

Laravel Filament-Based Driver Commission Management and Calculation System Automation

Alfito Aditya Trishardi¹

¹ Sekolah Tinggi Ilmu Komputer Banyuwangi, Jawa Timur, Indonesia.

Article Info

Article history:

Received : mm dd, yyyy

Revised : mm dd, yyyy

Accepted : mm dd, yyyy

Keywords:

Commission System;

Restaurant Drivers;

Laravel;

Laravel Filament;

Abstract

Mentari Restaurant relies on delivery services provided by drivers as part of its daily operations; however, the processes for recording deliveries and calculating driver commissions—ranging from data compilation to payment processing—are still handled manually. This approach leaves the system vulnerable to calculation errors, data opacity, and payment delays, while the lack of an integrated system hinders comprehensive monitoring of driver performance. This research developed a web-based Driver Commission System using the Laravel framework and the Laravel Filament administration panel. The system automates delivery logging, commission calculations, driver data management, and reporting within a single integrated platform, utilizing the Iterative Waterfall development method and data collection techniques such as observation, interviews, and document analysis. Black-box testing across sixteen scenarios demonstrated that all system functions (100%) operated as required; consequently, the system is expected to provide a concrete solution for improving commission calculation accuracy, accelerating payment processes, and strengthening trust between drivers and Mentari Restaurant management.

Corresponding Author:

Alfito Aditya Trishardi,
alfitoadityat@gmail.com

INTRODUCTION

In the digital era, information technology has transformed the way culinary businesses manage their operations. Web-based systems now enable integrated, real-time data processing, including the management of employee compensation. A Laravel-based information system, for example, has been shown to process and store data accurately while producing information that can be directly used for decision-making (Indah Melyani et al., 2023).

Mentari Restaurant, a culinary business that actively provides order delivery services, faces real challenges in managing its driver commissions. The process that has been running manually — from recording deliveries to calculating commissions — is not only time-consuming but also opens up opportunities for errors that affect drivers' trust in restaurant management (Gustina & Leidiyana, 2020). In fact, commission is a form of financial compensation given to employees based on the performance achieved, and a well-structured commission system has been proven to significantly boost work motivation (Suwandi et al., 2025).

Several studies have demonstrated the advantages of web-based systems for managing financial compensation. A system built using Laravel and MySQL is able to automatically calculate financial components, is equipped with employee data management and real-time

report generation, and has been shown to meet user needs through Black Box Testing (Popi pepmalisa & Abrar Hadi, 2025). Previous research has also shown that a web-based incentive calculation system can automate a process that was previously manual while improving the accuracy of employee data (Rachmanda & Azam, 2025). However, these studies have not yet addressed the specific needs of the restaurant industry, particularly a commission system based on delivery performance. On the other hand, Laravel Filament has proven effective as an administration panel that is easy to access and manage, making it a suitable choice for integrating the entire driver-commission management process into a single platform (Putri et al., 2026).

Using Laravel Filament is more efficient than plain Laravel for developing an administration panel because it provides ready-to-use components such as CRUD, tables, and forms. This speeds up the development process, produces a more standardized code structure, and makes system maintenance easier. Meanwhile, Laravel provides an MVC architecture and modern features for structured web application development, while Laravel Filament is a package built on top of Laravel designed to build responsive administration panels more quickly and declaratively. The use of Laravel with Filament as a Content Management System (CMS) supports efficient content management that is easily accessible through the admin panel, resulting in an information system that is easy to manage, responsive, and well-structured. This system's development process uses the Iterative Waterfall method, an approach that combines the regularity of the Waterfall model with the flexibility of repeating each stage as needed (Stinjak & Masya, 2021).

This method was chosen because the specifications of the driver commission system were relatively clear from the outset, yet still required evaluation and adjustment throughout the development process. System functionality was validated using Black Box Testing to ensure that all features operated as required. The developed system is expected to automate delivery logging, commission calculation, and driver performance reporting digitally, making the restaurant's operations more reliable and measurable.

METHOD

This research uses the Iterative Waterfall method as the system development framework. This approach combines the regularity of the Waterfall workflow with the flexibility of iteration at each stage, allowing the system to be continuously refined based on the evaluation results obtained during the development process.

1. System Requirements Analysis

The first stage is requirements analysis, the process of systematically identifying system requirements through several data-collection techniques (Priyoatmoko et al., 2023). The techniques used in this research include)

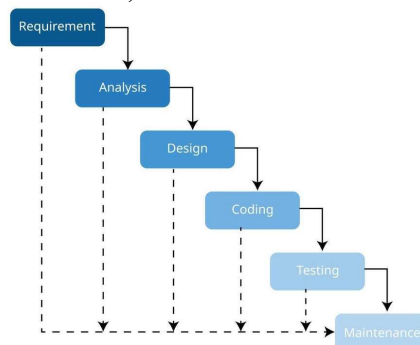


Figure 1. Iterative Waterfall Method

- a) Observation was carried out directly at Mentari Restaurant and revealed inconsistencies in delivery data due to the absence of a centralized recording system, which made performance monitoring difficult.

- b) Interviews were conducted because delivery and commission records were still kept manually, which created a risk of calculation errors, a lack of data transparency, and delays in commission payments.
- c) Document analysis was carried out to examine delivery records, driver data, and transaction data in order to understand data requirements and support system design in accordance with field conditions.

Based on the results of this data collection, the following list of system requirements was obtained.

Table 1. Functional Requirements of the System

Code	Functional Requirement
KF-01	The system provides login and logout features for admins and managers using email and password.
KF-02	The system can manage driver data (add, edit, delete, and view details).
KF-03	The system can record order-delivery data performed by each driver.
KF-04	The system automatically calculates driver commissions based on the number of deliveries and predefined parameters.
KF-05	The system displays delivery and commission history per driver over a specific period.
KF-06	The system provides filtering and search features for delivery data by driver, date, or period.
KF-07	The system generates periodic driver performance reports (daily, weekly, monthly).
KF-08	The system provides a summary dashboard displaying total deliveries, total commissions, and driver performance.
KF-09	Admins can configure commission calculation parameters (rate per delivery, bonuses, and deductions).
KF-10	The system provides a commission-report export feature in a printable or downloadable format.

Table 2. Non-functional Requirements of the System

Kode	Aspek	Kebutuhan Nonfungsional
KNF-01	Performa	Sistem mampu memproses dan menampilkan data pengantaran serta perhitungan komisi secara real-time tanpa penundaan yang signifikan.
KNF-02	Responsivitas	Antarmuka panel admin Laravel Filament dapat diakses dan tampil dengan baik pada berbagai ukuran perangkat (desktop, tablet, dan mobile).
KNF-03	Keamanan	Akses sistem dilindungi autentikasi berbasis Laravel dengan manajemen sesi yang aman, sehingga hanya pengguna terotorisasi yang dapat mengakses data komisi.
KNF-04	Kemudahan Penggunaan	Antarmuka panel admin dirancang intuitif menggunakan Laravel Filament sehingga pengelola restoran dapat mengoperasikan sistem tanpa pelatihan teknis khusus.
KNF-05	Ketersediaan	Sistem dapat diakses kapan saja melalui jaringan internet selama server aktif beroperasi.
KNF-06	Akurasi	Perhitungan komisi dilakukan secara otomatis oleh sistem untuk meminimalkan risiko kesalahan yang terjadi pada proses manual.
KNF-07	Pemeliharaan	Sistem dibangun menggunakan arsitektur MVC Laravel yang terstruktur sehingga memudahkan proses pemeliharaan dan pengembangan fitur di masa mendatang.

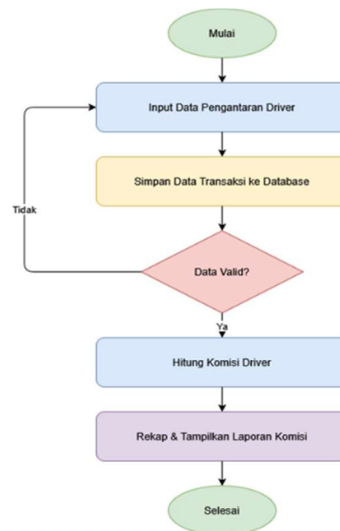


Figure 2. Business Process

2. System Design

The second stage is system design, the stage of designing the structure of the system to be built. At this stage, the website interface design, system flow, and database structure to be used were created. In addition, the system model was also designed using diagrams such as use case diagrams, activity diagrams, and class diagrams to describe the interaction between users and the system (Pristo et al., 2021).

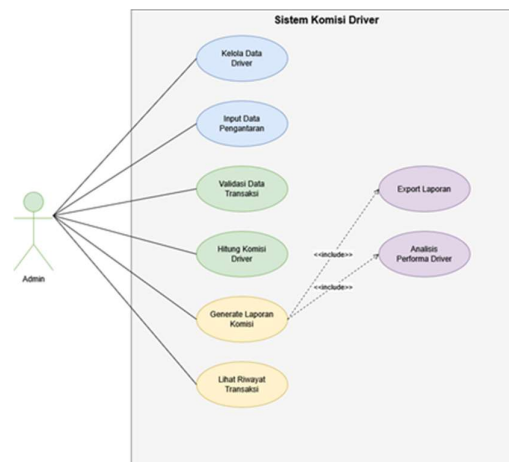


Figure 3. Use Case Diagram

- a. Use Case Diagram — The system involves two main actors, the Admin and the Driver. The Admin has full access to manage driver data, deliveries, transactions, commissions, reports, and transaction history. Drivers can view their delivery history and a recap of the commissions they have received. In addition, the system provides supporting features such as report export and driver performance analysis, so that it not only supports administrative needs but also improves information transparency for drivers.

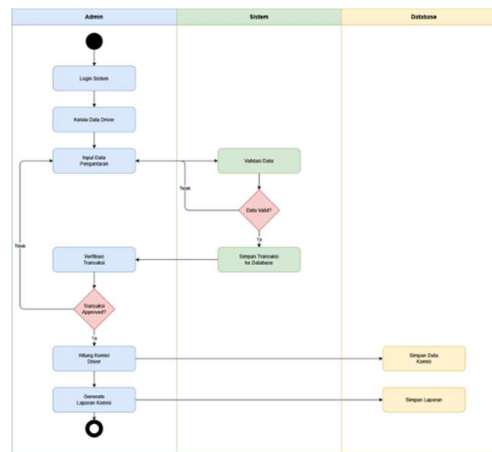


Figure 4. Activity Diagram

- b. Activity Diagram — The activity diagram describes the workflow between the Admin, the System, and the Database. The process begins with admin login, driver data management, and delivery data input. Each piece of data goes through automatic validation by the system and admin confirmation before the commission is calculated and the data is stored in the database. The process ends with the generation and storage of the commission report. This validation mechanism ensures that only valid data is processed further.

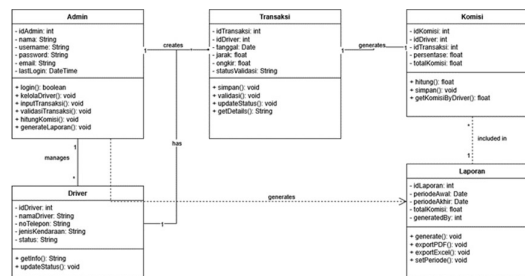


Figure 5. Class Diagram

- c. Class Diagram — The system's class diagram consists of five main classes: Admin, Driver, Transaction, Commission, and Report. The Admin manages driver and transaction data, each transaction generates one commission record, and several commission records are summarized into one periodic report. The relationships between these classes describe Mentari Restaurant's business flow, from recording deliveries to compiling commission reports.

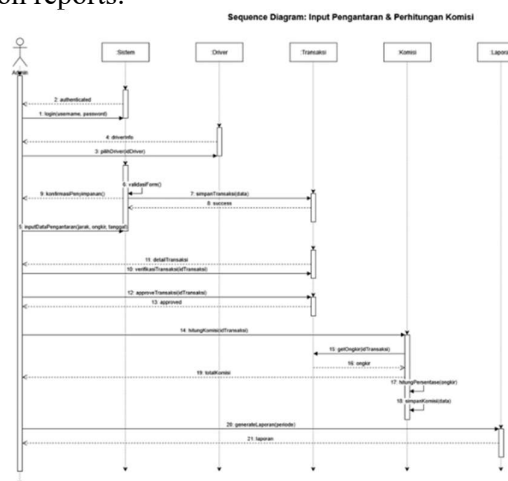


Figure 6. Sequence Diagram

- d. Sequence Diagram — The sequence diagram describes the interaction between the Admin, Driver, Transaction, Commission, and Report in the driver commission

management process. The flow begins with the admin entering and validating delivery data, after which the system automatically calculates and stores the commission data, and finally generates a commission report that can be accessed as needed. This diagram represents the sequence of business processes at Mentari Restaurant, from recording deliveries to producing commission reports.

3. System Implementation

The third stage is system implementation, the process of translating the design results into program code so as to produce a working system. In this research, implementation was carried out using the Laravel framework as the main foundation for web application development, with Laravel Filament as the administration panel for managing system data, and MySQL as the database for storing all operational data — from driver data and delivery recaps to commission history. At this stage, component integration was also carried out, namely connecting the Eloquent ORM-based commission calculation logic with the Filament panel interface, so that every recorded delivery automatically triggers a commission calculation according to the parameters set by the manager.

4. System Testing

The fourth stage is system testing, carried out to ensure that the system that has been built runs in accordance with the research's requirements and objectives. In this research, the testing method used is Black Box Testing, a software testing technique that focuses on the correspondence between the input given and the output produced, without needing to understand the internal program code structure (Uminingsih et al., 2022). Testing was carried out on several main system functions, namely: (a) admin authentication, to ensure that the login and logout processes work correctly; (b) driver data management, covering the add, edit, delete, and view-detail features; (c) delivery data recording, to ensure that every delivery transaction is stored accurately; (d) automatic commission calculation, to verify that the commission value generated by the system matches the configured parameters; and (e) driver performance reporting, to ensure that the generated reports reflect correct data for the selected period.

5. System Maintenance

The final stage is system maintenance, the process of repairing, updating, and developing the system after it has been completed and put into operational use at Mentari Restaurant. According to Br. Sihaloho et al. (2023), maintenance is the process of modifying software after the system is in use, in order to correct errors, improve performance, or adapt the system to new requirements (Br. Sihaloho et al., 2023). Maintenance includes corrective maintenance (bug fixes), adaptive maintenance (adjustment to changing requirements), perfective maintenance (quality and performance improvement), and preventive maintenance (prevention of potential future problems).

RESULTS AND ANALYSIS

This section presents the implementation results of the Restaurant Driver Commission System at Mentari Restaurant, developed using Laravel and Laravel Filament, along with an analysis of requirement fulfillment, Black Box Testing results, and a comparative discussion with previous research.

a. Implementation Results

The Restaurant Driver Commission System at Mentari Restaurant was successfully implemented using Laravel, Laravel Filament, and MySQL. The system provides admin authentication, driver data management, delivery recording, automatic commission calculation, and driver performance reporting for a specific period. The dashboard page (Figure 7) is the main display after the admin logs in, presenting a real-time summary of operational data including total deliveries for the day, total commissions calculated, the number of active drivers, and a driver performance trend chart. This feature fulfills requirement KF-08 while also addressing the observation

finding that the manager previously had no centralized means of monitoring driver performance.

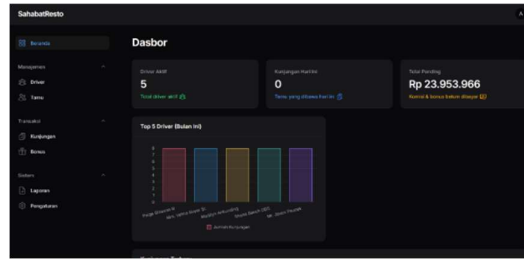


Figure 7. Dashboard Page Display

The driver bonus management page (Figure 8) displays the commission details calculated for each driver, with calculation parameters such as the rate per delivery and additional bonuses that admins can configure through a form. This feature fulfills requirement KF-09 and forms the core of the system, as it automates a process that was previously performed manually and prone to calculation errors.

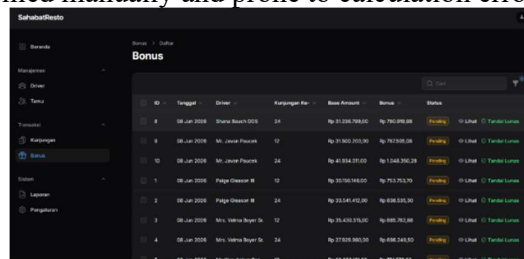


Figure 8. Driver Bonus Management Page Display

In addition to these two main pages, the system is also equipped with a driver management page (CRUD for driver data, fulfilling KF-02), a delivery/visit management page (delivery records that automatically trigger commission calculation, fulfilling KF-03 and KF-04), and a commission report page (a commission recap that can be filtered and exported, fulfilling KF-07 and KF-10). These three pages replace the manual recording process previously carried out through physical notes or text messages, while also addressing the interview finding that delayed commission payments were caused by a lengthy and unstructured recap process.

b. Conformity of the Implementation with System Requirements

The following table presents the verification results for the fulfillment of all functional requirements defined during the analysis stage.

Table 3. Verification of Functional Requirement Fulfillment

Code	Functional Requirement	Status	Description
KF-01	Admin login and logout	✓ Fulfilled	Laravel authentication with active session management.
KF-02	Manage driver data (CRUD)	✓ Fulfilled	Add, edit, and delete driver features are available.
KF-03	Delivery data recording	✓ Fulfilled	Every delivery is recorded with complete details.
KF-04	Automatic commission calculation	✓ Fulfilled	Commission is calculated automatically when delivery data is saved.
KF-05	Delivery and commission history per driver	✓ Fulfilled	Data can be filtered by driver and period.

Code	Functional Requirement	Status	Description
KF-06	Filter and search delivery data	✓ Fulfilled	Filtering by driver, date, and period is available.
KF-07	Driver performance report	✓ Fulfilled	Daily, weekly, and monthly reports can be generated.
KF-08	Operational summary dashboard	✓ Fulfilled	Total deliveries, commissions, and driver performance are displayed.
KF-09	Commission parameter configuration	✓ Fulfilled	Rates and bonuses can be configured through the admin panel.
KF-10	Commission report export	✓ Fulfilled	Reports can be downloaded for payment purposes.

c. Black Box Testing Results

System testing was carried out using Black Box Testing to ensure that each feature produces output corresponding to the given input, without testing the internal code structure. Testing covered positive scenarios (valid input) and negative scenarios (invalid input) to ensure the system could handle both conditions correctly.

Table 4. Black Box Testing Results

No.	Test Scenario	Input	Expected Output	Actual Output	Conclusion
1	Admin login with valid data	Correct email and password	Admin is logged into the dashboard with an active session	Admin successfully logs in and is redirected to the dashboard	✓ Valid
2	Admin login with incorrect password	Correct email, incorrect password	An error message appears and access is denied	Error message is displayed, login is rejected	✓ Valid
5	Add driver data (valid input)	Admin fills in the complete driver form	Driver data is saved and displayed in the list	Data is saved and appears in the driver table	✓ Valid
9	Add delivery data (valid input)	Admin fills in the delivery form with complete data	Data is saved and commission is calculated automatically	Data is saved and the commission value appears immediately	✓ Valid
13	Filter report by driver	Admin selects a specific driver name in the report filter	The report only displays data for the selected driver	Data is filtered according to the selected driver	✓ Valid
16	Admin panel responsiveness	Admin accesses the panel via a mobile device	The panel display adjusts to the screen size	Layout is responsive across various device sizes	✓ Valid

Based on the Black Box Testing results in Table 4, all functionality of the Restaurant Driver Commission System at Mentari Restaurant operated as expected. Of the 16 test scenarios carried out — comprising 11 positive scenarios (valid input) and 5 negative scenarios (invalid input) — all showed actual outputs matching the expected outputs, giving the system a success rate of 100%.

d. Discussion

This research uses Laravel Filament as the administration panel, as an alternative to a manually built admin-panel approach. The application of Laravel Filament has proven to deliver satisfactory results in building a system with an easily accessible and manageable administration panel, as confirmed by Hidayat et al. (2025) in their development of a Laravel Filament-based information module, which demonstrated the effectiveness of a declarative approach in building complex admin interfaces. Laravel was chosen as the development foundation because its structured MVC architecture clearly separates business logic, data management, and the interface. The use of Laravel with Filament as a CMS supports efficient content management that is easily accessible through the admin panel, resulting in an information system that is easy to manage, responsive, and well-structured (Hilmi et al., 2025). Laravel's Eloquent ORM also simplifies the management of relationships between entities — such as the relationships between drivers, delivery data, and commissions — so that data integrity can be maintained consistently.

Compared with previous research that developed a web-based payroll system using Laravel and MySQL without an integrated admin panel — where the system could automatically calculate financial components but its data management still required additional technical configuration (Attasiky & Utami, 2024) — this research offers better manageability through a Filament panel that restaurant managers can operate directly without technical expertise. The automatic commission calculation system built in this study also addresses a problem similar to that found in manual compensation systems, namely the risk of calculation errors due to unstructured data recaps and payment delays that affect employee trust, as reported by Syaifudin et al. (2024) in their development of a web-based payroll system using Laravel.

CONCLUSIONS

The Restaurant Driver Commission System at Mentari Restaurant was successfully developed using Laravel and Laravel Filament with the Iterative Waterfall method. The system includes admin authentication, driver management, delivery recording, automatic commission calculation, commission configuration, and driver performance reporting. The Black Box Testing results across 16 scenarios showed a 100% success rate, with all functional requirements successfully implemented and verified. This system is able to address the problems of manual calculation, a lack of transparency in commission data, and payment delays. However, this research still has limitations, namely that usability testing using the System Usability Scale (SUS) has not been conducted, and automatic notification features and integration with a Point of Sale (POS) system are not yet available. Future research is recommended to add these features and conduct a quantitative usability evaluation.

REFERENCES

- Attasiky, I. A. M., & Utami, L. L. (2024). Perancangan Sistem Informasi Penggajian Karyawan Berbasis Web Menggunakan Laravel Pada Bms Mart Grosir Herbal Kota Tasikmalaya. *Jurnal Tika*, 9(1). <https://doi.org/10.51179/tika.v9i1.2550>
- Br. Sihaloho, Y. N., Fauzi, A., & Safitri, S. T. (2023). IMPLEMENTATION OF DESIGN THINKING APPROACH METHODS IN CURVING SELL MAINTENANCE WEBSITES (CASE STUDY: KAI DAOP 5 PURWOKERTO). *Jurnal Sistem Informasi Dan Ilmu Komputer Prima(JUSIKOM PRIMA)*, 6(2). <https://doi.org/10.34012/jurnalsisteminformasidanilmukomputer.v6i2.3401>

- Gustina, R., & Leidiyana, H. (2020). SISTEM INFORMASI PENGGAJIAN KARYAWAN BERBASIS WEB MENGGUNAKAN FRAMEWORK LARAVEL. *JSiI (Jurnal Sistem Informasi)*, 7(1). <https://doi.org/10.30656/jsii.v7i1.1726>
- Hidayat, F. S., Yusuf, A., & Yasin, M. (2025). Pengembangan Modul Informasi pada Website Jatim Parkir Center Menggunakan Laravel Filament. *Jurnal Komputer Teknologi Informasi Sistem Informasi (JUKTISI)*, 4(3). <https://doi.org/10.62712/juktisi.v4i3.704>
- Hilmi, H. ilhamsyah, M. Rafli Supiyan Ananda, & M. Rifki Firmansyah. (2025). Perancangan Sistem Informasi Company Profile Berbasis Website Pada PT. Bagus Jaya Sejahtera. *JURNAL MULTIDISIPLIN ILMU AKADEMIK*, 2(4). <https://doi.org/10.61722/jmia.v2i4.5311>
- Indah Melyani, R., Rosita, R., & Aji, S. (2023). Pengembangan Sistem Informasi Penggajian Berbasis Web Menggunakan Framework Laravel dengan Metode Agile Software Development. *Jurnal Sistem Informasi Akuntansi (JASIKA)*, 3(1). <https://doi.org/10.31294/jasika.v3i01.2195>
- Popi pepmalisa, & Abrar Hadi. (2025). Perancangan Sistem Informasi Penggajian Karyawan Berbasis Web pada PT Cakrawala Telekomunikasi Indonesia dengan Laravel dan MySQL. *JEKIN - Jurnal Teknik Informatika*, 5(2). <https://doi.org/10.58794/jekin.v5i2.1480>
- Pristo, H., Harian Sitorus, R., asyari, T., Syafira Fatoni, S., & Rizky Lazuardi, M. (2021). Web-Based Attendance Information System Design At Uswatun Hasanah SMP Foundation. *JURNAL ABDIMAS GORONTALO*, 4(2).
- Priyoatmoko, W., Kapti, K., Waluyo, S., & Handayani, R. D. (2023). ANALISA DAN PERANCANGAN SISTEM INFORMASI PENGELOLAAN SURAT PADA SMP NEGERI 1 KALORAN BERBASIS WEB. *TRANSFORMASI*, 19(2). <https://doi.org/10.56357/jt.v19i2.389>
- Putri, D. W., Harahap, R. R., & Mayasari, N. (2026). Pengembangan Profil Perusahaan Berbasis Website Menggunakan Framework Laravel pada PT. Konsultan Media Digital. *Jurnal Komputer Teknologi Informasi Sistem Komputer (JUKTISI)*, 5(1), 44–54. <https://doi.org/10.62712/juktisi.v5i1.892>
- Rachmanda, I. S., & Azam, M. N. Al. (2025). Sistem Perhitungan Insentif Pegawai Berbasis Web Kantor Kementerian Agama Kota Surabaya. *KERNEL: Jurnal Riset Inovasi Bidang Informatika Dan Pendidikan Informatika*, 6(2). <https://doi.org/10.31284/j.kernel.2025.v6i2.8020>
- Stinjak, M. L., & Masya, F. (2021). PERANCANGAN DAN IMPLEMENTASI SISTEM INFORMASI INVENTORY BERBASIS WEBSITE MENGGUNAKAN ITERATIVE WATERFALL. *Rabit : Jurnal Teknologi Dan Sistem Informasi Univrab*, 6(2). <https://doi.org/10.36341/rabit.v6i2.1687>
- Suwandi, S., Hatta, M., Turini, T., Akbari, S., & Yanti, L. A. (2025). Pengembangan Aplikasi Insentif dan Komisi Salesman sebagai Strategi Peningkatan Kinerja SDM Marketing. *JDMIS: Journal of Data Mining and Information Systems*, 3(1). <https://doi.org/10.54259/jdmis.v3i1.4066>
- Syaifudin, A., Risqiati, R., & Sugianti, D. (2024). PERANCANGAN SISTEM INFORMASI PENGGAJIAN PADA PT. ABC. *IC-Tech*, 19(2). <https://doi.org/10.47775/icttech.v19i2.308>
- Uminingsih, Nur Ichsanudin, M., Yusuf, M., & Suraya, S. (2022). PENGUJIAN FUNGSIONAL PERANGKAT LUNAK SISTEM INFORMASI PERPUSTAKAAN DENGAN METODE BLACK BOX TESTING BAGI PEMULA. *STORAGE: Jurnal Ilmiah Teknik Dan Ilmu Komputer*, 1(2). <https://doi.org/10.55123/storage.v1i2.270>