in Fig. 8, showing the land ID and the number of land owners. These features are complemented with a detail button that links to features 3-8, as depicted in Fig. 9. Specifically, feature 6 will be directed to the farmer registration menu to accommodate integration with the existing registration feature.

Fig. 6. New metadata for farmer in IAIS.

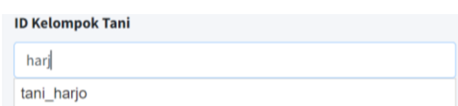

Fig. 7. Autocomplete on farmer's group search.

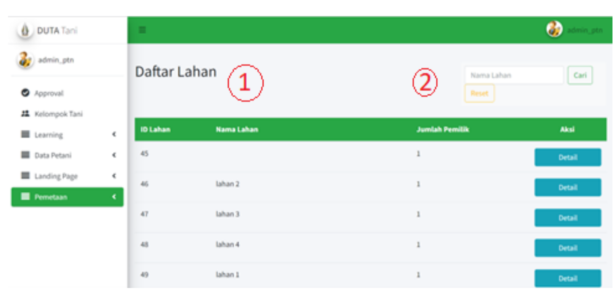

Fig. 8. The list of agricultural sites feature.

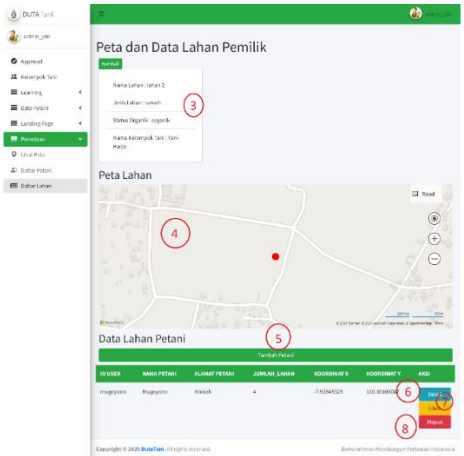

Fig. 9. Agricultural land data management feature.

In this study, the research team proposed an NFR-oriented classification to bridges the gap between the real world and information systems. The research team uses a BPMN model for the NFR classification to represent the business side and use case diagram to represent the information system side in the integrated agricultural information system case study.

Currently, previous studies have only classified or integrated NFRs for individual features either in real-world business processes or information systems. While there are many case studies on NFR and information systems, few have written about the sequence of classification, making it unclear when seeking the source of requirements. This has resulted in a gap between real-world business processes and information systems.

The proposed method in this study has resulted in:

1) NFR classification, each determined in relation to the requirements of the business process to be translated into the information system as either supporting or main requirements (system and data).

2) Higher number of NFRs compared to BPMN and use case diagram transformation results. This happened because the NFR side contains quality-related manual tasks (in addition to automated tasks), whereas the system side is controlled by the business and has only NFR-related automated tasks. This reduces the number of NFRs towards the BPMN and use case level.

3) A traceable flow diagram of the translation process between NFRs and the Information System. The diagram allows forward tracing (NFR to Metadata) and backward tracing (Metadata to NFR). This is applied to help map any business process requirements into the information system, and vice versa, when stakeholders decide to enhance the information system features. Additionally, the classification helps identify any NFR incompleteness between business processes and the information system by tracking each identified NFR in the business on the flow diagram. This can be seen in Fig. 10. There is a translation sequence that facilitates both forward and backward tracing.

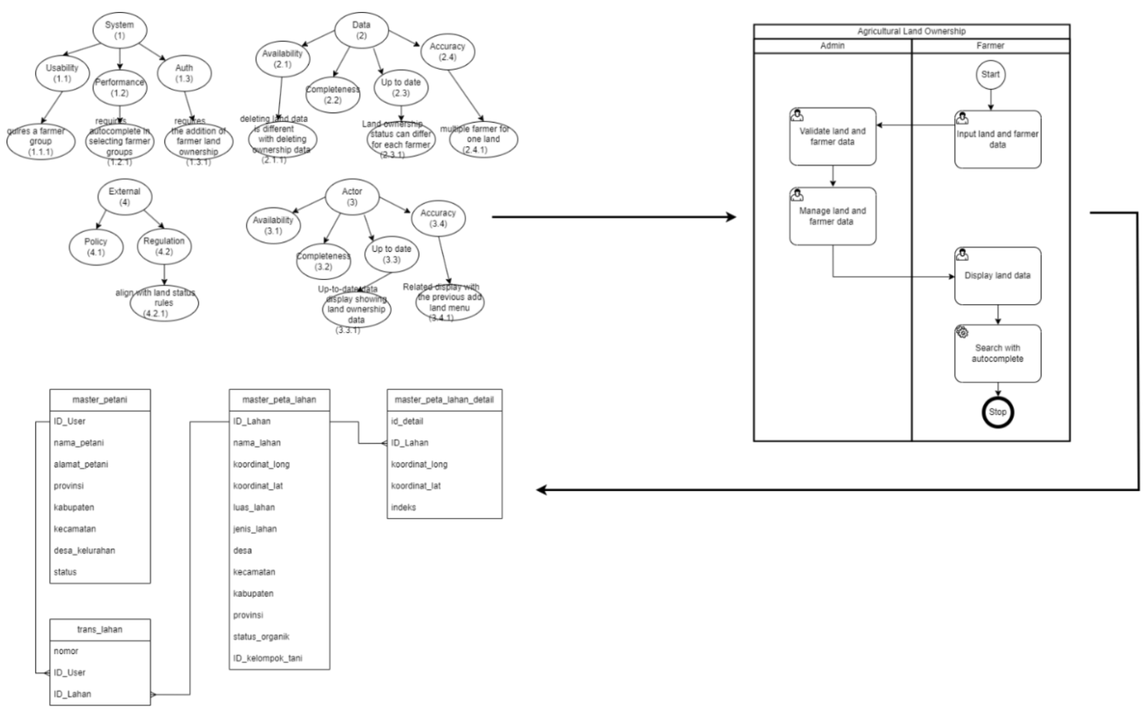

Fig. 10. Flowchart of information system requirements mapping.

The proposed method can perform good mapping between NFR, BPMN and also produce metadata. It has not been tested whether mapping can be done with other diagram notations. This can be studied in more depth with different cases because the purpose of each notation is different. The focus of this research is non-functional requirements that are compatible with BPMN's function, namely managing business process flows.

V. CONCLUSION

The study successfully identified and being bridge between the business process side and the information system side. The aim of this study is therefore to fill this gap through the proposed NFR-oriented classification and its translation into metadata, use case diagrams and functional design. NFRs can enrich the use case diagrams, metadata, and system features by integrating the successfully identified NFRs. The conclusions of this research are as follows:

1) NFRs variable can be integrated into use case diagrams and BPMN workflows. NFR classification that proposed in this study can strengthens the relationship between BPMN in business processes and its proper use in system metadata models.

2) Achieving forward and backward traceability from the business process to the information system model. The proposed classification has facilitated the mapping of NFRs from the business process into the information system. This contributes to tracking NFRs and system features. If NFRs can be tracked, it enables the development of features for future development in case of any NFR changes.

For future work, the research team plans to develop a matrix to connect NFRs to information system testing. Additionally, the matrix is expected to automatically assist in generating testing tables from BPMN to use case diagrams. Furthermore, the research team plans to investigate the effectiveness of NFRs on other system requirements such as load testing and system usability.

ACKNOWLEDGMENT

The authors would like to express their gratitude to Farmers in Ngudi Makmur and Mutiara for their support in this research and community service. The authors would also express their appreciation to the Faculty of Information Technology at Duta Wacana Christian University and the University's Institute for Research and Community Service (Lembaga Penelitian dan Pengabdian kepada Masyarakat/LPPM) for their support in this research and community service.

REFERENCES

[1] M. A. Gondal, N. A. Qureshi, H. Mukhtar, and H. F. Ahmed, “An engineering approach to integrate Non-Functional Requirements (NFR) to achieve high quality software process,” ICEIS 2020 - Proc. 22nd Int. Conf. Enterp. Inf. Syst., vol. 2, no. Iceis, pp. 377–384, 2020, doi: 10.5220/0009568503770384.

[2] A. R. Chrismanto, A. Wibowo, L. Chrisantyo, and M. N. A. Rini, “Implementasi Feature Driven Development untuk Mempermudah Ekualitas Fitur dan Adaptasi pada Pengembangan Portal Dutatani Web dan Mobile,” JEPIN (Jurnal Edukasi …, vol. 8, no. 1, pp. 62–73, 2022, [Online]. Available: https://jurnal.untan.ac.id/index.php/jepin/article/view/50715%0Ahttps://jurnal.untan.ac.id/index.php/jepin/article/viewFile/50715/75676592891.

[3] L. Chrisantyo, A. Wibowo, M. N. Anggiarini, and A. R. Chrismanto, “Blackbox Testing on the ReVAMP Results of The DutaTani Agricultural Information System,” in Proceedings of 11th International Congress on Advanced Applied Informatics, Sep. 2022, pp. 407–417, doi: https://doi.org/10.29007/1sx8.

[4] A. R. Chrismanto, H. B. Santoso, A. Wibowo, and R. Delima, “Studi Kelayakan Penerapan Web Mapping System Menggunakan Metode Telos (Studi Kasus : Kelompok Tani Harjo dan Rahayu),” in Seminar Nasional Dinamika Informatika, 2020, no. May, pp. 67–73.

[5] A. R. Chrismanto, R. Delima, H. B. Santoso, A. Wibowo, and R. A. Kristiawan, “Developing agriculture land mapping using Rapid Application Development (RAD): A case study from Indonesia,” Int. J. Adv. Comput. Sci. Appl., vol. 10, no. 10, pp. 232–241, 2019, doi: 10.14569/ijacsa.2019.0101033.

[6] M. Rondhi and A. H. Adi, “The Effects of Land Ownership on Production, Labor Allocation, and Rice Farming Efficiency,” J. Agribus. Rural Dev. Res., vol. 4, no. 2, 2018.

[7] C. Kartiko, A. C. Wardhana, and W. A. Saputra, “Requirements Engineering of Village Innovation Application Using Goal-Oriented Requirements Engineering (GORE),” J. Infotel, vol. 13, no. 2, pp. 38–46, 2021, doi: 10.20895/infotel.v13i2.602.

[8] F. Baskoro, R. A. Andrahsmara, B. R. P. Darnoto, and Y. A. Tofan, “A Systematic Comparison of Software Requirements Classification,” IPTEK J. Technol. Sci., vol. 32, no. 3, p. 184, 2021, doi: 10.12962/j20882033.v32i3.13005.

[9] A. F. B. Arbain, D. N. A. Jawawi, W. M. N. Bin Wan Kadir, and I. Ghani, “Requirement traceability model for agile development: Results from empirical studies,” Int. J. Innov. Technol. Explor. Eng., vol. 8, no. 8, pp. 402–405, 2019.

[10] I. Sommerville, Software engineering (10th edition), Tenth Edit. Pearson Education, 2016.

[11] E. K. Budiardjo and W. C. Wibowo, “Slr on Identification & Classification of Non-Functional Requirements Attributes , and Its Representation in Functional Requirements,” in International Conference on Computer Science and Artificial Intelligence (CSAI), 2018, pp. 151–157.

[12] P. Kochovski, P. D. Drobintsev, and V. Stankovski, “Formal Quality of Service assurances, ranking and verification of cloud deployment options with a probabilistic model checking method,” Inf. Softw. Technol., vol. 109, no. April 2018, pp. 14–25, 2019, doi: 10.1016/j.infsof.2019.01.003.

[13] S. Sumesh, A. Krishna, C. M. Subramanian, and F. Murtagh, “Game Theory-Based Reasoning of Opposing Non-functional Requirements using Inter-actor Dependencies,” Comput. J., vol. 62, no. 11, pp. 1557–1583, 2019, doi: 10.1093/comjnl/bxy143.

[14] B. H. C. C. B. DeVries, “Goal-Based Modeling and Analysis of Non-Functional Requirements,” in ACM/IEEE 22nd International Conference on Model Driven Engineering Languages and Systems, 2019, pp. 261–271.

[15] R.. Gnanasekaran, S. Chakraborty, J. Dehlinger, and L. Deng, “Using recurrent neural networks for classification of natural language-based non-functional requirements,” 2021.

[16] N. Handa, A. Sharma, and A. Gupta, “Framework for prediction and classification of non functional requirements: a novel vision,” Cluster Comput., vol. 25, no. 2, pp. 1155–1173, 2022.

[17] K. M. Bowers, E. M. Fredericks, R. H. Hariri, and B. H. C. Cheng, “Providentia: Using search-based heuristics to optimize satisficement and competing concerns between functional and non-functional objectives in self-adaptive systems,” J. Syst. Softw., vol. 162, 2020.

[18] J. Zubcoff, I. Garrigós, S. Casteleyn, J. Mazón, J. Aguilar, and F. Gomariz-castillo, “Evaluating different i ∗ -based approaches for selecting functional requirements while balancing and optimizing non-functional requirements : A controlled experiment,” Inf. Softw. Technol., vol. 106, 2019.

[19] Z. Zhou, Q. Zhi, S. Morisaki, and S. Yamamoto, “An Evaluation of Quantitative Non-Functional Requirements Assurance Using ArchiMate,” IEEE Access, vol. 8, 2020.

[20] H. Kaur, “NFR Types and its definitions NFR ’ s : Definition,” no. February, 2022.

[21] M. N. A. Rini, A. Wibowo, L. Chrisantyo, and A. R. Chrismanto, “Requirement validation approach using model and prototyping on agriculture information system case study: Dutatani agriculture information system,” AIP Conf. Proc., vol. 2508, 2023, doi: 10.1063/5.0130311.