Journal of Emerging Information System and Business Intelligence
ISSN: 2774-3993
Journal homepage: https://ejournal.unesa.ac.id/index.php/JEISBI/

Web-based Construction Project Application Development with ReactJS, NestJS, and Agile Scrum

Muhammad Zakia Avlach[1], Yuni Yamasari[2]

[1,2] *State University of Surabaya, Surabaya, Indonesia*

muhammadzakia.19054@mhs.unesa.ac.id, yuniyamasari@unesa.ac.id

ABSTRACT

Despite the rapid advancement of web-based information technology, the construction industry still heavily relies on manual project management methods, including budget tracking, worker coordination, and design inspiration searches. PT Media Kreasi Abadi, a construction company specializing in interior design and renovation, lacks a dedicated digital platform to streamline its project management processes. To address these inefficiencies, this study aims to develop a web-based construction project management application integrating a ReactJS frontend and a NestJS backend, supported by an API-driven architecture. The system enhances collaboration among project stakeholders while improving efficiency in project tracking and management. Additionally, this research evaluates the backend's performance to ensure seamless integration with the frontend. By adopting Agile Scrum, the development process remains flexible and continuously adaptive to user needs. The findings suggest that a well-designed digital platform can significantly optimize construction project management, reduce manual errors, and enhance team coordination. Future improvements may incorporate real-time analytics and AI-driven decision-making features to further enhance system capabilities.

Keyword: Construction project management, web application, rest api, react js, nest js, clean architecture, scrum.

Article Info:
Article history:
Received February 07, 2025
Revised July 01, 2025
Accepted July 09, 2025

Corresponding Author
Muhammad Zakia Avlach
State University of Surabaya, Surabaya, Indonesia
muhammadzakia.19054@mhs.unesa.ac.id

1. INTRODUCTION

In today's digital era, the development of internet-based information technology provides convenience in various aspects of life, including in sending and receiving information. Various sectors have adopted this technology, ranging from companies, educational institutions, to other organizations. One of the rapidly growing implementations of information technology is web applications, which allow users to access various services through a browser without the need to install additional software [1]. In general, web applications consist of two main parts, namely the frontend as the user interface and the backend that handles data processing and transactions securely.

The frontend or client-side is the part that interacts directly with users and is responsible for displaying information and carrying out the functionality provided by the backend. Meanwhile, the backend or server-side plays a role in managing data, including storage, processing, and sending data to the frontend. The connection between the two is generally done

through an Application Programming Interface (API) that allows communication between systems [2]. Along with technological developments, web applications are increasingly flexible and can be implemented on various platforms to improve work efficiency.

However, in the construction industry, the utilization of this technology is still limited. Many construction companies still use manual methods in project management, such as budget recording, coordination between workers, and searching for interior and exterior design inspiration [3]. One example is PT Media Kreasi Abadi, a construction company in Balikpapan engaged in interior design and renovation. Until now, the company has not had a dedicated digital platform to manage projects more efficiently.

Based on these problems, this research aims to develop a web-based application as a construction project management platform. The system will integrate an API-based backend with a web-based frontend to improve collaboration between project workers. In addition, this research will also analyze the quality of the backend to ensure the efficiency of the system in supporting the frontend [4].

In its development, the Agile Scrum method is used to ensure that application development runs iteratively and adaptively according to user needs. This method allows teams to work in structured sprints, so that applications can be developed more quickly and flexibly according to changes that occur during the development process [5]. This research carries the title "Web-based Construction Project Application Development with ReactJS, NestJS, and Agile Scrum."

2. METHODS

This research utilizes the Scrum method, an iterative and flexible approach to software development. This method is applied in the development of a web-based application designed as a construction project management system. Before implementation, several crucial stages are carried out to ensure the effectiveness of the Scrum method in the development process.

The first stage of this research is data collection, conducted through observation at PT Media Kreasi Abadi, a company engaged in interior design and home renovation. This observation aims to identify the primary issues faced in construction project management. The results indicate that the project management process has not been running efficiently, and a lack of inspirational sources for project designers—such as architects, interior designers, and exterior designers—remains a frequent challenge.

After identifying the problems, the research proceeds to the system design stage, where the workflow is defined before being developed into an application. The design process uses use case diagrams to illustrate the interactions between users and the system. In the construction project management system, there are three main user types with different access rights: project owners, project members, and guests. Project owners have full access to the system, including authentication mechanisms to prevent unauthorized access. Project members can manage financial aspects and project documentation but must obtain approval from the project owner before making changes. Meanwhile, guests can only view the list of available projects without making any modifications. Additionally, in the inspiration management system, there are two user types: inspiration design owners and guests. Inspiration design owners have full access to upload and manage inspirations, while guests can only view the inspiration list without requiring authentication. To ensure security and access control in both systems, an admin plays a key role in managing user access rights according to their needs.

The next process is system development, which is carried out gradually in accordance with the iterative principles of the Scrum method. Development begins with the creation of a minimum viable product (MVP) that includes basic features such as user registration and authentication, project management, budget and documentation management, and inspiration design management. Each development iteration is conducted in two-week sprints, with evaluations at the end of each sprint to ensure that the developed features meet user requirements. If deficiencies or bugs are found, corrections are made in the next iteration, allowing the development process to adapt dynamically to changing needs.

Once the system is developed, the next stage is system testing to ensure that every feature functions properly. Testing is conducted using the black-box testing method, which focuses on system functionality without inspecting the source code. Testing includes validation of user authentication, project management, and inspiration design management to ensure that each feature operates according to the access rights assigned to each user. If any bugs or errors are found during testing, corrections are made before the system is officially launched to ensure system stability.

After the system is implemented, the maintenance phase follows to ensure that the system continues to operate optimally and evolves according to user needs. Maintenance includes periodic performance monitoring to detect potential issues or performance degradation, bug fixes, and feature updates to align with user requirements, as well as security enhancements through authentication algorithm updates and data encryption to protect user information. By implementing the Scrum method, each stage of the development process allows for continuous evaluation and improvement, ensuring that the developed system is not only responsive to user needs but also flexible enough to adapt to evolving requirements in the field.

3. RESULTS AND DISCUSSION

3.1 Sprint 1

3.1.1 Sprint Planning

The Sprint Planning phase aims to plan each sprint based on a predetermined period. In this phase, the estimated completion time for each feature is determined based on the previously defined product backlog.

Table 1. Estimated work on all features in sprint 1

Feature	Estimated Work Time (days)
Registration or Sign Up	1
Login	0.5
Project Management Dashboard	0.5
Create New Project	1
Project List	0.5
Update Existing Project	1
Delete Project	0.5
View Specific Project Data	0.5
View Project-Specific Reports	0.5

Search Users	0.5
Add Project Members	1
Remove Project Members	0.5

Based on the table, the total estimated work time for all features in Sprint 1 is 8 days, covering both frontend and backend development.

3.1.2 Daily Scrum

The Daily Scrum phase is used to implement the development of each feature planned during Sprint Planning. This process is carried out iteratively, with daily documentation to monitor progress and encountered obstacles.

Table 2. Daily progress on sprint 1

Date	Completed Work	Next Task	Obstacles
May 31, 2024	None	Register feature	None
June 3, 2024	Register feature	• Login feature • Dashboard project management	None
June 4, 2024	• Login feature • Dashboard project management	Create New Project feature	None
June 5, 2024	Create New Project feature	• Project List feature • Update Project feature	None
June 6, 2024	• Project List feature • Update Project feature	• Delete Project feature • Detail Specific Project	None
June 7, 2024	• Delete Project feature • Detail Specific Project	• Project reports feature data	None
June 10, 2024	Project reports feature data	• Search users • Add members as collaborator	None
June 11, 2024	• Search users	Remove members feature	None

	• Add members as collaborator		

3.1.3 Sprint Review

The Sprint Review aims to evaluate the developed features in Sprint 1. Testing is conducted using the black box testing method to ensure that each feature functions as expected. The results of the feature testing in Sprint 1 are as follows:

Table 3. Black-box testing results in sprint 1

Feature	Expected Outcome	Test Result
Sign Up (Valid Data)	Success message, redirected to login	Valid
Sign Up (Invalid Data)	Error message displayed	Valid
Login (Correct Credentials)	Success message, redirected to dashboard	Valid
Login (Incorrect Credentials)	Error message displayed	Valid
Login (Invalid Email Format)	Error message displayed	Valid
Create New Project (Valid Data)	Success message, project created	Valid
Create New Project (Empty Data)	Error message displayed	Valid
Update Project (Valid Data)	Success message, project updated	Valid
Update Project (Empty Data)	Error message displayed	Valid
Delete Project (Confirm Yes)	Project deleted	Valid
Delete Project (Confirm No)	Project remains	Valid
Search Users	Displays matching users	Valid
Add Project Members	Member added successfully	Valid
Remove Project Members (Confirm Yes)	Member removed	Valid
Remove Project Members (Confirm No)	Member remains	Valid

All features have been tested and function correctly without any issues.

3.1.4 Sprint Retrospective

Sprint Retrospective is conducted to evaluate the progress of Sprint 1 and determine improvements for the next sprint.

Table 4. Sprint retrospective sprint 1

Evaluation Aspect	Results
What went well?	All features were completed on time and functioned correctly.
What needs improvement?	No major issues, but efficiency in development can be enhanced.
Solutions for the next sprint?	Optimize work processes to complete tasks faster than estimated.

3.2 Sprint 2

3.2.1 Sprint Planning

The Sprint Planning phase aims to plan each sprint based on a predetermined period. In this phase, the estimated time for completing each feature is determined based on the previously defined product backlog.

Table 5. Estimated work on all features in sprint 2

Feature	Estimated Completion (Days)
Project budget list	1
Add new budget	1
Delete budget	1
Update total budget value	1
Inspiration dashboard	1
Work documentation list	1
Add work documentation	0.5
Delete work documentation	0.5

Based on the table, the total estimated work time for all features in Sprint 2 is 7 days, covering both frontend and backend development.

3.2.2 Daily Scrum

The Daily Scrum phase is used to implement the development of each feature planned during Sprint Planning. This process is carried out iteratively, with daily documentation to monitor progress and encountered obstacles.

Table 6. Daily progress on sprint 2

Date	Completed Tasks	Upcoming Tasks	Obstacles
June 12, 2024	Remove project members	Project budget list	None
June 13, 2024	Project budget list	Add new budget	None
June 14, 2024	Add new budget	Delete budget	None
June 17, 2024	Delete budget	Update total budget value	None
June 18, 2024	Update total budget value	Work documentation list	None
June 19, 2024	Work documentation list	Add and delete work documentation	None
June 20, 2024	Add and delete work documentation	Inspiration dashboard	None

3.2.3 Sprint Review

The Sprint Review aims to evaluate the developed features in Sprint 2. Testing is conducted using the black box testing method to ensure that each feature functions as expected. The results of the feature testing in Sprint 2 are as follows:

Table 7. Black-box testing results in sprint 2

Test Scenario	Expected Outcome	Status
Add a new budget with valid data	A success message appears and redirects to the finance page	Valid
Add a new budget with empty data	A data invalid message appears	Valid
Delete a budget and confirm 'Yes'	A success message appears and the data is deleted	Valid
Delete a budget and confirm 'Cancel'	Data remains unchanged	Valid
Update total budget value with valid data	A success message appears and redirects to the finance page	Valid
Update total budget value with empty data	A data invalid message appears	Valid
Add work documentation with valid data	A success message appears and redirects to the resume page	Valid
Add work documentation with an invalid file type	A data invalid message appears	Valid

All features have been tested and function correctly without any issues.

3.2.4 Sprint Retrospective

Sprint Retrospective is conducted to evaluate the progress of Sprint 2 and determine improvements for the next sprint.

Table 8. Sprint retrospective sprint 2

Evaluation Aspect	Result
What went well?	All features were completed on time and functioned correctly.
What needs improvement?	No major issues, but development efficiency can be improved.
Solution for the next sprint?	Optimizing work time to complete features faster than estimated.

3.3 Sprint 3

3.3.1 Sprint Planning

The Sprint Planning phase aims to plan each sprint based on the predetermined period. In this phase, the estimated completion time for each feature is determined based on the previously defined product backlog.

Table 9. Estimated work on all features in sprint 3

Feature	Estimated Completion Time (Days)
Profile	1
Profile Data Update	1
Profile Picture Update	0.5
Create New Inspiration Data	1
Inspiration List	0.5
Specific Inspiration Data	0.5
Update Inspiration	1
Delete Inspiration	0.5

Based on the table, the total estimated work time for all features in Sprint 3 is 6 days, covering both frontend and backend development.

3.3.2 Daily Scrum

The Daily Scrum phase is used to implement the development of each feature planned during Sprint Planning. This process is carried out iteratively, with daily documentation to monitor progress and encountered obstacles.

Table 10. Daily progress on sprint 3

Date	Completed Tasks	Upcoming Tasks	Obstacles
June 21, 2024	Remove project member	Profile	None
June 24, 2024	Profile	Profile data update, including picture	None
June 25, 2024	Profile data update, including picture	Create new inspiration data	None
June 26, 2024	Create new inspiration data	Inspiration list, Specific inspiration data	None
June 27, 2024	Inspiration list, Specific inspiration data	Update inspiration	None
June 28, 2024	Update inspiration	Delete inspiration	None

3.3.3 Sprint Review

The Sprint Review aims to evaluate the developed features in Sprint 3. Testing is conducted using the black box testing method to ensure that each feature functions as expected. The results of the feature testing in Sprint 3 are as follows:

Table 11. Black-box testing results in sprint 3

Feature & Test Scenario	Expected Result	Status
Profile Data Update - Entering data in the correct format	Success message appears, redirected to profile page	Valid
Profile Data Update - Entering incorrect format (empty/wrong email)	Invalid data message appears	Valid
Profile Picture Update - Uploading an image in the correct format	Success message appears, redirected to profile page	Valid
Profile Picture Update - Uploading a non-image file	Invalid data message appears	Valid
Create New Inspiration Data - Uploading data in the correct format (image file)	Success message appears, redirected to inspiration list	Valid
Create New Inspiration Data - Uploading incorrect format (empty/non-image file)	Invalid data message appears	Valid
Update Inspiration - Uploading data in the correct format (image file)	Success message appears, redirected to inspiration list	Valid
Update Inspiration - Uploading incorrect format (empty/non-image file)	Invalid data message appears	Valid
Delete Inspiration - Clicking delete button and confirming 'Yes'	Success message appears, data is removed from database	Valid

Delete Inspiration - Clicking delete button and confirming 'Cancel'	Data remains unchanged	Valid

All features have been tested and function correctly without any issues.

3.3.4 Sprint Retrospective

Sprint Retrospective is conducted to evaluate the progress of Sprint 3 and determine improvements for the next sprint.

Table 12. Sprint retrospective sprint 3

Evaluation Aspect	Result
What went well?	All features were completed on time and functioned correctly.
What needs to be improved or removed?	No major issues, but development efficiency can be improved.
Solution for the next sprint?	Optimizing work time to complete features faster than estimated.

3.4 Sprint 4

3.4.1 Sprint Planning

The Sprint Planning phase aims to plan each sprint according to the predetermined period. In this phase, the estimated time required for each feature is determined based on the previously defined product backlog.

Table 13. Estimated work on all features in sprint 4

Feature	Estimated Completion (Days)
Add favorite inspirations	0.5
List of favorite inspirations	1
Remove inspiration from favorites	0.5
Update password (send email)	1
Update password	1
Total data list	1

Based on the table, the total estimated work time for all features in Sprint 4 is 5 days, covering both frontend and backend development.

3.4.2 Daily Scrum

The Daily Scrum phase is used to monitor the progress of feature implementation planned during Sprint Planning. This process is conducted iteratively with daily documentation to track progress and identify obstacles.

Table 14. Daily progress on sprint 4

Date	Completed Tasks	Upcoming Tasks	Obstacles
July 1, 2024	Remove inspiration	Add favorite inspirations, List of favorite inspirations	None
July 2, 2024	Add favorite inspirations, List of favorite inspirations	Continue listing favorite inspirations, Remove inspiration from favorites	None

July 3, 2024	Continue listing favorite inspirations, Remove inspiration from favorites	Update password (send email)	None
July 4, 2024	Update password (send email)	Update password	None
July 5, 2024	Update password	Total data list (total project and inspiration data)	None

3.4.3 Sprint Review

The Sprint Review phase is conducted to evaluate the results of feature development in Sprint 4. Testing is performed using the black-box testing method to ensure that each feature functions as required. The results of the feature testing in Sprint 4 are as follows:

Table 15. Black-box testing results in sprint 4

Feature & Test Scenario	Expected Outcome	Status
Add favorite inspiration - Click heart button	The heart button changes to red and remains on the inspiration details page	Valid
Remove inspiration from favorites - Click heart button	The heart button changes to white and remains on the inspiration details page	Valid
Update password (send email) - Enter email in correct format	Message appears to check email	Valid
Update password (send email) - Enter email in incorrect format	Message appears indicating invalid email	Valid
Update password - Enter matching password and confirmation	Message appears confirming password update was successful	Valid
Update password - Enter mismatched password and confirmation	Message appears indicating passwords do not match, update fails	Valid

All features have been tested and function correctly without any issues.

3.4.4 Sprint Retrospective

The Sprint Retrospective phase is conducted to evaluate the progress of Sprint 4 and determine improvement steps for the next sprint.

Table 16. Sprint retrospective sprint 4

Evaluation Aspect	Result
What went well?	All features were completed on time and functioned correctly.
What needs improvement or removal?	No major issues, but development efficiency can be improved.
Solutions for the next sprint?	Optimizing work time to complete features faster than estimated.

3.5 Sprint 5

3.5.1 Sprint Planning

The Sprint Planning phase aims to plan each sprint based on the predetermined period. In this phase, the estimated time for each feature development is determined based on the previously defined product backlog.

Table 17. Estimated work on all features in sprint 5

Feature	Estimated Completion (Days)
Add new level or role data along with assigned permissions	1
List of created roles	0.5
Specific role data along with assigned permissions	0.5
Update role data and reconfigure assigned permissions	1
Delete role data along with related permissions	0.5
Add role selection during sign-up and update profile	0.5
Handle errors when users access unauthorized pages (permission guard)	1

Based on the table above, the total estimated time required for completing all features in Sprint 5 is 5 days, including both frontend and backend development.

3.5.2 Daily Scrum

The Daily Scrum phase is used to monitor the progress of feature implementation planned in Sprint Planning. This process is conducted iteratively with daily documentation to track progress and encountered obstacles.

Table 18. Daily progress on sprint 5

Date	Work Completed	Work To Be Done	Obstacles
July 11, 2024	List total data (total project and inspiration data)	Add new level or role data along with assigned permissions	None
July 12, 2024	Add new level or role data along with assigned permissions	List roles, Specific role data along with permissions	None
July 15, 2024	List roles, Specific role data along with permissions	Update role data and reconfigure assigned permissions	None
July 16, 2024	Update role data and reconfigure assigned permissions	Delete role data, Add role selection during sign-up and update profile	None
July 17, 2024	Delete role data, Add role selection during sign-up and update profile	Handle errors when users access unauthorized pages (permission guard)	None

3.5.3 Sprint Review

The Sprint Review phase is conducted to evaluate the development results of Sprint 5. Testing is performed using the black-box testing method to ensure that each feature functions according to the requirements.

Table 19. Black-box testing results in sprint 5

Feature & Test Scenario	Expected Outcome	Status
Adding/updating a role with a name exceeding 128 characters	Error: Invalid data	Valid
Adding/updating a role without selecting any permissions	Error: Invalid data	Valid

Deleting role data via trash icon or delete button	Success message appears, data is deleted from database, redirected to role list page	Valid
Canceling role deletion after confirmation	Data remains unchanged	Valid

All features have been tested and function correctly without any issues

3.5.4 Sprint Retrospective

The Sprint Retrospective phase is conducted to evaluate the execution of Sprint 5 and determine improvement steps for the next sprint.

Table 20. Sprint retrospective sprint 5

Evaluation Aspect	Results
What went well?	All features were completed on time and functioned correctly.
What needs to be improved or removed?	No major issues, but development efficiency can be improved.
Solutions for the next sprint?	Optimizing work time to complete features faster than estimated.

3.6 Discussion

This research developed key features in a web-based construction project management application using ReactJS, NestJS, and the Agile Scrum methodology. The features include project management, team monitoring, budgeting, documentation, and project inspiration for prospective owners.

The findings are consistent with previous research but introduce new contributions through the adoption of Agile Scrum and modern technologies, which enhance efficiency and reduce application errors. This integration is proposed as a new approach to technology-based construction project management.

Although beneficial, the application still has limitations, such as dependence on infrastructure and internet connectivity. However, with further development, it has the potential to significantly improve the efficiency of construction project management.

CONCLUSION

This research successfully developed a web-based construction project management application using ReactJS and NestJS TypeScript with the Agile Scrum methodology. The use of NestJS offers advantages in minimizing bugs through static typing and enabling a more structured development process with a modular system. Additionally, Agile Scrum enhances flexibility and responsiveness in development, though performance improvements could be explored using alternative technologies like Go.

As recommendations, it is advised to continue exploring new technologies, consider alternatives that enhance application performance, and deepen the understanding of Agile Scrum to improve development effectiveness. Keeping up with industry advancements is also crucial to ensure the application remains relevant and competitive in the market. By implementing these suggestions, the web-based construction project management application can continue to evolve and provide greater benefits to its users.

ACKNOWLEDGEMENTS

The author expresses gratitude to PT Media Kreasi Abadi for its support and assistance in carrying out this research until its completion. The author also appreciates the permission granted to conduct observations in order to obtain the necessary information during the research process.

REFERENCES

[1] Rahmadhani, S., Dyvani, M., Wildana, W., Arumdanie, H. W., Hakim, L., Rekayasa, T., Lunak, P., & Banyuwangi, N. (2024). Penerapan React JS dan Axios untuk Pengembangan Front-end Aplikasi iCare. Software Development, Digital Business Intelligence, and Computer Engineering, 2(02), 40–46.

[2] Kaniya, I. A., Paramitha, P., Made Wiharta, D., Made, I., Suyadnya, A., Raya, J., Unud, K., Jimbaran, B., & Selatan, K. (2022). Perancangan dan Implementasi Restful Api pada Sistem Informasi Manajemen Dosen Universitas Udayana. Jurnal Spektrum, 9(3), 15–23.

[3] Widjojo, J. F., Rusdianto, E., Kartika, F., & Dewi, S. (2020). Pembangunan Sistem Manajemen Proyek pada Perusahaan Konstruksi Bangunan Berbasis Website. Jurnal Informatika Atma Jogja, 1(1), 60–70.

[4] Putri, M. P., & Bobby, B. (2020). Sistem Informasi Manajemen Proyek PT. Samudera Perkasa Konstruksi Berbasis Web. Matrik: Jurnal Manajemen, Teknik Informatika Dan Rekayasa Komputer, 20(1), 85–96.

[5] Hadji, S., Taufik, M., & Mulyono, S. (2020). Implementasi Metode Scrum pada Pengembangan Aplikasi Delivery Order Berbasis Website (Studi Kasus pada Rumah Makan Lombok Idjo Semarang). Prosiding Konstelasi Ilmiah Mahasiswa Unissula (KIMU) Klaster Engineering, 0(0).

[6] Andipradana, A., & Dwi Hartomo, K. (2021). Rancang Bangun Aplikasi Penjualan Online Berbasis Web Menggunakan Metode Scrum. Jurnal Algoritma, 18(1), 161–172.

[7] Duc Pham, A. (2020). Developing back-end of a web application with Nest JS framework: Case: Integrify Oy's student management system. Lab University of Applied Sciences.

[8] Hasanuddin, Asgar, H., & Hartono, B. (2022). Rancang Bangun Rest Api Aplikasi Weshare Sebagai Upaya Mempermudah Pelayanan Donasi Kemanusiaan. Jurnal Informatika Teknologi Dan Sains, 4(1), 8–14.

[9] Maulana, Moch. A., Haryoko, H., Santoso, B., & Lukman, L. (2022). Penerapan Teknologi Stack Mern pada Aplikasi Service Manajemen Bengkel Berbasis Web. Jurnal Media Informatika Budidarma, 6(3), 1536–1544.

[10] Astana, I. N. Y., Dharmayanti, G. A. P. C., & Sumarni, N. K. (2020). Strategi Peningkatan Kinerja Pengelolaan Proyek Konstruksi pada Dinas Pekerjaan Umum dan Penataan Ruang Kabupaten Karangasem. Spektran, 8(2), 179–186.

[11] Wena, M., Sugiyanto, S., & Pribadi, P. (2020). Penggunaan Aplikasi Manajemen Proyek Berbasis Ict pada Pelaksanaan Proyek Konstruksi untuk Meningkatkan Produktivitas Usaha. Prosiding Hapemas, *1*(1), 157–161.

[12] Sulistyorini, T., Sova, E., & Ramadhan, R. (2022). Pemantauan Kasus Penyebaran Covid-19 Berbasis Website Menggunakan Framework React js dan Api. Jurnal Ilmiah Multidisiplin, *1*(04), 01–13.

[13] Kim, M., Corradini, D., Sinha, S., Orso, A., Pasqua, M., Tzoref-Brill, R., & Ceccato, M. (2023). Enhancing Rest Api Testing with Nlp Techniques. ISSTA 2023 - Proceedings of the 32nd ACM SIGSOFT International Symposium on Software Testing and Analysis, 1232–1243.

[14] Soplanit, A. R., Supiyanto, S., Saputro, A. D., Kmurawak, R. M., & Sampebua, M. R. (2023). Penerapan Version control system Berbasis Web Menggunakan Next.JS, Nest.JS, Node.JS, dan MongoDB Pada Proses Pengerjaan Skripsi Mahasiswa. Jurasik (Jurnal Riset Sistem Informasi Dan Teknik Informatika), *8*(2), 361–370.

[15] Warkim, W., Muslim, M. H., Harvianto, F., & Utama, S. (2020). Penerapan Metode Scrum dalam Pengembangan Sistem Informasi Layanan Kawasan. Jurnal Teknik Informatika Dan Sistem Informasi, *6*(2).

[16] Annisa, R., Ananda, R. A., & Sulistiono, W. E. (2024). Implementasi Golang Clean Architecture Pada Perancangan Backend Point of Sales Website. Jurnal Informatika Dan Teknik Elektro Terapan, *12*(2), 2830–7062.