



Development of a Web-Based E-Commerce System for Amelia Hantaran Using the V-Model

Muhammad Mahardicka Alfattah Zelda¹, Anargya Rabbani Aslam², Juanda For Trio³, Maulana Zulfan Azka⁴, Aditya Wicaksono^{5*}, Muhammad Nasir⁶

^{1,2,3,4,5,6} Teknologi Rekayasa Perangkat Lunak, Sekolah Vokasi IPB University, Indonesia

¹5464zelda@apps.ipb.ac.id, ²rbannaslam@apps.ipb.ac.id, ³veronikajuanda@apps.ipb.ac.id,

⁴zulfanazka@apps.ipb.ac.id, ^{5*}adityawicaksono@apps.ipb.ac.id, ⁶m_nasir@apps.ipb.ac.id

Abstract: The rapid growth of the digital economy has encouraged Micro, Small, and Medium Enterprises (MSMEs) to adopt technology in order to improve competitiveness and business performance. However, many wedding gift (seserahan) businesses still rely on conventional methods and social media platforms such as WhatsApp and Instagram for marketing and ordering processes, which often lead to inefficiencies and limited access to real-time information. This study aims to design and develop a web-based wedding gift rental information system for Amelia Hantaran to improve service quality and operational efficiency. The system is developed using the V-Model, which consists of requirement analysis, system design, implementation, testing, and maintenance stages. The proposed system provides features such as user registration, product catalog access, and online ordering, as well as administrative functions for managing product and order data. The results show that the developed system facilitates customers in accessing product information and placing orders more efficiently, while also assisting business owners in managing data more structured and organized. Therefore, implementing this system can enhance business performance and support the digital transformation of MSMEs in the wedding gift sector.

Keywords: Wedding Gift Rental; Information System; V-Model

1. INTRODUCING

The digital economy era has become a key driver of innovation and competition among business actors. Currently, only about 12% of Micro, Small, and Medium Enterprises (MSMEs) in Indonesia have adopted digitalization, even though implementing digital technologies significantly contributes to increasing national income [1]. In addition, global trends indicate that the COVID-19 pandemic has accelerated digital transformation, with companies that adapt to technology tending to be more competitive [2].

A similar phenomenon also occurs in Indonesia's wedding gift (seserahan) business sector. Most MSME actors still rely on social media platforms such as WhatsApp and Instagram for marketing and ordering processes [3]. However, this method has several limitations, such as customers' inability to access real-time product availability and the order management process still being conducted manually, which is error-prone [4].

These problems indicate the need for a technology-based solution that can improve the effectiveness and efficiency of business processes. An e-commerce system is one solution that can be implemented to expand market reach, increase sales, and simplify the management of product and order data [5].



Given these issues, this study aims to develop a web-based wedding-gift rental information system for Amelia Hantaran. Unlike previous approaches that primarily rely on social media for product promotion and ordering, this system integrates product catalog access, product availability information, online ordering, and administrative management of products, stock, orders, and customer testimonials within a single platform [3][4]. This integration is intended to provide customers with a more structured ordering experience while enabling business owners to manage their business data more effectively and systematically.

2. RESEARCH METHODOLOGY

This study applies the Verification and Validation Model (V-Model), a highly structured extension of the traditional Waterfall model [6]. The V-Model is particularly suitable for developing the Amelia Hantaran web-based system because it associates a specific testing phase with each corresponding development stage, ensuring high reliability for critical features such as order processing and stock management. As illustrated in the methodology framework, the process is systematically divided into the Verification Phase (development), Coding, the Validation Phase (testing), and concludes with deployment and maintenance.

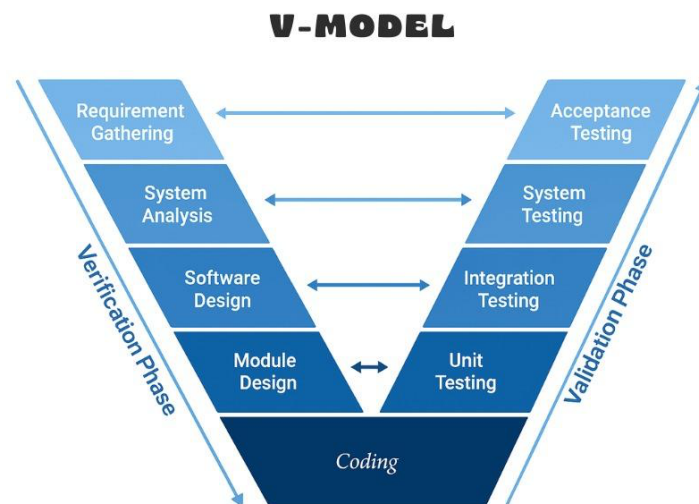


Figure 1. V-Model Waterfall

2.1. Verification Phase

The verification phase focuses on understanding system requirements and designing the solution progressively before actual coding begins [7]. The process starts with Project Initiation, marked by a kickoff meeting to establish the initial scope. This is followed by Requirement Gathering, which involves identifying existing problems and analyzing both functional and non-functional requirements to produce a finalized Software Requirements Specification (SRS). Next, System Analysis is conducted to evaluate the current workflow and identify proposed system solutions. The process then moves to Software Design, where the overall architecture, Use Case and Activity Diagrams, Entity-Relationship Diagrams (ERD), and User Interface (UI) wireframes are systematically designed. Finally, Module Design is performed to specify the detailed structure and specifications of each individual module before development.

2.2. Coding Phase

Positioned at the bottom of the V-Model, the coding phase serves as the execution stage where the designs are translated into functional software. This stage involves setting up the development environment and writing code for core features such as the product catalog and ordering system, followed by the development of additional functional features. The phase concludes with system integration and rigorous debugging to finalize the application, ensuring it is fully prepared for the subsequent validation phases.

2.3. Validation Phase

The validation phase ensures that the developed system strictly meets the specifications defined during the verification phase through a structured testing hierarchy [8]. It begins with Unit Testing, evaluating individual modules to identify and fix bugs at the earliest stage. This is followed by Integration Testing to validate the seamless interaction between the combined modules. System Testing is then conducted to verify that the integrated system aligns with the initial requirements and functions correctly. Finally, Acceptance Testing is performed, involving user acceptance trials to ensure final system validation from the client's perspective before the system is released.

3.RESULT AND DISCUSSIONS

3.1. Requirement Gathering

The requirements-gathering phase begins with problem identification to establish a clear understanding of the existing business limitations in the conventional wedding-gift rental process. This is followed by a comprehensive analysis of the system's requirements. The functional requirements define 14 core capabilities tailored for two main actors: the User and the Admin. Non-functional requirements are also strictly established, encompassing data security, cross-browser compatibility, and a maximum page load time of 3 seconds. To visually represent the system's functional boundaries and the interactions between the actors and the system, a Use Case Diagram is developed [9]. All these elements, from problem identification to functional specifications and diagrams, are formally consolidated into the Software Requirements Specification (SRS) document [10].

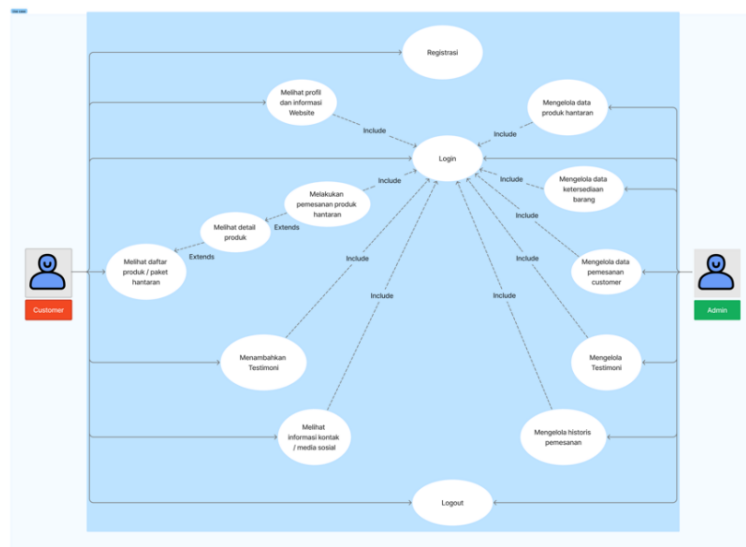


Figure 2. Use Case Diagram

As illustrated in Figure 2, the Use Case Diagram outlines the interaction between the system and its two primary actors: the User (customer) and the Admin (business owner). The User is authorized to perform customer-centric actions, such as browsing the company profile, managing the shopping cart, and placing orders. Conversely, the Admin is granted elevated privileges to manage the product catalog, process incoming orders, and update system configurations. Both actors must complete a secure login process before accessing their respective main functionalities.

3.2. System Analysis

Following the requirements definition, the system analysis phase focuses on outlining the system process flow design and conducting system requirements validation. This stage translates the finalized SRS into a logical operational framework. By designing the overarching process flows, the development team can validate that the proposed systemic solutions align perfectly with Amelia Hantaran's predefined business needs. The goal of this validation step is to ensure that all logical scenarios are viable and sound before moving into detailed architectural modeling [11].

3.3. Software Design

The software design phase provides a comprehensive blueprint of the software's dynamic behavior, static structure, and visual interface. The dynamic workflows, such as checkout and order verification, are modeled using Activity Diagrams and Sequence Diagrams to map the step-by-step execution of user actions and system responses. The system's object-oriented structure is then mapped using a Class Diagram. Furthermore, the database architecture is constructed through an Entity-Relationship Diagram (ERD). The ERD is meticulously designed with complete cardinality and strictly uses standard diamond symbols to accurately define the relationships between entities, ensuring high data integrity [12]. Finally, this phase involves creating UI wireframes and mockups to guide the front-end implementation.

Activity Diagrams detail the dynamic operational workflow of the ordering process within the system. It maps the sequential steps starting from the user selecting a product and proceeding to the checkout interface. The diagram illustrates how the system verifies the input data, calculates the total transaction amount, and subsequently updates the order status. It also shows the backend response, in which the Admin is notified to verify the payment and process the rental delivery, ensuring a transparent and traceable transaction loop.

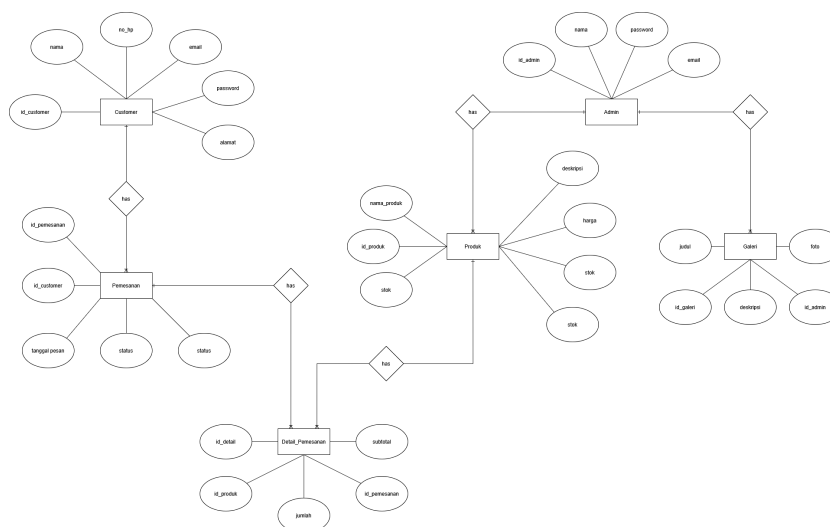


Figure 3. Entity Relationship Diagram

Based on Figure 3, the Entity-Relationship Diagram defines the relational database architecture supporting the application. It highlights the connections between core data entities, such as Customers, Admins, Products, Orders, Order Details, and Gallery. The diagram specifies cardinalities, such as the one-to-many relationship between a single User and multiple Orders. This structural mapping guarantees that the stored data remains consistent and prevents data anomalies.

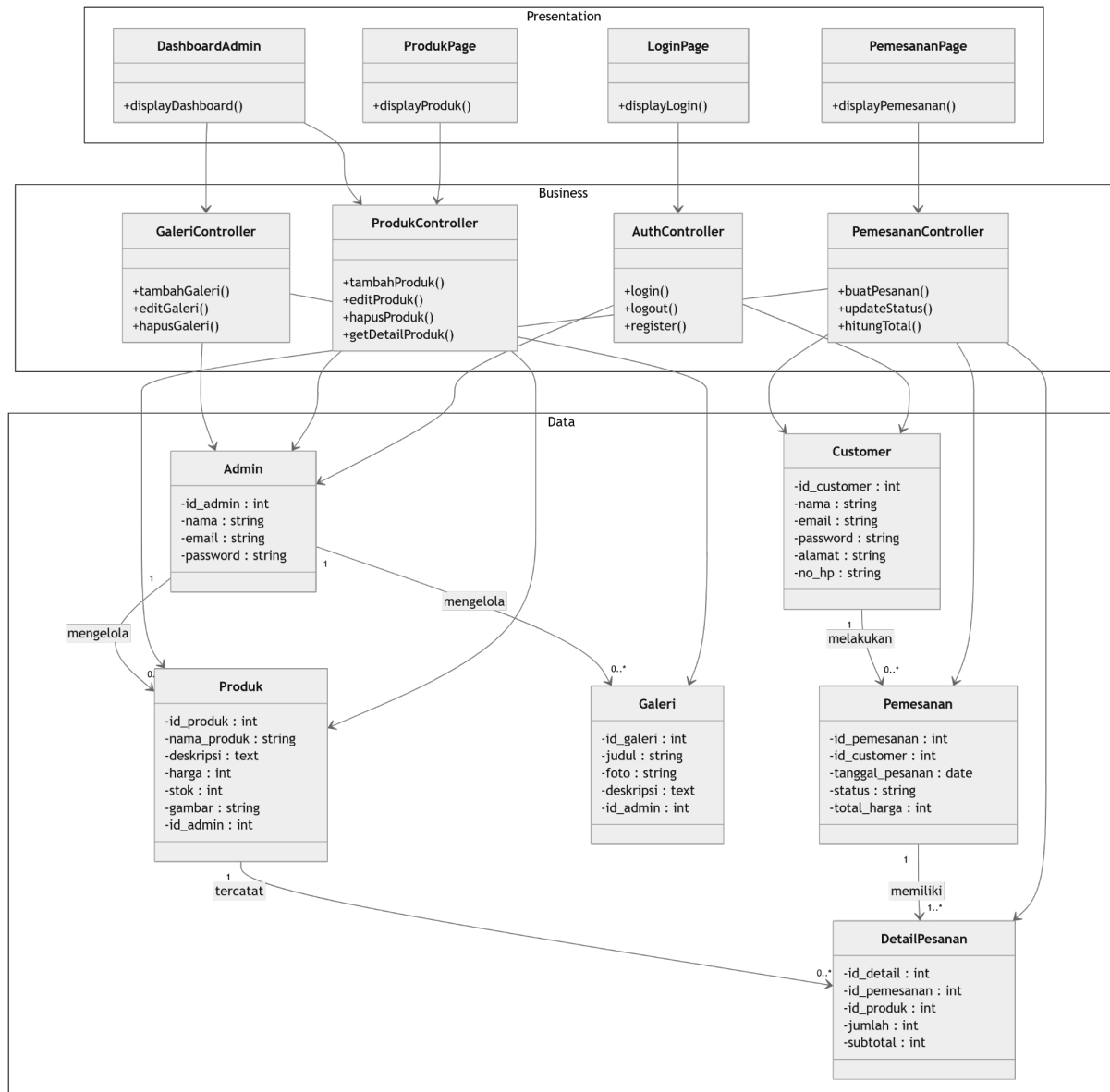


Figure 4. Class Diagram

Figure 4 visualizes the Class Diagram, which translates the database entities into an object-oriented software structure. It defines the primary classes used in the application's backend, detailing their attributes and methods. This diagram serves as the direct foundation for the subsequent coding phase, bridging the gap between data storage and application logic.

3.4. Module Design

Based on the architectural design and the Software Design Description, the system is decomposed into smaller, highly cohesive, and manageable modules. The core system is divided into several primary modules, including the Authentication Module, the Product and Category Management Module, the Shopping Cart and Order Processing Module, and the Admin Dashboard Module. This modularization strategy isolates specific functionalities, thereby simplifying the subsequent coding phase, facilitating independent unit testing, and ensuring future system maintainability and scalability.

3.5. Coding

The User Interface aims to translate the system's logical design into a visual interface that is intuitive, aesthetically pleasing, and easy for users to operate. The website interface is divided into two main categories: the interface for Customers and the interface for Admins.

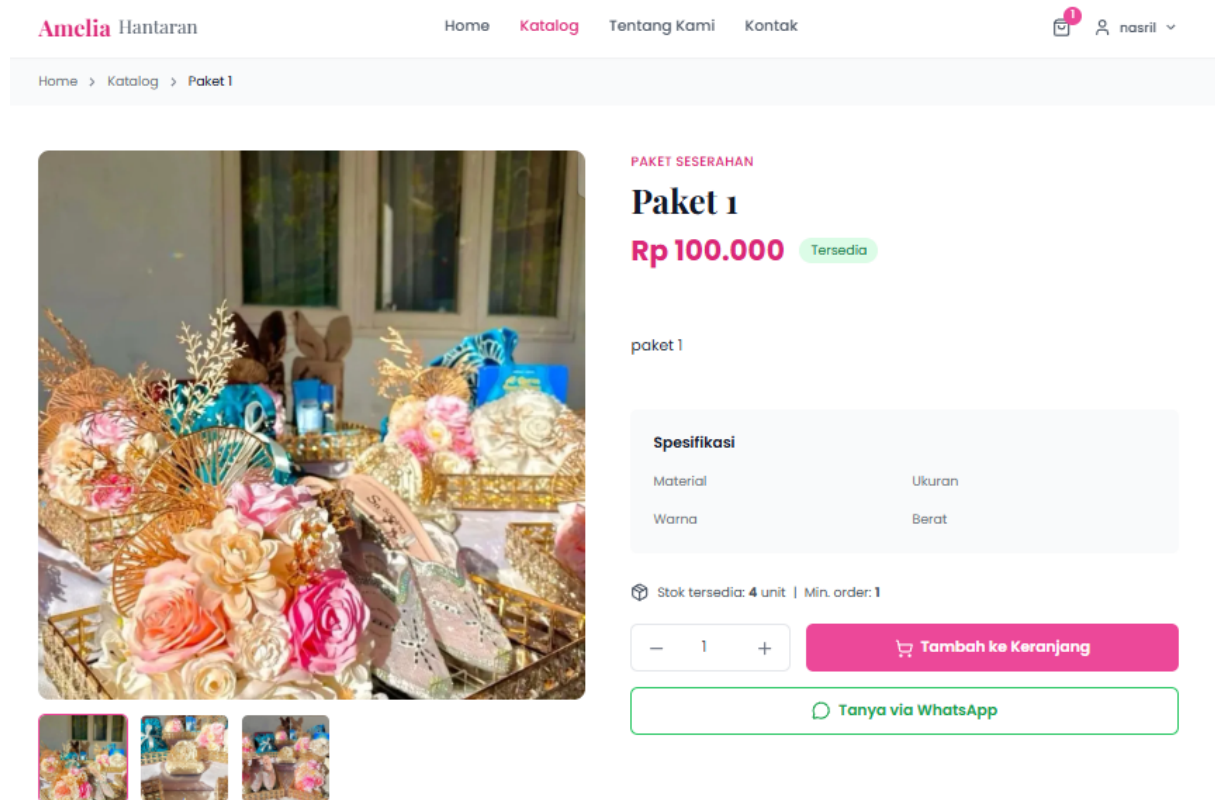


Figure 5. Customer Product Page

The customer product page provides an in-depth presentation of a specific rental item to facilitate informed purchasing decisions. The layout is logically divided into two main sections: the left panel focuses on visual representation through a high-resolution primary image and a supplementary thumbnail gallery, while the right panel displays structured product metadata, including the category, title, explicit pricing, real-time availability, and detailed specifications. To streamline the transaction process, the interface integrates transparent inventory data, a quantity adjustment tool, and a primary "Tambah ke Keranjang" (Add to Cart) button, along with a "Tanya via WhatsApp" button that provides a direct communication channel for immediate, personalized customer support.



JURNAL INFORMATIKA DAN REKAYASA PERANGKAT LUNAK (JATIKA)

Volume 7, Number 3, September 2026, Page 187-201

E-ISSN 2797-2011

P-ISSN 2797-3492

<https://publikasi.teknokrat.ac.id/index.php/jatika/index>

DOI: <https://doi.org/10.33365/jatika.v7i3.2737>

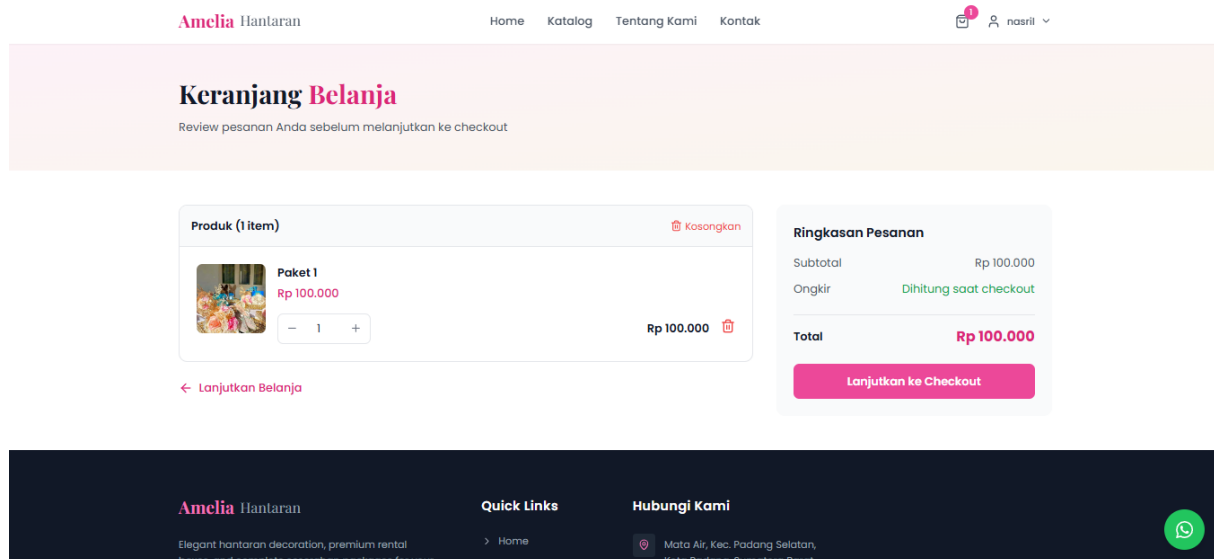


Figure 6. Customer Cart Page

The customer cart page serves as a critical intermediary interface where customers can systematically review and manage their selected items before finalizing a transaction. To ensure optimal clarity for users, the interface is divided into two main sections. The left panel displays a detailed breakdown of queued products, with intuitive controls that let users dynamically adjust item quantities, remove individual items, or clear the entire cart. Concurrently, the right panel presents a transparent order summary. This section calculates the real-time subtotal and explicitly states that shipping costs will be determined during the subsequent step, ensuring financial transparency. This clean, structured layout provides customers with full visibility and control over their order, culminating in a prominent call-to-action button that smoothly guides them to the final checkout process.

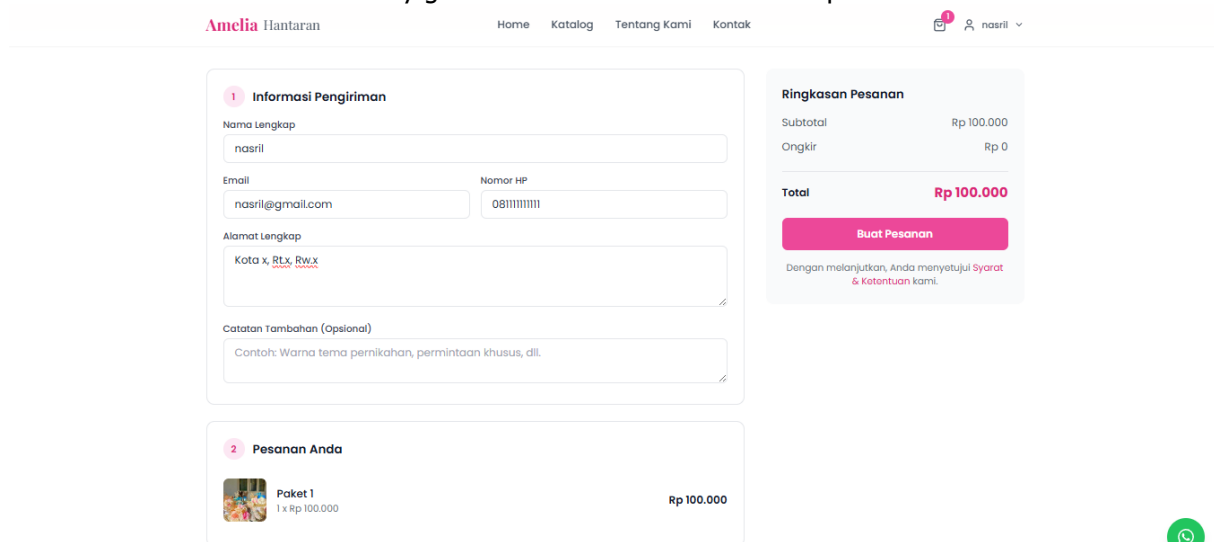
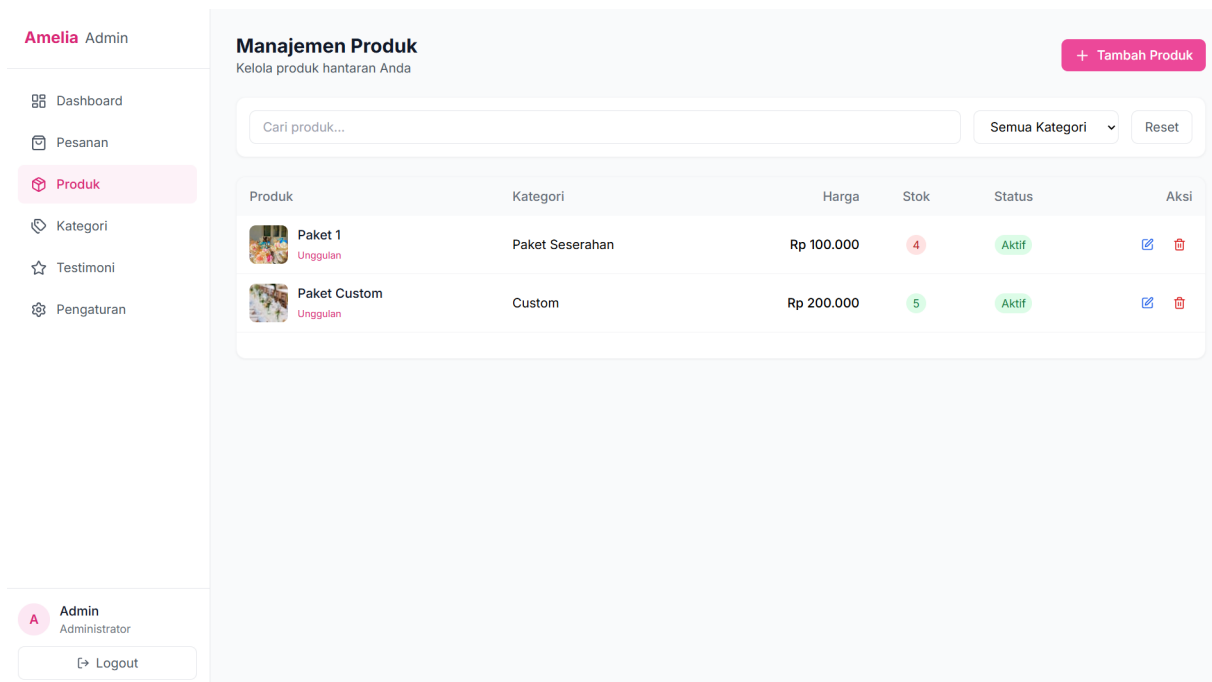


Figure 7. Customer Order Page

The customer order page, serving as the final checkout interface, is designed to securely capture essential transaction and delivery data. To minimize cognitive load during the final purchasing step, the



layout employs a structured, two-column grid. The primary left panel is systematically divided into two sequential steps: the first section requires the user to input critical shipping information (contact details and full address) and includes a dedicated text area for customized notes, such as specific wedding theme color requests. The second section provides a final visual verification of the selected rental items to ensure order accuracy. Concurrently, the right-hand panel displays a persistent order summary that transparently calculates the subtotal, final shipping costs, and the total payable amount. The transaction is concluded through a prominent "Buat Pesanan" (Place Order) button, which is integrated with explicit terms and conditions agreement, ensuring a secure and legally compliant checkout process.



The screenshot shows the 'Manajemen Produk' (Product Management) interface. On the left is a sidebar with a user profile for 'Amelia Admin' and a menu with options: Dashboard, Pesanan, Produk (highlighted), Kategori, Testimoni, and Pengaturan. The main content area is titled 'Manajemen Produk' with the subtitle 'Kelola produk hantaran Anda'. It features a search bar, a category filter set to 'Semua Kategori', and a 'Reset' button. Below this is a table with columns: Produk, Kategori, Harga, Stok, Status, and Aksi. The table lists two products: 'Paket 1' (Unggulan) with a price of Rp 100.000 and stock of 4, and 'Paket Custom' (Unggulan) with a price of Rp 200.000 and stock of 5. Both are 'Aktif'. Each row has edit and delete icons in the 'Aksi' column. A '+ Tambah Produk' button is in the top right corner.

Produk	Kategori	Harga	Stok	Status	Aksi
 Paket 1 Unggulan	Paket Sederahan	Rp 100.000	4	Aktif	 
 Paket Custom Unggulan	Custom	Rp 200.000	5	Aktif	 

Figure 8. Admin Product Page

The product management page provides the administrative mechanism for maintaining the rental catalog. Search, category filtering, and the add-product button are positioned above the product table, allowing administrators to either locate existing items or create new ones from the same workspace. The table structure covers core product attributes such as category, price, stock, status, and action. Through this arrangement, catalog maintenance becomes more systematic because product visibility and stock-related information can be managed from a single interface.



JURNAL INFORMATIKA DAN REKAYASA PERANGKAT LUNAK (JATIKA)

Volume 7, Number 3, September 2026, Page 187-201

E-ISSN 2797-2011

P-ISSN 2797-3492

<https://publikasi.teknokrat.ac.id/index.php/jatika/index>

DOI: <https://doi.org/10.33365/jatika.v7i3.2737>



Amelia Admin

Dashboard

Pesanan

Produk

Kategori

Testimoni

Pengaturan

Admin Administrator

Logout

Manajemen Pesanan

Kelola dan verifikasi pesanan pelanggan

Cari nomor/nama/email...

Semua Status

Reset

Nomor	Pelanggan	Tanggal	Total	Status	Aksi
ORD-20260525-ECJYM1	zulfan 088923893223	25 May 2026 14:13	Rp 300.000	Selesai	Detail

Figure 9. Admin Order Page

Order management is represented via a table-based interface with search and status filtering. The available columns, including order number, customer, date, total, status, and action, indicate that each transaction can be reviewed in a structured manner. Even when no order data is available, the empty-state message still clearly communicates the system's condition to the administrator. This design is important for operational reliability, since order verification and follow-up actions depend on accurate and easily traceable transaction records.

3.6. Unit Testing

The Unit Test phase is conducted to ensure that every feature of the Amelia Hantaran website meets the defined requirements. Testing is performed on each module individually to verify its functionality before the modules are integrated into the complete system. The objective of this phase is to identify defects, validate feature performance, and improve system reliability. The testing process includes evaluating customer-facing features as well as administrative functions. All testing activities, issue findings, corrective actions, and validation results are documented in the Document of Unit Programming and Testing (DUPL) as evidence that each module has met the expected quality standards.

Table 1. Cart Function Testing Results

Identification	[DUPL-AMELIAHANTARAN.K-006]
Test Case Name	Cart Function
Case Description	The Cart function on the Amelia Hantaran Website was tested for adding item quantities, reducing item quantities, deleting products, clearing the cart, calculating the total purchase amount, and the checkout process.
Initial Condition	Internet access is available, and the user has added products to the shopping cart.
Testing Date	May 14, 2026
Tester	Muh Fahrul Fahrezi

Test Scenario

Aditya Wicaksono: *Corresponding Author



Copyright © 2026, All Authors.



Test procedure steps for case [DUPL-AMELIAHANTARAN.K-006]:

1. The user opens the cart page.
2. The system displays the list of products in the shopping cart.
3. The user changes the item quantity, removes a product, or clears the cart.
4. The system automatically updates the subtotal and total purchase amount.
5. The user clicks the checkout button to proceed with the ordering process.

Test Results		
Expected Result	Observation	Conclusion
1. The system successfully increases the item quantity and automatically updates the subtotal.	1. The system successfully updated the item quantity and subtotal.	OK
2. The system successfully decreases the item quantity and automatically updates the subtotal.	2. The system successfully updated the item quantity and subtotal.	OK
3. The system prevents the item quantity from being less than 1.	3. The system successfully enforced the minimum item quantity.	OK
4. The system successfully removes a product from the cart.	4. The system successfully removed the product and updated the total.	OK
5. The system successfully removes all products from the cart.	5. The system successfully cleared the cart.	OK
6. The system successfully updates the total purchase amount according to the item quantity.	6. The system successfully updated the total purchase amount.	OK
7. The system successfully redirects the user to the checkout page.	7. The system successfully opened the checkout page.	OK

Table 2. Order Function Testing Results

Identification	[DUPL-AMELIAHANTARAN.K-007]
Test Case Name	Order Function
Case Description	The Order function on the Amelia Hantaran Website was tested under normal conditions, with empty required fields, invalid order quantities, and the order data submission process.
Initial Condition	Internet access is available, and the user has logged into the system.
Testing Date	May 14, 2026
Tester	Muh Fahrul Fahrezi

Test Scenario

Test procedure steps for case [DUPL-AMELIAHANTARAN.K-007]:

1. The user opens the order page.
2. The system displays the product data to be ordered.
3. The user fills in the order information in the provided fields.
4. The user clicks the Create Order button.
5. The system validates the data and stores the order information in the database.





Test Results		
Expected Result	Observation	Conclusion
1. The system successfully stores the order data when all required information is entered correctly.	1. The system successfully received and stored the order data.	OK
2. The system displays an error message when required fields are left empty.	2. The system successfully displayed a validation message.	OK
3. The system rejects invalid order quantities.	3. The system successfully displayed an error message.	OK
4. The system displays the order details after the process is completed successfully.	4. The system successfully displayed the order details.	OK

Table 3. Percentage of Testing Results

No	Feature Tested	Number of Test Scenarios	Passed	Failed	Percentage
1	Cart	7	7	0	100%
2	Pesanan	4	4	0	100%
Total		11	11	0	100%

Based on the results of functional testing of the Cart and Order features on the Amelia Hantaran website, all test scenarios were successfully executed in accordance with the system requirements. The Cart feature testing included adding item quantities, reducing item quantities, deleting products, clearing the cart, calculating the total purchase amount, and the checkout process. Meanwhile, the Order feature testing covered input validation, order data storage, and the display of order details. Of the 11 test scenarios conducted, all were completed successfully, resulting in a 100% success rate. These results indicate that the Cart and Order functions comply with the established functional requirements.

In addition to the functional test results, the overall testing documented in the DUPL consisted of 79 test scenarios, with 77 scenarios successfully passed and 2 scenarios failed, resulting in an overall success rate of 97.47%. The successful results indicate that most of the tested functions operated according to the specified requirements. The two failed scenarios were identified in the non-functional testing, particularly in responsive display and website performance, and were used as indicators for further system improvement.

3.7. Integration Testing

The Integration Test phase is conducted to verify that all modules within the Amelia Hantaran website can interact and function properly as a unified system. After each feature has successfully passed unit testing, the modules are integrated and tested collectively to ensure seamless data exchange and process execution. This phase focuses on validating the interaction between the user

interface, application logic, and database components. The integration testing process aims to identify communication errors, data inconsistencies, and workflow disruptions that may occur when multiple modules operate simultaneously. The results of the testing activities are documented and used as a basis for further system refinement.

Table 4. Integration Testing

No	Integration Scenario	Integrated Modules	Result
1	Adding products to the cart and changing product quantities	Product - Cart	Success
2	Changing product quantities and updating subtotal/total	Cart - Transaction	Success
3	Proceeding from cart to checkout	Cart - Order	Success
4	Creating an order based on selected product	Product - Order	Success
5	Displaying order details after a successful transaction	Order - Details	Success

3.8. System Testing

The System Test phase evaluates the Amelia Hantaran website as a complete, integrated system. This phase aims to ensure that all functional and non-functional requirements specified in the Software Requirements Specification (SRS) have been fulfilled. Testing is performed in an environment that closely resembles actual operational conditions to assess the system's overall performance, reliability, usability, and compatibility. The test results are analyzed to identify potential issues and determine whether the system is ready for deployment and real-world use.

The system testing results also revealed several limitations that should be considered. On smartphone devices, some image elements slightly exceeded the screen boundaries, indicating that further responsive layout optimization is still required. In addition, several product images required a longer loading time during performance testing. These findings did not prevent the main functions from operating, but they indicate that image optimization and responsive interface refinement are still needed to improve the overall quality of the system.

Table 5. System Testing

No	System Testing Scenario	Result
1	Customer registers and logs into the system	Success



2	Customer views the catalog and product details	Success
3	Customer adds products to the cart and places an order	Success
4	Admin manages product data and availability	Success
5	Admin views and manages customer order data	Success
6	Customer provides a testimonial after using the service	Success
7	System tested on multiple browsers	Success

3.9. Acceptance Testing

The Acceptance Testing phase is conducted to determine whether the Amelia Hantaran website meets the business requirements and user expectations. This phase represents the final evaluation stage before the system is officially deployed for operational use. Testing is conducted by representative users and stakeholders who assess the system using real business scenarios and workflows. The objective of acceptance testing is to ensure that all implemented features function correctly, support the rental process effectively, and provide a satisfactory user experience. The results obtained during this phase serve as the basis for the system's final approval. Acceptance Testing was conducted for user to evaluate whether the developed system met the functional needs of Amelia Hantaran from the perspective of its intended users. The testing involved customers and the business owner as respondents, covering key activities such as viewing product information, placing orders, and managing product, stock, and order data. The test results showed that all evaluated scenarios were accepted by the respective respondents, indicating that the developed system was able to support the main business processes of Amelia Hantaran.

Table 6. Acceptance Testing

No	Test Scenario	Respondent	Result
1	View product catalog and details	Customer	Accepted
2	Place a product order	Customer	Accepted
3	View contact and social media information	Customer	Accepted
4	Manage products	Owner	Accepted
5	Manage product stock	Owner	Accepted
6	Manage customer orders	Owner	Accepted
7	View order history	Owner	Accepted



3.10 System Impact and Limitations

The implementation of the Amelia Hantaran website has an impact on both customer service and business administration. From the customer perspective, the availability of an online product catalog, product details, product availability information, and online ordering provides a more structured access to information compared with relying solely on conventional communication channels. From the business owner's perspective, product, stock, and order management features support more systematic data management and reduce the need to handle these processes manually. The integration of these functions also provides a centralized platform for managing the main rental activities.

However, the impact of the system in this study is evaluated based on functional and technical testing rather than direct measurements of business performance. Therefore, this study does not yet provide quantitative evidence regarding increases in sales, customer numbers, or revenue after implementation. In addition, the testing identified minor limitations related to responsive image display on smartphones and longer loading times for some product images. These limitations indicate opportunities for further optimization in future development.

4. CONCLUSION

This research successfully designed and implemented a web-based wedding gift information and rental system for Amelia Hantaran using the V-Model method. The system addresses the limitations of conventional marketing by allowing customers to access product catalogs and place orders in real time, while enabling business owners to manage products, availability, and order data more systematically. Its implementation improves service quality, operational efficiency, and market reach, supporting the digital transformation of MSMEs in the wedding gift rental sector. Future improvements could include integrating third-party payment gateways, adding email or WhatsApp order notifications, and developing a mobile application to further enhance accessibility and customer experience.

5. CREDIT AUTHORSHIP CONTRIBUTION STATEMENT

Muhammad Mahardicka Alfattah Zelda: Conceptualization, Methodology, Writing - Original Draft. **Anargya Rabbani Aslam, Juanda For Trio, Maulana Zulfan Azka:** Software, Validation, Investigation, Data Curation, Visualization, Writing - Review & Editing. **Aditya Wicaksono, Muhammad Nasir:** Supervision, Project Administration.

6. DECLARATION OF COMPETING INTEREST

The authors declare that there are no financial, personal, or professional relationships that could have influenced or potentially affected the research, analysis, or publication of this article.

7. FUNDING STATEMENTS

This research received no external funding for the research, authorship, and/or publication of this article.

8. DECLARATION OF GENERATIVE AI AND AI-ASSISTED TECHNOLOGIES IN THE WRITING PROCESS

During the writing process, the authors used Elicit AI to search for and analyze relevant literature references used in this manuscript. The authors reviewed, verified, and selected the relevant sources, and take full responsibility for the accuracy and originality of the final content of the manuscript.





9. REFERENCES

- [1] Yuli Setiawan, "DIGITALISASI UMKM MELALUI E-COMMERCE SEBAGAI PENINGKATAN PENDAPATAN NASIONAL," *Jurnal Manajemen*, vol. 7, no. 2, pp. 76–83, Feb. 2023, doi: 10.54964/manajemen.v7i2.250.
- [2] K. Jakšić, S. Trajković, M. Delečić, and V. Delečić, "Digital transformation of business through e-commerce in small and medium-sized enterprises (SME's)," *International Review*, no. 3–4, pp. 154–162, 2023, doi: 10.5937/intrev2304151J.
- [3] Ustadus Sholihin and Imam Mukhlis, "Penggunaan Media Sosial dan Market Place Untuk Mengembangkan Pemasaran UMKM," *Jurnal Bintang Manajemen*, vol. 1, no. 4, pp. 204–218, Nov. 2023, doi: 10.55606/jubima.v1i4.2266.
- [4] F. Akbar Ramadhan, G. Gunawan, Y. Erwadi, and C. Prihantoro, "Sistem Informasi Penyewaan Kotak Seseheran Berbasis Web," *JUSIBI (Jurnal Sistem Informasi dan E-Bisnis)*, vol. 4, no. 1, pp. 41–48, Jan. 2022, doi: 10.54650/jusibi.v4i1.424.
- [5] Dicha Putri Rezkia, "Analisis Penggunaan E-Commerce Bagi UMKM Di Era Digital," *Prosiding Seminar Nasional Ilmu Manajemen Kewirausahaan dan Bisnis*, vol. 1, no. 1, pp. 151–163, Jun. 2024, doi: 10.61132/prosemmasimkb.v1i1.13.
- [6] S. A. Putri and F. Candra, "Website-Based Development of Learning Management System using V-Development Method Model (Case Study: UNRI, Informatics Engineering Study Program)," *JINAV: Journal of Information and Visualization*, vol. 4, no. 1, pp. 79–92, Feb 2023, doi: 10.35877/454RI.jinav1575.
- [7] S. Pargaonkar, "A Comprehensive Research Analysis of Software Development Life Cycle (SDLC) Agile & Waterfall Model Advantages, Disadvantages, and Application Suitability in Software Quality Engineering," *International Journal of Scientific and Research Publications*, vol. 13, no. 8, pp. 120–124, Aug 2023, doi: 10.29322/IJSRP.13.08.2023.p14015.
- [8] C. H. Lim and J. F. Chin, "V-model with fuzzy quality function deployments for mobile application development," *Journal of Software: Evolution and Process*, vol. 35, no. 1, Jan 2023, doi: 10.1002/smr.2518.
- [9] H. Sitompul, Z. Matondang, E. Daryanto, and F. Syahputra, "Use Case Diagram Design For Information System Services To Students At The Faculty Of Engineering Universitas Negeri Medan," in *Proceedings of the 5th International Conference on Innovation in Education, Science, and Culture, ICIESC 2023*, 24 October 2023, Medan, Indonesia, EAI, 2024. doi: 10.4108/eai.24-10-2023.2342345.
- [10] N. Qamar, N. Sabahat, and A. Mosavi, "Evaluating the Impact of Pair Documentation on Requirements Quality in Agile Software Development," *IEEE Access*, vol. 13, pp. 45784–45794, 2025, doi: 10.1109/ACCESS.2025.3550123.
- [11] S. Chang, C. Chang, X. Liu, and X. Gao, "A Scenario-Based Visual Modeling Method for the Complex Products Lifecycle," *Electronics (Basel)*, vol. 15, no. 6, p. 1198, Mar. 2026, doi: 10.3390/electronics15061198.
- [12] S. M. Pulungan, R. Febrianti, T. Lestari, N. Gurning, and N. Fitriana, "Analisis Teknik Entity-Relationship Diagram Dalam Perancangan Database," *Jurnal Ekonomi Manajemen dan Bisnis (JEMB)*, vol. 1, no. 2, pp. 98–102, Feb. 2023, doi: 10.47233/jemb.v1i2.533.
- [13] B. Hardika, "Blackbox Testing of the Garuda Farm Website Using the Equivalence Partitioning Technique," *Kst, Kri, Kridatama*, vol. 6, no. 02, pp. 740–753, Dec. 2024
- [14] I. A. Shaleh, P. J. Prayogi, R. Syawal, and A. Saifudin, "Penguji Black Box pada sistem informasi penjualan buku berbasis web dengan teknik Equivalent Partitions," *Jurnal Teknologi Sistem Informasi dan Aplikasi*, 2021.
- [15] N. Hidayati and S. Sismadi, "Application of Waterfall Model In Development of Work Training Acceptance System," *INTENSIF: J. Ilm. Penelit. dan Penerap. Tek. Sist. Inf.*, vol. 4, no. 1, pp. 75–89, Feb. 2020, doi: [10.29407/intensif.v4i1.13575](https://doi.org/10.29407/intensif.v4i1.13575).

