DECODE: JURNAL PENDIDIKAN TEKNOLOGI INFORMASI
P-ISSN: 2775-2984 | E-ISSN: 2775-1813

Vol. 4, No 3, November 2024, Hal: 898-914
Doi: http://dx.doi.org/10.51454/decode.v4i3.752
https://journal.umkendari.ac.id/index.php/decode

Implementation Of Dynamic Role-Based Access Control and Record Log-In Dependency Injection Based VRMS System

Astrid Noviana Paradhita[1*], Mutiara Auliya Khadija[1], Agus Purbayu[1], Sahirul Alim Tri Bawono[1], Abdul Aziz[1], Sri Haryati[2], Daaniis Raditya Pramoda Wardani[1], Rafael Kurniawan Albyseptra Pratama[1]

[1]Vocational School, Universitas Sebelas Maret, Indonesia.
[2]English Education, Universitas Sebelas Maret, Indonesia.

Article Info

Keywords:

Dependency Injection;
Dynamic Role-Based Access Control;
Human Resource Management System;
Hybrid Agile;
Record Log.

Article History:

Submitted: August 28, 2024
Accepted: October 12, 2024
Published: October 29, 2024

Corresponding Author:
Astrid Noviana Paradhita
Email: astrid.noviana@staff.uns.ac.id

Abstract: Human resource management is one of the critical factors for a company in order to realize the company's vision, mission, and goals. This requires effective and efficient human resource management by integrating the entire management process well. This is to manage the company's internal and external human resources. VRMS is an integrated application that can manage the company's external resources. VRMS is built based on dependency injection. VRMS has two superior features: role-based access control (RBCA) and record log. RBCA is a feature that manages all resources involved in a company project. At the same time, the record log allows super admins to monitor and evaluate the performance of all external company resources. The VRMS system is built using hybrid methodoly in software development life cycles from initiation phase, planning, execution, and testing. The testing process used in this system is the user acceptance test (UAT) method. The test results show that all features on VRMS can run 80% well.

INTRODUCTION

Human resource management is one of the critical factors for companies in realizing the company's vision, mission, and goals. It is undeniable that the view of human resource management applied by companies is constantly undergoing changes that are needed to adapt to changes in the environment, both internal and external (Boon et al., 2019). In another opinion, Abuhantash, 2023) said that the company's increased business scale can also influence changes in human resource management. The era's development requires every company to survive, not only in domestic competition but also in global competition (Diallo et al., 2021). This requires effective and efficient human resource management by properly integrating all management processes.

Human resource integration can be done by utilizing technological developments. In general, large companies will implement special procedures to monitor and evaluate the performance of their human resources (Apascaritei & Elvira, 2022). This is done to assess whether the human resources have carried out their duties and responsibilities properly. Human resources who can carry out their duties and responsibilities with full awareness are sure to be able to lead the company to success. Conversely, if the human resources department needs awareness or even a sense of responsibility for the work it has to do, it can lead the company to success (Jagarwar, 2022).

Integrated human resource management provides benefits not only for the company but also for employees. Companies with good human resource management can guarantee the careers of their employees (International Labour Organization, 2023). Employees can get various benefits ranging from performance points and career advancement to salary increases based on their performance. Beer (2022) also added that transparent, integrated human resource management could improve the quality of products and services owned by the company to increase consumer reach to foreign countries.

Nowadays, companies can utilize many types of information technology to manage their human resources more optimally. This technology is often called a human resource management system (HRMS). HRMS will generally have several features, such as attendance, administrative filing, payroll, leave, out-of-town travel, and so on (El Idrissi et al., 2021). These features can be integrated into the company's system and information technology, which policymakers can access to assess their employees' performance. This can be well controlled to manage the internal resources owned by the company. However, today, many companies also have external company resources. This means that the employees they have not only work within the company but are also spread across various regions worldwide to work on specific projects. HRMS has yet to have a prominent feature specifically designed to manage human resources from outside the company (Osezua & Nkogbu, 2016).

As an international company with its parent company located in Switzerland, the company relies heavily on cloud computing technology to coordinate with each other. However, the company still often needs help managing its human resources. The obstacles experienced by PT Star Software are related to the absence of a log record of its vendors' performance and external resources. PT Star Software must refrain from dealing with and resolving problems caused by irresponsible vendors infrequently. In addition, PT Star Software also experiences delays in project completion because of the external resources it has been using for several other projects. Therefore, PT Star Software requires an integrated system to monitor and evaluate the performance of its vendors and external resources.

VRMS is one of the solutions used to manage the company's external human resources. VRMS stands for Vendor Resource Management System. VRMS is a system that manages, assesses, and records the company's external human resources performance. Several studies have been conducted previously to manage the company's external resources. The first study was conducted by Edwards et al. (2022), who examined the strategy of managing international resources in multinational companies. This study concluded that international resource management needs to be accompanied by contextualized, personalized, and contested standardization. The second study was conducted by Hamouche (2021), which examined the company's human resources management after the Covid-19 crisis. From this study, it can be seen that to ensure that business activities in the company continue to run smoothly, the company needs to transform human resource management. This management includes human resources management, which is carried out directly in and outside the office. Another study was

conducted by Pattrick & Mazhar (2019), who studied human resource management by implementing a talent search strategy. This study concludes that skills and talents are the key to optimizing business processes in the company. Companies can take on roles in business based on their internal talent and offer projects to other professional talents outside of the company's internal employees.

Based on the research that has been conducted, the company needs to find talent outside the company to run business activities. However, previous research has shown that there needs to be more focus on developing an integrated system to manage external human resources. Therefore, this study was specifically conducted to develop a VRMS system with the main features of managing vendors and the company's external human resources. VRMS development is carried out by utilizing the backend module. The citation states that the backend module is one of the development modules in computer technology that develops a system or application that users cannot see on a running system (Bangare et al., 2016). The backend module has several advantages, such as ensuring that the system algorithm can run adequately, ensuring optimal performance, and making it easier for developers to troubleshoot and debug (Kaluža et al., 2019).

METHOD

This research was developed by adopting the system development stages of the backend module. The VRMS system was developed to have two main features: administrator system management called Role Based Access Control (RBAC) and the record log feature. The administrator feature is implemented dynamically to manage role-based user access. RBAC in the VRMS system has the flexibility to adjust to needs without changing the program code that has been built (Marquis, 2024).

Meanwhile, the VRMS system has a record log feature to see how good the qualifications and performance of its external resources are. The record log feature has the main function of automatically recording every activity carried out by vendors and human resources without making data changes. This feature applies the concept of dependency injection. Dependency injection by Deursen & Seemann (2019) is defined as a particular module linked to several predetermined parameters to monitor specific variables. In this study, dependency injection is used to assess the performance of vendors and human resources based on assessments, project completion accuracy, and testimonials.

This research was conducted by combining the hybrid Agile method. Yahya & Maidin (2023) said that the hybrid agile method is a software development method that combines agile and waterfall methods. The agile method is developing software carried out repeatedly in stages (Žužek et al., 2020). The waterfall method is a software development method carried out sequentially and sequentially (Imani et al., 2017). The combination of agile and waterfall methods makes developing the VRMS system more effective and efficient. This opinion is supported by Nikpay et al. (2017) stating that the hybrid method can overcome the flexibility weakness of waterfall and the weakness in agile that cannot document milestones well. Hybrid method carried out separately to develop specific modules combined into one VRMS system that can run well. Figure 1 shows the flow of methods used in this research.

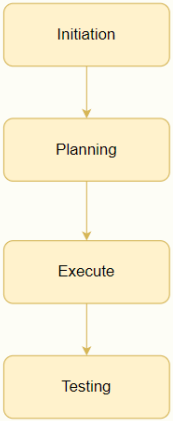

Figure 1. Method Flow

Initiation

Initiation is the initial stage in system development, which includes developing ideas and collecting mature requirements (Verma et al., 2021). From the initiation stage, researchers gain insight into the VRMS system to be built. Researchers also determine the main functions and features of the system, namely RBCA and record log.

Planning

The planning stage is the stage for processing ideas and requirement data obtained for research purposes (Senarath, 2021). At this stage, researchers create several models to facilitate system development at the execution stage. The models created in the planning stage are use case diagrams, activity diagrams, class diagrams, entity relationship diagrams, and sitemaps.

A use case diagram is a diagram that can show system users and the access rights owned by each user (Setiaji & Sastra, 2021). From this diagram, application developers can create and organize the types of users required by the system, along with the features that specific users can access. Next is the activity diagram, which describes a system's workflow and sequence of activities (Kurniawan, 2018). In other words, the activity diagram describes the activities that users can carry out and the responses from the system.

Another diagram developed at the planning stage is the class diagram. A class diagram is a diagram that shows the data model needed in system development (Ahmad et al., 2019). Class diagrams can provide developers with an overview of data, data types, data types, and the functions of the data. Researchers can build an entity relationship diagram (ERD) from the modeled class diagram. ERD is a diagram that describes data, attributes, relationships, and relationships between data (Tabrani et al., 2021). Next, create a sitemap. A sitemap is a map that describes essential pages in the system to be built (Ríos & Pedreira-Souto, 2019). Sitemaps have several functions, such as facilitating navigation, speeding up indexing, emphasizing the site's direction, organizing the system's architecture, and increasing system visibility.

Execute

The next stage in this study is execution. The execution stage is the programming implementation stage, which is when program code is built to create a system (Dhivya & Nirmala, 2018). This study's programming was carried out using Nest js, Postgre SQL, and JSON Web Token.

Testing

The next stage is testing. The system will be tested in the testing stage based on several aspects, such as user acceptance, unit, component, smoke, integration, regression, sanity, and system testing (Dhivya & Nirmala, 2018). In the study, the system was tested using the user acceptance test (UAT) method. UAT is a type of testing that checks whether the features and displays on the system meet the user's requirements. In addition, UAT also ensures that all buttons and features on the system can run properly (Chamida et al., 2021).

RESULTS AND DISCUSSIONS

Requirement

Researchers surveyed several stakeholders in developing the VRMS system, such as vendors, job seekers, and vocational education universities. The survey was conducted by distributing questionnaires to 10 stakeholders, 20 respondents from the career development field of Universitas Sebelas Maret, and 20 students at the vocational school, Universitas Sebelas Maret. From the total survey, a complete response was obtained from 6 stakeholders, 13 respondents in the field of career development at Sebelas Maret University, and 20 students. From the survey results, we see the need for the VRMS system. Details of system requirements are shown in Table 1.

Table 1. Functional Requirement

Kode	Functional Requirement	Actor
FR001	Provides an endpoint for login returning JWT token	Superadmin, *Admin, User*
FR002	Provides an endpoint to delete JWT tokens on logout	Superadmin, *Admin, User*
FR003	Provides endpoints and data to display a list of data on the System Administrator page with search, pagination, and sorting filters.	Superadmin, *Admin, User*
FR004	Provides endpoints for adding new accounts and storing account data on the System Administrator page.	Superadmin
FR005	Provides dynamic access control mechanisms based on user roles and stores access settings in a database.	Superadmin
FR006	Provides endpoints for editing account details and updating account data in the database on the System Administrator page.	Superadmin
FR007	Provides an endpoint to delete accounts and associated data from the database on the System Administrator page	Superadmin
FR008	Provides endpoints and data to display log records with search, pagination, sort, menu filter, and date filter features.	Superadmin

Planning

Based on the functional needs of the VRMS system, researchers can create several designs and models to facilitate the development of the VRMS system during the execution stage. The first design is the use case diagram shown in Figure 2.

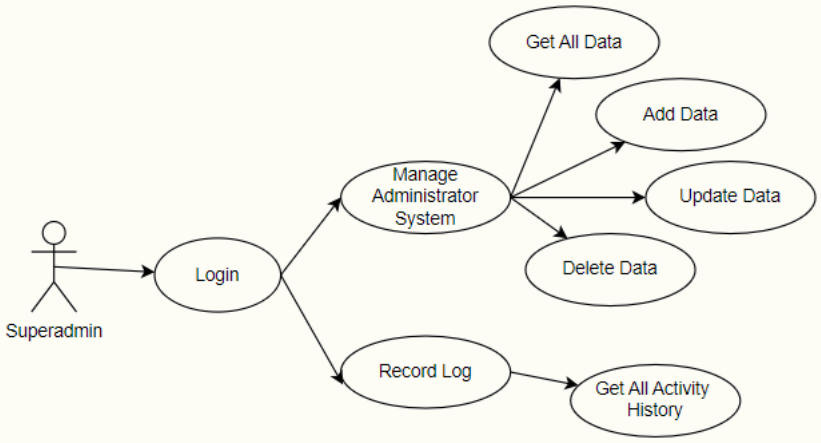

Figure 2. Use Case Diagram

Figure 2 shows that after logging in to the VRMS system, the super admin has access rights to manage the administrator management feature. The super admin can do administrator management to display all administrator data and add new, update, and delete administrator data. In addition, the super admin also has access rights to record all information activities and existing data transactions. Administrators in VRMS are vendors and human resources involved in all business projects run directly or indirectly by PT Star Software. Furthermore, the researcher also modeled an activity diagram for developing the VRMS system in the study. The activity diagram consists of several activities, such as

login and getting all users, as shown in Figure 3. In addition, the design of adding users (administrators), editing users, and deleting users is shown in Figure 4, and getting record logs is shown in Figure 5.

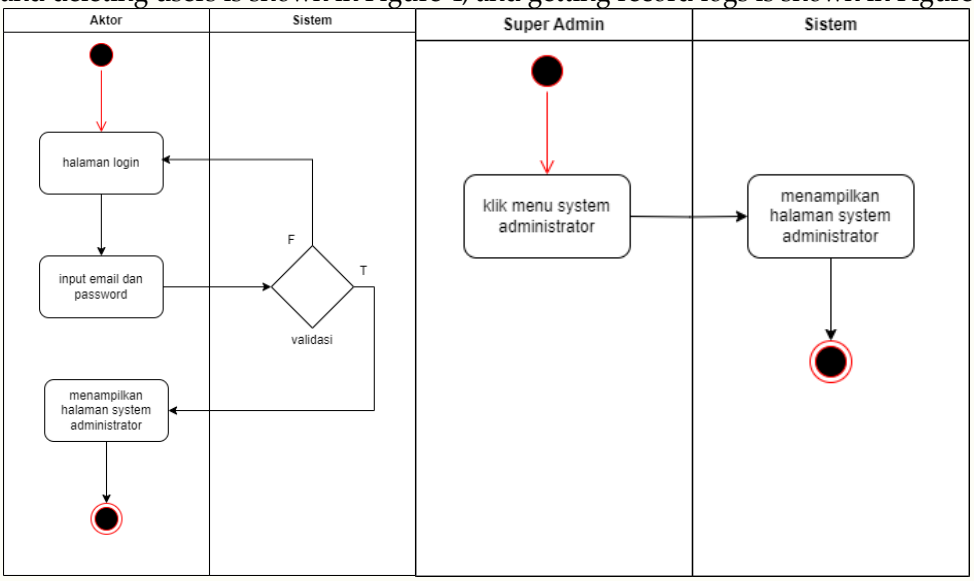

Figure 3. Login (left), Show homepage (right)

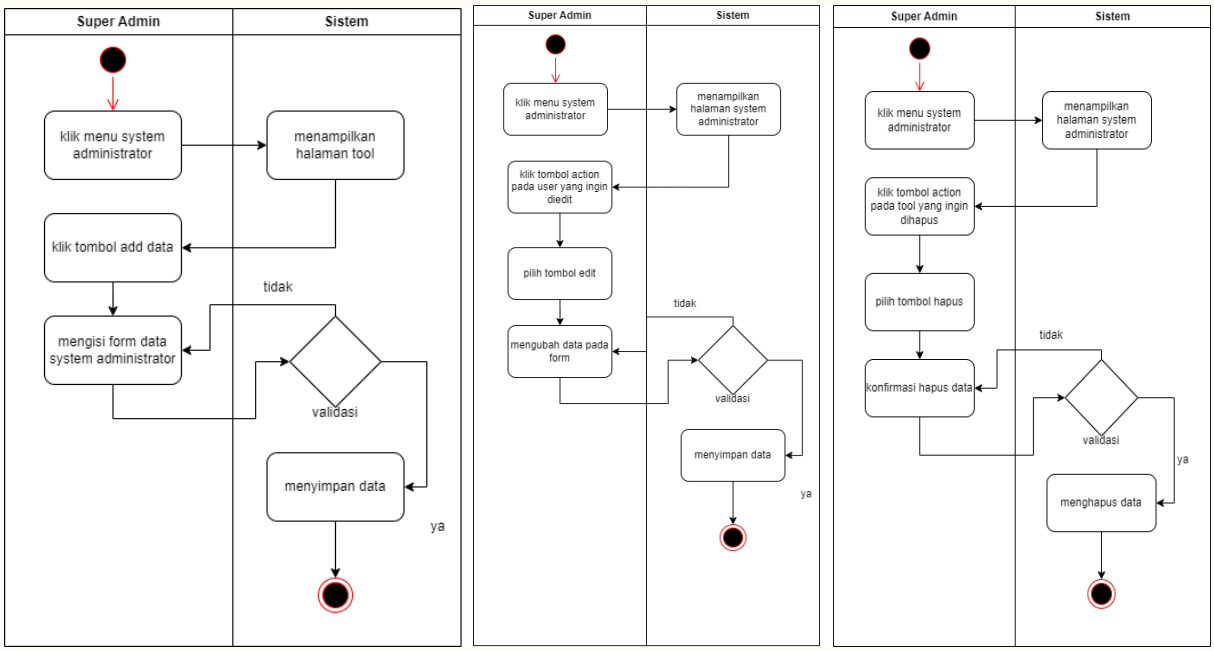

Figure 4. Add user (left), edit user (center), and delete user (right)

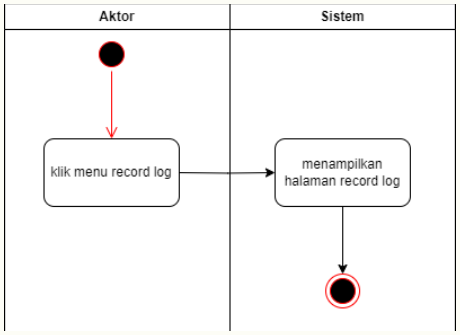

Figure 5. Get All Log

By modeling activity diagrams, researchers build class diagrams as data models that will be used to model databases on the VRMS system. The class diagram design is shown in Figure 6.

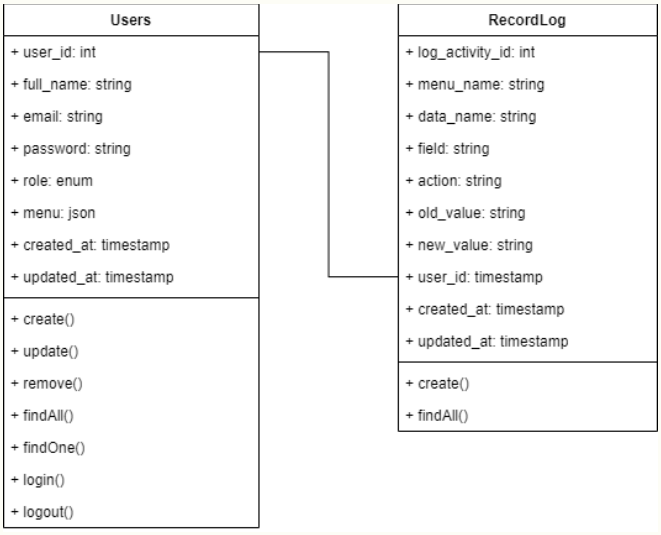

Figure 6. Class Diagram

The next step in the planning stage carried out in this study is to model the database design, which is described by an entity relationship diagram (ERD). The ERD in the VRMS system is shown in Figure 7.

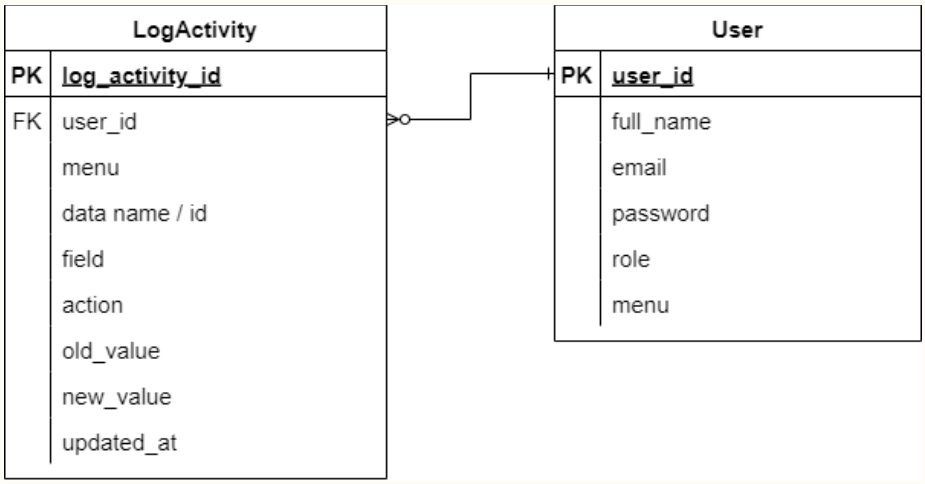

Figure 7. Entity Relationship Diagram

The final step in the planning stage is to build a sitemap. The sitemap in the VRMS system describes the data and information needed by the super admin to monitor and evaluate all business activities and transactions. The sitemap design in the VRMS system is shown in Figure 8.

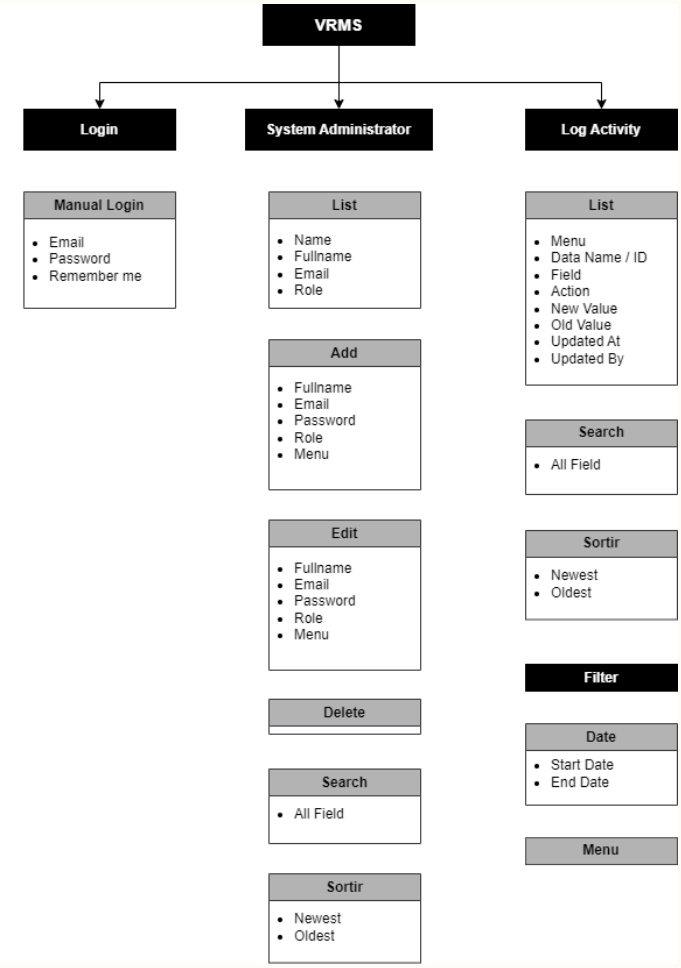

Figure 8. Sitemap VRMS

Execution

The next stage is to execute all models and designs that have been built in the planning stage. After executing the program code, the superadmin display is obtained as a way to log in as shown in Figure 9.

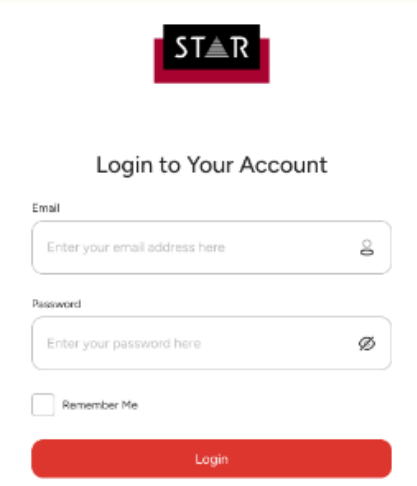

Figure 9. Login View

After logging in, the super admin will have a homepage display to select several needed features. The super admin homepage display is shown in Figure 10.

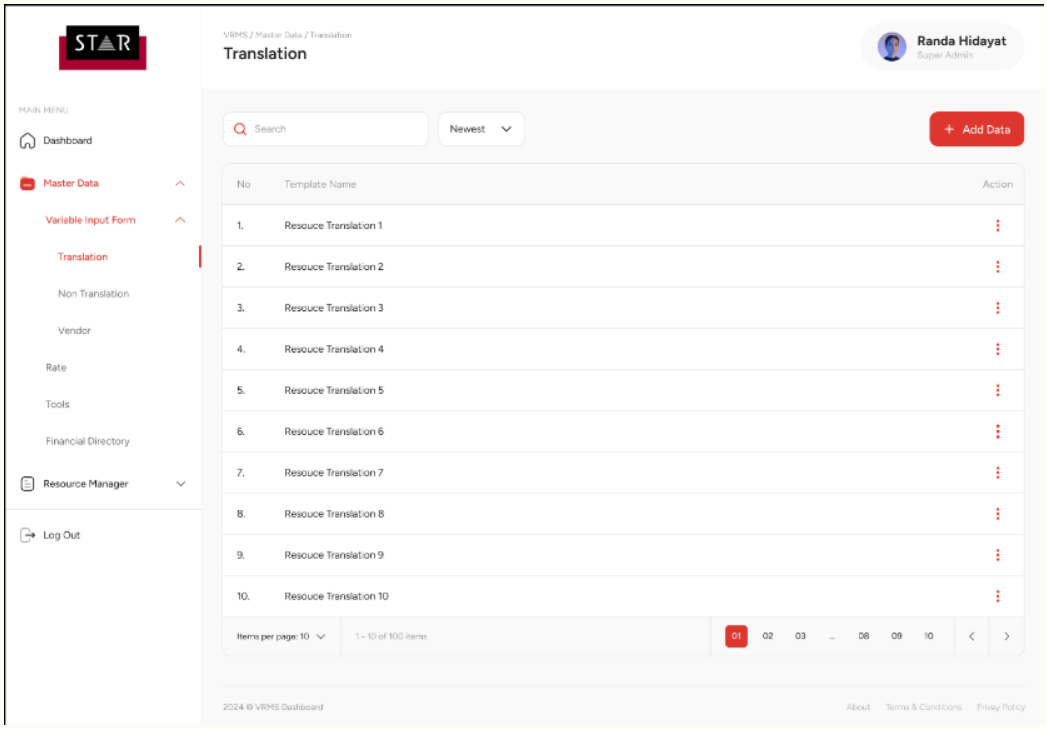

Figure 10. Super admin Home

Figure 10 shows the VRMS system display when the superadmin successfully logs in after entering the correct username and password. On the left side, there is a navigation menu showing the features the super admin owns to manage users. The first feature is adding users, as shown in Figure 11.

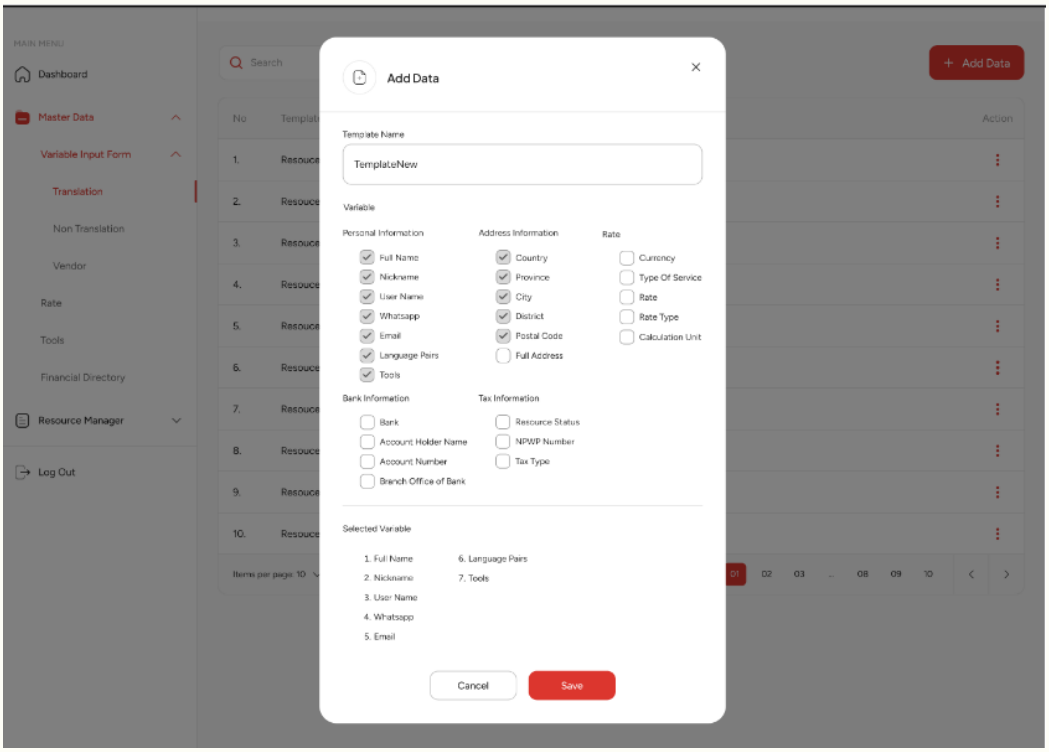

Figure 11. Add User

If there is a change in user data, the super admin also has the authority to edit and delete data if the user data is not validated. The superadmin display for editing data is shown in Figure 12, while the feature for deleting user data is shown in Figure 13.

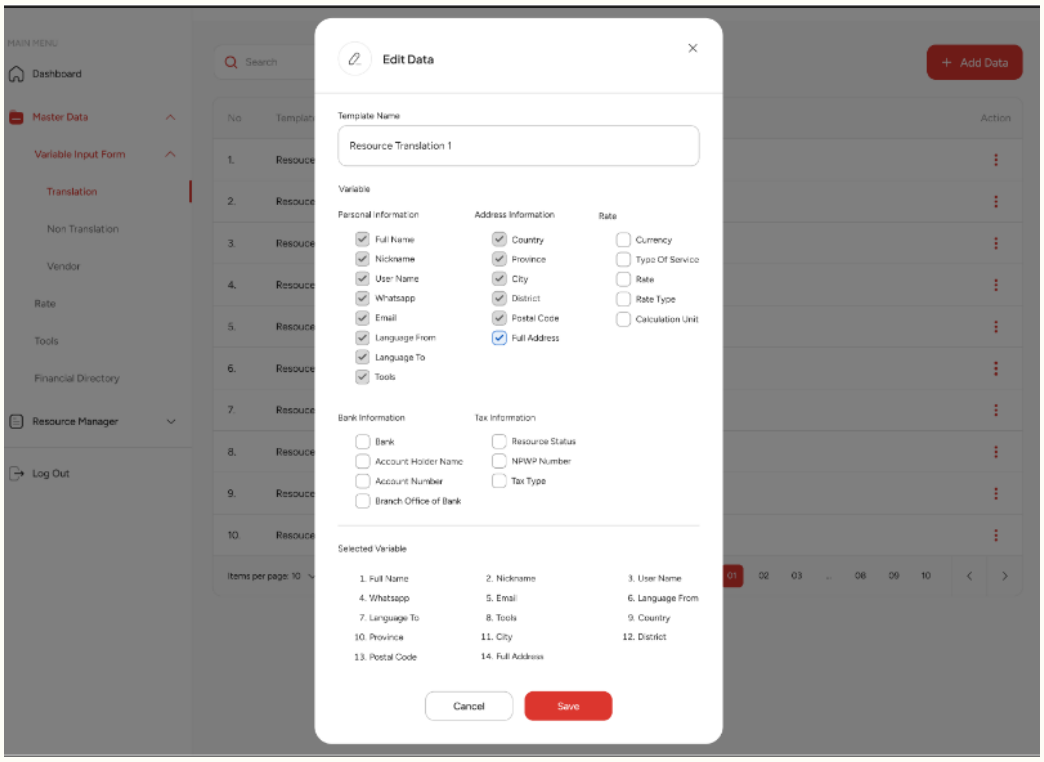

Figure 12. Edit User

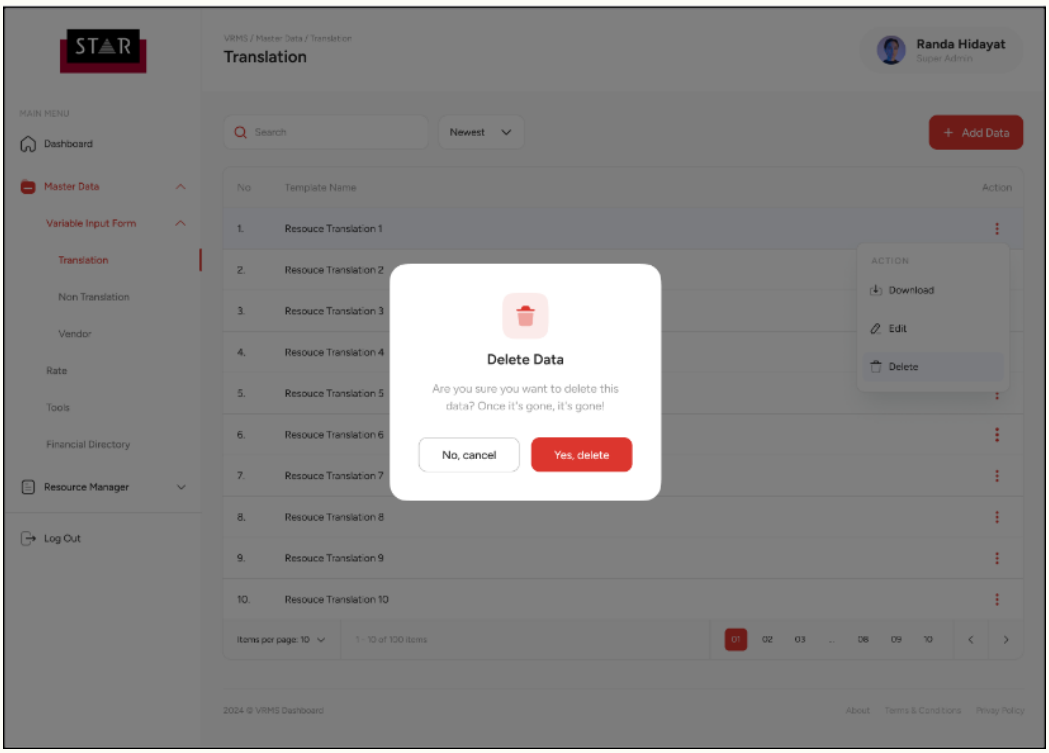

Figure 13. Delete User

This unique feature owned by the super admin will show the RBCA feature on the built VRMS system. The super admin settings illustrate that RBCA can add, edit, and delete user data if needed. Meanwhile, the record log is another superior feature to monitor and evaluate users.

Testing

The testing method used by the author is the User Acceptance Test (UAT) method. The User Acceptance Test method is a software acceptance test carried out at the user's location by the software user. The user of this system consists of vendors managed by PT Digital Software Yogyakarta, the career development field of Universitas Sebelas Maret, and several students from several study programs at Universitas Sebelas Maret. The total number of users who tested this system was 37 users.This test aims to test whether the system is in accordance with what is stated in the system's functional specifications. The results of the tests carried out for the master data menu can be seen in Table 2.

Table 2. UAT Result

Tested Function	Scenario	Result	Status
Menu add data Variable Input Form - Translation	Check the add function in the Variable Input Form - Translation menu	Data added successfully	Succeed
Menu add data - Variable Input Form - Translation (Check Error)	Check Mandatory Input	The data was not added successfully and an alert appears	Succeed
Menu edit data Variable Input Form - Translation	Check the edit function in the Variable Input Form - Translation menu	Data changed successfully	Succeed
Menu edit data Variable Input Form - Translation (Check Error)	Check Mandatory Input	The data was not successfully changed and an alert appears	Succeed
Menu delete data Variable Input Form - Translation	Check the delete function on the Variable Input Form - Translation menu	Data deleted successfully	Succeed
Menu sort data Variable Input Form - Translation	Check the sort function on the Variable Input Form - Translation menu	Data has been successfully sorted	Succeed
Menu search data Variable Input Form - Translation	Check the search function on the Variable Input Form - Translation menu	Data successfully found	Succeed
Menu export data Variable Input Form - Translation	Check the export function in the Variable Input Form - Translation menu	Data downloaded successfully	Succeed
Menu pagination data Variable Input Form - Translation	Check the pagination function in the Variable Input Form - Translation menu	Pagination was successful	Succeed
Menu add data Variable Input Form - Non Translation	Check the add function on the Variable Input Form - Non Translation menu	Data added successfully	Succeed

Tested Function	Scenario	Result	Status
Menu add data - Variable Input Form - Non Translation (Check Error)	Check Mandatory Input	The data was not added successfully and an alert appears	Succeed
Menu edit data Variable Input Form - Non Translation	Check the edit function on the Variable Input Form - Non Translation menu	Data changed successfully	Succeed
Menu edit data Variable Input Form - Non Translation (Check Error)	Check Mandatory Input	The data was not successfully changed and an alert appears	Succeed
Menu delete data Variable Input Form - Non Translation	Check the delete function on the Variable Input Form - Non Translation menu	Data deleted successfully	Succeed
Menu sort data Variable Input Form - Non Translation	Check the sort function on the Variable Input Form - Non Translation menu	Data has been successfully sorted	Succeed
Menu search data Variable Input Form - Non Translation	Check the search function on the Variable Input Form - Non Translation menu	Data successfully found	Succeed
Menu export data Variable Input Form - NonTranslation	Check the export function on the Variable Input Form - Non Translation menu	Data downloaded successfully	Succeed
Menu pagination data Variable Input Form - Non Translation	Check the pagination function on the Variable Input Form - Non Translation menu	Pagination was successful	Succeed
Menu add data Variable Input Form - Vendor	Check the add function on the Variable Input Form - Vendor menu	Data added successfully	Succeed
Menu add data - Variable Input Form - Vendor (Check Error)	Check Mandatory Input	The data was not added successfully and an alert appears	Succeed
Menu edit data Variable Input Form - Vendor	Check the edit function on the Variable Input Form - Vendor menu	Data changed successfully	Succeed
Menu edit data Variable Input Form - Vendor (Check Error)	Check Mandatory Input	The data was not successfully changed and an alert appears	Succeed
Menu delete data	Check the delete function on	Data deleted	Succeed

Tested Function	Scenario	Result	Status
Variable Input Form - Vendor	the Variable Input Form - Vendor menu	successfully	
Menu sort data Variable Input Form - Vendor	Check the sort function on the Variable Input Form - Vendor menu	Data has been successfully sorted	Succeed
Menu search data Variable Input Form - Vendor	Check the search function on the Variable Input Form - Vendor menu	Data successfully found	Succeed
Menu export data Variable Input Form - Vendor	Check the export function on the Variable Input Form - Vendor menu	Data downloaded successfully	Succeed
Menu pagination data Variable Input Form - Vendor	Check the pagination function on the Variable Input Form - Vendor menu	Pagination was successful	Succeed
Menu add data Rate Type	Check the add function in the Rate Type menu	Data added successfully	Succeed
Menu add data - Rate Type (Check Error)	Check Mandatory Input	The data was not added successfully and an alert appears	Succeed
Menu edit data Rate Type	Check the edit function in the Rate Type menu	Data changed successfully	Succeed
Menu edit data Rate Type (Check Error)	Check Mandatory Input	The data was not successfully changed and an alert appears	Succeed
Menu delete data Rate Type	Check the delete function in the Rate Type menu	Data deleted successfully	Succeed
Menu sort data Rate Type	Check the sort function in the Rate Type menu	Data has been successfully sorted	Succeed
Menu search data Rate Type	Check the search function in the Rate Type menu	Data successfully found	Succeed
Menu pagination data Rate Type	Check the pagination function in the Rate Type menu	Pagination was successful	Succeed
Menu add data Tools	Check the add function in the Tools menu	Data added successfully	Succeed
Menu add data - Tools (Check Error)	Check Mandatory Input	The data was not added successfully and an alert appears	Succeed

Tested Function	Scenario	Result	Status
Menu edit data Tools	Check the edit function on the Tools menu	Data changed successfully	Succeed
Menu edit data Tools (Check Error)	Check Mandatory Input	The data was not successfully changed and an alert appears	Succeed
Menu delete data Tools	Check the delete function in the Tools menu	Data deleted successfully	Succeed
Menu sort data Tools	Check the sort function in the Tools menu	Data has been successfully sorted	Succeed
Menu search data Tools	Check the search function in the Tools menu	Data successfully found	Succeed
Menu pagination data Tools	Check the pagination function on the Tools menu	Pagination was successful	Succeed
Menu add data Financial Directory	Check the add function on the Financial Directory menu	Data added successfully	Succeed
Menu add data - Financial Directory (Check Error)	Check Mandatory Input	The data was not added successfully and an alert appears	Succeed
Menu edit data Financial Directory	Check the edit function on the Financial Directory menu	Data changed successfully	Succeed
Menu edit data Financial Directory (Check Error)	Check Mandatory Input	The data was not successfully changed and an alert appears	Succeed
Menu delete data Financial Directory	Check the delete function on the Financial Directory menu	Data deleted successfully	Succeed
Menu sort data Financial Directory	Check the sort function on the Financial Directory menu	Data has been successfully sorted	Succeed
Menu search data Financial Directory	Check the search function on the Financial Directory menu	Data successfully found	Succeed
Menu pagination data Financial Directory	Check the pagination function in the Financial Directory menu	Pagination was successful	Succeed
Menu download file Financial Directory	Check the file download function on the Financial Directory menu	File downloaded successfully	Succeed
Menu delete file Financial Directory	Check the delete file function in the Financial Directory	File deleted successfully	Succeed

Tested Function	Scenario	Result	Status
	menu		

Table 2 shows the results of testing the VRMS system using the UAT method. The test results shows that all scenario and function of the VRMS can run sesuai dengan kebutuhan dari pengguna. This shows that the VRMS system performs quite well. Based on the test, VRMS provides 100% system performance. Although all functions in the system are already running according to the request from the user, the quality of backend programming in VRMS development still needs to be improved.

CONCLUSIONS

This research successfully produced VRMS to manage integrated human resources based on dynamic role-based access control and record logs. The dependency injection basis successfully provides convenience in real-time managing, monitoring, and assessing all VRMS users. From the simulation results, the system can centrally manage all external resources, both vendors and talents, by the super admin. The system can also record the activities of all external resources to provide ratings for both vendors and talents based on the activities that have been carried out. The system was tested using the User Acceptance Testing method, which found that 100% of the system's features ran well. So, although it has achieved good system performance, the quality of backend programming in VRMS development still needs to be improved.

REFERENCES

Abuhantash, A. (2023). The Impact of Human Resource Information Systems on Organizational Performance: A Systematic Literature Review. *European Journal of Business and Management Research*, *8*(3), 239–245. https://doi.org/10.24018/ejbmr.2023.8.3.1992

Ahmad, T., Iqbal, J., Ashraf, A., Truscan, D., & Porres, I. (2019). Model-based testing using UML activity diagrams: A systematic mapping study. *Computer Science Review*, *33*, 98–112. https://doi.org/10.1016/j.cosrev.2019.07.001

Apascaritei, P., & Elvira, M. M. (2022). Dynamizing human resources: An integrative review of SHRM and dynamic capabilities research. *Human Resource Management Review*, *32*(4). https://doi.org/10.1016/j.hrmr.2021.100878

Bangare, S. L., Gupta, S., Dalal, M., & Inamdar, A. (2016). Using Node.Js to Build High Speed and Scalable Backend Database Server. *International Journal of Research in Advent Technology*. www.ijrat.org

Beer, M. (2022). Developing strategic human resource theory and making a difference: An action science perspective. *Human Resource Management Review*, *32*(1). https://doi.org/10.1016/j.hrmr.2017.11.005

Boon, C., Den Hartog, D. N., & Lepak, D. P. (2019). A Systematic Review of Human Resource Management Systems and Their Measurement. *Journal of Management*, *45*(6), 2498–2537. https://doi.org/10.1177/0149206318818718

Chamida, M. A., Susanto, A., & Latubessy, A. (2021). Analisa User Acceptance Testing Terhadap Sistem Informasi Pengelolaan Bedah Rumah di Dinas Perumahan Rakyat Dan Kawasan Permukiman Kabupaten Jepara. *Indonesian Journal of Technology, Informatics and Science (IJTIS)*, *3*(1), 36–41. https://doi.org/10.24176/ijtis.v3i1.7531

Deursen, S. Van, & Seemann, M. (2019). *Dependency Injection Principles, Practices, and Patterns* (Vol. 1).

Dhivya, D., & Nirmala, K. (2018). Study on Integration Testing and System Testing. *International Journal of Creative Research Thoughts*, *6*(2), 2320–2882.

Diallo, M. F., Djelassi, S., & Kumar, V. (2021). Marketing and globalization: Relevance, trends and future research. *Recherche et Applications en Marketing (English Edition)*, *36*(3), 2-7. https://doi.org/10.1177/20515707211027326

Edwards, T., Almond, P., Murray, G., & Olga, T. (2022). International human resource management in multinational companies: Global norm making within strategic action fields. *Human Resource Management Journal*, *32*(3), 683–697. https://doi.org/10.1111/1748-8583.12422

El Idrissi, F. E. H., Benabdelhadi, A., & Kabaili, H. (2021). Adoption and impact of electronic Human Resource Management: A systematic literature review. *Technium Social Sciences Journal*, *21*, 594–610. https://doi.org/10.47577/tssj.v21i1.3744

Hamouche, S. (2023). Human resource management and the COVID-19 crisis: Implications, challenges, opportunities, and future organizational directions. *Journal of Management & Organization*, *29*(5), 799-814. https://doi.org/10.1017/jmo.2021.15

Imani, T., Nakano, M., & Anantatmula, V. (2017). Does a Hybrid Approach of Agile and Plan-Driven Methods Work Better for IT System Development Projects? *International Journal of Engineering Research and Applications*, *07*(03), 39–46. https://doi.org/10.9790/9622-0703043946

International Labour Organization. (2023). *Human Resource Management: A manual for employer and business membership organizations Tool 4: Performance management*. www.ilo.org/publns.

Jagarwar, P. (2022). Effective Human Resource Management as Tool For Organizational Success and Development. *International Journal for Research Trends and Innovation (Www.Ijrti.Org)*, *7*, 1.

Kaluža, M., Kalanj, M., & Vukelić, B. (2019). A comparison of back-end frameworks for web application development. *Zbornik Veleučilišta u Rijeci*, *7*(1), 317–332. https://doi.org/10.31784/zvr.7.1.10

Kurniawan, T. A. (2018). Pemodelan Use Case (UML): Evaluasi Terhadap beberapa Kesalahan dalam Praktik. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, *5*(1), 77–86. https://doi.org/10.25126/jtiik.201851610

Marquis, Y. A. (2024). From Theory to Practice: Implementing Effective Role-Based Access Control Strategies to Mitigate Insider Risks in Diverse Organizational Contexts. *Journal of Engineering Research and Reports*, *26*(5), 138–154. https://doi.org/10.9734/jerr/2024/v26i51141

Nikpay, F., Ahmad, R., & Yin Kia, C. (2017). A hybrid method for evaluating enterprise architecture implementation. *Evaluation and Program Planning*, *60*, 1–16. https://doi.org/10.1016/j.evalprogplan.2016.09.001

Osezua, E., & Nkogbu, G. O. (2016). Managing Changes In Organizational External Environment: The Role Of Human Resource Managers. *Ilorin Journal of Administration and Development*, *2*(1), 45–61.

Pattrick, P., & Mazhar, S. (2019). Core Functions of Human Resource Management and its Effectiveness on Organization: A Study. *International Journal of Research in Economics and Social Sciences*, *9*(5). http://www.euroasiapub.org

Ríos, J. M., & Pedreira-Souto, N. (2019). Approach of agile methodologies in the development of web-based software. *Information , 10*(10). https://doi.org/10.3390/info10100314

Senarath, U. S. (2021). *Waterfall Methodology, Prototyping and Agile Development*. https://doi.org/10.13140/RG.2.2.17918.72001

Setiaji, & Sastra, R. (2021). Implementasi Diagram UML (Unified Modelling Language) Pada Perancangan Sistem Informasi Penggajian. *Jurnal Teknik Komputer AMIK BSI*, *7*(1). https://doi.org/10.31294/jtk.v4i2

Tabrani, M., Suhardi, & Priyandaru, H. (2021). Sistem Informasi Manajemen Berbasis Website Pada Unl Studio Dengan Menggunakan Framework Codeigniter. *Jurnal Ilmiah M-Progress*, *11*(1), 13-21. https://doi.org/10.35968/m-pu.v11i1.598

Verma, S., Popli, R., & Kumar, H. (2021). The Agile Deployment Using Machine Learning in Healthcare Service. In *Proceedings of the International Conference on Paradigms of Computing, Communication and Data Sciences: PCCDS 2020* (pp. 879-890). Springer Singapore.

Yahya, N., & Maidin, S. S. (2023). Hybrid agile development phases: The practice in software projects as performed by software engineering team. *Indonesian Journal of Electrical Engineering and Computer Science*, *29*(3), 1738–1749. https://doi.org/10.11591/ijeecs.v29.i3.pp1738-1749

Žužek, T., Kušar, J., Rihar, L., & Berlec, T. (2020). Agile-Concurrent hybrid: A framework for concurrent product development using Scrum. *Concurrent Engineering Research and Applications*, *28*(4), 255–264. https://doi.org/10.1177/1063293X20958541