JURNAL TEKNOLOGI DAN OPEN SOURCE
Vol. 8, No. 1, June 2025, pp. 142~152
e-ISSN: 2622-1659, accredited Four Grade by Kemenristekdikti, Decree No: 152/E/KPT/2023
DOI: 10.36378/jtos.v8i1.4385

Frontend Design and Development of Thesis Management Application Using NextJS in Physics Study Program UPN "Veteran" Jatim

Adimas Syiraa Setiabudhi[1], Eka Dyar Wahyuni[2], Nur Cahyo Wibowo[3]
[1,2,3]Department of Information System, Faculty of Computer Science, UPN "Veteran" Jawa Timur, Surabaya

Article Info

Article history:

Received 05 20, 2025
Revised 05 25, 2025
Accepted 06 5, 2025

Keywords:

Application
Administration
Management
Thesis
SCRUM

ABSTRACT

In higher education institutions, managing undergraduate thesis administration remains a challenge, particularly in the Physics Study Program, which currently lacks an system that can manage the process of students thesis. To address this issue, the Thesis Management Application was developed as a solution to manage thesis administration through a web-based system accessible to Students, Lecturers, Thesis Coordinators, and Study Program Coordinator. This study focuses on frontend web development using the Next.js framework and APIs. The application offers key features including authentication, pre-proposal submission, thesis advisor assignment, seminar and examination management, and submission of graduation documents. The development process followed the Scrum methodology over six sprint iterations. The application was tested using both black-box testing and user acceptance testing. A total of 177 were tested using black-box testing, all of which received a passed status. User Acceptance Testing was conducted across the four user roles mentioned earlier, with all features receiving accepted status Finally, the system was deployed on the Vercel cloud hosting server, allowing public access for users.

Corresponding Author:

Adimas Syiraa Setiabudhi
Department of Information System
Faculty of Computer Science
UPN "Veteran" Jawa Timur Surabaya
Email: adimas.id@gmail.com

1. Introduction

The advancement of information technology has significantly transformed the way humans access and manage information quickly [1]. One prominent form of this technology is the website, which has evolved into an information system supporting decision-making processes across various fields, including higher education. In the context of higher education, one crucial process that requires an integrated system is the preparation of undergraduate theses, which involves numerous stages and administrative documents [2], [3]. A thesis is a term used to describe a scientific paper written by undergraduate students (S1) that presents

the results of research addressing a specific problem or phenomenon within a particular field of study, following established academic standards. It is typically based on field research or literature studies conducted by experts in the relevant discipline, and serves as a final requirement for graduation [4]. The documentation and management of theses through a Thesis Management Application is a crucial aspect that must be considered by academic programs as higher education providers. Although each study program may have different procedures, the thesis process generally consists of several phases: Thesis Title Submission, Proposal Submission, Proposal Seminar, Research, and Result Seminar [2].

The Physics Study Program at UPN "Veteran" Jawa Timur, established in 2022, currently lacks a digital system to support the thesis management process, which consists of several phases: Thesis Title Submission, Supervision Sessions, Proposal Seminar Registration and Execution, Research Implementation, Result Seminar Registration and Execution, Oral Examination (Comprehensive Exam), and Final Graduation Assessment (Yudisium). Currently, the oldest students in the Physics Study Program are those in their 5th semester who have not yet begun their thesis work. These students are expected to start their thesis in the upcoming 7th semester. The Coordinator of the Physics Study Program has stated the need for a system to manage students' thesis processes, aiming to facilitate the organization of documents and administrative tasks for students who are preparing to undertake their thesis. Therefore, an application is needed to support the implementation of the thesis process that will be undertaken by students in the upcoming semester.

The Thesis Management Application is proposed as a solution to streamline administrative processes from the title submission stage to the final graduation assessment. This application supports five main user roles: Student, Lecturer, Thesis Coordinator, Program Coordinator, and Administrator. In this Thesis Management Application, there are five distinct roles: Students, Lecturers (who can serve as either Thesis Supervisors or Examiners), the Thesis Coordinator, the Study Program Coordinator (who can also act as a Supervisor or Examiner), and the Study Program Administrator. Several features are included in this application. Students can register for the proposal seminar by submitting the required documents. The Thesis Coordinator can then assign examiners based on their area of expertise to assess the student's Proposal Seminar. The Coordinator also has the authority to publish the examiner assignments and the seminar schedule. Examiners are able to provide assessments and suggest revisions for the student. After completing the revisions, students may resubmit their proposal documents for re-evaluation by the examiners. The supervision process during the research phase is conducted face-to-face between the student and their Thesis Supervisor. Once this phase is completed, students can register for the Result Seminar by submitting the necessary documents. The Thesis Coordinator can then reschedule the seminar and assign examiners accordingly. During the Result Seminar, examiners again provide evaluations and feedback. Following this, students are required to take a Comprehensive Oral Examination as a prerequisite for graduation (Yudisium). After completing the Oral Exam and Yudisium, students are required to upload proof of Yudisium completion to the system. The presence of this application is expected to accelerate administrative workflows and assist supervisors in monitoring student thesis progress [5], [6].

The application is web-based due to its advantages in accessibility, scalability, and ease of maintenance [7]. In web-based application development, there are two main technologies: Front-End and Back-End, each with distinct responsibilities. Broadly speaking, Front-End technology focuses on providing an intuitive user interaction and an attractive interface, while Back-End technology is primarily concerned with server-side logic and database management [8]. Front-End development plays a crucial role in website creation. The Front-End is responsible for handling user interactions through the system interface. A well-designed front-end interface can attract users by providing comfort, ease of use, and access to various available features [9]. Front-End development requires a solid understanding of HTML for structure, CSS for styling, and JavaScript for interactivity, along with libraries and frameworks to streamline the development process [10]. This study focuses on the Front-End module, as it aims to develop a system that offers an interactive user experience, ease of data management, and a responsive interface [11].

The separation between the Front-End and Back-End modules is carried out due to the complexity of the application and to facilitate the development process, as it allows each module to be developed independently and enables better collaboration among team members. Another reason for adopting this modular separation is that it enables the Front-End to make Asynchronous JavaScript and XML (AJAX) requests to the Back-End, allowing data to be retrieved from the database and dynamically displayed on the web page without the need to reload the entire page. In contrast, the traditional Model-View-Controller (MVC) approach processes data requests by rendering a new web page on the server, which requires a full page reload and can be time-consuming [12]. Several client-side frameworks and libraries commonly used in Front-End web development include Vue.js, Angular.js, and React.js [13]. The selection of appropriate technology to create a user experience that is engaging, interactive, and functional forms the basis for developing the application using the Next.js framework, which is built on the React.js library [14].

Next.js is a framework built on top of React.js that uses the JavaScript programming language. Major companies such as Netflix, Nike, and PlayStation currently use Next.js as the framework for developing their web applications [15]. In terms of routing, Next.js adopts a file-based routing approach, where each file created within the app directory becomes a route in the website. This approach allows developers to better understand the navigation structure within the project [16]. Using the Next.js framework to develop a web based application can accelerate development time and enhance the quality of the web application [17]. Next.js was chosen as the framework based on React.js because it offers features such as image optimization, Server Side Rendering (SSR), Static Site Generation (SSG), and compatibility with TypeScript [18]. By default, Next.js uses the JavaScript programming language. However, due to its type analysis, JavaScript has weak and dynamic typing. This kind of type system often leads to various issues related to data type mismatches [19]. To address this issue, the TypeScript programming language was chosen, as TypeScript is an extension of JavaScript designed to facilitate the development of large-scale JavaScript applications also for its ability to detect errors early through a static typing system that JavaScript does not have [20], [21]. Based on the aforementioned issues, this research aims to design and develop the Front-End of a web-based Thesis Management Application for the Physics Study Program at UPN "Veteran" Jawa Timur to support the digital and centralized administration of student theses.

2. Research Method

The research methodology outlines the steps taken to achieve the objectives of this study. It consists of data collection, requirements analysis, and an application development framework using Scrum, as illustrated in Figure 1.

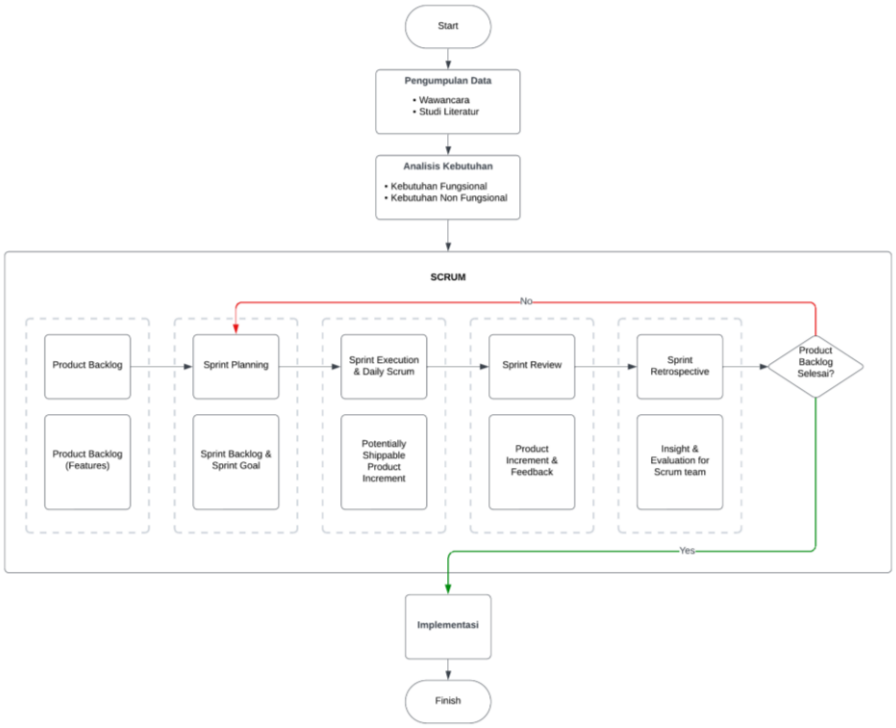

Figure 1. Research Design Diagram

The method include these following steps below:

2.1. Data Collection

Data collection was carried out using two methods: interviews and literature study. Interviews were conducted with Coordinator of the Physics Study Program at UPN "Veteran" Jawa Timur and Thesis Coordinator of the same program. The literature study involved searching for relevant references in the form of research journals and books accessed online, focusing on topics such as the development of Thesis Management Applications, the use of Next.js as a front-end web development framework, and the Scrum software development methodology. The gathered information was used to compile the literature review, design the research methodology, and support the development of the Thesis Management Application.

2.2. Requirement Analysis

The requirement analysis in this study involves processing data obtained from information about the current thesis process, which is still conducted manually and lacks a dedicated management application in the Physics Study Program. The analysis results are used to define the functional and non-functional requirements to be implemented in the Thesis Management Application under development.

2.3. Scrum

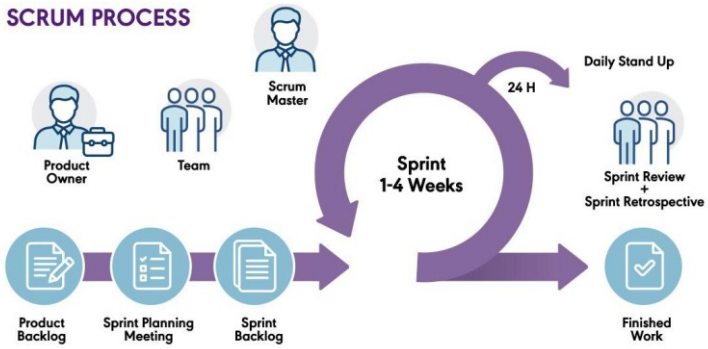

Figure 2. Scrum Methodology (Source: PM Partners)

Scrum is a structured framework used in the Software Development Life Cycle (SDLC) to manage complex product development [22]. It is based on a set of values, principles, and practices, involving three main roles: the Product Owner, who defines and prioritizes features; the Scrum Master, who monitors team progress; and the Development Team, responsible for design, development, and testing [23]. The Scrum method include these following steps below:

a) Product Backlog

In this stage, user requirements are categorized to determine the priority of each feature and product based on its level of importance [24].

b) Sprint Planning

This stage involves sprint planning, which includes defining tasks and goals to be achieved during the application development process [24].

c) Sprint

A sprint is a time-boxed iteration during which planned tasks from the sprint backlog are executed and developed [24].

d) Sprint Review

The Sprint Review is a meeting in Scrum held to showcase the application developed during the Sprint and to present the progress made [24].

e) Sprint Retrospective

The Sprint Retrospective is a phase where the team conducts self-evaluation and plans improvements to be implemented in the next Sprint [24].

2.4. Implementation

In the Implementation phase, the system that has successfully passed all testing stages will be uploaded to the server to make it accessible online.

3. Result and Discussion

The results of this study include the design and development of a thesis management application. The design phase involves creating a system design using Unified Modeling Language (UML), as UML provides a set of tools to describe various aspects of the system, such as classes, components, and the relationships between them [25]. The UML diagrams used include the Use Case Diagram, Sequence Diagram, and Class Diagram. The development phase consists of designing the application's user interface and consuming the backend API. Once all items in the product backlog are completed, the system will undergo Black Box Testing and User Acceptance Testing.

3.1. Requirement Analysis

This study aims to design and develop a Thesis Management Application to assist in managing student theses—from pre-proposal submission, proposal seminars, to final seminars. With this application, the thesis process in the Physics Study Program can be more streamlined and reduce errors in data recording. The application features four user levels: Student, Lecturer, Thesis Coordinator, and Program Coordinator. Following a business process analysis, 189 functional requirements were identified and then mapped into use case diagram.

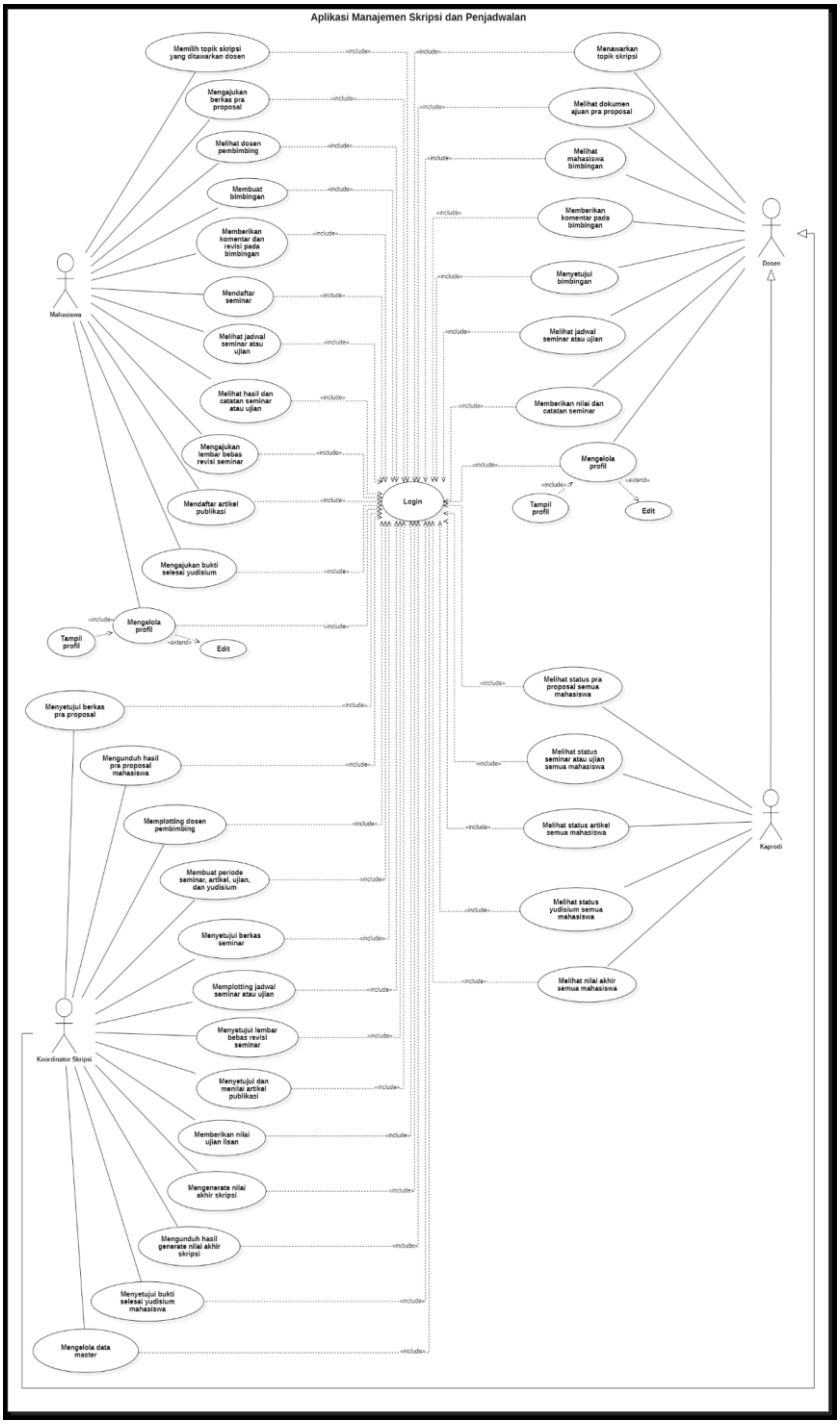

Figure 3. Use Case Diagram

3.2. Execution with Scrum

To define the steps in this research, the team created a task list or product backlog. Task priorities, shown in Table 4.4, were determined through an interview with the Thesis Coordinator of the Physics Study Program. The tasks were then further discussed and divided based on team roles. Prioritization is crucial to

ensure that high-priority features receive appropriate attention. The features listed in the product backlog in Table 1 were derived from the system requirements gathered through the data collection process and prioritized based on the system's requirements.

Table 1. Product Backlog

ID	Name	Priority
1	Interface Design	High
2	Authentication	High
3	Profile	High
4	Title Submission (Pre-Proposal)	High
5	Advisor Assignment	High
6	Thesis Guidance Card	High
7	Seminar Registration	High
8	Seminar Scheduling	High
9	Seminar Assesment	High
10	Seminar Revision	High
11	Publication Article Submission	High
12	Examiner Assignment (Oral Exam)	High
13	Oral Exam Assesment	High
14	Final Thesis Grade	High
15	Submission of Graduation Completion Proof	Medium

This study focuses on interface development and integration with the backend API, particularly for the seminar submission and scheduling modules, which are core features of the application. The development follows the SCRUM methodology, with this module scheduled for the third sprint. The integration aims to ensure the application can efficiently consume data from the API and function as expected by users.

Sequence diagrams

During the sprint planning phase, items from the product backlog are selected for execution, with a focus on implementing the seminar scheduling interface module. As this feature is high-priority, this sprint emphasizes its successful development. The sprint begins with creating sequence diagrams for the seminar submission and scheduling processes (see Figures 4 and 5), which serve as clear guides for the development team in implementing these features into the system.

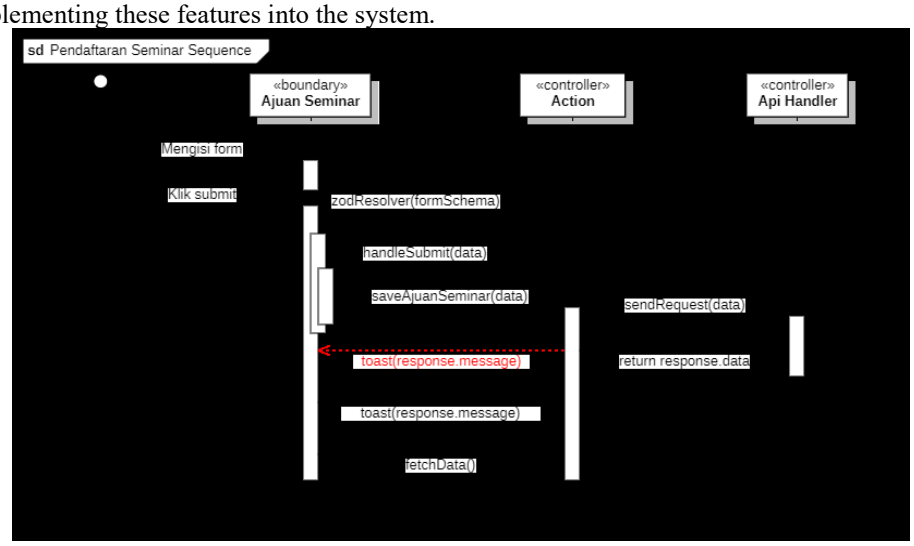

Figure 4. Seminar Submission Sequence Diagram

The image is a sequence diagram that illustrates the process flow of seminar registration by a student within the system. The process begins when the student fills out a form and clicks the submit button on the user interface. Once the button is clicked, the system initiates data validation through the zodResolver(formSchema) function to ensure that the input data complies with the predefined schema.

After successful validation, the data is passed to the handleSubmit(data) function, which then forwards it to the saveAjukanSeminar(data) function. This function is responsible for saving or managing the seminar registration data and subsequently calls sendRequest(data) to transmit the data to the server through the Api Handler component. This component processes the request and returns the response data back to the system. Once the response is received, the system provides feedback to the student in the form of a

notification or message using toast(response.message). This message may indicate either success or failure, which is represented in the diagram by two possible paths: a red dashed line for errors and a black line for success. Finally, the system invokes the fetchData() function to refresh the data displayed to the user, ensuring that any changes are immediately reflected in the interface.

This diagram provides a comprehensive overview of how the components within the system interact sequentially to handle the seminar submission process from start to finish, including validation, data submission, response handling, and interface update.

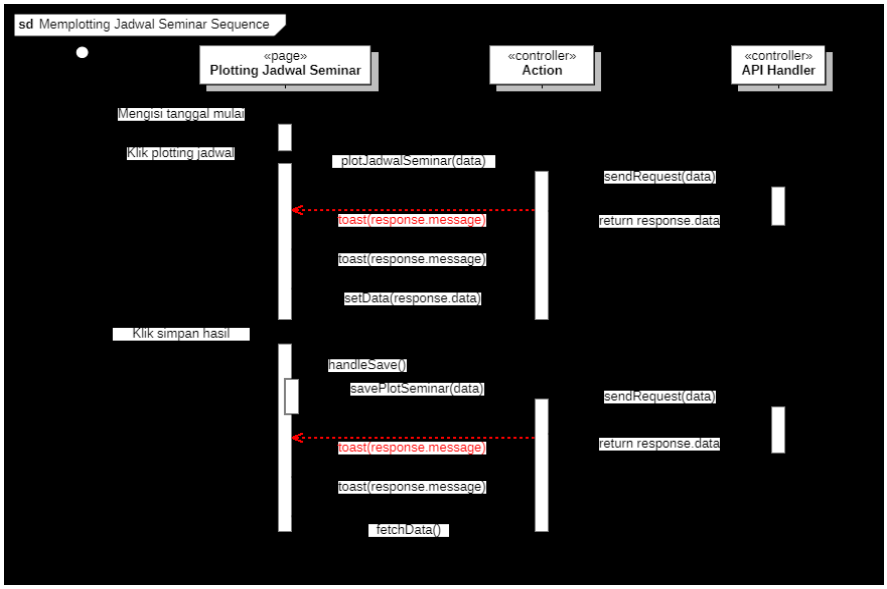

Figure 5. Seminar Scheduling Sequence Diagram

Interface results

When the Seminar Registration Use Case is executed, students can access the seminar proposal period page via the sidebar menu and click the submission button for the selected period to open the proposal submission form (Figure 6). This form includes several input fields, such as the required seminar registration documents. After completing the form, students can click "Save" to submit the data to the server. If any input does not meet the form requirements, validation error messages will appear beside the corresponding fields. Otherwise, the data is sent to the server. The server will then process and validate the data. If any issues are found, a failure response will be returned and displayed on the Seminar Submission page. If the data is valid, a success response will be returned, and the table will be updated accordingly.

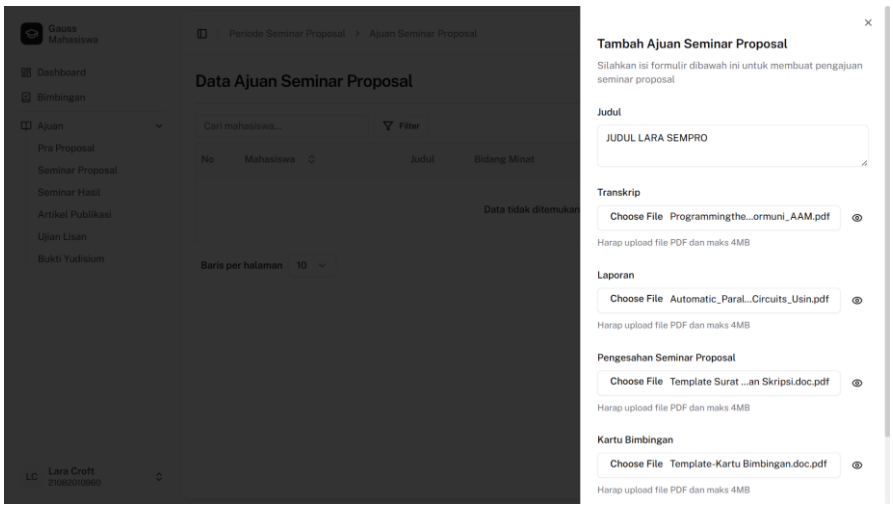

Figure 6. Seminar Submission Page

When the Seminar Schedule Plotting Use Case is executed, the thesis coordinator can access the Seminar page via the sidebar menu and navigate to the Plotting page as shown in Figure 7. This page is the default view when the coordinator has not yet scheduled any seminars. On this page, the coordinator can enter the seminar implementation date. In addition to setting the date, this interface provides a simple and intuitive layout that guides the coordinator through the initial steps of the seminar scheduling process. By offering a clean starting point, the system ensures that no prior data interferes with the input, reducing the risk of errors and helping the coordinator to initiate the plotting process smoothly and accurately. This setup is essential in ensuring that all seminars are planned systematically from the outset.

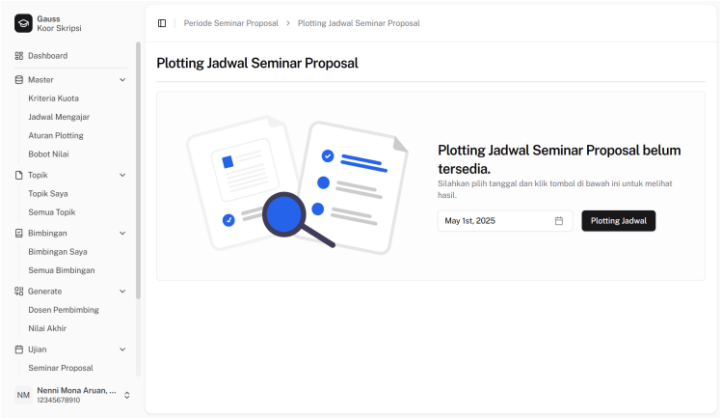

Figure 7. Seminar Scheduling Default Page

After clicking the schedule plotting button, the thesis coordinator can view the results of the advisor plotting as shown in Figure 8. This page displays a table that contains comprehensive information regarding the seminar schedule. The table includes essential details such as student names and identification numbers, the assigned seminar date, time, and location, as well as the names of the examiners and assigned student advisors. This structured presentation allows the coordinator to have a complete overview of all scheduled seminars in a single, centralized interface.

The primary purpose of this feature is to provide a clear, organized, and easily interpretable layout that enables the thesis coordinator to verify the scheduling data at a glance. With all the information presented in tabular form, the coordinator can quickly identify if any students are missing schedule assignments, if there are conflicts in time or location, or if an examiner has been double-booked. This greatly reduces the potential for human error and makes the process of seminar arrangement more efficient.

In addition, the table format promotes better decision-making, as the coordinator can easily compare different scheduling entries and make necessary adjustments if overlaps or inconsistencies are found. It also supports accountability and transparency, since all assignments are visible and traceable within the system.

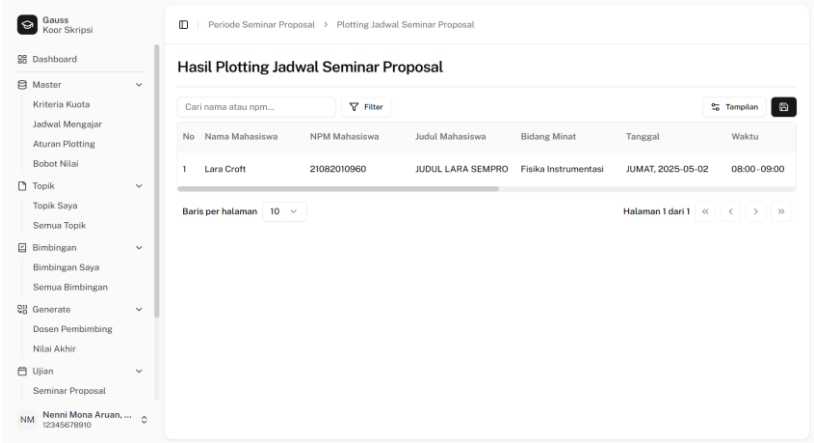

Figure 8. Seminar Scheduling Result Page

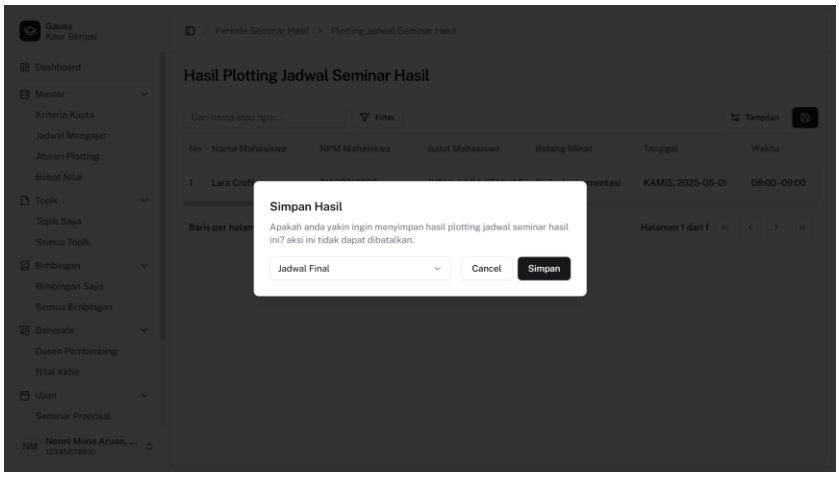

Figure 9. Save Seminar Scheduling Result Page

After the plotted schedule data is displayed, the thesis coordinator can save the results, as shown in Figure 9. Two saving options are available: as a draft or as scheduled. The request is then sent to the server for processing and validation. If the data is invalid, the server returns a failure response, which is shown on the Schedule Plotting page. Otherwise, a success response is returned.

Black Box Testing

After the user interface design and backend API integration were completed, black box testing was conducted, as shown in Table 2 below.

Table 2. Black Box Testing

No.	Test Scenario	Expected Result	Actual Result	Status
1.	User completes the seminar submission form with valid and complete data.	Seminar Submission Success	Seminar Submission Success	PASSED
2.	User leaves all or some fields in the seminar submission form empty.	Seminar Submission Failed	Seminar Submission Failed	PASSED
3.	User fills in the seminar title using non-capitalized letters.	Seminar Submission Failed	Seminar Submission Failed	PASSED
4.	User performs seminar schedule plotting by entering a seminar date after the plotting date.	Seminar Scheduling Success	Seminar Scheduling Success	PASSED
5.	User performs seminar schedule plotting without entering the seminar date.	Seminar Scheduling Failed	Seminar Scheduling Failed	PASSED
6.	User saves the seminar schedule plotting result by selecting a storage type (draft or scheduled).	Seminar Scheduling Success	Seminar Scheduling Success	PASSED
7.	User saves the seminar schedule plotting result without selecting a save type (draft or scheduled).	Seminar Scheduling Failed	Seminar Scheduling Failed	PASSED

The daily scrum held during the third sprint iteration showed significant progress in the application's development. It was concluded that students are now able to submit seminar registration requests, and the thesis coordinator can plot seminar schedules. Positive progress was also seen in the frontend development of this module, which reached 100% completion. All bugs identified during the third sprint were resolved within the same sprint.

3.2. Implementation

The implementation was carried out by deploying the code to the Vercel cloud hosting server, allowing the system to be publicly accessed. At the end of the development phase, the team conducted User Acceptance Testing (UAT), where the developers and QA team presented and demonstrated the Thesis Management Application to lecturers and students of the Physics Department at UPN “Veteran” Jawa Timur. UAT was performed for all modules of the Gauss application across all user roles: Student, Lecturer, Thesis Coordinator, and Program Coordinator as shown in Table 3 below.

Table 3. User Acceptance Testing

ID	Name	Status
1	Interface Design	Accepted
2	Authentication	Accepted
3	Profile	Accepted
4	Title Submission (Pre-Proposal)	Accepted
5	Advisor Assignment	Accepted
6	Thesis Guidance Card	Accepted
7	Seminar Registration	Accepted
8	Seminar Scheduling	Accepted
9	Seminar Assesment	Accepted
10	Seminar Revision	Accepted
11	Publication Article Submission	Accepted
12	Examiner Assignment (Oral Exam)	Accepted
13	Oral Exam Assesment	Accepted
14	Final Thesis Grade	Accepted
15	Submission of Graduation Completion Proof	Accepted

4. Conclusion

Based on the result of research and discussion that have been conducted, it can be concluded that:

1. The front-end development of the Gauss Thesis Management Application has been completed to support the thesis administration process in the Physics Study Program at UPN "Veteran" Jawa Timur. The development followed the Scrum software methodology over six sprints spanning approximately three months, covering stages from design to implementation. The design phase produced one Use Case Diagram, 50 Sequence Diagrams for core business processes, and one Class Diagram. The development phase resulted in a functional application implementing 189 functional requirements, covering key thesis processes: (1) Pre-Proposal Submission, (2) Supervisor Assignment, (3) Supervision, (4) Seminar Registration, Scheduling, Assessment, and Revision, (5) Publication Article Submission, (6) Examiner Assignment and Oral Exam Evaluation, (7) Final Thesis Grading, and (8) Yudisium Proof Submission. In the final stage, testing was conducted using two methods: black-box testing and user acceptance testing (UAT). The black-box test, performed by the programmer, covered 177 functional requirements, all of which passed. UAT was carried out by four user roles—students (47 requirements), lecturers (31), thesis coordinators (75), and program coordinators (24)—by four individual users. After testing, the application was deployed to the Vercel cloud hosting server, making it publicly accessible online.
2. One of the integration results between the Front-End and Back-End was implemented using a REST API architecture for the Seminar Schedule Plotting feature. In this process, the seminar schedule data retrieved from the backend API endpoint is displayed completely and accurately on the front-end without any truncation.

Acknowledgement

The author would like to express his gratitude to God Almighty for all His mercy and grace so that this research can be completed well. This research would not have been achieved without support and assistance from various parties, family and friends who always provided moral support and enthusiasm throughout the research process, all parties who cannot be mentioned one by one, who have contributed, helped and supported in completion of this research.

References

[1] R. M. Thamrin and R. Andriani, “Design Web-Based Registration And Data Management Of Student Thesis Information System,” *SISFOTENIKA*, vol. 11, no. 1, p. 101, Feb. 2021, doi: 10.30700/jst.v11i1.1111.

[2] T. P. Handayani, R. Maku, and R. Musa, “SISTEM INFORMASI MANAJEMEN SKRIPSI BERBASIS WEB DI UNIVERSITAS MUHAMMADIYAH GORONTALO,” *J. Ilmu Komput. JUIK*, vol. 1, no. 2, Oct. 2021, doi: 10.31314/juik.v1i2.1174.

[3] N. A. B. Saputra and H. S. Purba, “Rancangan Sistem Manajemen Skripsi Berbasis Web Menggunakan Metode Rapid Application Development (RAD),” *JURIKOM J. Ris. Komput.*, vol. 9, no. 5, p. 1621, Oct. 2022, doi: 10.30865/jurikom.v9i5.5012.

[4] M. Muslich and Maryaeni, *Bagaimana menulis skripsi*. PT Bumi Aksara, 2009.

[5] I. G. N. T. A. P. Wijaksana, I. P. Satwika, and I. N. Y. A. Wijaya, “SISTEM INFORMASI SKRIPSI STMIK PRIMAKARA BERBASIS WEBSITE MENGGUNAKAN FRAMEWORK LARAVEL,” *J. Teknol. Inf. Dan Komput.*, vol. 6, no. 2, Jan. 2020, doi: 10.36002/jutik.v6i2.1018.

[6] D. A. Kristiyanti and A. Mulyana, "Sistem Informasi Monitoring Skripsi Berbasis Web (Studi Kasus: Prodi Akuntansi Universitas Mercu Buana)," *J. Sist. Inf. BISNIS*, vol. 10, no. 1, pp. 56–63, Jun. 2020, doi: 10.21456/vol10iss1pp56-63.
[7] M. Farhan, E. D. Wahyuni, and A. R. E. Najaf, "SISTEM INFORMASI POINT OF SALE BERBASIS WEB DENGAN NODE.JS (STUDI KASUS: CWIMIE KAISAR)," 2024.
[8] Q. Behruz and U. Son, "WEB FRONT-END AND BACK-END TECHNOLOGIES IN PROGRAMMING," *Int. Sci.-Online Conf.*, 2024, doi: https://doi.org/10.5281/zenodo.10518360.
[9] M. F. S. Lazuardy and D. Anggraini, "Modern Front End Web Architectures with React.Js and Next.Js," vol. 7, no. 1, 2022.
[10] S. M. Prasetiyo, M. I. P. Nugroho, R. L. Putri, and O. Fauzi, "Pembahasan Mengenai Front-End Web Developer dalam Ruang Lingkup Web Development," vol. 01, no. 6, 2022.
[11] A. Baehaqi, M. S. Basit, R. E. Indrajit, and R. D. Kurniawan, "FRONT END LEARNING MANAGEMENT SYSTEM DEVELOPMENT USING THE NEXTJS FRAMEWORK," *J. Tek. Inform. Jutif*, vol. 4, no. 4, pp. 899–911, Aug. 2023, doi: 10.52436/1.jutif.2023.4.4.1273.
[12] Z. Lei, H. Zhou, W. Hu, and G.-P. Liu, "Toward an international platform: A web-based multi-language system for remote and virtual laboratories using react framework," *Heliyon*, vol. 8, no. 10, p. e10780, Oct. 2022, doi: 10.1016/j.heliyon.2022.e10780.
[13] R. Vyas, "Comparative Analysis on Front-End Frameworks for Web Applications," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 10, no. 7, pp. 298–307, Jul. 2022, doi: 10.22214/ijraset.2022.45260.
[14] H. H. B. Kora and M. S. Manita, "Modern Front-End Web Architecture Using React.js and Next.js," *Univ. Zawia J. Eng. Sci. Technol.*, vol. 2, no. 1, pp. 1–13, Aug. 2024, doi: 10.26629/uzjest.2024.01.
[15] A. Nugroho, K. Prihandani, and R. Mayasari, "RANCANG BANGUN SISTEM PEMBELIAN E-TICKET BERBASIS WEBSITE DENGAN KONSEP SERVER-SIDE RENDERING MENGGUNAKAN FRAMEWORK NEXT JS PADA WISATA TELAGA KUSUMA JUMANTONO," *JATI J. Mhs. Tek. Inform.*, vol. 8, no. 4, pp. 5771–5777, Jun. 2024, doi: 10.36040/jati.v8i4.9960.
[16] V. K. R. Ballamudi, K. Lal, H. Desamsetti, and S. Dekkati, "Getting Started Modern Web Development with Next.js: An Indispensable React Framework," vol. 1, no. 1, 2021.
[17] R. A. Ananda and G. F. Nama, "ANALISIS DAN PERANCANGAN LAYANAN STREAMING FILM BERBASIS WEB LANGGANAN MENGGUNAKAN FRAMEWORK NEXTJS," *J. Inform. Dan Tek. Elektro Terap.*, vol. 12, no. 1, Jan. 2024, doi: 10.23960/jitet.v12i1.3967.
[18] M. Riva, *Real-World Next.js: Build Scalable, High-Performance, and Modern Web Applications Using Next. Js, the React Framework for Production*. Packt Publishing, Limited., 2022.
[19] S. H. Jensen, A. Møller, and P. Thiemann, "Type Analysis for JavaScript," in *Static Analysis*, vol. 5673, J. Palsberg and Z. Su, Eds., in Lecture Notes in Computer Science, vol. 5673. , Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 238–255. doi: 10.1007/978-3-642-03237-0_17.
[20] G. Bierman, M. Abadi, and M. Torgersen, "Understanding TypeScript," *ECOOP 2014 – Object-Oriented Program.*, pp. 257–281, 2014, doi: https://doi.org/10.1007/978-3-662-44202-9_11.
[21] J. Bogner and M. Merkel, "To Type or Not to Type? A Systematic Comparison of the Software Quality of JavaScript and TypeScript Applications on GitHub," in *Proceedings of the 19th International Conference on Mining Software Repositories*, May 2022, pp. 658–669. doi: 10.1145/3524842.3528454.
[22] K. Schwaber and J. Sutherland, "The Scrum Guide The Definitive Guide to Scrum: The Rules of the Game," 2011.
[23] K. S. Rubin, *Essential Scrum: A practical guide to the most popular agile process*. Addison-Wesley, 2013.
[24] Y. Yarpriransa, D. Saripurna, and H. Santoso, "Implementasi Metode Scrum pada Pengembangan Aplikasi Bimbingan Skripsi Online," *Hello World J. Ilmu Komput.*, vol. 2, no. 1, pp. 42–57, Apr. 2023, doi: 10.56211/helloworld.v2i1.228.
[25] H.-E. Eriksson, Ed., *UML 2 toolkit*. Indianapolis, Ind: Wiley Pub, 2004.