Research Paper

Web-Based ISBN Registration System Development at LPPM UPN "Veteran" Yogyakarta Using the RAD Method

Dhimas Arief Dharmawan*, Antik Suprihanti, Sri Dwi Ari Ambarwati,
Muhammad Almas Farros Dhiyaulhaq, Noveanto Nur Akbar, Alva Raymon Yehudha
Universitas Pembangunan Nasional Veteran Yogyakarta, Indonesia

Received : Sept 27, 2025	Revised : Sept 30, 2025	Accepted : Sept 30, 2025	Online : October 14, 2025

Abstract

This research aims to develop an integrated web-based information system to support the registration process of the International Standard Book Number (ISBN) at the Institute for Research and Community Service (LPPM), UPN "Veteran" Yogyakarta. The background of this study lies in the limitations of the current system, which relies on Google Forms. This existing approach is considered inefficient, unstructured, and vulnerable to data security breaches. The research methodology includes several stages: data collection, requirement analysis, selection of system development methods, design, implementation, testing, and evaluation. The Rapid Application Development (RAD) approach was chosen as it enables fast and iterative system development with direct user involvement. The expected outcome of this study is a web-based system that enhances efficiency, effectiveness, and transparency in managing ISBN processes, while also contributing to the achievement of the institution's Key Performance Indicator (KPI) 2.3. Furthermore, the system is expected to generate positive impacts in technological, social, and economic aspects within the academic environment.

Keywords: *ISBN, Web-Based Information System, Rapid Application Development, Higher Education*

INTRODUCTION

The Institute for Research and Community Service (LPPM) of UPN "Veteran" Yogyakarta plays a vital role in supporting research and community service activities. One of its key responsibilities is managing International Standard Book Numbers (ISBNs), which are essential for scientific publications and academic productivity. Currently, ISBN submissions are handled through Google Forms. While convenient for data collection, this method has drawbacks, including limited structured data management, a lack of transparency in tracking submission status, and potential data security risks, all of which reduce service efficiency for lecturers and researchers.

To improve service quality and support academic productivity, LPPM has introduced programs such as the Technical Guidance on Book Writing and ISBN Registration held on October 9, 2024. This initiative aimed to equip academics with practical writing skills and knowledge of ISBN procedures. However, without an integrated system, such efforts cannot achieve optimal results. Recent data show a sharp increase in ISBN submissions, reflecting the growing research and publication climate at UPN "Veteran" Yogyakarta. However, the lack of an efficient management system limits the full impact of these activities.

In alignment with the Internal Research Grant Guidelines of LPPM UPN "Veteran" Yogyakarta, which emphasize the importance of enhancing research outputs through publications and academic books, the development of an integrated web-based system for ISBN management emerges as a critical necessity. Such a system is expected to provide a reliable platform for facilitating submission, monitoring, and management of ISBN registration. By doing so, it will not only improve the services offered to lecturers and researchers but also strengthen the university's

Corresponding author's email: dhimas.arief@upnyk.ac.id

contribution to scientific dissemination and academic excellence.

Previous studies on web-based academic publication systems have primarily focused on general research output management, library services, or productivity monitoring. However, very few have specifically addressed ISBN registration services at the institutional level, particularly in the context of Indonesian higher education. Most existing systems rely on basic content management or form-based submissions, which lack integrated tracking, transparency, and user validation mechanisms. This study fills the gap by proposing a Scrum-based development of a dedicated ISBN management system at LPPM UPN "Veteran" Yogyakarta (Alamsyah & Yuliansyah, 2017). The novelty of this work lies in its focus on (1) integrating submission, monitoring, and reporting features in a single platform, (2) applying an agile development process with active stakeholder involvement, and (3) aligning the system outcomes with institutional performance indicators (KPI 2.3). By doing so, the system provides a more efficient, transparent, and user-centered approach to ISBN management, which has not been sufficiently explored in prior works (Pappas et al., 2017).

LITERATURE REVIEW

ISBN Management in Higher Education

The International Standard Book Number (ISBN) is a unique identifier for book publications, facilitating the distribution and cataloging of books. In Indonesia, the National Library serves as the ISBN authority, setting ISBN service policies that include administrative sanctions for publishers who violate regulations, such as improper use of ISBNs or delays in registration. For universities, understanding ISBN service policies is crucial to ensure that academic publications comply with established standards and regulations. Compliance not only enhances the credibility of the institution's academic outputs but also improves their distribution and visibility.

Web-Based System Development for ISBN Management

Studies on the design of web-based ISBN management systems suggest that the use of Content Management Systems (CMS) combined with a SWOT analysis can significantly enhance the development of effective systems. Web-based platforms allow for easier access and more structured data management, enabling academic staff to submit and monitor the status of their applications more efficiently. Such systems ensure transparency, improve user engagement, and reduce administrative workload, making them an increasingly important solution for higher education institutions.

Benefits of Effective ISBN Management

Effective ISBN management within universities can enhance the quality of research and strengthen competitiveness in implementing the Tridharma. Furthermore, the optimal utilization and management of ISBNs can support entrepreneurial development within academic environments. Therefore, the development of a web-based system for ISBN management at LPPM UPN "Veteran" Yogyakarta is expected to increase service efficiency and effectiveness, facilitate lecturers and researchers in submission and monitoring, and support the growth in both the quality and quantity of academic publications.

Previous Studies

Several higher education institutions in Indonesia have developed web-based information systems to manage research outcomes. Universitas Raharja developed a CMS-based system supported by a SWOT analysis to simplify dissemination (Raharja et al., 2021), while Universitas Dhyana Pura utilized Symfony to generate research output reports and monitor lecturers'

productivity (Wibowo & Putri, 2020). Institut Teknologi Kalimantan transitioned from manual tools, such as email and Google Forms, to a structured Management Information System for greater efficiency (Santoso et al., 2022). Politeknik Negeri Padang emphasized information systems as a response to industry and globalization challenges (Hamzah & Nurhalim, 2019). Universitas Negeri Makassar adopted the SDLC model to develop a structured and accessible scientific publication management system (Universitas Negeri Makassar, 2021). Collectively, these studies highlight that implementing web-based systems for academic publication management, including ISBN, enhances efficiency, effectiveness, and transparency, supporting better research quality and innovation.

RESEARCH METHOD

This study aims to develop a web-based information system to enhance the management of International Standard Book Numbers (ISBN) at the Institute for Research and Community Service (LPPM) of UPN "Veteran" Yogyakarta. Each phase is described in detail as follows.

Data Collection

The first phase focuses on collecting data to understand current ISBN management practices within the institution. Direct observation was used to document workflows for submissions, verifications, and reporting. Interviews with LPPM staff, lecturers, and researchers provided insights into their experiences, challenges, and expectations, ensuring the design meets user needs. A literature review was also conducted to identify prior studies and best practices in academic information systems as references for the project.

Requirement Analysis

The collected data were then analyzed to identify both functional and non-functional requirements of the system. Functional requirements included features such as registration of ISBN requests, document verification, progress tracking, and report generation. Non-functional requirements involved aspects such as system security, usability, performance, and scalability. This analysis provided a comprehensive understanding of the organizational processes and helped to determine the scope and boundaries of the proposed system.

Selection of Development Method

After defining the requirements, an appropriate system development method was selected (Mishra & Dubey, 2013). The Scrum methodology was adopted as the framework for system development (Schwaber & Sutherland, 2020). Scrum is an agile-based approach that emphasizes iterative development, team collaboration, and continuous feedback (Hardiansyah et al., 2023; Alim et al., 2023). The methodology is implemented through sprint cycles, where each sprint delivers incremental features that can be tested and evaluated by users. This iterative process ensures that user requirements are met effectively while minimizing development risks (Rodríguez et al., 2012). The involvement of LPPM staff and researchers as stakeholders throughout the sprints strengthens the alignment between system features and organizational needs.

System Design

In this phase, the design of the system architecture, user interface, and database structure was carried out. The user interface was designed to be intuitive and user-friendly, reducing the learning curve for non-technical users such as lecturers and administrative staff. The database design focused on ensuring efficient storage, retrieval, and security of sensitive data, including ISBN records. The system architecture was developed to support scalability, allowing the application to accommodate future enhancements and potential integration with other institutional systems.

Wireframes and mockups were prepared to visualize the interaction between users and the system before moving to implementation.

System Implementation

The implementation stage involved translating the system design into functional code (Sukamto & Shalahuddin, 2018). The chosen programming languages, frameworks, and tools were selected to ensure stability, compatibility, and ease of maintenance (Fikri, 2023). Features were implemented in stages, beginning with core modules such as user registration, submission management, and verification, followed by additional features like reporting and notifications. Iterative prototyping was employed to incorporate user feedback at each development cycle. This approach helped to minimize errors and align the development output with user needs.

Testing and Evaluation

Once the system was implemented, it underwent comprehensive testing to verify its functionality and performance. Functional testing ensured that each feature functioned according to the requirements, while usability testing evaluated whether the interface was easy to navigate and met the expectations of end-users. Performance testing assessed the system's ability to handle concurrent access and data loads. Feedback from testing sessions was used to refine and improve the system prior to its final deployment. Evaluation also included discussions with stakeholders to identify potential improvements for future versions of the system, ensuring that it remains adaptable to evolving organizational needs.

FINDINGS AND DISCUSSION

After the design phase of the ISBN management website using the Scrum methodology, the next step is to implement the backend, transforming the design into a functional system. This involves developing system logic, configuring the database, and managing data security. The logic ensures that processes such as ISBN submission, verification, and status tracking follow the defined workflow, while database configuration provides structured storage for scalability and reliability. Security measures are also applied to protect sensitive information and maintain data integrity. At this stage, the website is expected to function according to specifications, enabling lecturers and researchers to manage ISBN applications effectively and securely, while laying the foundation for testing and deployment.

Implementation of Main Features

a. User Authentication (Login, Registration, and Password Recovery)

The authentication feature serves as the primary entry point for users to access the system. It is a critical component for both the security and usability of the ISBN management website. Its implementation begins with preparing a configuration file, config.php (Figure 1), which serves as the central connection between the application and the database. This file defines parameters such as hostname, username, password, and database name, ensuring proper connection and management of stored data. Additionally, config.php provides supporting functions that enhance efficiency and maintain consistency across modules.

url_dasar() Function

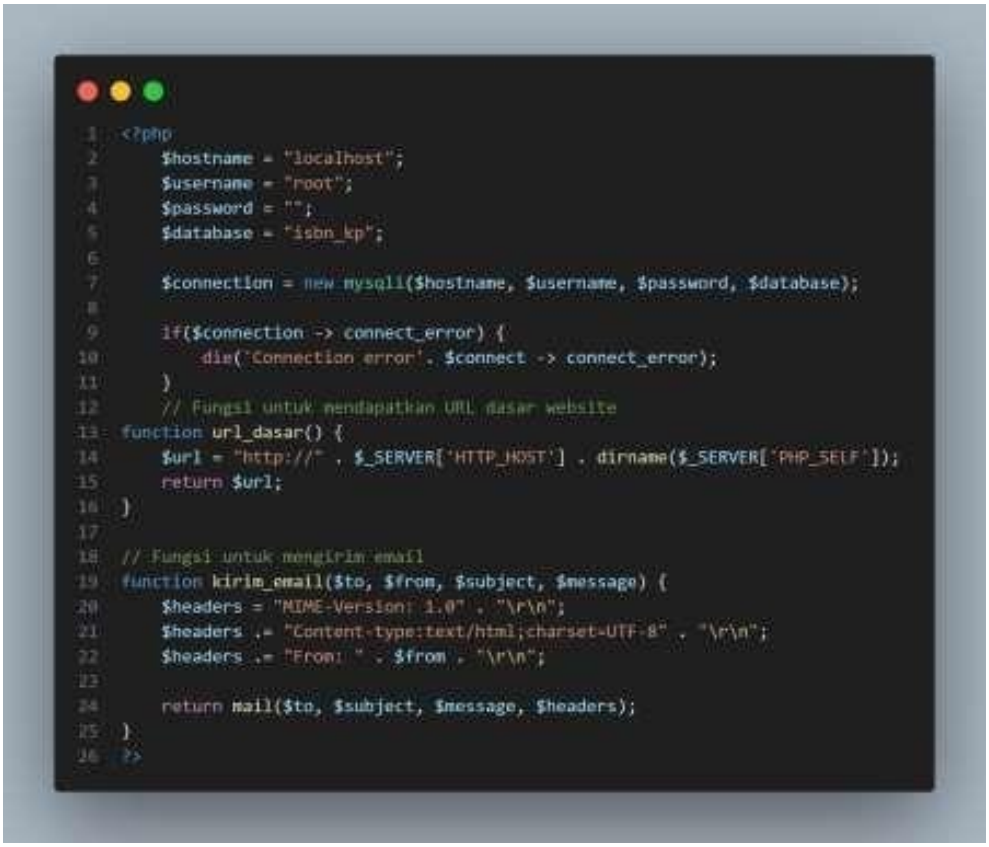
This function is designed to retrieve the base URL of the website dynamically. It simplifies the process of generating dynamic links throughout the system, ensuring that navigation and redirection are consistent even if the website's deployment environment changes.

kirim_email() Function

The kirim_email() function is implemented to facilitate email communication between the system

and its users. For instance, it is utilized in the password recovery feature (Forgot Password), where the system must securely send verification links or reset instructions to the user's registered email address.

By centralizing these configurations and utility functions in one file, the backend ensures consistency, reduces redundancy, and enhances maintainability. This approach also enables easier modifications to the database or service settings, as changes can be made in one location without affecting other parts of the application.



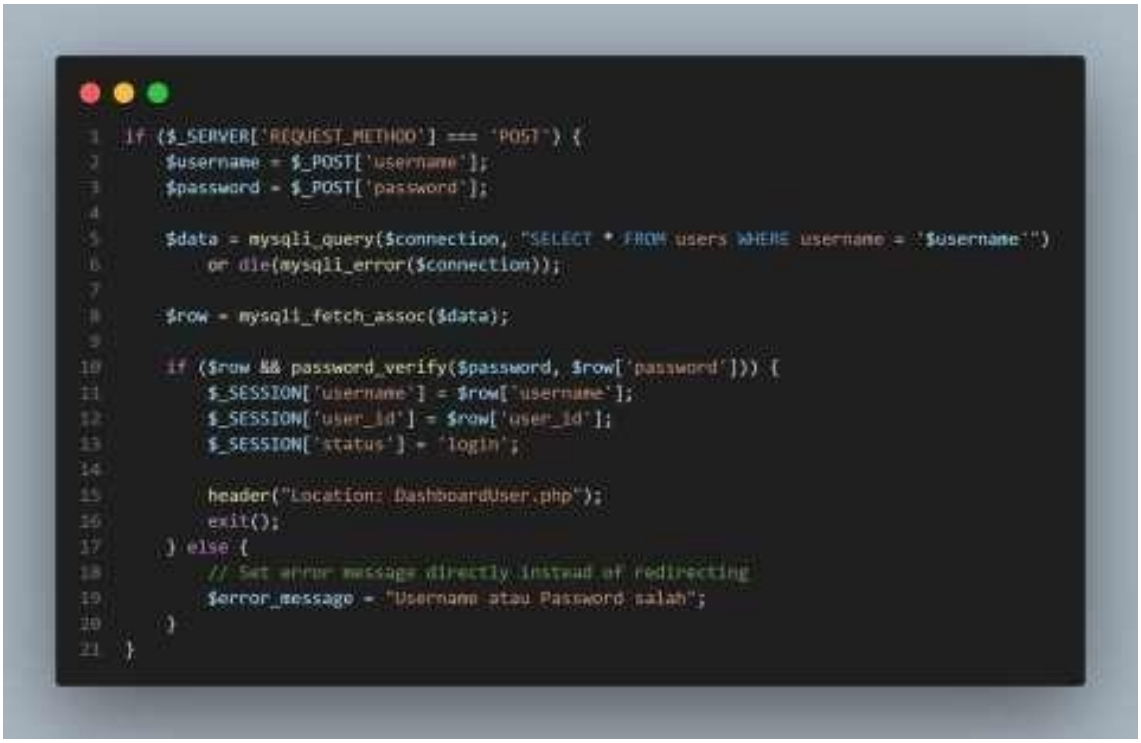
```php
<?php
    $hostname = "localhost";
    $username = "root";
    $password = "";
    $database = "isbn_kp";

    $connection = new mysqli($hostname, $username, $password, $database);

    if($connection -> connect_error) {
        die('Connection error'. $connect -> connect_error);
    }
    // Fungsi untuk mendapatkan URL dasar website
function url_dasar() {
    $url = "http://" . $_SERVER['HTTP_HOST'] . dirname($_SERVER['PHP_SELF']);
    return $url;
}

// Fungsi untuk mengirim email
function kirim_email($to, $from, $subject, $message) {
    $headers = "MIME-Version: 1.0" . "\r\n";
    $headers .= "Content-type:text/html;charset=UTF-8" . "\r\n";
    $headers .= "From: " . $from . "\r\n";

    return mail($to, $subject, $message, $headers);
}
?>
```

Figure 1. Our implementation for config.php.

From the backend perspective, authentication is implemented through three main processes:

Login Validation

The backend is responsible for verifying user credentials by matching the provided username and password against the stored data in the database. To enhance security, user passwords are encrypted using hashing algorithms before being stored, ensuring that sensitive information cannot be easily compromised. Session or token-based authentication mechanisms are also employed to maintain secure user sessions throughout system usage.



```php
if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    $username = $_POST['username'];
    $password = $_POST['password'];

    $data = mysqli_query($connection, "SELECT * FROM users WHERE username = '$username'")
        or die(mysqli_error($connection));

    $row = mysqli_fetch_assoc($data);

    if ($row && password_verify($password, $row['password'])) {
        $_SESSION['username'] = $row['username'];
        $_SESSION['user_id'] = $row['user_id'];
        $_SESSION['status'] = 'login';

        header("Location: DashboardUser.php");
        exit();
    } else {
        // Set error message directly instead of redirecting
        $error_message = "Username atau Password salah";
    }
}
```

Figure 2. Our implementation for the login verification.

User Registration

The registration process allows lecturers and researchers to create accounts, with backend validation ensuring completeness, uniqueness, and protection against attacks such as SQL injection or cross-site scripting. Validated data is securely stored in the database, and users receive confirmation of successful registration. Administrators can also add accounts through the backend, with checks for unique email addresses and strong passwords. User credentials are encrypted before storage to protect sensitive data and maintain database security.

```php
if (isset($_POST['email'], $_POST['username'], $_POST['password'], $_POST['user_id'])) {
        $email = mysqli_real_escape_string($connection, $_POST['email']);
        $username = mysqli_real_escape_string($connection, $_POST['username']);
        $password = $_POST['password'];
        $user_id = mysqli_real_escape_string($connection, $_POST['user_id']);

        // Cek apakah email sudah terdaftar
        $check_email_query = "SELECT * FROM users WHERE email = '$email'";
        $result = mysqli_query($connection, $check_email_query);
```

Figure 3. The user registration source code.

Password Recovery (Forgot Password)

The Forgot Password feature is designed to help users recover their accounts securely when credentials are lost or forgotten. The process begins when a user submits their registered email address, which the system verifies against the database. If valid, a temporary security token with

an expiration time is generated and sent to the user via a reset link. When accessed, the system validates the token before granting access to the reset page, where the user can create a new password. The new password is encrypted before being stored in the database, ensuring confidentiality and security. This mechanism simplifies password recovery while maintaining strong protection against unauthorized access.

Figure 4. Our implementation for the forgot password feature.

Figure 5. The source code that handles the reset password feature.

Logout

The logout feature is implemented to terminate a user's active session securely. It works by clearing all session data associated with the authenticated user. Once the session is destroyed, the system redirects the user to the login page with a confirmation message indicating successful logout. This ensures that access to the system is properly closed, maintaining security and preventing unauthorized use of the account after the user has exited.

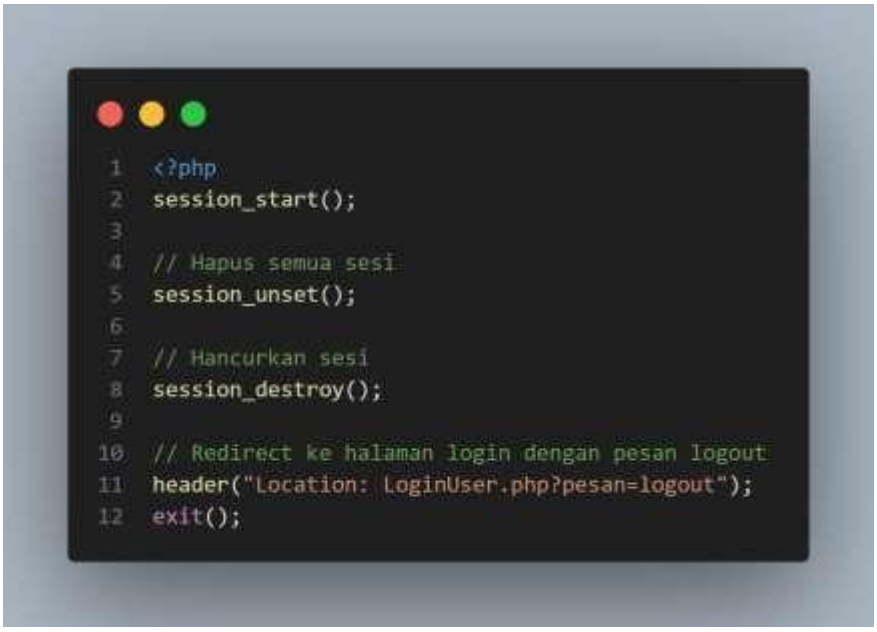


Figure 6. Our implementation for the Logout feature.

b. ISBN Application Management

This feature is used to manage ISBN applications submitted by users. The backend is responsible for handling data storage, data validation, and updating the application status. The ISBN application form enables users to submit requests by completing the required information. The backend validates the input to ensure it follows the correct format (e.g., date, author's name, or book title) before storing the data in the database.

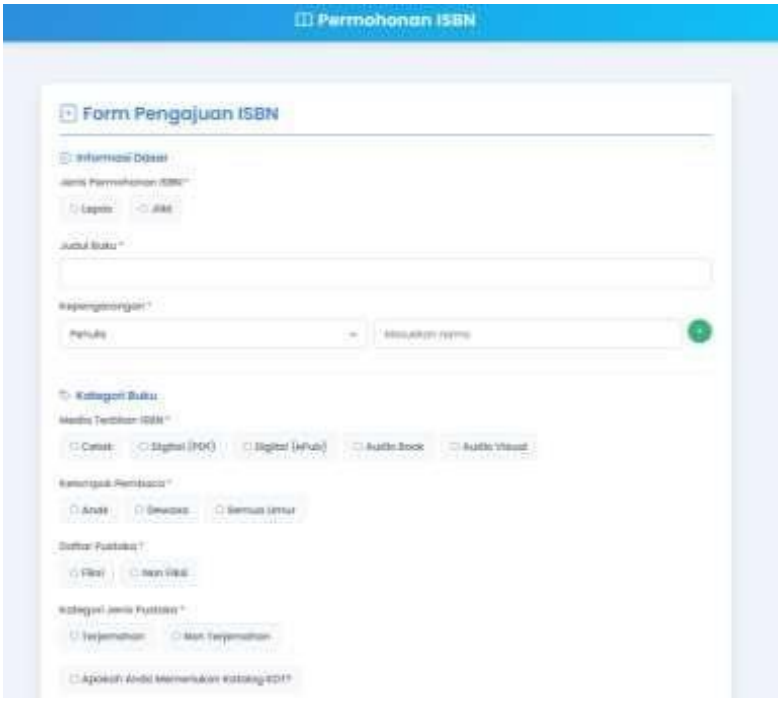


Figure 7. An overview of the applcation form in our proposed ISBN management system.

The history feature displays all ISBN applications previously submitted by the user. The backend retrieves data from the permohonan_isbn table according to the logged-in account, ensuring that each user can only view their own application records.

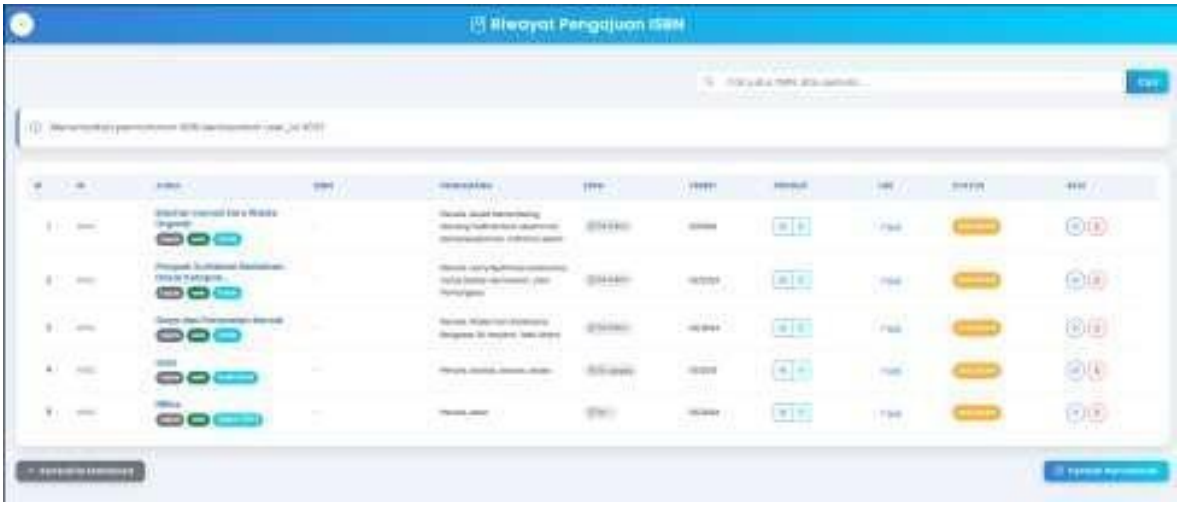
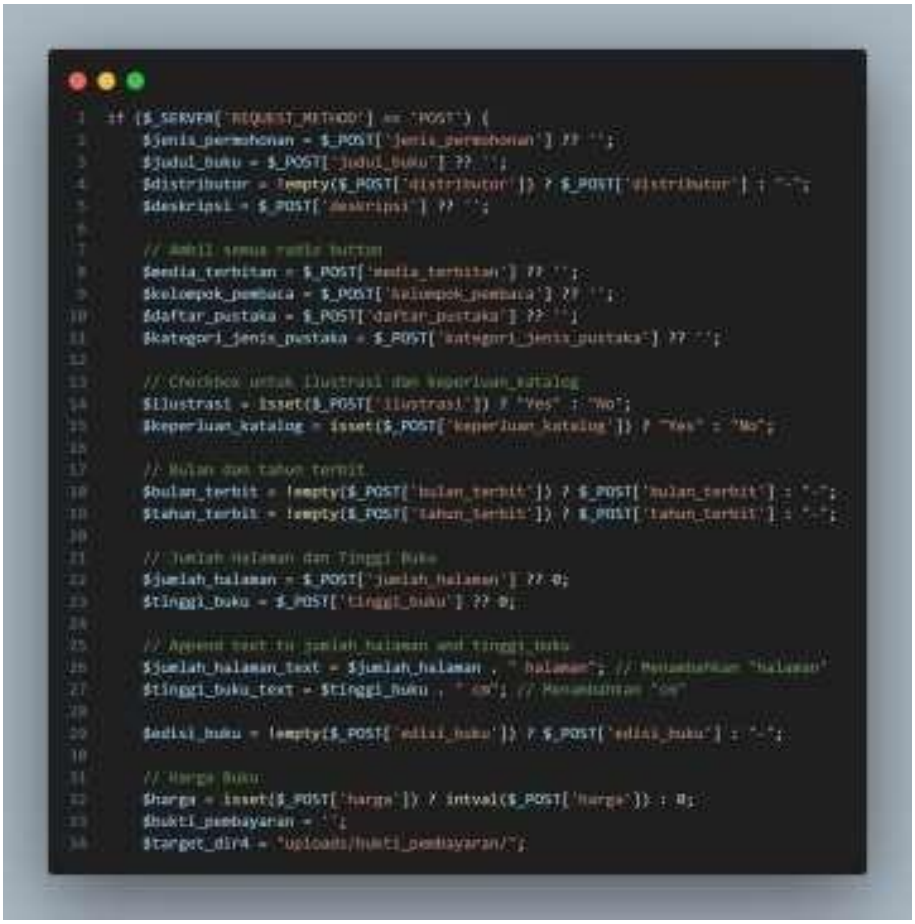


Figure 8. An overview of the applcation form in our proposed ISBN management system.



```
if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    $jenis_permohonan = $_POST['jenis_permohonan'] ?? '';
    $judul_buku = $_POST['judul_buku'] ?? '';
    $distributor = !empty($_POST['distributor']) ? $_POST['distributor'] : '-';
    $deskripsi = $_POST['deskripsi'] ?? '';

    // Ambil semua radio button
    $media_terbitan = $_POST['media_terbitan'] ?? '';
    $kelompok_pembaca = $_POST['kelompok_pembaca'] ?? '';
    $daftar_pustaka = $_POST['daftar_pustaka'] ?? '';
    $kategori_jenis_pustaka = $_POST['kategori_jenis_pustaka'] ?? '';

    // Checkbox untuk ilustrasi dan keperluan_katalog
    $ilustrasi = isset($_POST['ilustrasi']) ? "Yes" : "No";
    $keperluan_katalog = isset($_POST['keperluan_katalog']) ? "Yes" : "No";

    // Bulan dan tahun terbit
    $bulan_terbit = !empty($_POST['bulan_terbit']) ? $_POST['bulan_terbit'] : '-';
    $tahun_terbit = !empty($_POST['tahun_terbit']) ? $_POST['tahun_terbit'] : '-';

    // Jumlah halaman dan Tinggi Buku
    $jumlah_halaman = $_POST['jumlah_halaman'] ?? 0;
    $tinggi_buku = $_POST['tinggi_buku'] ?? 0;

    // Append text to jumlah halaman and tinggi buku
    $jumlah_halaman_text = $jumlah_halaman . " halaman"; // Menambahkan "halaman"
    $tinggi_buku_text = $tinggi_buku . " cm"; // Menambahkan "cm"

    $edisi_buku = !empty($_POST['edisi_buku']) ? $_POST['edisi_buku'] : '-';

    // Harga Buku
    $harga = isset($_POST['harga']) ? intval($_POST['harga']) : 0;
    $bukti_pembayaran = '';
    $target_dir4 = "uploads/bukti_pembayaran/";
```

Figure 9. The sourcode that enables users to submit their applications (Add).

Figure 10. Our implementation that handles the Edit Application feature.

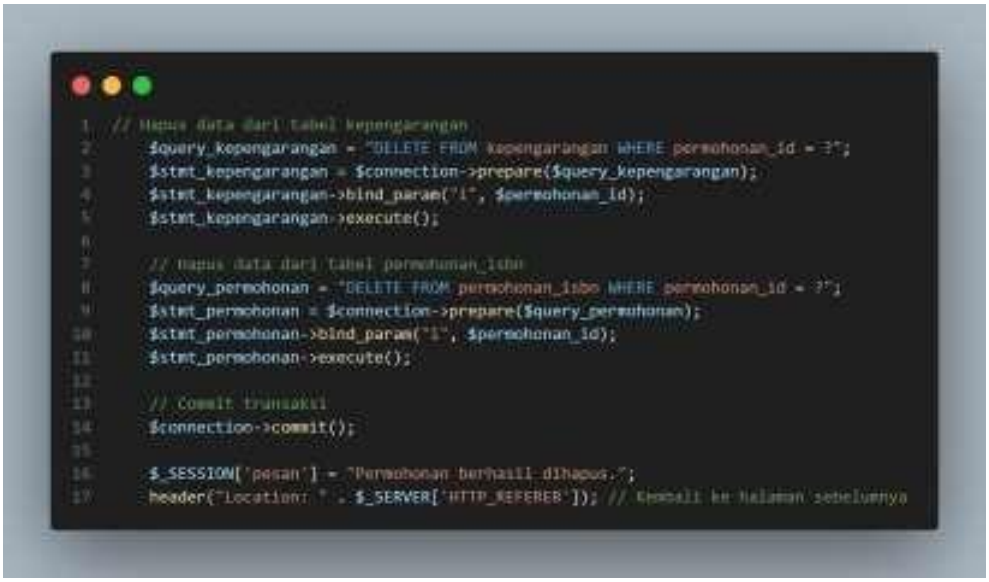


Figure 11. Our implementation that handles the Delete Application feature.

As illustrated in Figures 9, 10, and 11 (Add, Edit, and Delete Application Data), the backend manages ISBN application data through three main functions: add, edit, and delete. In the add process, the application data entered by the user is validated before being stored in the database using an INSERT query. The edit feature allows users or administrators to update existing records with an

UPDATE query, while the delete feature removes specific data based on its ID using a DELETE query. All three processes are equipped with validation mechanisms to ensure data security and integrity, allowing only authorized users to perform modifications.

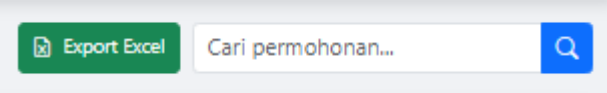


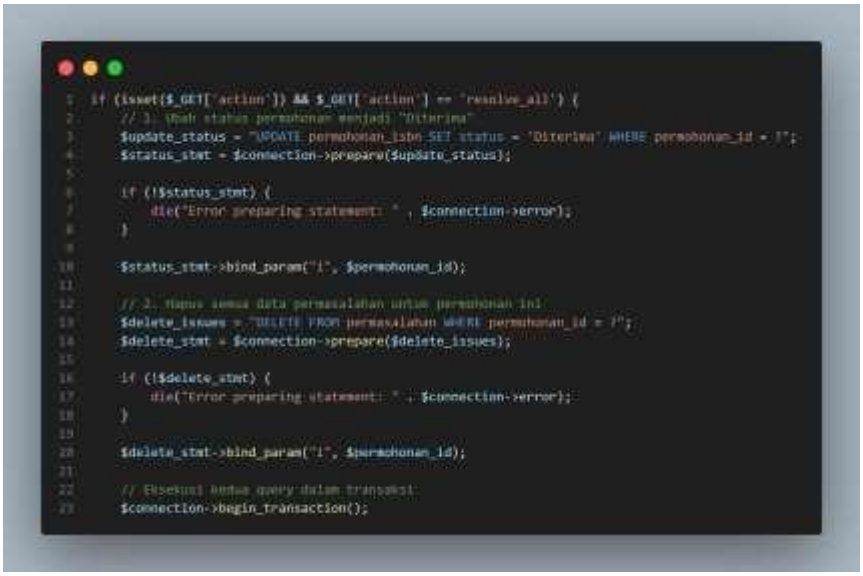
Figure 12. The export ISBN and search features in our system.

The system allows users or administrators to search for ISBN applications by status (e.g., in process, approved, rejected) or keyword, with results displayed in a table for easier analysis. It also provides an "Export to Excel" feature that retrieves all or filtered records and converts them into .xlsx format using PHP libraries or third-party extensions, enabling efficient reporting and data storage. Additionally, the module supports flagging problematic records by assigning special statuses in the database, helping administrators manage and validate ISBN data more effectively.

```php
// Set filename based on export type
$filename = "Data_ISBN_";
switch ($export_type) {
    case 'diajukan':
        $filename .= "Diajukan_";
        break;
    case 'diproses':
        $filename .= "Diproses_";
        break;
    case 'diterima':
        $filename .= "Diterima_";
        break;
    case 'bermasalah':
        $filename .= "Bermasalah_";
        break;
    default:
        $filename .= "Semua_";
        break;
}
$filename .= date('Y-m-d') . ".xls";
```

Figure 13. Our implementation for the "Export Excel" feature.

The last feature related to ISBN applications is the management of problematic ISBN data, which is implemented through the code shown in Figure 14. This code snippet is used to flag or handle ISBN records that contain issues, such as incomplete or invalid information. The backend processes this by assigning a special status to a specific field in the *permohonan_isbn* table. Consequently, administrators can clearly distinguish between valid and problematic data, allowing the ISBN validation process to be more structured and reliable.



```php
if (isset($_GET['action']) && $_GET['action'] == 'resolve_all') {
    // 1. Ubah status permohonan menjadi "Diterima"
    $update_status = "UPDATE permohonan_isbn SET status = 'Diterima' WHERE permohonan_id = ?";
    $status_stmt = $connection->prepare($update_status);

    if (!$status_stmt) {
        die("Error preparing statement: " . $connection->error);
    }

    $status_stmt->bind_param("i", $permohonan_id);

    // 2. Hapus semua data permasalahan untuk permohonan ini
    $delete_issues = "DELETE FROM permasalahan WHERE permohonan_id = ?";
    $delete_stmt = $connection->prepare($delete_issues);

    if (!$delete_stmt) {
        die("Error preparing statement: " . $connection->error);
    }

    $delete_stmt->bind_param("i", $permohonan_id);

    // Eksekusi kedua query dalam transaksi
    $connection->begin_transaction();
```

Figure 4.14. The sourcode that handles ISBN records with issues.

c. User Profile Editing Feature

Figure 15. User Profile Editing Interface

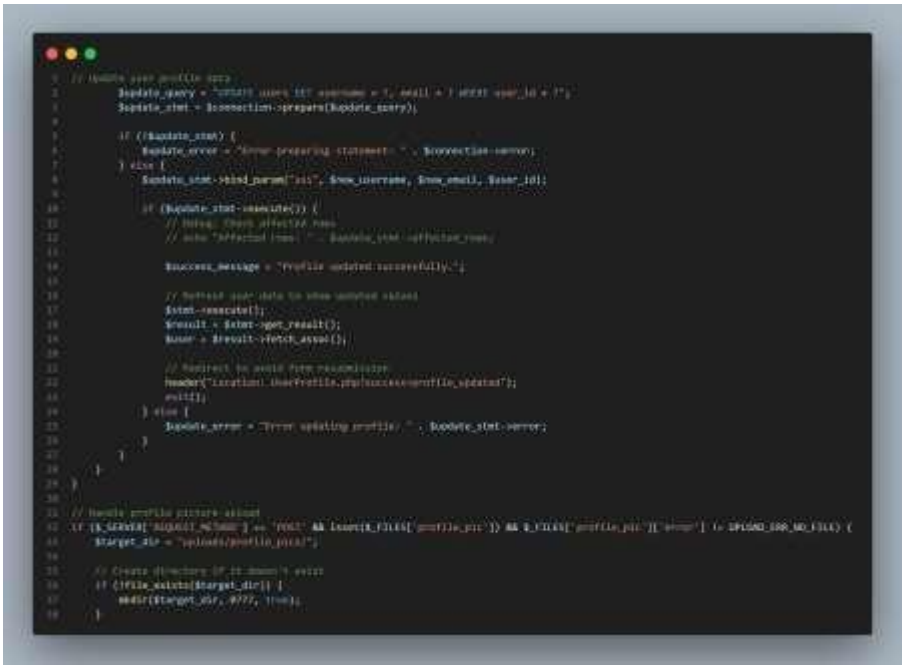


Figure 16. User Profile Editing Source Code

This feature enables users to update their personal information, including name, email address, and password. As shown in Figure 15, the profile editing form can be accessed after logging into the system. The backend processes the data submitted through this form by performing validations, such as ensuring email uniqueness and encrypting the password before saving it. Figure 16 presents the backend source code that handles updating user data in the users table, ensuring that the changes are stored in the database and immediately reflected in the user's account.

d. Admin Dashboard

The Admin Dashboard serves as the primary page, providing a summary of information related to managing ISBN applications. In the top section, information cards display the total number of applications, pending approvals, applications in process, approved applications, and problematic entries. In addition, a "Latest Applications" table is presented, listing the most recent submissions complete with ID, title, author, date, status, and actions available to administrators, such as editing or deleting data.

At the bottom section, a Quick Access menu is provided to add new ISBN applications or create user accounts directly. The main navigation is located on the left side of the page, offering menu options to manage applications based on their status as well as user management. With its informative yet straightforward interface, the Admin Dashboard enables administrators to monitor submission statuses and manage data efficiently.

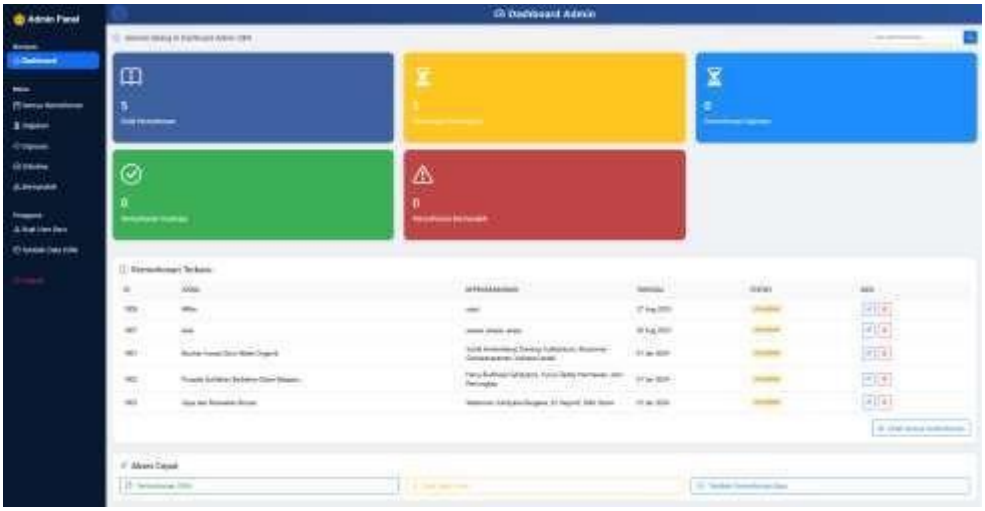


Figure 17. Admin Dashboard Interface

Database Integration

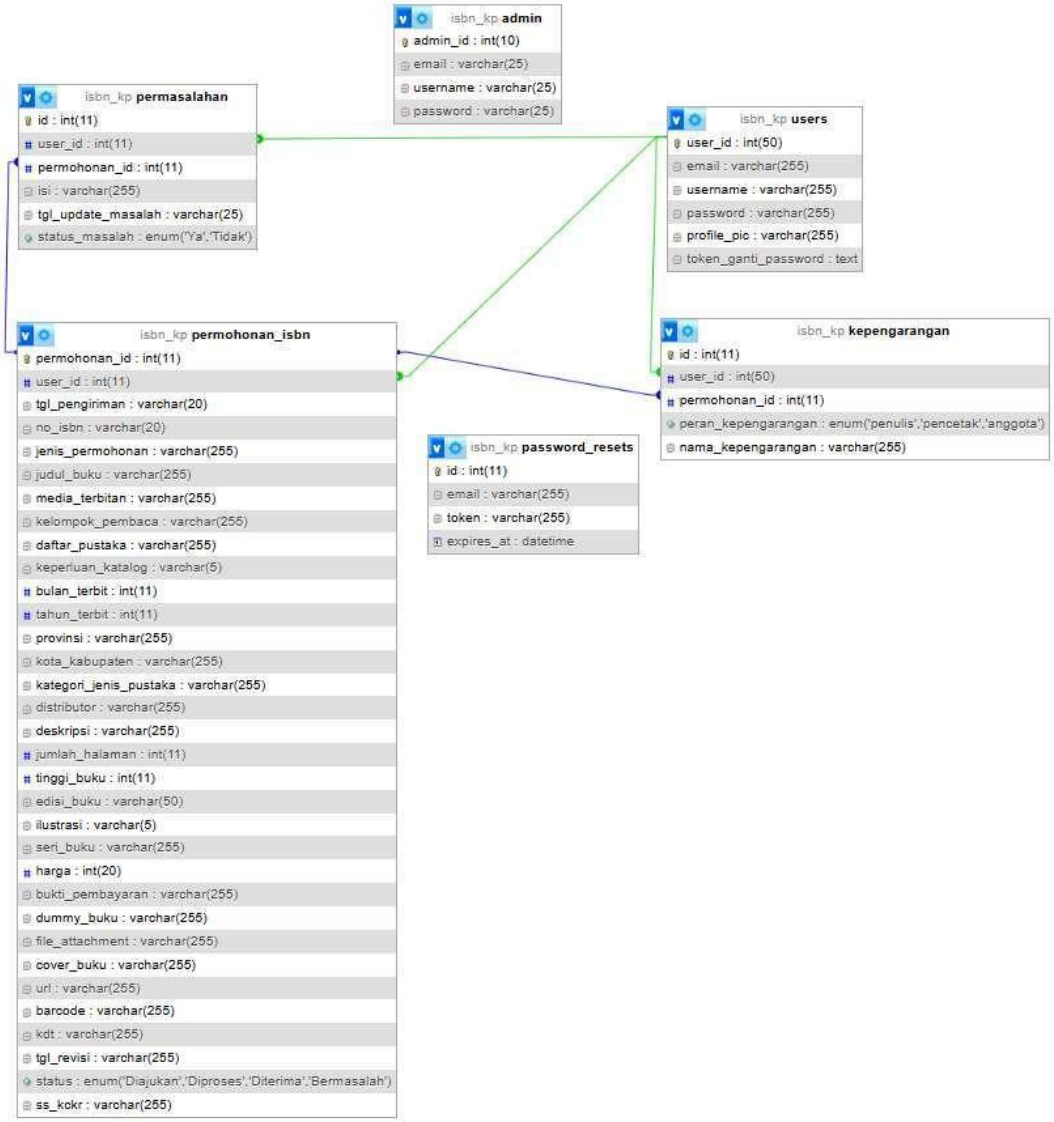


Figure 18. ISBN Submission System Database

Figure 18 illustrates the database of the ISBN submission system, which is designed using MySQL. MySQL is an open-source Relational Database Management System (RDBMS) used to store, manage, and process application data. The database consists of several interrelated tables that support the system's functionality, particularly in managing ISBN applications, user information, and administrative activities.

The main tables used in the system are as follows:

users

This table stores user information, such as email, username, encrypted password, and additional profile details. It also serves as the link between ISBN applications and authorship data.

permohonan_isbn

This table stores ISBN application data submitted by users. The recorded information includes book title, author, category, year of publication, publisher, and attached files such as book drafts, cover files, and barcodes. The application status is also stored in this table, with values such as Submitted, In Process, Approved, or Problematic.

kepengarangan

This table records authorship details related to ISBN applications, including author names, roles (e.g., writer, printer, or contributor), and their association with the application ID.

permasalahan

This table is used to record issues or problems related to an ISBN application. The stored data includes application ID, user ID, problem description, update date, and problem status.

admin

This table stores administrator data with access to manage the system. The information includes username, email, and password for authentication.

password_resets

This table is used in the forgot password process. It stores user email addresses, temporary security tokens, and the expiration times of these tokens. This ensures that the password reset process is conducted securely.

CONCLUSIONS

Based on the implementation and discussion, it can be concluded that the development of the PHP-based backend system for the ISBN website has successfully achieved its primary objective: to provide a structured, accessible, and user-oriented platform for managing ISBN data (Fajar et al., 2024). The system is capable of handling user registration and login, profile management, password recovery (forgot password), and ISBN data export. All of these features were tested using the black-box testing method, and the results were consistent with expectations. The application of the Scrum methodology in the development process facilitated measurable workflow management through sprint phases, ensuring that each feature was completed on time while minimizing the risk of delays (Etrariadi & A'inunisya, 2023). Consequently, the developed backend system not only meets functional requirements but also enhances the efficiency and effectiveness of ISBN data management.

LIMITATIONS & FURTHER RESEARCH

Despite these promising results, this study has several limitations. First, the system has only been tested within the scope of LPPM UPN "Veteran" Yogyakarta, so its scalability and adaptability to other higher education institutions remain unverified. Second, the evaluation primarily focused on functional and usability testing, while broader performance metrics, such as system load handling and security stress testing, were not fully addressed. Third, integration with other institutional information systems (e.g., research repositories or academic portals) has not yet been

implemented.

Future research could explore extending the system to multiple institutions, conducting large-scale usability studies with diverse user groups, and improving interoperability through API integration with existing academic platforms. Furthermore, additional features such as automated plagiarism detection, integration with national library databases, and advanced analytics dashboards could be incorporated to enhance the system's relevance and impact further.

REFERENCES

Alamsyah, A., and Yuliansyah, H. (2017). Rancang Bangun Aplikasi Web untuk Manajemen Proyek Berbasis Scrum. *Jurnal Sarjana Teknik Informatika*, *5*(2), 58-67, https://doi.org/10.12928/jstie.v5i2.10834

Alim, K. S., Ekowati, N. A., Kisworini, R. Y., & Riyandari, L. (2023). Design and Development of Web-Based Application Cangkingan Using Scrum Method. *Jurnal Teknik Informatika, 4*(4), 953-965. https://doi.org/10.52436/1.jutif.2023.4.4.1311

Etrariadi, N.,, & A'inunisya, E. S. P. (2023). Pengembangan Website Manajemen Proyek Menggunakan Metode Agile Scrum (Studi Kasus Diskopindag Kota Malang). *Jurnal TEKNOSI, 9*(1), 55–66. https://doi.org/10.25077/TEKNOSI.v9i1.2023.55-66

Fajar, F. S., Kusumaningsih , N., & Mu'is , A. (2024). Rekayasa Sistem Informasi Perpustakaan Berbasis Web Dengan Menggunakan PHP Native. *Curtina*, *5*(2), 74–85. https://doi.org/10.55719/curtina.v5i2.1493

Fikri, A. D. (2023). Pengembangan Sistem Perizinan Santri Berbasis Website Menggunakan PHP Native di Pondok Pesantren Raudlatul Musthofa. *Jurnal Informatika dan Teknik Elektro Terapan (JITET), 13*(3). https://doi.org/10.23960/jitet.v13i3.7108

Hamzah, F., & Nurhalim, W. (2019). Sistem Informasi Pengelolaan Jurnal dan Konferensi Tingkat Internasional (Studi Kasus: Politeknik Negeri Padang). *Jurnal Vokasi Informatika, 7*(1), 1-7.

Hardiansyah, F., Rizal, A., & Purnamasari, I. (2023). Implementasi Metode Agile Scrum dalam Pengembangan Aplikasi Pembelajaran Olahraga. *JATI – Jurnal Mahasiswa Teknik Informatika, 7*(2). https://ejournal.itn.ac.id/index.php/jati/article/download/6734/4050

Mishra, A., & Dubey, D. (2013). A Comparative Study of Different Software Development Life Cycle Models in Different Scenarios. *International Journal of Advance Research in Computer Science and Management Studies, 1*(5), 64–69.

Pappas, I. O., Pateli, A. G., & Giannakos, M. N. (2017). Designing and Implementing Agile Information Systems: A Review and Research Agenda. *Information Systems Frontiers, 19*(2), 259–277. https://doi.org/10.1007/s10796-015-9593-4

Raharja, U., Suryani, H., & Ananda, A. (2021). Perancangan Sistem Informasi iLearning Jurnal Berbasis Content Management System (CMS) Menggunakan Analisis SWOT. *Jurnal Media Informatika Budidarma, 5*(2), 548-557.

Rodríguez, P., Markkula, J., Oivo, M., & Turula, K. (2012). Survey on Agile and Lean Usage in Finnish Software Industry. *Proceedings of the ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*, 139–148. https://doi.org/10.1145/2372251.2372275

Santoso, A. B., Huda, M., Wathon, M. I., & Lestari, D. (2022). Rancang Bangun Sistem Informasi Manajemen Luaran Penelitian dan Pengabdian kepada Masyarakat (SIM-LUARAN) di Institut Teknologi Kalimantan. *Jurnal Teknologi dan Sistem Informasi, 3*(2), 241-250.

Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide: The Definitive Guide to Scrum: The Rules of the Game*. Scrum. https://www.scrum.org/resources/scrum-guide

Sukamto, R. A., & Shalahuddin, M. (2018). *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Informatika Bandung.

Universitas Negeri Makassar. (2021). *Laporan Kinerja Universitas Negeri Makassar Tahun 2021*. Universitas Negeri Makassar. https://unm.ac.id/laporan-kinerja-perguruan-tinggi/

Wibowo, A., & Putri, A. A. (2020). Pengembangan Sistem Informasi Manajemen Penelitian dan Pengabdian kepada Masyarakat Menggunakan Framework Symfony (Studi Kasus: Universitas Dhyana Pura). *Jurnal Sistem dan Informatika, 15*(1), 55-64.