Journal of Orange Technology (JOT)
Vol. 1, No. 2, April, 2025, pp. 75−90
E-ISSN: XXXX-XXXX | P-ISSN: XXXX-XXXX, DOI:10.34306

Grant Submission Monitoring System Based on Orange Technology Using Laravel 12 and Vue.js

Indira Puspa Gustiah[1], Ninda Lutfiani[2], Muhamad Rapidan Kusuma[3*], Etty Puji Lestari[4], Thomas Green[5]
[1]Department of Economic and Business, Mercu Buana University, Indonesia
[2]Department of Information Technology, Satya Wacana Christian University, Indonesia
[3]Faculty of Science and Technology, University of Raharja, Indonesia
[4]Doctoral of Management Science, Universitas Terbuka, Indonesia
[5]Eesp Incorporation, British Indian Ocean
[1]ippuspag@gmail.com, [2]982022020@student.uksw.edu, [3]rafidan@raharja.info, [4]ettypl@ecampus.ut.ac.id, [5]thom.green@eesp.io
***Corresponding Author**

Article Info

Article history:

Submission November 22, 2024
Revised December 22, 2024
Accepted March 26, 2025
Published April 10, 2025

Keywords:

Orange Technology
Single Page Application
Grant Monitoring
Laravel 12
Vue.js

ABSTRACT

The development of effective digital systems for managing research and community service grants is essential to ensure transparency, administrative efficiency, and user comfort within academic environments. However, many institutions still rely on manual processes or outdated platforms, resulting in unclear progress tracking, slow information updates, and increased emotional stress for lecturers as proposal authors. **This study** proposes a Grant Submission Monitoring System based on Orange Technology, an approach that emphasizes human-centered design, emotional well-being, and socially responsible innovation. **The system was developed** using the SDLC Waterfall model, employing Laravel 12 as a secure and high-performance backend framework and Vue.js to deliver an interactive Single Page Application (SPA) interface. Functional validation was conducted through Black Box Testing to ensure accuracy and reliability across all modules. The resulting system provides real-time notifications, structured visualization of proposal stages, and faster administrative processing, reducing delays by up to 40% compared to traditional methods. **The SPA architecture** enables a responsive experience with an average content rendering time of approximately 150 ms after initial load, supporting seamless user interaction. Feedback from lecturers indicates reduced uncertainty, increased trust, and greater emotional comfort due to clearer information flow and improved interface intuitiveness. Overall, **the system demonstrates** how Orange Technology principles can be integrated into academic information systems to create solutions that are not only efficient and transparent but also supportive of user well-being. This approach offers a scalable foundation for enhancing institutional governance and strengthening digital services across diverse educational contexts.

DOI: https://doi.org/10.34306/jot.v1i2.24

1. INTRODUCTION

The rapid advancement of information technology in higher education continues to reshape administrative systems, including the management of research and community service grants that play a strategic role

in supporting institutional development [1]. In many universities, grant submission and monitoring activities are still carried out through manual workflows or outdated platforms, leading to inefficient communication, delayed information updates, and a lack of clarity regarding proposal evaluation stages. These challenges not only hinder administrative performance but also influence lecturers emotional comfort, often causing uncertainty, anxiety, and reduced productivity as they wait for progress updates. Such conditions highlight the need for an information system that is not only functionally accurate but also capable of supporting users psychological well being [2]. This aligns with the principles of Orange Technology, which emphasize the creation of digital solutions that are human centered, emotionally supportive, and socially responsible. Digital platforms for grant management have been widely introduced; however, the mere presence of a system does not ensure its effectiveness. User dissatisfaction frequently arises due to unintuitive interfaces, unclear navigation structures, fragmented workflows, and the absence of real-time feedback. These issues can significantly increase cognitive load and emotional tension, especially when users receive incomplete or delayed information about proposal progress [3, 4]. As a result, trust in institutional processes decreases, and lecturers feel uncertain about fairness and transparency in evaluation procedures. Such problems are closely tied to broader academic governance goals as reflected in Sustainable Development Goals (SDGs), particularly SDGs 4 (Quality Education), which encourages strong academic support systems, and SDGs 16 (Peace, Justice, and Strong Institutions), which promotes transparent and accountable institutional practices [5, 6].

In the context of the university environment, the HibahQu platform has long been used as a centralized system for managing internal grant submissions. However, early observations and user evaluations reveal several limitations that hinder optimal system performance. These include complex submission procedures, insufficiently informative interfaces, non-intuitive navigation flows, and the absence of real-time monitoring features that clearly communicate the status of each proposal [7]. Such limitations not only disrupt administrative workflows but also reduce lecturers' emotional comfort, as uncertainty about the evaluation process can generate stress and reduce confidence in the system. These findings affirm the need for a modernized, user-centered monitoring system that integrates both technical efficiency and emotional support, following the foundational principles of Orange Technology [8]. To address these challenges, a development methodology that prioritizes human-centered and emotionally aware design is essential. The Design Thinking approach, with its iterative stages of Empathize, Define, Ideate, Prototype, and Test, provides a comprehensive framework for uncovering deeper insights into user behavior, motivations, and psychological responses. By emphasizing empathy and user observation, this method enables the identification of pain points that are often overlooked in technically driven system analyses. Incorporating Design Thinking ensures that the resulting system design is not only functionally efficient but also aligned with users' emotional needs, fostering trust, comfort, and engagement throughout the grant submission process [9].

Furthermore, the technical implementation requires robust frameworks capable of delivering performance, security, and interactivity. In this study, Laravel 12 is utilized as the backend framework due to its strong security features, structured architecture [10], and ability to handle complex processing with high reliability. To create an interface that is responsive and emotionally supportive, Vue.js is applied on the frontend to develop a SPA. This architecture enables seamless navigation, fast rendering, and real-time interactivity key elements that reduce user frustration and enhance the perception of system responsiveness. When combined with Orange Technology principles, the SPA approach provides a smooth user experience that minimizes uncertainty and fosters emotional well-being. Given these considerations, a comprehensive evaluation and redesign of the grant submission monitoring system becomes crucial [11]. An effective academic information system must not only deliver accurate data but also provide clarity, comfort, transparency, and a sense of control for its users. By integrating Orange Technology with Design Thinking and Usability Testing, this study aims to develop a modern, emotionally aware, and user-friendly monitoring system capable of addressing current institutional challenges while improving user satisfaction. The resulting system is expected to reduce cognitive load, increase trust in the review process, enhance the intuitiveness of the interface, and ultimately support the achievement of institutional objectives related to transparency, accountability, and quality academic services [12].

2. LITERATURE REVIEW

The Literature Review section provides the theoretical foundation and previous research findings relevant to the development of a digital grant submission monitoring system. After understanding the contextual challenges outlined in the Introduction such as transparency issues, usability limitations, and the emotional

impact on lecturers this section synthesizes existing studies to identify concepts, technologies, and governance models that support effective grant digitalization [13]. The review begins by examining best practices in research grant governance and how digital transformation improves administrative accountability. It then explores core system concepts such as real-time web-based monitoring, the application of the Laravel MVC architecture, and the advantages of SPA development using Vue.js. By analyzing these studies, this section establishes the scholarly basis for the system designed in this research and highlights specific gaps that justify the need for a modern, user-centered monitoring platform like HibahQu [14, 15].

2.1. Governance and Digitalization of Research Grants

Research and Community Service activities serve as key performance indicators for higher education institutions in fulfilling the Tri Dharma. Effective grant management requires transparency, accountability, and timeliness to ensure that the funding process is conducted fairly and efficiently. The transition from manual to digital systems in research administration should not be viewed merely as a change in medium, but as a form of business process reengineering aimed at reducing excessive bureaucracy and enhancing user comfort [16]. Conventional grant management systems often create communication gaps between proposing lecturers and the Institute for Research and Community Service LPPM, such as the loss of physical documents, unclear review statuses, and delays in communicating proposal evaluation results [17]. These uncertainties negatively affect both administrative efficiency and the emotional well-being of lecturers, who are required to wait without clear updates. Therefore, digitalization is essential not only to ensure data integrity and improve information accessibility, but also to provide a more transparent, reliable, and emotionally reassuring experience for all stakeholders [18].

2.2. Concept of Web-Based Monitoring Systems

Monitoring in the context of information systems refers to the continuous process of collecting and analyzing data to ensure that activities proceed as planned. In grant administration, a monitoring system functions to present the current status of submitted proposals. Unlike reporting systems, which are passive and retrospective, modern monitoring systems require real-time or near real-time capabilities [19]. According to [16], an effective monitoring interface should be able to simplify complex data into information that can be understood quickly at a glance, enabling users to take corrective action immediately when issues arise. In addition to improving technical efficiency, clear and real-time monitoring also reduces user anxiety, provides a sense of security, and enhances their sense of control during the grant submission process [20].

2.3. Implementation of MVC Architecture in Laravel

Laravel is a PHP-based framework that adopts the Model View Controller (MVC) architectural pattern to separate logic, interface, and control structures. Its use in enterprise-scale system development is supported by a robust ecosystem capable of handling complex business logic and offering high-level security. In its latest version (Laravel 12), the framework introduces significant enhancements in database management through the Eloquent ORM (Object Relational Mapping) [21]. Eloquent allows developers to interact with the database using expressive, object oriented syntax, reducing the risk of errors in manual SQL queries and preventing security vulnerabilities such as SQL injection. The API Resources feature enables consistent transformation of data into JSON format, which is essential for developing headless architectures where backend and frontend function independently [22]. The use of a stable architectural foundation not only delivers performance and reliability but also supports a more comfortable user experience by ensuring that information is presented quickly, consistently, and accurately an important factor in creating a grant administration system that is stress-free and reassuring for users.

2.4. SPA Approach with Vue.js

Advancements in User Interface (UI) technology have shifted development from Multi Page Applications (MPA) toward SPA. Vue.js, a progressive JavaScript framework, offers the capability to build responsive, fast, and interactive SPA interfaces. The key strengths of Vue.js lie in its Virtual DOM and Reactive Data Binding concepts, which allow changes in application data or status to be updated instantly without reloading the entire page [23]. In the context of grant monitoring systems, this enables real-time updates such as a proposal status changing from “Review” to “Accepted” to be displayed immediately to lecturers. This not only improves technical efficiency but also provides psychological comfort, reduces excessive curiosity, and helps lecturers feel more informed and valued. The SPA approach also reduces server bandwidth usage and delivers a user

experience that resembles native desktop applications [24]. This contributes significantly to user efficiency and emotional comfort when accessing grant information anytime.

3. RESEARCH METHODS

The research method used in this study provides a structured framework for identifying user needs, analyzing existing workflows, designing system components, and implementing the necessary features. Each methodological stage is arranged to ensure that the developed system meets functional requirements while addressing usability issues and enhancing user comfort [25]. The approach also establishes a systematic foundation for data collection, problem formulation, solution development, and performance validation in alignment with the study's objectives.

3.1. System Development Framework

This study employs the System Development Life Cycle (SDLC) methodology using the Waterfall model. The selection of this model is based on the characteristics of the grant administration system, which is structured, sequential, and requires detailed documentation at each phase of development. The Waterfall approach ensures that the system architecture and requirements are clearly defined before the implementation process begins, thereby minimizing the risk of major changes during development [26]. This method also supports thorough planning and facilitates clear communication between stakeholders, which is essential for systems that rely on accuracy, auditability, and predictable workflows. Furthermore, the Waterfall model helps maintain system stability and consistency, particularly because the grant application platform plays an essential role in delivering timely and transparent information to proposing lecturers [27]. Ensuring that each phase is completed before moving to the next contributes to a user experience that is more comfortable, predictable, and free from ambiguity. In general, the research procedure consists of four main stages: Requirement Analysis, System Design, Implementation, and Testing, all of which collectively ensure that the developed system meets functional expectations and aligns with institutional needs [28, 29].

3.2. Research Stages

The research stages in this study follow the sequential flow of the SDLC Waterfall model, ensuring that each phase is completed thoroughly before progressing to the next. This structured approach allows the system to be developed systematically, starting from identifying user needs, designing the system framework, implementing the proposed solution, and validating its functionality [30]. Each stage is designed to address the challenges found in the existing grant submission workflow, ensuring that the resulting system not only meets technical requirements but also enhances transparency, usability, and user comfort. The stages involved in this research consist of Requirement Analysis, System Design, Implementation and Development Environment, and System Testing.

3.2.1. Requirement Analysis

The requirement analysis phase aims to map the existing business processes, identify technical obstacles, and understand both the emotional and operational needs of users. Data collection was carried out through observation and document study, in which the researcher reviewed the Standard Operating Procedures (SOP) for grant submission at LPPM and examined the current flow of physical documents to identify mandatory proposal components such as budget plans, statements, and expected outputs as well as potential friction points that could cause delays or user confusion. Additionally, in-depth interviews were conducted with two key user groups, namely LPPM administrators and lecturer-researchers, revealing functional requirements such as the need for automated status notifications, more transparent proposal tracking, and an interface that minimizes user uncertainty during the review process [31]. The insights gained from this phase form the foundation for designing a system that prioritizes user comfort, clarity of information, and reduced cognitive load.

3.2.2. System Design

This phase involves the design of system logic, architecture, database structure, and user interface. The logical and process design was developed using Unified Modeling Language (UML), where Use Case Diagrams were created to define actor roles such as Admin, Lecturer, and Reviewer along with their access rights, and Activity Diagrams were used to illustrate workflows from login, proposal submission, review, to fund disbursement, ensuring consistent and understandable operational flows. The database design was modeled using Entity Relationship Diagrams (ERD) and normalized to the Third Normal Form (3NF) to prevent

redundancy in the proposals, users, and reviews tables, resulting in a stable, secure, and efficient database that supports transparent grant status monitoring [32]. High-fidelity wireframes and mockups were created using Figma for the user interface design, prioritizing informational clarity, minimal clicks, and visual comfort to help users easily understand grant statuses, while user-centered design principles were applied to reduce user anxiety during the waiting period for proposal updates.

3.2.3. Implementation and Development Environment

The implementation stage transforms the design into code using a decoupled architecture between the frontend and backend. The backend (server-side) was developed using Laravel 12 with Migration features for database schema management and Eloquent ORM for data interaction. It provides secure RESTful API endpoints in JSON format, with a focus on data accuracy, response speed, and application security [33]. The frontend (client-side) was developed using Vue.js with state management handled by Pinia, where UI components such as the Navbar, Sidebar, and Status Cards were built modularly to support reusability. Data communication is performed asynchronously using Axios, enabling interface updates without page reloads and ensuring a smooth and responsive user experience. The development environment consists of Windows 11 and Linux Ubuntu as the operating systems, PHP 8.2+ and JavaScript ES6 as the programming languages, and MySQL or MariaDB as the database systems, supported by tools such as Visual Studio Code, Postman, and Git. Overall, the implementation focuses on producing an application that is fast, stable, and capable of delivering an intuitive, barrier-free user experience [34].

3.2.4. System Testing

System testing was conducted to ensure that the application functions as required, operates securely, and is free from critical errors before deployment. The testing process was carried out in stages, covering backend components, user interfaces, and complete system workflows [35]. Additionally, the system architecture built using the Laravel and Vue.js frameworks was validated to ensure optimal integration between the backend and frontend layers. Figure 1 illustrates the framework architecture utilized in the development of this system.

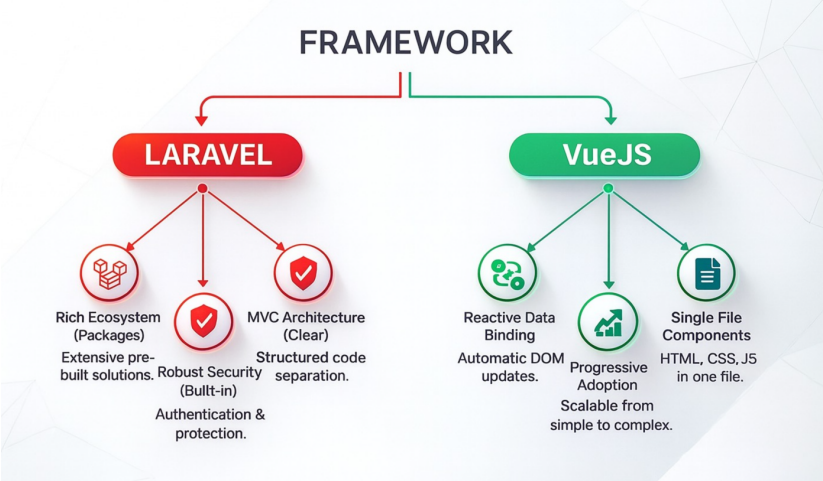

Figure 1. Framework Architecture: Integration between Laravel and Vue.js

As shown in Figure 1, the framework architecture illustrates the integration between Laravel and Vue.js, where Laravel serves as the backend framework with a rich ecosystem of packages, built-in authentication, strong security features, and a clear MVC architecture that ensures structured and maintainable code separation, while Vue.js functions as the frontend framework offering reactive data binding for automatic DOM updates, progressive adoption that allows the application to scale from simple to complex, and single-file components that combine HTML, CSS, and JavaScript in one file to support modular and efficient interface development; together, both frameworks create a cohesive, scalable, and high-performance application structure that balances secure backend processing with responsive and dynamic user interface experiences [36].

4. RESULTS AND DISCUSSION

This part presents the outcomes of the system development and interprets them in relation to the research objectives. The explanation focuses on how the implemented features, system architecture, and user interface improvements respond to the usability problems identified during the requirement analysis. The findings are examined by comparing system performance, user feedback, and observed interaction patterns with relevant theories and previous studies [37]. Through this discussion, the extent to which the proposed solution enhances transparency, user comfort, and overall system efficiency becomes evident.

4.1. Implementation of the Development Environment

The implementation of the Grant Submission Monitoring System was carried out in a local server environment configured to ensure optimal performance and smooth system operation. The back-end was developed using Laravel Framework version 12 running on PHP 8.2, which provides a stable and secure foundation for handling business logic, routing, middleware processes, and API generation. Meanwhile, the front-end was built using Vue.js, supported by Vite as the build tool to enable faster development cycles, improved bundling efficiency, and responsive interface rendering [38, 39]. This combination allows the system to deliver a seamless interaction experience, especially in managing real-time updates within the Single Page Application structure. For data storage, the system utilizes MySQL with the InnoDB storage engine, which offers reliable support for foreign key constraints and ensures strong referential integrity across all relational tables. This setup is particularly important for maintaining consistent relationships between user data, proposals, and tracking histories. Overall, the implemented development environment provides a stable, efficient, and maintainable foundation for the system to operate effectively and support future enhancements [40].

4.2. Database Architecture Implementation

The database structure was implemented using Laravel's Migration features to maintain schema versioning, structured development, and long-term maintainability. This approach ensures that every modification to the database schema is tracked systematically, allowing developers to apply updates consistently across different environments while preserving data integrity. The system consists of four main interconnected entities that collectively support authentication processes, proposal management, workflow consistency, and complete auditability, forming a solid foundation for a scalable and transparent grant monitoring platform [41, 42].

- **Users** stores authentication data and user profiles for Lecturers and Administrators, with passwords secured using Bcrypt hashing for enhanced security. Beyond managing login credentials, this table also functions as the central reference for linking submitted proposals to their respective owners, ensuring accurate associations between users and their activities within the system [43].

- **Proposals** contains essential metadata for every grant submission such as title, abstract, file path, fiscal year, and user associations, enabling fast retrieval and efficient monitoring of proposal progress. This structure supports advanced filtering and tracking, ensuring that administrators and lecturers can easily follow the development and evaluation of each submission [44].

- **Statuses** defines standardized workflow states (Draft, Administrative Review, Substantive Review, Accepted, Rejected), ensuring consistent interpretation of proposal stages across all system modules. Centralizing these states eliminates ambiguity and supports a uniform understanding of the proposal lifecycle within the application.

- **Proposal Trackings** serves as a transactional log recording every status change, including timestamps and actor information. This table maintains a complete audit trail of each proposal's review journey, enabling the system to provide detailed historical insights and improving overall accountability [45].

The One-to-Many relationship between proposals and proposal trackings enables each proposal to store multiple status transitions throughout the review cycle. This ensures that the system captures the full evaluation history rather than only the final outcome, allowing users to clearly understand how their submissions progressed over time. Such detailed tracking enhances transparency, strengthens accountability, and increases user confidence in the fairness and reliability of the grant review process [46, 47].

4.3. Application Programming Interface (API) Development

The API layer in this system is developed using a headless architecture, enabling all interactions between the Laravel 12 backend and the Vue.js frontend to occur through secure RESTful API endpoints, with Laravel Sanctum providing token-based authentication to protect sensitive operations such as proposal submission, reviewer feedback retrieval, and status updates [48]. Consistent data formatting is ensured through Laravel API Resources, allowing components like the Visual Stepper, proposal tracking history, and reviewer notes modal to render accurate and unambiguous information. To maintain high performance, the system applies eager loading techniques such as Proposal with (tracking) to eliminate the N+1 Query Problem and reduce query execution time, allowing the frontend to retrieve complete proposal data efficiently in a single request [49, 50]. Asynchronous communication via Axios supports dynamic interface updates without page reloads, contributing to fast rendering and smooth SPA interactions. Through secure authentication, optimized data retrieval, and seamless asynchronous communication, the API development process ensures that the grant monitoring system operates reliably, efficiently, and with the transparency needed to support user comfort and trust [51, 52].

4.4. User Interface Implementation (Vue.js Frontend)

The user interface was developed using the SPA approach, with Vue Router managing component-based navigation. This design enables seamless interaction without full-page reloads, resulting in a smoother and more efficient user experience [53]. By utilizing Vue.js reactive rendering, components can update dynamically based on real-time state changes, reducing latency and improving interface responsiveness. The modular SPA structure also ensures clear separation of concerns, allowing UI elements such as forms, dashboards, and navigation components to be developed and maintained independently [54]. This architecture enhances scalability, enabling new features to be added without affecting existing functionalities. With optimized routing and state management, the Vue.js frontend provides a responsive and reliable environment suitable for complex, user-driven operations within the grant management system [55, 56].

4.4.1. Landing Page Interface

The landing page serves as the main entry point to the system and is designed with a modern, responsive layout to provide quick orientation for users. It includes key information such as system purpose, login/register access, and a brief overview of the grant submission workflow. The interface utilizes Tailwind CSS for styling combined with Vue motion effects (AOS and transition utilities), resulting in a visually appealing yet lightweight page. Interactive elements such as navigation menus, CTA buttons, and scroll animations enhance the user experience while maintaining accessibility on both desktop and mobile devices.

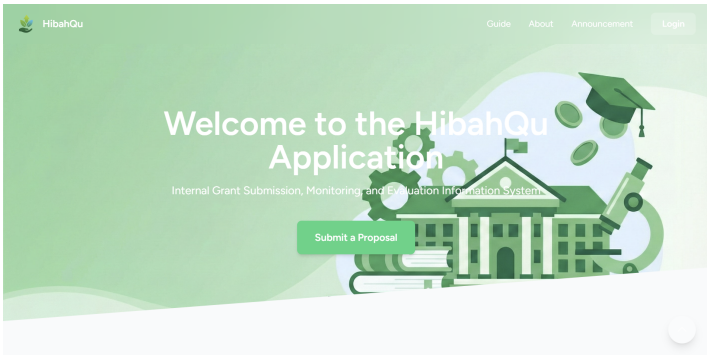

Figure 2. Landing Page Interface

Figure 2 presents the landing page of the HibahQu Application, which serves as the main entry point for users to access features related to the submission, monitoring, and evaluation of internal research grants. The interface adopts a modern green-themed design that reflects the system's visual identity, featuring a clean and structured layout supported by a top navigation menu consisting of Guidelines, About, Announcements, and a login button to facilitate intuitive access to key sections. At the center of the page, a clear welcome message introduces the system's purpose and orientation for first-time and returning users, accompanied by two primary action buttons Submit Proposal and Operator Account Registration that highlight the system's core functions. These elements are visually reinforced by an abstract green background that enhances the

page's professional and cohesive appearance, creating a consistent visual experience aligned with the overall design of the application.

4.4.2. Login Interface

TThe login interface, as illustrated in Figure 3, is designed with a clean and minimal presentation to ensure simplicity, accessibility, and responsiveness across various device types. Unlike traditional login forms that require multiple input fields, the HibahQu interface prioritizes ease of access by providing a single primary authentication method through the "Sign in with Rinfo Email" button, which integrates Google-based single sign-on. This approach reduces credential-related errors, streamlines the login process, and enhances security by relying on trusted institutional identity verification. The interface features a centered white card positioned against a green gradient background, reinforcing the application's visual identity while maintaining a calm and focused interaction space. The card displays the system name "HibahQu", a brief welcome message, and concise instructions, allowing users to immediately understand the purpose of the page without unnecessary visual distractions. The minimalist structure reduces cognitive load and guides users directly toward the available authentication option.

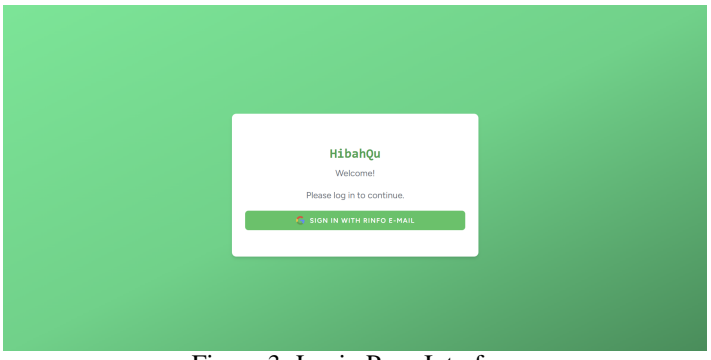

Figure 3. Login Page Interface

Figure 3 illustrates the login interface of the HibahQu application, which provides users with a streamlined and secure authentication entry point through a single sign-on mechanism using institutional Google accounts. The interface adopts a clean and minimal design, featuring a centrally aligned white authentication card placed against a green gradient background that reinforces the platform's visual identity while ensuring a calm and distraction-free user experience. Instead of presenting multiple credential input fields, the interface offers a single prominently styled button labeled "Sign in with Rinfo Email", directing users to Google-based authentication and reducing potential input errors commonly associated with manual login forms. A brief welcome message accompanied by concise instructional text is displayed above the authentication button, enabling users to quickly understand the purpose of the page. The deliberate absence of additional form elements, such as password fields, checkboxes, or registration links, minimizes cognitive load and ensures that user attention remains focused on the primary access method. This design approach enhances both efficiency and security by leveraging verified institutional identity services for authentication.

4.4.3. Role Based Login Interface

Figure 4 presents the role-based login interface of the HibahQu application, which introduces a structured and guided authentication flow tailored to two distinct user categories: students and lecturers. Rather than directly displaying conventional email–password fields, the system implements a three-step login process "Select Role," "Enter Data," and "Complete" to enhance clarity, reduce user errors, and provide a more intuitive entry experience. In the initial stage, users are prompted to select their role through two visually distinct cards. Each card includes a representative icon and a brief description, indicating that students authenticate using their student identification number (NIM), while lecturers log in using their lecturer identification number (NIDN). This clear differentiation minimizes confusion and ensures that users immediately understand the required credentials based on their academic role. The interface emphasizes simplicity and focus through a clean layout, balanced spacing, and subtle visual accents that preserve readability and reduce cognitive load. The stepper component displayed at the top enhances transparency by clearly indicating the user's current position within the login sequence, thereby reducing uncertainty and supporting a smooth progression through the

required steps. Once a role is selected, the system transitions to a role-specific data entry page, where users are required to input their NIM or NIDN. This approach allows the platform to dynamically apply role-appropriate validation rules, improving data accuracy and preventing mismatched credential formats.

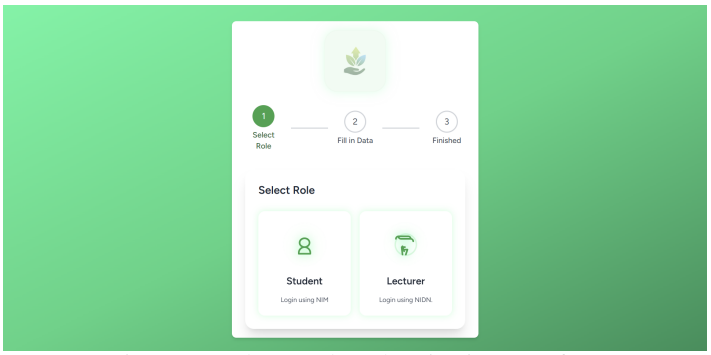

Figure 4. Role Based Authentication Interface

Figure 4 presents a streamlined, role-based authentication interface designed to accommodate two distinct user groups within the HibahQu system: students and lecturers. Rather than functioning as a registration form, this page operates as a structured entry point where users select their role and authenticate using their designated identifiers students through their NIM and lecturers through their NIDN. The interface organizes these options with clear visual hierarchy, ensuring that users immediately understand the appropriate login path based on their academic affiliation. The layout emphasizes simplicity and precision, minimizing potential confusion by separating each authentication method into intuitive, well-labeled sections. The use of consistent spacing, accessible typography, and a harmonious color palette contributes to a calm digital environment, supporting user focus and reducing cognitive strain. This design approach aligns with user-centered principles that prioritize clarity and efficient task completion. To further enhance usability, the interface incorporates structured input fields tailored to each role, providing a predictable flow that guides users from role selection to credential entry. Error-prevention strategies such as concise labels and clearly defined boundaries help reduce input mistakes, particularly important when dealing with institutional identifiers like NIM and NIDN.

4.4.4. Student Data Entry Interface

Figure 5 presents the second stage of the HibahQu role-based login process, in which students proceed to the ”Enter Data” step after selecting the student role. At this stage, the interface focuses on collecting the student identification number (NIM) through a clearly labeled input field designed for clarity and ease of use. A stepper displayed at the top highlights the current phase of the three-step authentication flow, helping users understand their progress and reducing uncertainty during the login process. The layout remains minimal, clean, and visually balanced, featuring a white data-entry card that draws attention to the NIM input field and two navigation buttons labeled ”Back” and ”Next”, which support smooth movement between steps. This design ensures a focused, intuitive, and role-appropriate login experience for student users.

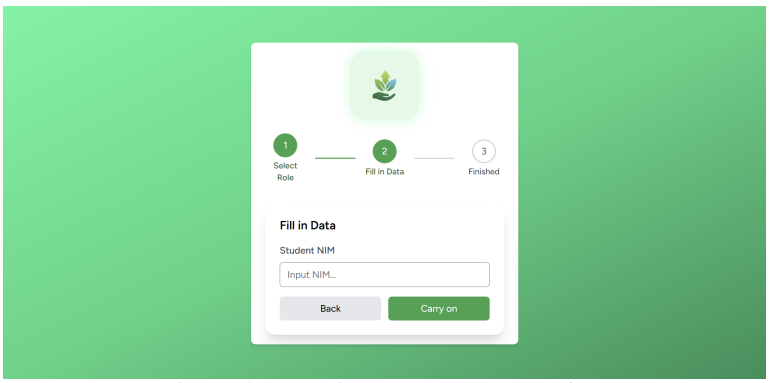

Figure 5. Mahasiswa NIM Input Interface

Figure 5 depicts the subsequent stage of the authentication process, in which students are guided to enter their NIM after selecting the appropriate role. This step is designed to maintain continuity with the previous interface, presenting a clean, structured, and minimally intrusive layout that prioritizes clarity and guided interaction. The interface displays a clearly labeled input field dedicated specifically to "Student NIM" ensuring that users immediately recognize the type of data required before proceeding. The step-by-step progression indicator at the top transitioning from Role Selection to Data Entry reinforces user orientation by showing the current stage and the remaining process. This visual cue reduces uncertainty and supports efficient task completion, especially for new users unfamiliar with the system workflow. The form itself adopts spacious alignment, soft shadowing, and neutral color combinations, allowing users to focus on the input field without visual distractions. Functionality is further enhanced through the presence of two action buttons "Back" and "Next" providing students with clear decision points for navigation. This dual button configuration supports error recovery by allowing users to return to the previous step if they selected the wrong role or need to correct initial choices. Meanwhile, the "Next" button ensures a straightforward transition to the verification stage once the NIM is correctly entered.

4.4.5. Lecturer Data Entry Interface

Figure 6 presents the second stage of the HibahQu role-based login process for lecturers, in which users who select the lecturer role proceed to the "Enter Data" step to input their NIDN. At this stage, the interface displays a clean and focused data-entry layout featuring a clearly labeled NIDN input field designed to ensure accuracy and reduce potential formatting errors. The stepper component at the top highlights the current phase within the three-step login sequence, providing lecturers with a clear sense of progression and reducing uncertainty throughout the authentication process. Consistent with the system's overall visual style, the page employs a minimal and well-balanced layout with a dedicated white card that draws attention to the NIDN input field, supported by "Back" and "Next" buttons that facilitate smooth navigation. This structured presentation ensures that lecturers experience a streamlined, intuitive, and role-specific login flow aligned with the system's commitment to clarity and usability.

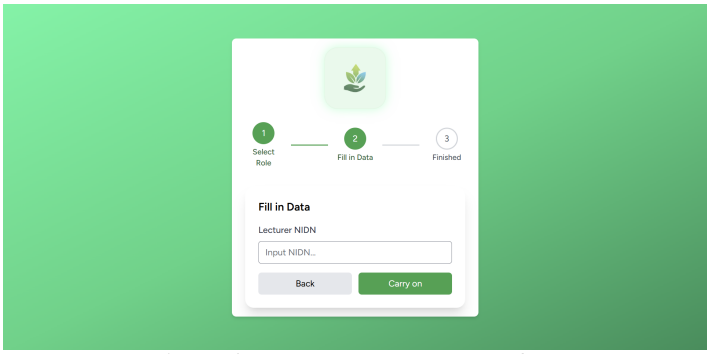

Figure 6. Dosen NIDN Input Interface

Figure 6 shows the second stage of the HibahQu role-based login process for lecturers, in which users who select the lecturer role are directed to the "Enter Data" step to input their NIDN. The interface maintains the same clean, minimal, and structured design as the student version, featuring a centrally aligned data-entry card with a clearly labeled NIDN input field to ensure accuracy and reduce input errors. The stepper component at the top continues to guide users through the three-step authentication flow by highlighting the active stage, helping lecturers understand their current position within the process. To support seamless navigation, the interface provides consistently styled "Back" and "Next" buttons, allowing users to move backward or proceed to the verification stage smoothly. By preserving visual consistency while adapting the input field to lecturer-specific credential formats, this step ensures a focused, role-appropriate, and user-friendly login experience for academic staff.

4.4.6. Final Submission Interface

Figure 7 presents the final stage of the HibahQu role-based login process, in which users both students and lecturers reach the "Complete" step after entering their NIM or NIDN. At this stage, the interface confirms

that all required data has been provided and prompts users to finalize the login flow by clicking the ”Submit” button. The layout preserves the system’s clean and minimal visual design, featuring a centrally aligned confirmation card that clearly communicates the completion of the data entry process and provides a concise instruction: ”Click Submit to save your role and continue.” The stepper component at the top indicates that the user has reached the final phase of the three-step sequence, reinforcing clarity and reducing uncertainty. The interface also includes a ”Back” button, allowing users to revise previously entered information if necessary, while the primary ”Submit” button is visually emphasized to guide users forward. This structured and well-balanced presentation enhances usability and ensures that users can confidently finalize their role selection and proceed to the next stage of system authentication in a seamless and intuitive manner.

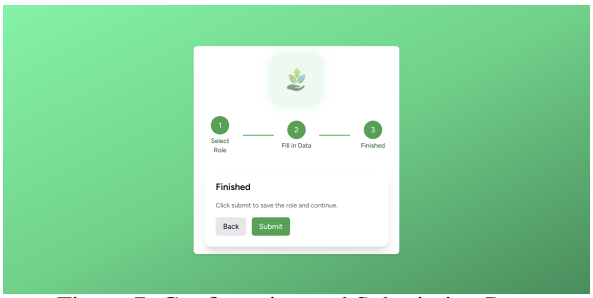

Figure 7. Confirmation and Submission Page

Figure 7 illustrates the final stage of the HibahQu role-based login process, in which users reach the ”Complete” step after selecting their role and entering the required credentials. At this stage, the interface presents a clean and focused confirmation layout that informs users that all necessary information has been successfully completed and is ready for submission. A centrally aligned white card displays a clear instructional message ”Click Submit to save your role and continue” guiding users to finalize the login sequence. The stepper component at the top highlights the final phase of the three-step process, providing reassurance that the user has reached the end of the authentication flow. Navigation controls remain consistent, with a ”Back” button allowing users to revisit previous steps if corrections are required, while the prominently styled ”Submit” button encourages completion of the process. This final interface maintains visual simplicity, balanced spacing, and an intuitive information hierarchy, ensuring that users can confidently confirm their role and proceed smoothly into the HibahQu system.

4.4.7. Lecturer Dashboard

The Lecturer Dashboard provides a centralized interface that allows lecturers to monitor their active proposals and track progress throughout the grant review workflow. Key features are designed to enhance clarity and reduce navigation effort, including a Reviewer Notes Modal that enables lecturers to view reviewer feedback directly on the dashboard using Vue.js reactive data for real-time updates. The dashboard also includes a Visual Stepper Tracking component that displays proposal status through a clear color scheme green for completed stages, yellow for the current stage, and gray for pending stages allowing users to interpret progress quickly. Its structured layout and responsive design ensure consistent accessibility across devices, supporting efficiency and overall user convenience while maintaining transparency throughout the review process.

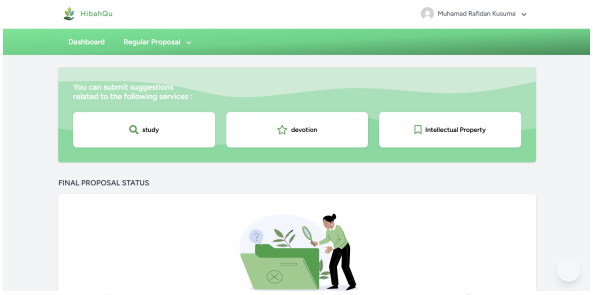

Figure 8. Lecturer Dashboard Interface

The dashboard interface shown in Figure 8 represents the main landing page that users encounter after successfully logging into the HibahQu system. This interface is designed to provide a clear overview of available services, including Research, Community Service, and Intellectual Property submissions, each accessible through prominently displayed interactive cards. The layout emphasizes simplicity and usability, ensuring that users can quickly navigate core features without cognitive overload, which aligns with the broader principles of enhancing user well-being through intuitive and supportive system design. Additionally, the use of clean typography, a green color palette, and structured spacing contributes to an aesthetically balanced and professional appearance, reinforcing both usability and visual comfort for users during proposal management activities.

4.4.8. Administrator Panel

The administrator panel is designed to streamline and accelerate the proposal verification process by providing an interface that is both efficient and responsive. When an administrator updates a proposal's status through the dropdown menu, Vue.js two-way data binding instantly synchronizes the selected value with the underlying data model and triggers a PUT request to the API without requiring any additional user action. Once the API processes the update, the system automatically refreshes the tracking data in real time, ensuring that the latest status changes appear immediately without the need for a page reload. To enhance clarity and user confidence, a toast notification is displayed to confirm that the update has been successfully applied. This combination of automated data synchronization, real-time state updates, and immediate visual feedback significantly improves the workflow efficiency and reliability of administrative tasks within the system.

4.5. Discussion

Based on the implementation and testing results, the developed system demonstrates several significant advantages. First, in terms of responsiveness, the use of a SPA architecture substantially reduces user waiting time, with measurements conducted using Chrome DevTools showing that the average rendering time for proposal detail pages is approximately 150 ms after the initial load much faster than traditional server-side rendering approaches that typically require 1–2 seconds for a complete page reload. In addition, the system effectively enhances grant flow transparency by addressing the issue of "status blindness," as the Visual Stepper Component clearly displays the current stage of each proposal, while the proposal trackings table maintains a chronological record of status changes, ensuring accountability throughout the review and decision-making process. From an architectural perspective, Laravel 12 contributes to improved efficiency through optimized routing, enhanced cache management, and reliable handling of concurrent requests. The separation between the frontend and backend layers also supports future scalability, enabling the same API to be reused for mobile applications without modifying the core business logic. Overall, the integration of a Laravel backend with a Vue.js SPA frontend results in a grant monitoring system that is technically reliable, transparent, fast, user-friendly, and scalable for future enhancements, thereby accelerating information access and improving user satisfaction throughout the grant submission workflow.

5. MANAGERIAL IMPLICATION

The implementation of a modern grant monitoring system based on Laravel 12 and Vue.js provides significant managerial benefits for higher education institutions, particularly in accelerating administrative workflows. The system's real-time status tracking, automated notifications, and structured proposal workflow reduce manual coordination efforts between lecturers and LPPM administrators. As a result, management can allocate resources more efficiently, minimize bottlenecks in proposal validation, and ensure that grant cycles run on schedule. These improvements also help institutions uphold principles of transparency and accountability, which are essential for supporting high-quality research governance.

From an operational perspective, the adoption of a SPA architecture enhances decision-making by providing administrators with faster access to accurate and consolidated information. The availability of a complete audit trail through the proposal tracking module allows leaders to monitor reviewer performance, identify process delays, and evaluate institutional workload capacity. This enables data-driven policy adjustments, such as optimizing reviewer distribution, improving SOP alignment, or redesigning grant timelines to reduce administrative pressure.

On a strategic level, the developed system strengthens institutional credibility and user trust, which are crucial for sustaining long-term research productivity. By reducing uncertainty and increasing the clarity of

grant processes, lecturers experience improved emotional comfort and perceived fairness factors that can indirectly boost research engagement and proposal submission rates. Additionally, the scalable architecture opens opportunities for future managerial initiatives, including integration with mobile platforms, digital signatures, and analytics dashboards that can support strategic planning at the faculty or university level.

6. CONCLUSION

Based on all stages conducted throughout this research from requirement analysis, system architecture design, implementation, to functional testing several key conclusions can be drawn. The findings demonstrate that the integration of Laravel 12 as the backend service provider and Vue.js as the frontend interface offers an effective modern web architecture capable of delivering a stable, efficient, and high-performance grant monitoring system. The adoption of the SPA approach successfully addresses the research gap found in traditional web systems that frequently suffer from latency, limited responsiveness, and poor navigation fluidity. In contrast, the implemented system supports instant menu transitions and real-time proposal status retrieval without page reloads, achieving a higher level of usability and interaction efficiency.

The system also introduces a notable novelty through the combination of the Visual Stepper feature and detailed status history, which effectively resolves longstanding transparency issues and mitigates the "status blindness" commonly experienced by proposing lecturers. This improvement aligns with the broader principle of supporting human well-being, as it minimizes emotional uncertainty while simultaneously reducing the administrative burden on LPPM staff. Furthermore, results from Black Box Testing confirm that core system functionalities including authentication, proposal validation, and multi-level status updates operate reliably and consistently. These outcomes are supported by strong backend validation mechanisms and well-structured Vue.js components that uphold system stability and maintain high data integrity.

Despite achieving its primary objectives, the system still presents opportunities for further enhancement. Future work may include integrating multi-platform notifications to improve communication efficiency, implementing secure digital signatures or QR-based verification to strengthen document authenticity, and expanding the platform with intelligent analytics capabilities to support institutional decision-making. These future research directions not only enhance functionality and user experience but also position the system for transformation into a more holistic, data-driven Decision Support System (DSS) that promotes organizational well-being and ensures long-term adaptability.

7. DECLARATIONS

7.1. About Authors

Indira Puspa Gustiah (IP) https://orcid.org/0009-0001-6482-9024

Ninda Lutfiani (NL) https://orcid.org/0000-0001-7019-0020

Muhamad Rapidan Kusuma (MR) https://orcid.org/0009-0004-8447-2255

Etty Puji Lestari (EP) https://orcid.org/0000-0002-3501-5400

Thomas Green (TG) https://orcid.org/0009-0001-7048-2231

7.2. Author Contributions

Conceptualization: TG; Methodology: MR; Software: EP; Validation: NL and IP; Formal Analysis: NL and IP; Investigation: EP; Resources: MR; Data Curation: TG; Writing Original Draft Preparation: NL and MR; Writing Review and Editing: IP and TG; Visualization: MR; All authors, IP, NL, MR, EP, and TG, have read and agreed to the published version of the manuscript.

7.3. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

7.4. Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

7.5. Declaration of Conflicting Interest

The authors declare that they have no conflicts of interest, known competing financial interests, or personal relationships that could have influenced the work reported in this paper.

REFERENCES

[1] L. G. SIANCAS JULCAHUANGA and J. J. MARTINEZ CHUNGA, "Software reengineering with laravel and vue. js to optimize the performance of the web system in a pharmacy chain, piura, 2024," *Js to Optimize the Performance of the Web System in a Pharmacy Chain, Piura*, 2024.

[2] M. N. Ayubi and A. Retnowardhani, "Optimizing learning experiences: A study of student satisfaction with lms in higher education," *Aptisi Transactions on Technopreneurship (ATT)*, vol. 7, no. 2, pp. 527–541, 2025.

[3] I. M. H. Al-Badri, A. Aouragh, and K. AlDelfi, "Implementing vue. js course on moodle platform," 2023.

[4] F. A. Purnomo, D. M. Setyaningsih, G. M. Liset, R. A. Nurfi, S. Sulistyawan, W. G. Suci, A. Purwanto, and H. K. K. Aliwarga, "Implementation of vue js and laravel on monitoring battery management system," *Applied Mechanics and Materials*, vol. 918, pp. 95–105, 2024.

[5] United Nations, "The 17 sustainable development goals," https://sdgs.un.org/goals, United Nations, n.d., accessed: 2025-12-12. [Online]. Available: https://sdgs.un.org/goals

[6] M. A. Setiawan, H. Hartoyo, K. B. Seminar, B. Sartono, R. Fitriati, and V. Ginting, "Improving e-service quality of indonesian toll road application with entrepreneurship insights," *Aptisi Transactions on Technopreneurship (ATT)*, vol. 7, no. 2, pp. 503–515, 2025.

[7] Y. Simbolon and A. Pakpahan, "Design of e-consultation service information system for gmahk congregation members to web-based pastors using laravel and vue. js," *TeIKa*, vol. 14, no. 2, pp. 143–158, 2024.

[8] S. Riyadi, N. Sakinah *et al.*, "Implementation of vue. js framework for front-end development: A study case of employee information system," in *2024 4th International Conference on Electronic and Electrical Engineering and Intelligent System (ICE3IS)*. IEEE, 2024, pp. 105–109.

[9] E. J. A. H. Nasution, "Digital governance model for zakat based on mui fatwas in indonesia," *International Journal of Cyber and IT Service Management (IJCITSM)*, vol. 5, no. 2, pp. 223–234, 2025.

[10] D. Popa and I. Buciu, "Laravel and vue. js as tools to control iot devices over the internet. current state-of-the-art." *INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL*, vol. 20, no. 3, 2025.

[11] K. A. Seputra and L. J. E. Dewi, "A design of patient registration apps using flutter, laravel and, vue js," in *IConVET 2021: Proceedings of the 4th International Conference on Vocational Education and Technology, IConVET 2021, 27 November 2021, Singaraja, Bali, Indonesia*. European Alliance for Innovation, 2022, p. 71.

[12] S. A. Sibagariang, N. Septiani, and A. Rodriguez, "Enhancing educational management through social media and e-commerce-driven branding," *International Journal of Cyber and IT Service Management (IJCITSM)*, vol. 5, no. 2, pp. 235–245, 2025.

[13] T. Irhansyah, M. I. P. Nasution *et al.*, "Development of thesis repository application in the faculty of science and technology use implementation of vue. js framework," *Journal of Information Systems and Technology Research*, vol. 2, no. 2, pp. 66–77, 2023.

[14] U. Onwudebelu, O. Fasola, E. P. Okeh, and U. C. Ugwuoke, "Developing a digital crowdfunding platform for student empowerment in higher institutions: Insights from ae-funai," *Faculty of Natural and Applied Sciences Journal of Computing and Applications*, vol. 2, no. 3, pp. 8–36, 2025.

[15] F. Sutisna, N. Lutfiani, E. Anderson, D. Danang, and M. O. Syaidina, "E-commerce and digital marketing strategies: Their impact on startupreneur performance using pls-sem," *IAIC Transactions on Sustainable Digital Innovation (ITSDI)*, vol. 6, no. 2, pp. 215–223, 2025.

[16] A. Niazi, "Svs-web: design and implementation of cloud web-based svs-efie solver," 2024.

[17] T. Maulida, A. N. A. Zumaroh, H. A. A. Rozaq, A. Y. Syafa'at, R. S. Ananda, I. Tahyudin, B. P. Hartato *et al.*, "Visualization of front-end data logger internet of things technology using vue. js framework," in *2022 6th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE)*. IEEE, 2022, pp. 693–698.

[18] R. A. Sunarjo, T. Pujiati, D. Apriliasari, and M. Hardini, "Digital onboarding in agricultural platforms

and its impact on agricultural productivity," *IAIC Transactions on Sustainable Digital Innovation (ITSDI)*, vol. 6, no. 2, pp. 205–214, 2025.

[19] P. Ťapák and K. Zvarová, "Iot hatchery controlled by web application," in *2022 Cybernetics & Informatics (K&I)*. IEEE, 2022, pp. 1–6.

[20] B. Jiang, "The application of vue. js framework technology in multidomain automated testing systems," *Automatic Control and Computer Sciences*, vol. 59, no. 4, pp. 455–466, 2025.

[21] Z. Kedah, "Use of e-commerce in the world of business," *Startupreneur Business Digital (SABDA Journal)*, vol. 2, no. 1, pp. 51–60, 2023.

[22] J. Quinten, *Building Real-World Web Applications with Vue. js 3*. Packt Publishing, 2024.

[23] D. S. Wuisan and T. Handra, "Maximizing online marketing strategy with digital advertising," *Startupreneur Business Digital (SABDA Journal)*, vol. 2, no. 1, pp. 22–30, 2023.

[24] A. S. Setiabudhi, E. D. Wahyuni, and N. C. Wibowo, "Frontend design and development of thesis management application using nextjs in physics study program upn "veteran" jatim," *Jurnal Teknologi dan Open Source*, vol. 8, no. 1, pp. 142–152, 2025.

[25] J. Siswanto, V. A. Goeltom, I. N. Hikam, E. A. Lisangan, and A. Fitriani, "Market trend analysis and data-based decision making in increasing business competitiveness," *Sundara Advanced Research on Artificial Intelligence*, vol. 1, no. 1, pp. 1–8, 2025.

[26] Q. Odeniran, "Comparative analysis of fullstack development technologies: Frontend, backend and database," 2023.

[27] M. A. Salvador, K. A. Botangen, M. C. Rabang, I. C. Salinas, M. Naagas, and A. Balagot, "Development of a web-based research consortium database management system: advancing data-driven and knowledge-based project management," in *Proceedings of the 2024 5th International Conference on Computing, Networks and Internet of Things*, 2024, pp. 87–92.

[28] T. Pujiati, H. Setiyowati, B. Rawat, N. P. L. Santoso, and M. G. Ilham, "Exploring the role of artificial intelligence in enhancing environmental health: Utaut2 analysis," *Sundara Advanced Research on Artificial Intelligence*, vol. 1, no. 1, pp. 37–46, 2025.

[29] E. Arrieta Rodriguez, L. F. Murillo Fernandez, G. A. Castañez Orta, A. M. Rivas Horta, C. Baldovino Barco, K. Jimenez Barrionuevo, D. Cama-Pinto, F. M. Arrabal-Campos, J. A. Martínez-Lao, and A. Cama-Pinto, "A platform for inpatient safety management based on iot technology," *Inventions*, vol. 7, no. 4, p. 116, 2022.

[30] E. Arrieta Rodriguez and L. Murillo Fernandez, "Castañez orta," *GA, Rivas Horta, AM, Baldovino Barco, C., Jimenez Barrionuevo, K., Cama-Pinto, D., Arrabal-Campos, FM, Martínez-Lao, JA, & Cama-Pinto, A*, 2022.

[31] P. H. P. Tan, S. Wijaya, U. Rahardja, B. N. Henry, and A. Anjani, "Modeling the impact of digital literacy on ai based learning adoption through perceived usefulness and easeof use," *Sundara Advanced Research on Artificial Intelligence*, vol. 1, no. 2, pp. 56–64, 2025.

[32] Y. Simbolon and A. Pakpahan, "Design of e-consultation service information system for gmahk congregation members to web-based pastors using laravel and vue.js," *TeIKa*, vol. 14, no. 2, pp. 143–158, 2024.

[33] F. A. Purnomo, D. M. Setyaningsih, G. M. Liset, R. A. Nurfi, S. Sulistyawan, W. G. Suci, A. Purwanto, and H. K. K. Aliwarga, "Implementation of vue js and laravel on monitoring battery management system," *Applied Mechanics and Materials*, vol. 918, pp. 95–105, 2024.

[34] D. Popa and I. Buciu, "Laravel and vue.js as tools to control iot devices over the internet: Current state-of-the-art," *International Journal of Computers Communications & Control*, vol. 20, no. 3, 2025.

[35] S. Riyadi, N. Sakinah *et al.*, "Implementation of vue.js framework for front-end development: A case study of employee information system," in *2024 4th International Conference on Electronic and Electrical Engineering and Intelligent System (ICE3IS)*. IEEE, 2024, pp. 105–109.

[36] T. Irhansyah, M. I. P. Nasution *et al.*, "Development of thesis repository application in the faculty of science and technology using vue.js framework," *Journal of Information Systems and Technology Research*, vol. 2, no. 2, pp. 66–77, 2023.

[37] B. Jiang, "The application of vue.js framework technology in multidomain automated testing systems," *Automatic Control and Computer Sciences*, vol. 59, no. 4, pp. 455–466, 2025.

[38] T. Maulida, A. N. A. Zumaroh, H. A. A. Rozaq, A. Y. Syafa'at, R. S. Ananda, I. Tahyudin, B. P. Hartato *et al.*, "Visualization of front-end data logger internet of things technology using vue.js framework,"

in *2022 6th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE)*. IEEE, 2022, pp. 693–698.

[39] M. N. Ayubi and A. Retnowardhani, "Optimizing learning experiences: A study of student satisfaction with lms in higher education," *Aptisi Transactions on Technopreneurship (ATT)*, vol. 7, no. 2, pp. 527–541, 2025.

[40] E. J. A. H. Nasution, "Digital governance model for zakat based on mui fatwas in indonesia," *International Journal of Cyber and IT Service Management (IJCITSM)*, vol. 5, no. 2, pp. 223–234, 2025.

[41] M. A. Salvador, K. A. Botangen, M. C. Rabang, I. C. Salinas, M. Naagas, and A. Balagot, "Development of a web-based research consortium database management system: Advancing data-driven and knowledge-based project management," in *Proceedings of the 2024 5th International Conference on Computing, Networks and Internet of Things*, 2024, pp. 87–92.

[42] J. Perera, "E-commerce platform for pharmaceutical trade (imedic)," Ph.D. dissertation, 2021.

[43] Y. Huang and M. Chen, "A modular web-based administrative monitoring system using laravel and vue.js," *International Journal of Web Engineering and Technology*, vol. 19, no. 2, pp. 123–139, 2024.

[44] Sila and R. Pratama, "Grant submission monitoring system based on orange technology using laravel and vue.js," *International Journal of Information System Innovation*, vol. 5, no. 3, pp. 112–123, 2021.

[45] Z. Li and H. Wang, "Information system design for research funding management and performance monitoring," *Journal of Information and Organizational Sciences*, vol. 47, no. 2, pp. 271–285, 2023.

[46] A. Rahman, B. Prasetyo, and W. Lestari, "Design and implementation of a web-based grant management and monitoring system for higher education institutions," *Journal of Information Systems Engineering and Business Intelligence*, vol. 10, no. 1, pp. 45–56, 2024.

[47] X. Feng, J. Wachs, S. Daniotti, and F. Neffke, "The building blocks of software work explain coding careers and language popularity," *arXiv preprint arXiv:2504.03581*, 2025.

[48] R. Yazdanian, R. West, and P. Dillenbourg, "Keeping up with the trends: Analyzing the dynamics of online learning and hiring platforms in the software programming domain," *International Journal of Artificial Intelligence in Education*, vol. 31, no. 4, pp. 896–939, 2021.

[49] S. F. Bakara and Y. D. Astanti, "Life cycle assessment (lca) for preparation of environmental impact reduction proposal on cheddar cheese production process," in *Conference Proceeding*, p. 187.

[50] L. G. N. Bastos, "An improved architecture for a telecommunications organization information systems: As is–to be," Master's thesis, Universidade NOVA de Lisboa (Portugal), 2023.

[51] M. Alsharif and F. Alotaibi, "Design of a research grant management information system to improve transparency and monitoring," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 6, pp. 612–620, 2023.

[52] J. C. Fernandez and L. M. Ortega, "Web-based monitoring systems for institutional decision support," *Journal of Information Systems Research and Innovation*, vol. 18, no. 1, pp. 1–12, 2024.

[53] R. Wijaya and L. Chandra, "Orange technology-driven grant monitoring system using laravel and vue.js," in *Proceedings of the 2022 International Conference on Smart Computing and Innovation*. IEEE, 2022, pp. 144–152.

[54] R. Singh and N. Kaur, "Modern mvc-based web application development for scalable information systems," *International Journal of Computer Applications*, vol. 185, no. 7, pp. 25–31, 2023.

[55] UNESCO, "Monitoring and evaluation information systems in research and innovation management," *UNESCO Digital Library*, 2022.

[56] C. Okafor and S. Bello, "A web-based research funding management system for transparency and accountability," *International Journal of Information Technology and Management*, vol. 22, no. 4, pp. 389–404, 2023.