JURNAL TEKNOLOGI DAN OPEN SOURCE
Vol. 9, No.1, June 2026, pp. 228~241
e-ISSN: 2622-1659, accredited Four Grade by Kemenristekdikti, Decree No: 152/E/KPT/2023
DOI: https://doi.org/10.36378/jtos.v9i1.5223

Design and Development of a Web-Based Booking and Employee Management System Using Agile Methodology

Kesya Sakha Nesya Arimawan[1], Reisa Permatasari[2], Seftin Fitri Ana Wati[3]
[1,2,3] Information Systems, Faculty of Computer Science, Pembangunan Nasional "Veteran" Jawa Timur, Surabaya, Indonesia

Article Info

Article history:

Received 01 09, 2026
Revised 03 14, 2026
Accepted 06 16, 2026

Keywords:

Booking System
Employee Management
Agile Methodology
User Acceptance Testing (UAT)
Web-based System

ABSTRACT

The rapid advancement of information technology has compelled various business sectors, including the creative industry, to undergo digital transformation to enhance their market competitiveness. April Photo Studio previously relied on manual and conventional methods for customer booking and employee management, which frequently resulted in operational issues such as unrecorded orders, slow response times, and inaccurate scheduling. To address these systemic problems, this study adopted the Agile development methodology to design and build a comprehensive web-based information system. The selection of Agile facilitated an iterative and incremental development lifecycle, allowing for continuous feedback and refinement through stages of requirement analysis, system design, implementation, testing, and maintenance. The system was developed utilizing the Laravel framework and the Model-View-Controller (MVC) architecture to ensure scalability and organized code management. The resulting platform integrates several critical features, including a real-time online booking engine, secure digital payment gateways, and an automated employee shift management module. System testing confirmed that all functional requirements were met. Furthermore, quantitative evaluation through User Acceptance Testing (UAT) involving administrators, staff, and customers yielded a 100% acceptance rate. These results indicate that the implementation of the system successfully streamlined the entire booking process, significantly improved data accuracy, and increased transparency in employee workload management. In conclusion, the transition from manual to digital systems at April Photo Studio has enhanced overall operational effectiveness, reduced human error, and established a more professional and structured workflow for both staff and customers.

Corresponding Author:

Kesya Sakha Nesya Arimawan
Department of Information Systems
Universitas Pembangunan Nasional "Veteran" Jawa Timur
Surabaya, Indonesia
Email: Kesyasakha14@gmail.com

1. Introduction

The photography industry has experienced significant growth in recent years, driven by the increasing demand for creative visual documentation such as weddings, pre-wedding sessions, and special events [1]. This sector contributes significantly to the creative economy, pushing businesses to adapt to modern market trends [2]. This expansion has encouraged many studios to enhance their service quality and operational efficiency through digital transformation. Studies indicate that adopting digital technology is no

longer optional but essential for Small and Medium Enterprises (SMEs) to maintain competitiveness and sustainability [3], [4]. However, not all businesses have fully adopted digital systems to manage their daily operations. April Photo Studio, for instance, still carries out several core activities such as service booking and employee management manually. These traditional practices often result in issues such as unstructured bookings, delayed customer responses frequently cited as major hindrances to SME growth [5], [6].

In service-oriented industries, efficient workforce scheduling plays a vital role in ensuring smooth operations. Unlike manufacturing firms, where scheduling focuses on machinery and production flow, service-based companies prioritize fair and balanced staff management to prevent workload inequality and excessive overtime [7]. Manual scheduling is widely recognized as inefficient; it consumes valuable administrative time and increases the risk of overlapping schedules [8], [9]. At April Photo Studio, manual scheduling and leave applications remain time-consuming and prone to human errors.

Specifically, the studio faces complex challenges in calculating payroll and managing leave fairness. April Photo Studio operates two distinct areas: "Stand 1" (two rooms) and "Stand 2" (five rooms), each with different daily revenue targets. The current payroll system includes a base salary plus daily incentives (meal allowances and performance bonuses) that fluctuate based on achieving these targets. Calculating these variable bonuses manually is highly prone to calculation errors. Research confirms that manual payroll processing is one of the most common sources of financial discrepancies in small businesses [10]. Additionally, the current "first-come, first-served" system for leave requests has resulted in an unfair distribution of off-days, necessitating a digital rotation logic.

Furthermore, from the customer's perspective, accessing service information solely through social media platforms lacks structured details regarding pricing and availability. While social media is a powerful marketing tool, it is often insufficient for handling complex transactional data. Relying on WhatsApp for bookings creates information asymmetry, making it difficult for customers to make informed booking decisions [11].

Several prior studies have utilized the Agile method and Laravel framework to address similar operational challenges. In the domain of customer reservations, [12] developed an e-booking system for a photo studio (4 People Studio), while [13] focused on meeting room reservations, and [14] built a marketplace for sports venue rentals. These studies emphasize that web-based booking systems significantly improve user satisfaction and booking accuracy [15], [16]. Conversely, other research focused on internal administrative efficiency. For example, [17] developed a personnel information system for payroll and leave, while [18]designed a system for student organization administration. Furthermore, recent studies suggest that integrating sales data with HR systems provides better insights for business decision-making [19], [20].

Based on these studies, a significant research gap is identified. While [12] successfully digitized the booking process, it did not address the internal management of employees. Similarly, while [17] optimized payroll, it operated independently of customer transaction data. There is a lack of research developing an integrated system for small-scale creative industries that connects customer transaction data directly with internal HR processes.

To address this gap, this research proposes the development of a web-based information system that integrates service booking, digital payment, and employee management features. Unlike generic systems, this study introduces a unified architecture where customer booking data directly influences dynamic shift scheduling and payroll calculation. The system is implemented using the Laravel framework, which supports modern web development through a clear MVC architecture, high security, and scalability [21], [22].

The Agile methodology is adopted to enable iterative progress and continuous evaluation. Unlike the Waterfall model, which is rigid and risky when requirements change during development [23], Agile offers a flexible approach that allows for the gradual testing of complex features such as the daily incentive logic and shift rotation ensuring the final system aligns with the studio's dynamic business rules [24]. Finally, Black Box testing will be employed to ensure all functional requirements are met without errors [25], while User Acceptance Testing (UAT) will be conducted to validate that the system acts in accordance with the user's expectations and business needs . The objective of this study is to design a low-cost, integrated digital ecosystem for April Photo Studio that bridges the gap between client services and backend workforce management

2. Research Method

The Agile method is a software development approach that prioritizes flexibility, collaboration, and adaptability to shifting user requirements. Agile utilizes an iterative framework where development is executed incrementally through concise cycles known as iterations or sprints. Each iteration encompasses requirements analysis, design, coding, testing, and evaluation. The output of each cycle serves as the

foundation for the subsequent phase, allowing the system to be continuously refined based on user feedback [24].

A defining characteristic of Agile Development is the ability to conduct ongoing reviews and revisions throughout the information system's construction. Theoretically, this approach offers a higher project success rate compared to traditional development methods [18]. Agile emphasizes intensive communication between developers and users, aiming to produce software that is responsive to change and superior in quality. As illustrated in Figure 2.8, this concept entails five distinct execution stages:

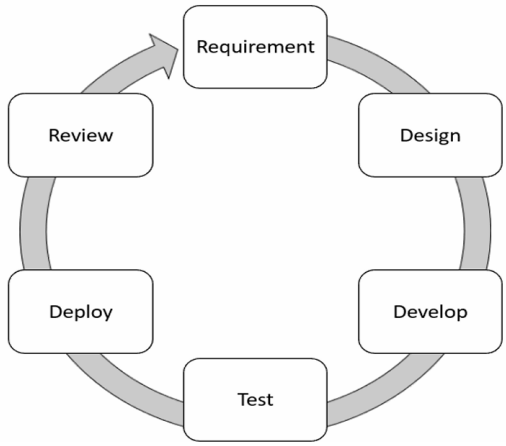

Figure 1. Agile Method

2.1 Requirement

This stage aims to comprehend the user requirements for the system. Information is gathered from relevant sources to ensure that complete and accurate data is obtained regarding the needs of the system to be developed.

2.2 Design

This phase involves creating a design based on the data acquired in the previous step. The objective is to provide a comprehensive visualization or blueprint of the work that needs to be executed.

2.3 Development

This stage is the process of translating the results of system design and analysis into programming code using specific programming languages understandable by computers [10]. During this phase, the information system is built according to the established specifications, and functional checks are conducted to ensure that every part of the system operates as intended.

2.4 Testing

Once the system has been successfully developed, the subsequent step is verification. This involves a comprehensive inspection and testing of the system to identify any potential failures, bugs, or errors.

2.5 Deployment

The goal of this stage is to deploy or distribute the completed application created by the developers, making it available for use.

2.6 Review

This is the final stage of the Agile Development concept. It involves evaluating the system by checking the responses and feedback from users who utilize the application.

2. Result and Discussion

This section presents the results of the web-based information system developed for April Photo Studio using the Laravel framework and MVC architecture [22]. he findings are organized into two main categories: system implementation and testing. The implementation subsection displays the user interfaces of key features, including online booking and shift scheduling, to demonstrate the system's usability. Meanwhile, the testing subsection verifies that all functionalities operate according to the initial

requirements. Finally, the discussion evaluates the system's effectiveness in resolving previous operational challenges, highlighting the improvements in data accuracy and workflow efficiency compared to the prior manual methods.

3.1. Requirement

The Requirement Phase The requirement phase aimed to establish a new workflow and identify the necessary system features using an Agile approach. The analysis prioritized functional requirements for the Admin role to serve as the primary data foundation while establishing global non functional requirements. Subsequently, the requirement definition expanded to cover interactive features for Customer and Staff roles, specifically focusing on booking workflows and operational management. This comprehensive analysis ensured that the system design effectively integrated core data management with end-user interactions.

3.2. Design

The Design phase translates the identified requirements into a technical blueprint for system development. This study utilizes the Unified Modeling Language (UML) standard to visualize the system's architecture, behavior, and data structure [26]. The design process is comprehensively presented through several modeling tools: Use Case Diagrams to define actor interactions, Activity and Sequence Diagrams to detail the operational logic and data flow, and Class Diagrams to structure the database relationships. Additionally, Wireframes are designed to visualize the user interface and ensure an intuitive user experience prior to implementation.

A. Use Case Diagram

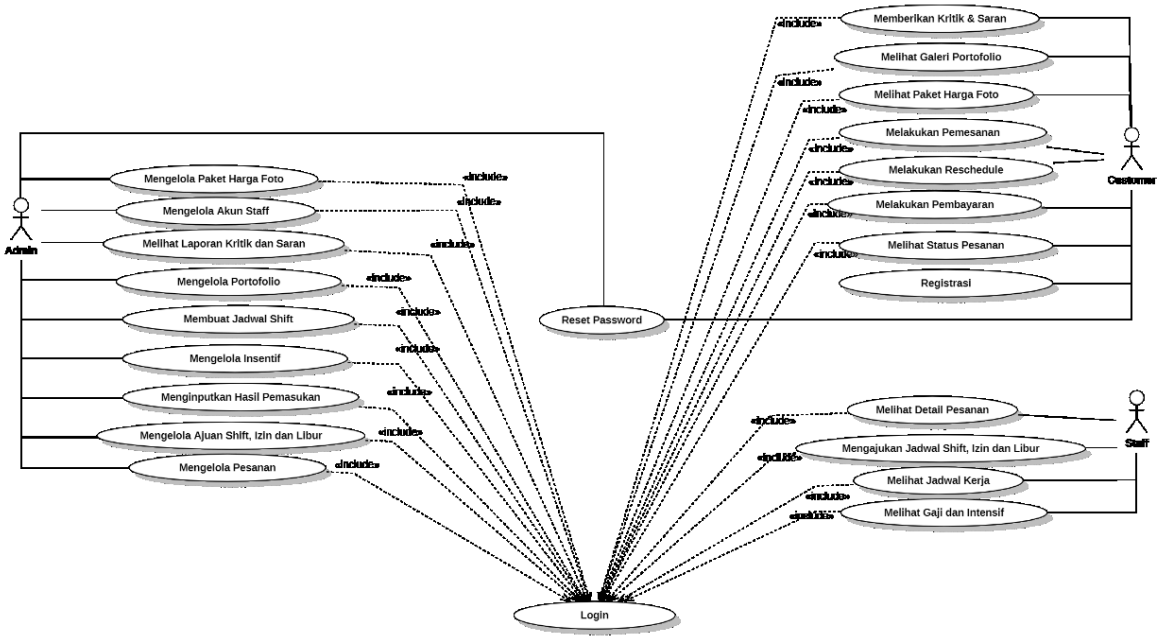

Figure 2. Use Case System

Figure 2 illustrates the comprehensive Use Case Diagram for the Web-Based Booking and Personnel Information System at April Photo Studio. This diagram delineates the system's functional requirements and the interactions between the system and its three primary actors: the Admin, the Customer, and the Staff.

The Admin is vested with extensive administrative privileges to manage the system's backend operations. These responsibilities include managing master data (such as photo pricing packages and staff accounts), overseeing operational schedules (shift management and leave approvals), and monitoring financial performance through revenue inputs and incentive management. Conversely, the Customer actor interacts with the front-end services. Their functionalities range from account registration and portfolio viewing to transactional activities, including booking packages, processing

payments, and rescheduling appointments. Finally, the Staff actor utilizes the system for internal operational coordination, such as verifying order details, checking work rosters, and viewing compensation data.

A critical architectural feature depicted in Figure 2 is the implementation of the <<include>> relationship. As shown, all sensitive and functional use cases, ranging from the Admin's management tasks to the Customer's transactions and the Staff's operational requests are directly linked to the "Login" use case. This structure mandates that user authentication is a prerequisite for all system interactions, thereby ensuring data security and enforcing access control protocols across all user levels.

B. Activity Diagram

1. Scheduling Management

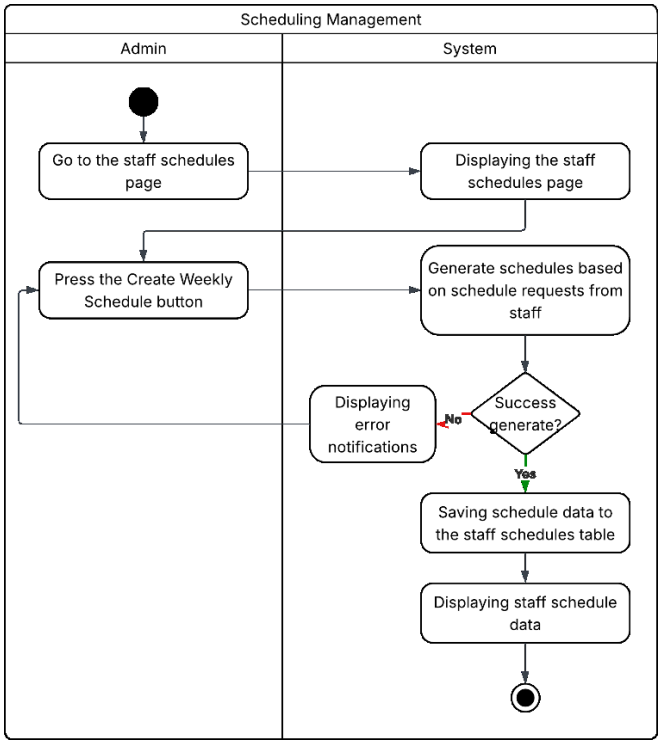

Figure 3. Activity Diagram Scheduling Management

The Activity Diagram for Scheduling Management provides a detailed visualization of the backend operational flow concerning workforce allocation and leave administration. The process commences when the Administrator accesses the scheduling dashboard, triggering the system to retrieve and display the current roster and calendar view. A core component of this workflow is the shift assignment process, where the Administrator distributes duties based on the studio's unique rotation policy. The diagram highlights the decision logic required to rotate staff between "Stand 1" (lower capacity) and "Stand 2" (higher capacity), ensuring an equitable distribution of workload and potential daily incentives.

Parallel to the shift assignment process, the workflow also illustrates a structured decision-making mechanism for handling employee leave requests. This process begins when an employee submits a leave application through the system. The request is then forwarded to the Administrator, who is responsible for reviewing all pending applications and ensuring that each request complies with organizational policies and operational requirements. During this stage, the system automatically evaluates the leave request against predefined minimum staffing thresholds, ensuring that sufficient personnel remain available to maintain operational continuity and service quality.

The evaluation process plays a critical role in preventing workforce shortages that could negatively affect daily operations. If the requested leave period does not compromise staffing levels, the application proceeds through the approval workflow and is incorporated into the employee scheduling plan. However, if the system detects a staffing conflict—such as an insufficient number of employees available to cover essential shifts—the request is redirected into a rejection workflow. In such cases, the employee is notified of the decision, and the request status is updated accordingly within the system.

Following the completion of both shift assignment and leave management activities, the workflow enters a comprehensive validation phase. During this stage, the scheduling algorithm performs a series of consistency and integrity checks to identify potential logical errors. These checks include detecting double-booking situations, preventing employees from being assigned to overlapping shifts, and ensuring that no employee is scheduled to work during an approved leave period. The system also verifies compliance with scheduling rules and organizational constraints. Only after all validation checks have been successfully completed are the schedule updates permanently committed to the database, ensuring the accuracy, reliability, and integrity of the final workforce schedule..

2. Booking

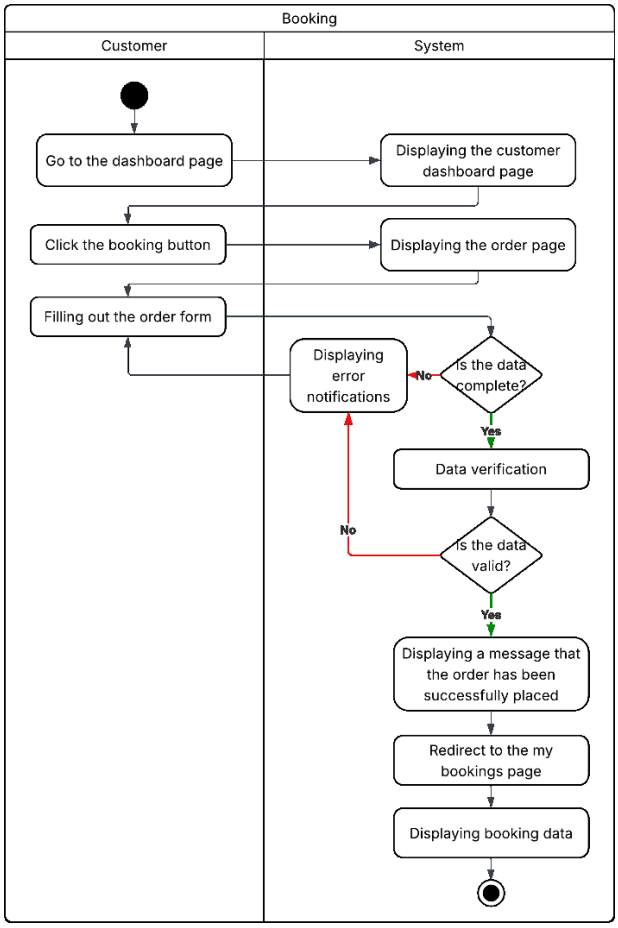

Figure 4. Activity Diagram Booking

The Activity Diagram for the Booking Process provides a comprehensive visual representation of the end-to-end workflow regarding customer reservations. The process commences when the customer accesses the booking interface and browses the available photography packages. Upon selecting a specific package, the customer is prompted to input their preferred date and session time. At this stage, a critical decision point occurs where the system

automatically queries the database to verify slot availability. If the selected time slot clashes with an existing reservation, the system rejects the request and prompts the user to select an alternative time, creating a loop until a valid slot is chosen. Conversely, if the slot is available, the workflow advances to the payment phase. Here, the customer submits the payment proof or completes a digital transaction. The system then performs a final validation of the payment status. Upon successful verification, the process concludes by generating a digital booking confirmation for the customer and simultaneously updating the master transaction records to block the schedule, thereby preventing any potential double bookings.

C. *Sequence Diagram*

1. Schedule Management

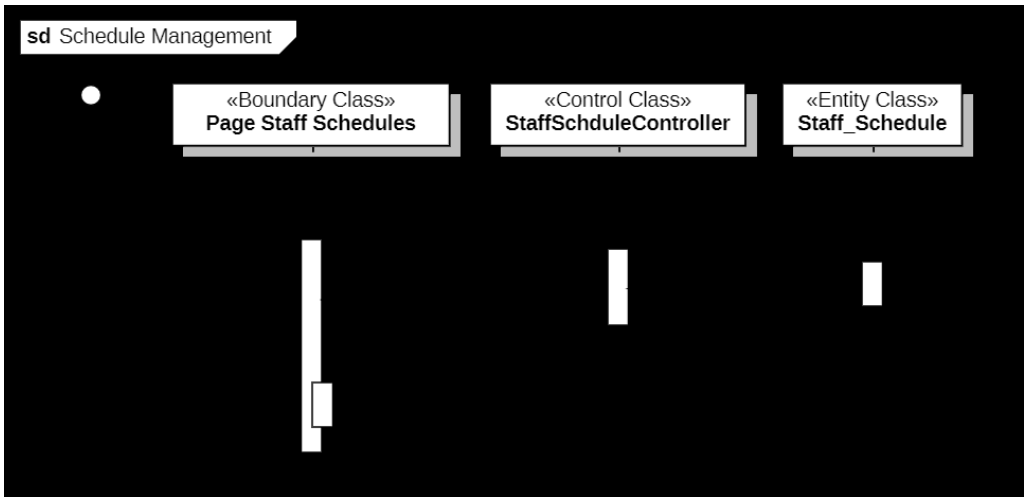

Figure 5. Sequence Diagram Schedule Management

The Sequence Diagram for Schedule Management details at figure 5. The time-ordered interactions between the Administrator and the system components during shift allocation. The process begins when the Administrator submits a shift assignment or leave approval request via the user interface (View). This request is transmitted to the Controller, which executes the validation logic to ensure no overlapping shifts occur. Upon successful validation, the Controller instructs the Model to update the database, and a confirmation message is returned to the Administrator's view.

2. Booking

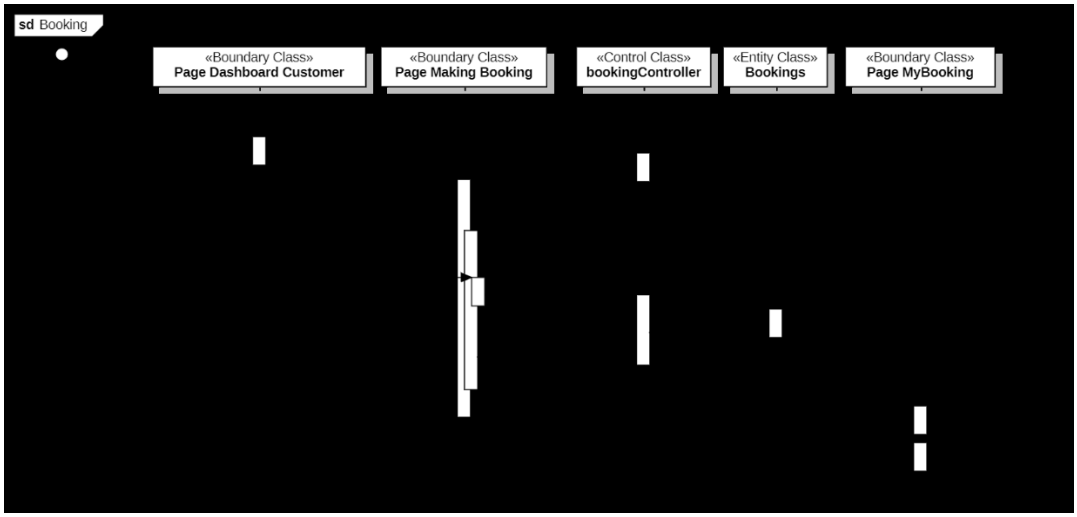

Figure 6. Sequence Diagram Booking

The Sequence Diagram for the Booking process illustrates at figure 6. The sequential message flow initiated by the Customer to secure a photography session. The interaction starts with the Customer selecting a package and date through the Booking Page. The Controller receives this input and queries the database (Model) to verify real-time slot availability. Once availability is confirmed, the system processes the transaction data, records the new reservation in the database, and sends a booking confirmation response back to the Customer.

D. *Class Diagram*

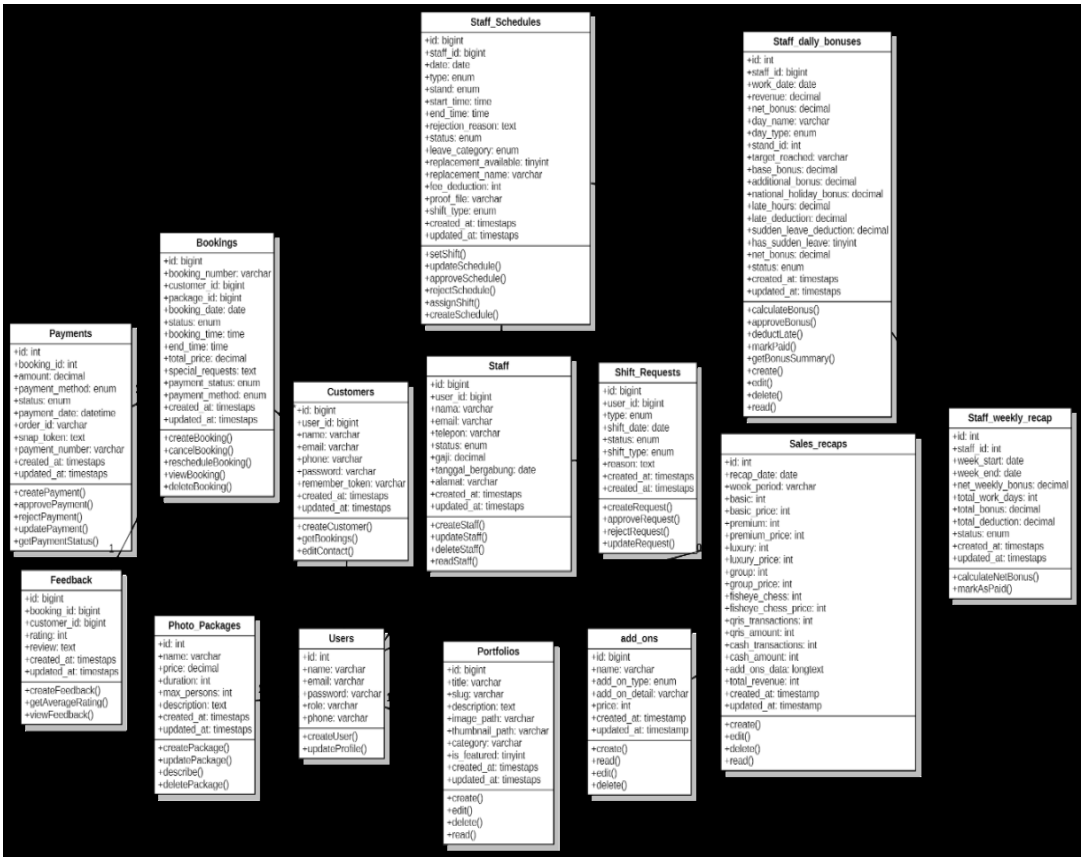

Figure 7. Class Diagram System

Figure 3 presents the comprehensive Class Diagram for the April Photo Studio Booking and Personnel Information System, delineating the static structure and object-oriented architecture of the application. At the core of this structure is an inheritance strategy where the Users class serves as a superclass, generalizing common attributes for the Customers and Staff subclasses. The diagram encompasses critical operational entities such as Photo_Packages, Bookings, and Portfolios, which are essential for defining the studio's service offerings. Furthermore, the inclusion of financial classes like Payments and Sales_recaps, alongside service modifiers like add_ons, ensures that the system accurately models the complexity of business transactions and revenue tracking.

The relationships defined in the diagram illustrate the seamless integration of client-facing services with internal management workflows. The association chain linking Customers to Bookings and Payments establishes a robust end-to-end workflow for reservation and transaction processing. On the administrative side, the Staff entity is intimately connected to management modules including Shift_Requests, Staff_Schedules, and Staff_daily_bonuses, facilitating precise tracking of employee availability, performance, and compensation. Additionally, the integration of the Feedback class allows for the systematic collection of user reviews, ensuring that both operational efficiency and customer satisfaction are structurally supported within the system’s design.

E. Wireframe

The Wireframe serves as a low-fidelity visual guide that represents the skeletal framework of the website. It focuses on the layout, arrangement of content, and interface elements without the distraction of colors or detailed graphics. This section illustrates the structural design for key user interfaces, including the Customer Booking Page and the Admin Dashboard. The purpose of the wireframe is to establish the user experience (UX) flow and ensure that all functional requirements are properly placed before the high-fidelity interface design and actual coding begin.

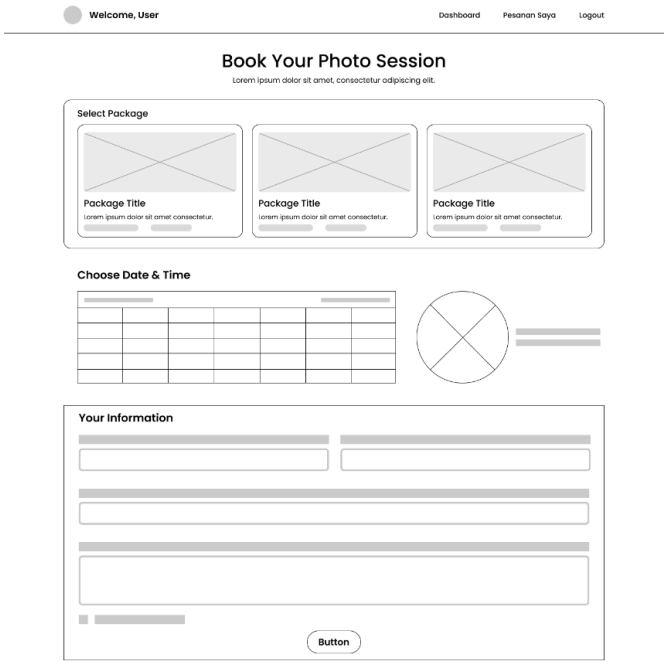

Figure 8. Wireframe Booking System

Figure 8 presents the wireframe for the Booking System, displaying the structural layout of the customer-facing interface. It highlights essential components, including the service package selection area, the availability calendar, and the reservation form, designed to facilitate a streamlined and intuitive booking process for customers.

Figure 9. Wireframe Schedule Management System

Figure 9 depicts the wireframe for the Schedule Management System, designed specifically for the Administrator's dashboard. The layout focuses on a comprehensive calendar view that allows for the efficient visualization of employee shifts and roster distribution. Key functional areas include controls for assigning daily shifts, monitoring staff availability, and processing leave requests.

3.3. Development

The Development phase involves the translation of design specifications and wireframes into executable source code. In this stage, the system is constructed using the Laravel framework, adhering to the Model-View-Controller (MVC) architecture to ensure modularity and maintainability. This process includes setting up the database schema through migrations, implementing the backend logic for booking and scheduling algorithms, and building the frontend interfaces. The objective is to transform the conceptual design into a functional application ready for testing.

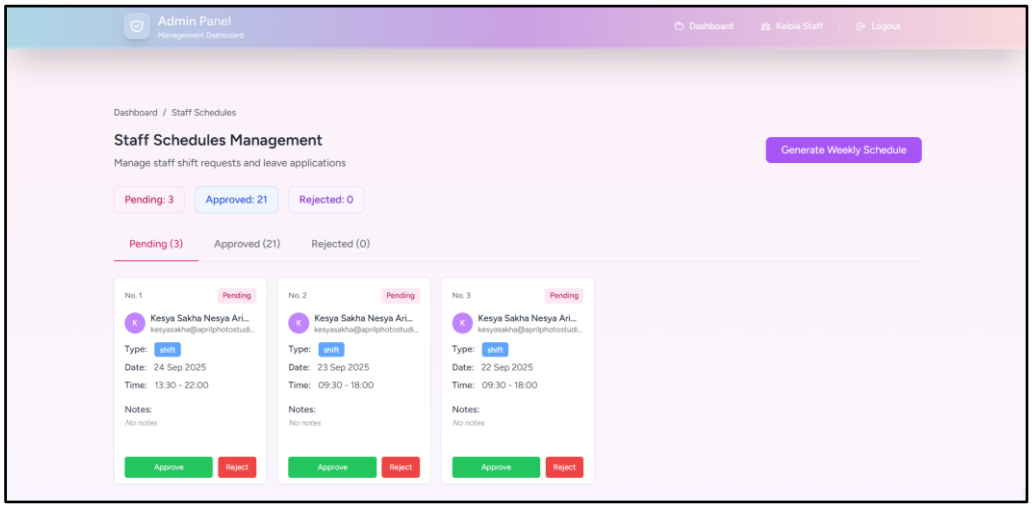

Figure 10. View Page Schedule Managament

Figure 10 presents the implemented user interface for the Schedule Management module. This page displays the operational roster, allowing the Administrator to view and adjust employee shift assignments in real-time. Key features include a calendar-based view of staff distribution across different stands and status indicators for approved leave or off-days, ensuring clear visibility of workforce availability.

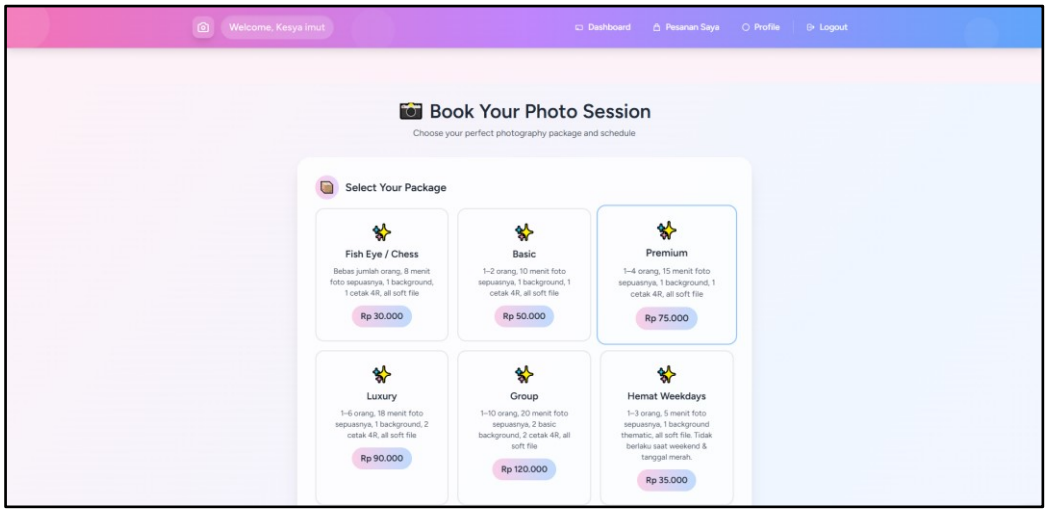

Figure 10. View Page Booking

Figure 10 illustrates the fully implemented user interface for the Booking module. This page is designed for customer interaction, featuring an intuitive layout where users can browse photography packages, view detailed descriptions, and select their preferred service. It integrates a real-time availability calendar that prevents double-booking by only displaying open time slots for reservation.

3.4. Testing

The testing phase employed the Black Box Testing methodology to rigorously evaluate the system's functionality, focusing on verifying that external inputs produce the expected outputs without examining the internal code structure. The primary objective of this assessment was to validate critical operational workflows, with specific emphasis placed on the accuracy and reliability of creating shift schedules, managing incoming bookings, and initiating new orders. By simulating various usage scenarios for these key features, the testing process aimed to identify and rectify potential discrepancies or errors, ensuring that the system reliably supports the core operational needs of the studio prior to full deployment.

Table 1. System Testing

Test Case Code	Test Case	Expected Results	Test Results	Status
P_Admin_Jadwal_C01	Clicking the "Generate Weekly Schedule" button.	A new weekly schedule is automatically generated for all available Staff.	A new weekly schedule is automatically generated for all available Staff.	Success
P_Admin_Pesanan_R01	Viewing the list of all Customer orders.	Complete order data is displayed (ID, Customer name, status, etc.).	Complete order data is displayed (ID, Customer name, status, etc.).	Success
P_Admin_Pesanan_U01	Verifying or updating a Customer's order status.	The order status is successfully updated to 'Confirmed'.	The order status is successfully updated to 'Confirmed'.	Success
P_Admin_Pesanan_D01	Canceling or deleting a Customer's order.	The order is successfully canceled, and its status changes to 'Cancelled'.	The order is successfully canceled, and its status changes to 'Cancelled'.	Success
P_Cust_Pesan_01	Selecting an available date, session, and photo package, then completing the order.	The order is successfully created with the status 'Pending Payment'.	The order is successfully created with the status 'Pending Payment'.	Success
N_Cust_Pesan_02	Attempting to book a session on a date or time slot that is fully booked/unavailable.	The system displays a "Schedule Unavailable" notification and prevents the booking from proceeding.	The system displays a "Schedule Unavailable" notification and prevents the booking from proceeding.	Success
N_Cust_Pesan_03	Attempting to book a session on a past date.	Past dates are disabled (unselectable) on the calendar interface.	Past dates are disabled (unselectable) on the calendar interface.	Success

3.5. Deployment

The deployment phase was executed by establishing the application on a local development server utilizing Laravel and MySQL to ensure the stability of the core architecture and database integration. As the system expanded to unify the Customer, Staff, and Admin modules, specific attention was given to the integration of the digital payment gateway via Midtrans. To facilitate this within a local environment, tunneling services (such as Ngrok) were employed to bridge the server with the public internet, thereby enabling the processing of real-time payment simulations and the reception of status callbacks. The results of this comprehensive deployment demonstrated that all system functionalities from user authentication to complex transaction processing were fully active, confirming seamless integration with third-party services and accurate data synchronization across all user roles.

3.6. Review

The system evaluation was conducted through User Acceptance Testing (UAT) involving 11 stakeholders, comprising 1 Administrator, 4 Staff members, and 6 Customers. The respondents tested key functionalities, specifically creating shift schedules, managing bookings, and initiating orders, to ensure the technical workflows aligned with operational needs. The quantitative results, as summarized in Table 2, demonstrate a 100% acceptance rate, providing empirical evidence that the system functions correctly and effectively meets the requirements of all user groups.

Table 2. User AcceptanceTesting

User Group	Number of Respondents	Key Features Tested	Successful Tests	Acceptance Rate (%)
Administrator	1	Dashboard, Employee Management, Order Management,	1/1	100%
Staff	4	Shift Scheduling	4/4	100%
Customer	6	Online Booking	6/6	100%

3. Conclusion

Based on the research conducted at April Photo Studio, the development process has successfully aligned the initial objectives outlined in the introduction with the final results. The design phase yielded comprehensive blueprints, including UML diagrams and database schemas, which guided the construction of the Web-Based Booking and Employee Management System. This application is now fully operational, featuring distinct modules that allow customers to book and pay for sessions online, staff to manage schedules and leave requests digitally, and administrators to oversee orders and revenue. Consequently, the system has successfully shifted the studio's workflow from manual processes to a digital environment. This impact is validated by quantitative evaluation results, where User Acceptance Testing (UAT) involving 11 stakeholders (administrators, staff, and customers) yielded a 100% acceptance rate. These findings empirically demonstrate that the system has improved data accuracy, streamlined management, and optimized customer service performance compared to the previous manual methods.

Regarding the prospects for future development, several enhancements are recommended to further improve the system's usability and user experience. It is suggested that future iterations include comprehensive user guides in both text and video formats to facilitate independent learning for administrators and staff, thereby streamlining the onboarding process for new employees. Additionally, the development should consider incorporating interface customization features, such as a dark mode or alternative color themes, to provide visual comfort and adaptability for users who interact with the system for extended periods.

References

[1] N. F. Rahmadhani and D. N. Istiandari, "Media Sosial Dan Digitalisasi: Analisis Perkembangan Fotografi Serta Industri Kreatif Indonesia," *Ekopedia: Jurnal Ilmiah Ekonomi*, vol. 1, pp. 2700–2712, 2025, doi: 10.63822/cn4a9q74.

[2] Y. Agustina, A. Winarno, H. Pratikto, B. S. Narmaditya, and F. Filianti, "A Creative Economy Development Strategy: The Case of Trenggalek Creative Network for Trenggalek Regency,

Indonesia," *Journal of Asian Finance, Economics and Business*, vol. 7, no. 12, pp. 1111–1122, 2020, doi: 10.13106/JAFEB.2020.VOL7.NO12.1111.

[3] A. Solechan, T. W. Ap, and B. Hartono, "Transformasi Digital Pada UMKM Dalam Meningkatkan Daya Saing Pasar," *JURNAL INFORMATIKA UPGRIS*, vol. 9, no. 1, 2023.

[4] M. E. Atmojo, "Pemberdayaan UMKM Melalui Pemanfaatan Teknologi Informasi," *Dinamisia : Jurnal Pengabdian Kepada Masyarakat*, vol. 6, no. 2, pp. 378–385, Apr. 2022, doi: 10.31849/dinamisia.v6i2.8214.

[5] P. P. Hermawan *et al.*, "Perancangan Sistem Informasi Pengolahan Data Rekam Medis Elektronik Guna Menunjang Tata Kelola Pelaporan Rawat Jalan," 2024. [Online]. Available: https://journal.stmiki.ac.id

[6] M. Rizky Firmansyah, A. Cristina Santoso, A. Farah, U. Monalissa, and M. Reza Adiyanto, "Pengaruh Pencatatan Akuntansi Manual Dengan Pencatatan Digital Di Era Globalisasi Dalam Suatu Usaha Snack Rehan Demangan Bangkalan," *JURNAL MEDIA AKADEMIK (JMA)*, vol. 2, no. 7, pp. 3031–5220, Jul. 2024, doi: 10.62281.

[7] D. Susanto, F. Huda Aminuddin, T. Djauhari, and W. Adriana, "RANCANG BANGUN SISTEM APLIKASI PENJADWALAN KARYAWAN PADA RUMAH SAKIT ST. THERESIA JAMBI BERBASIS ANDROID," *Jurnal TIMES*, vol. XI, no. 1, Jun. 2022, [Online]. Available: http://ejournal.stmik-time.ac.id

[8] M. Cahyo and W. Sulistyo, "PENJADWALAN KARYAWAN (APLIKASI METODE CYCLICAL SCHEDULING DI LAUNDRY ZONE)," *Jurnal Profit*, vol. 12, no. 2, 2018, doi: https://doi.org/10.21776/ub.profit.2018.012.02.5.

[9] A. Raya Suhari, A. Faqih, and F. M. Basysyar, "Sistem Informasi Kepegawaian Mengunakan Metode Agile Development di CV. Angkasa Raya," *Jurnal Teknologi dan Informasi*, 2022, doi: 10.34010/jati.v12i1.

[10] M. Dedi Irawan, L. Hasni, U. Asahan, J. Ahmad Yani Kisaran Telp, and K. Kunci, "SISTEM PENGGAJIAN KARYAWAN PADA LKP GRACE EDUCATION CENTER," *Jurnal Teknologi Informasi)*, vol. 1, no. 2, 2017.

[11] F. S. Bufra, P. Antari, and D. Yuza, "Rancang Bangun E-Commerce Jasa Fotografi di Kota Padang Berbasis Web," *INFORMASI (Jurnal Informatika dan Sistem Informasi)*, vol. 15, no. 1, May 2023.

[12] S. Ranti and A. D. Putra, "Penerapan Framework Laravel Pada Sistem E-Booking (Studi Kasus: 4people Studio)," *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 9, no. 3, pp. 1643–1656, Aug. 2024, doi: 10.29100/jipi.v9i3.5481.

[13] R. Rahardian *et al.*, "Agile Software Development on Design and Layout of Booking Room Website (Case Study: Witel Telkom Yogyakarta)," *Jurnal Teknik Informatika (JUTIF)*, vol. 3, no. 1, pp. 61–67, 2022, doi: 10.20884/1.jutif.2022.3.1.132.

[14] K. Anwar, L. D. Kurniawan, M. I. Rahman, and N. Ani, "Aplikasi Marketplace Penyewaan Lapangan Olahraga Dari Berbagai Cabang Dengan Metode Agile Development," *Jurnal Sisfokom (Sistem Informasi dan Komputer)*, vol. 9, no. 2, pp. 264–274, Aug. 2020, doi: 10.32736/sisfokom.v9i2.905.

[15] R. Kivania, A. Novianti, and R. Firmansyah, "Analisis Implementasi Peranan Sistem Reservasi Pada Bisnis Di Sektor Industri," *Sosial dan Humaniora*, vol. 1, no. 1, 2023.

[16] Julianto Simatupang and Setiawan Sianturi, "Perancangan Sistem Informasi Pemesanan Tiket Bus Pada Po. Handoyo Berbasis Online," *Intra-Tech*, vol. 3, Oct. 2019.

[17] R. Santika, R. Ayuni, M. Tri, and I. Rahmayani, "RANCANG BANGUN SISTEM INFORMASI KEPEGAWAIAN BERBASIS WEBSITE PADA MA MIFTAAHUL'ULUM KABUPATEN BENGKALIS," *Jurnal Teknologi Informasi*, vol. 4, no. 1, 2023, doi: 10.46576/djtechno.

[18] A. C. Hutauruk and A. F. Pakpahan, "Perancangan Sistem Informasi Organisasi Kemahasiswaan Berbasis Web pada Universitas Advent Indonesia Menggunakan Metode Agile Development (Studi Kasus: Universitas Advent Indonesia)," *Cogito Smart Journal*, vol. 7, no. 2, p. 2021, 2021.

[19] N. Dengen Heliza Rahmania Hatta, "Program Studi I lmu Komputer Universitas Mulaw arman Perancangan Sistem Informasi Terpadu Pemerintah Daerah Kabupaten Paser," 2009.

[20] I. Prayoga Silalahi, "Pengaruh Enterprise Resource Planning (ERP) Terhadap Kinerja Perusahaan Usaha Kecil Menengah," *Jurnal Syntax Admiration*, vol. 3, no. 6, pp. 768–775, Jun. 2022, doi: 10.46799/jsa.v3i6.440.

[21] "Laravel, 'Laravel documentation,' Laravel. [Online]. Available: https://laravel.com/docs [Accessed: Dec. 9, 2024]."

[22] E. Siswanto and M. Kom, *Belajar Laravel*. 2023.

[23] Y. Handrianto and B. Sanjaya, "Model Waterfall Dalam Rancang Bangun Sistem Informasi Pemesanan Produk Dan Outlet Berbasis Web," *JII: Jurnal Inovasi Informatika Universitas Pradita*, vol. 5, no. 2, p. 8462039, Sep. 2020.
[24] S. Pratasik and I. Rianto, "Pengembangan Aplikasi E-DUK Dalam Pengelolaan SDM Menggunakan Metode Agile Development The Development Of E-DUK Application in HR Management Using Agile Development Method," *Cogito Smart Journal* |, vol. 6, no. 2, 2020.
[25] Soetam Rizky Wicaksono, "Blackbox Testing (Teori dan Studi Kasus)," 2022.
[26] L. P. Sumirat, D. Cahyono, Y. Kristyawan, and S. Kacung, *Dasar - Dasar Rekayasa Perangkat Lunak*. Madza Media, 2023. [Online]. Available: www.madzamedia.co.id