

# Web-based Cashier Information System for Car and Motorcycle Washing Company

Abednego Wijaya Adiwibowo<sup>1</sup>, Agus Cahyo Nugroho<sup>2</sup>, T. Bernadinus Harnadi<sup>3</sup>

<sup>1,2,3</sup> Information Systems Department, Soegijapranata Catholic University

<sup>1,2,3</sup> Jl. Pawiyatan Luhur Sel. IV No. 1, Bendan Duwur, Semarang City, Central Java 50234

<sup>1</sup> 20n10014@student.unika.ac.id

<sup>2</sup> agus.nugroho@unika.ac.id

<sup>3</sup> bharnadi@unika.ac.id

**Abstract**—The motorbike and car wash company is one of the SMEs in the service sector. The company needs recording their transactions better. Previously, the company used a recording system that is less efficient in manually written on paper, so that they have difficulty in calculating revenue and transaction amounts. Their system also has a risk of losing transaction records and less secure. Therefore, this system was created using the RAD (Rapid Application Development) method. This system was built using Laravel and Filament to further shorten production time. The research results a cashier information system. With this system, the company can record the necessary data and transactions. The transactions recorded in the system are automatically stored in the database. This system was tested by company owner and their admin using interview model. The interview results stated that the admin and owner felt a positive impact after implementing this cashier information system. Admins feel safety when record customer data including transactions because the data is stored safely in the database. The company owner can also analyze revenue through the number of transactions so that he can determine a more accurate business strategy.

**Keywords:** Cashier Information System, RAD, PHP, Laravel, Filament

## I. INTRODUCTION

This research was conducted to provide solutions for motorcycle and car wash companies that still record their transactions by writing manually using paper. This method is less effective and less secure because of the risk of being lost or tucked away and it is difficult for company owners to analyze the

income earned from the amount of income so that it can harm the company. Therefore, an information system is needed that can manage data involving hardware, software, and brain ware, so that companies can make the right decisions [1]. This information system can monitor the development of customer data and the number of transactions [2], and manage these data [3]. API is also needed as an interface that connects between the server and the application [4], one of the roles of the API in this application is to display the location of the company using Google Maps. This cashier information system was created so that the admin can record customer data and transactions more neatly, integrated, and secure [5]. In addition, this system can display recorded data such as customer data and transaction data accurately [6]. Transactions can be made using cash, bank transfers, QRIS, and Gopay. QRIS (Quick Response Code Indonesian Standard) is a national payment method and standard in Indonesia [7].

QRIS is included in the type of payment because it is flexible and available in many fintech services [8]. With this cashier information system, it is hoped that company owners can more easily review transaction history and process customer data to plan and decide the right strategy for the development of their company.

RAD (Rapid Application Development) was chosen to be the design method for developing this cashier information system because this methodology can accelerate the development of web applications [9]. The advantages of this method are its speed in system development and more agile to modify the system being built.

This information system was built using the PHP programming language. PHP (Hypertext Preprocessor) is a high-level programming language that is usually used to develop web applications and has the ability to manage and interact with databases [10]. Object-Oriented Programming is a flexible programming technique because it can change the code independently, making it easier for developers to develop, maintain, and optimize the system. One of the PHP frameworks, Laravel, is also used in this web application because there are many features in the library that make it easier for developers to develop the system [11]. There is also MVC (Model, View, Controller) which makes the code neat and makes system development more efficient [12]. Filament is also used to build this information system because there are admin panel packages that greatly facilitate and speed up system development. Filament is a collection of components to support the development of Laravel projects, which includes several components from Tailwind CSS, Alpine.js, Laravel, and Livewire [13]. These components need to be adjusted to what the system needs.

## II. METHOD

The methodology used in this research is RAD (Rapid Application Development). RAD is a system development flow design that prioritizes short production time, so that web applications can be built quickly [14]. RAD has 4 stages in system development, including: planning, system design, development, and implementation. RAD performance in making this system is better and more suitable than other methodologies because this project prioritizes making cashier information systems in a short time and is flexible to repair and maintain.

## III. DESIGN AND IMPLEMENTATION

Design and Results contains the results and design diagrams related to the system being built.

### A. Design

Flowchart is a workflow design of a system so that it can be understood easily. Figure 3.1

shows the flowchart of motorcycle or car wash transactions.



Figure 3.1 Transaction Flowchart Diagram

The transaction process in Figure 3.1 begins with the customer coming to the cashier to use the motorcycle/car washing service. If the customer is not yet a member, the cashier records the customer as a member so that they can use the motorcycle/car wash service. Then when the customer is registered, the customer orders the type of service. Then, the cashier records the selected order and shows QRIS, Gopay, or account number. Then, the customer can pay for the order and be checked by the cashier whether it has been proven to pay or not. If it has been proven to pay, then the transaction process is complete.

Figure 3.2 shows the flowchart of the admin in operating the system.

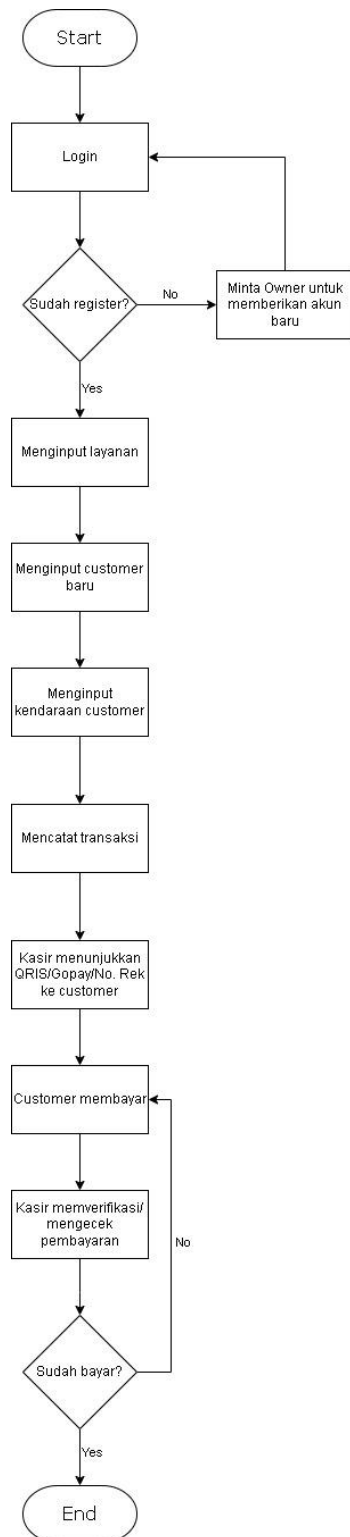


Figure 3.2 Admin Flowchart Diagram

The Admin Flowchart in Figure 3.2 shows the Admin process in performing its role through the Admin panel system. The process begins with the admin logging in, if it has not been registered or given an account by the Owner, then the admin must ask the Owner to be given access through the account given by the

Owner. Admin can add new customers, new services, new vehicles, and new transactions. In addition, Admin is responsible for checking payments made by customers.

Figure 3.3 shows the flowchart of the owner in operating the system.



Figure 3.3 Owner Flowchart Diagram

The Owner Flowchart in Figure 3.3 shows the Owner's process of monitoring and checking important parts, one of which is the number of transactions. The process starts with logging in, if it has not been registered or given an account by the Developer, then the Owner must ask the Developer to be given access through the account given by the Developer. Owner can add services through their access rights to access the Admin system. In the Owner system, there are 3 main menus, namely: Dashboard, Customers, and Users. Dashboard displays the number of users/accounts, customers, services,

transactions, vehicles, and customer rankings from the most to the least frequent use of motorcycle/car washing services.

Figure 3.4 shows the Admin Usecase. The Admin Usecase consists of login, dashboard, and section. Section consists of Customer, Services, Transaction, Vehicle. In the login section, Admin can enter their account and password to be detected by the system whether they really have access to the cashier system.

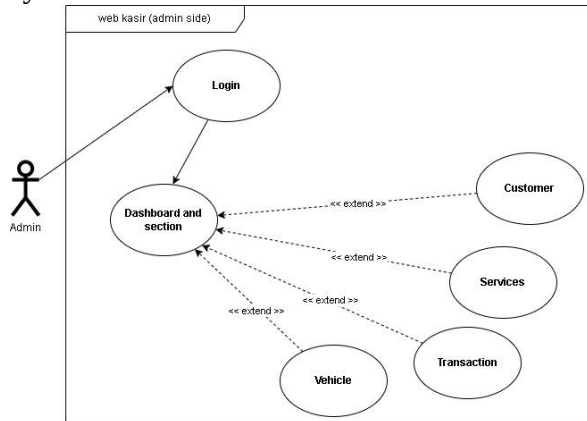


Figure 3.4 Admin Use Case Diagram

After the Admin enters the system, the Admin can add and record the data needed during the transaction. In the Dashboard section, data that can help Admins see data in a concise manner can be displayed, such as Customer, Service, Transaction, and Vehicle.

Figure 3.5 shows the Owner Usecase. The Owner Usecase consists of dashboards and sections. Section consists of revenue resume.

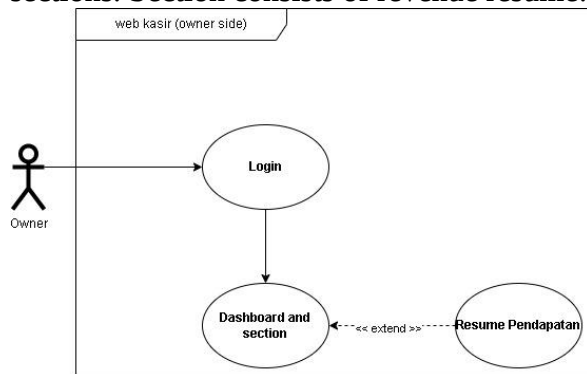


Figure 3.5 Owner Use Case Diagram

The login page in Figure diagram 3.5 serves to verify whether or not the Owner's account is accessing the Owner's system. If correct, then the Owner can log in and see the revenue

summary from the transaction amount view. In addition, the owner can see a summary of the amount of User, Transaction, Vehicle, Service, Customer, and customer ranking.

Activity Diagram in Figure 3.6 shows the activity process that occurs in the system.

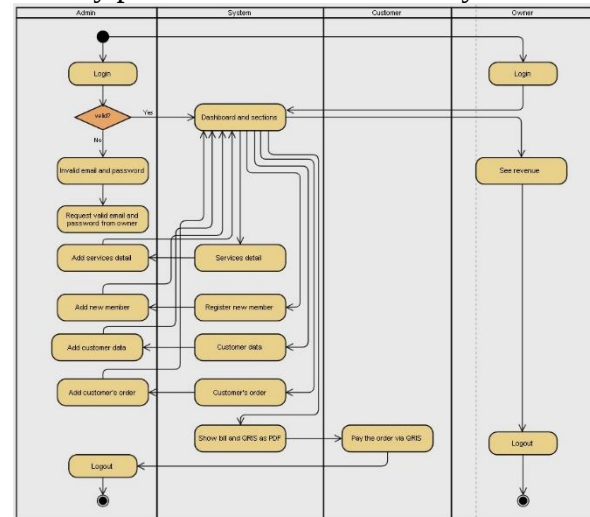


Figure 3.6 Activity Diagram

Activity Diagram in Figure 3.6 is divided into 4 parts, namely Admin, System, Customer, and Owner. On the Admin side there is an important workflow in recording transactions. If the Admin is successfully logged in, then the Admin can add new customers, customer vehicles, services, and transactions. Customers only take on the role of payer through existing payment methods. Then, on the Owner side can find out the summary of income and can manage users / accounts that can access the system.

Entity Relationship Diagram shown in Figure 3.7 contains relationships between relationships in the cashier information system.

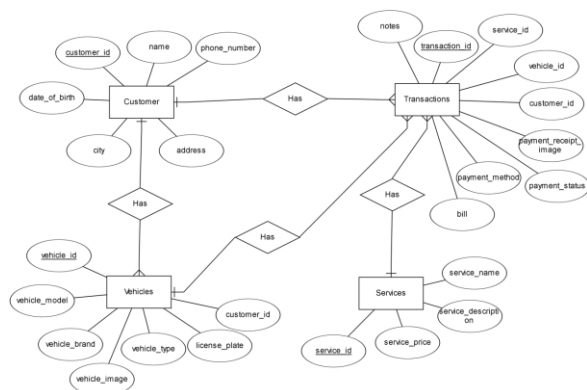


Figure 3.7 Entity Relationship Diagram

The Entity Relationship Diagram in Figure 3.7 shows the Customer entity consists of customer\_id, name, no\_hp, address, city, and date\_birth. Customers can have many vehicles and transactions. The Vehicle entity consists of vehicle\_id, customer\_id, vehicle model, vehicle brand, vehicle image, vehicle type, and license plate\_number. A vehicle can be owned by one customer and one transaction. Service entity consists of service\_id, service\_name, service\_description, and service\_price. 1 service can have many transactions. The Transaction entity consists of transaction\_id, service\_id, vehicle\_id, customer\_id, image\_proof\_of\_payment, payment\_status, type\_of\_payment, amount\_paid, and notes. 1 Transaction is owned by 1 customer, 1 service, and 1 vehicle.

## B. Implementation

### Landing Page

The landing page is shown in Figure 3.8. This page displays a picture of the company's location, an explanation of motorcycle and car washing services, and frequently asked questions.



Figure 3.8 Landing Page

## Admin Panel

The admin dashboard is shown in Figure 3.9. The menu displays the number of transactions and the number of registered customers. The number of transactions and customers is displayed using a chart. The chart can be set in several specific time periods, including: per day, per week, per month, per quarter, per 6 months, and per year.

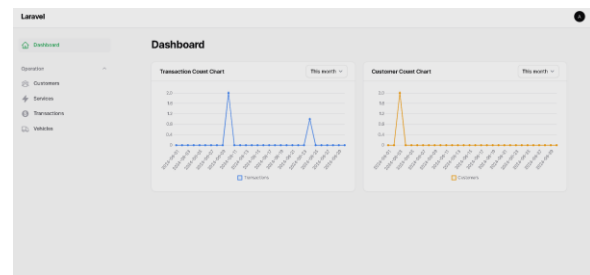


Figure 3.9 Dashboard Admin

The Customers menu is shown in Figure 3.10. The menu displays customer data details. Admin can add new customers.

| Name              | Phone number  | Address          | City     | Date of birth | Created at        |
|-------------------|---------------|------------------|----------|---------------|-------------------|
| Abderrigo Wilaya  | 08132339023   | Jl. Maju Jaya    | Semarang | 09-May-2024   | 22-May-2024-00:50 |
| Vania Polika      | 085432123939  | Jl. Sura Merdeka | Semarang | 14-May-2024   | 22-May-2024-00:52 |
| Kivir Hartono     | 0873547656966 | Jl. Pahlawan     | Semarang | 07-Jul-1993   | 03-Jun-2024-08:42 |
| Jessica Citravana | 085756377853  | Jl. Insan Rongit | Semarang | 29-May-2024   | 03-Jun-2024-07:08 |

Figure 3.10 Menu Customers

The Services menu is shown in Figure 3.11. The menu displays details of available service data. Admin can add services.

| Service name      | Service price | Service description                  | Created at        |
|-------------------|---------------|--------------------------------------|-------------------|
| cuci motor tipe 1 | IDR 12,000.00 | Bobok, Merik Kocil                   | 22-May-2024-09:29 |
| cuci motor tipe 2 | IDR 15,000.00 | Merik Sedarig                        | 22-May-2024-09:35 |
| cuci motor tipe 3 | IDR 20,000.00 | Sport, Merik                         | 22-May-2024-09:38 |
| cuci mobil tipe 1 | IDR 40,000.00 | LOGO, Standar Wagne, Sedan, Pickup   | 22-May-2024-09:38 |
| cuci mobil tipe 2 | IDR 45,000.00 | Hardback, Offroad, SUV, Double Cabin | 22-May-2024-09:30 |
| cuci mobil tipe 3 | IDR 50,000.00 | MPV, Sport, Elektrik, Convertible    | 22-May-2024-09:32 |

Figure 3.11 Menu Services

The Transactions menu is shown in Figure 3.12. The menu displays transaction data details. Admin can add transactions.

Transactions > List

**Transactions** Add transaction

Search

| Service                                    | Customer name  | Vehicle   | Payment method | Payment status | Notes         | Bill          | Payment receipt image |
|--------------------------------------------|----------------|-----------|----------------|----------------|---------------|---------------|-----------------------|
| <input type="checkbox"/> cuci motor tipe 1 | Abdrego Wijaya | H 0001 SD | Cash           |                |               | IDR 50,000.00 |                       |
| <input type="checkbox"/> cuci mobil tipe 1 | Abdrego Wijaya | H 0001 SD | Cash           |                |               | IDR 50,000.00 |                       |
| <input type="checkbox"/> cuci motor tipe 3 | Abdrego Wijaya | H 0001 SD | Cash           |                |               | IDR 40,000.00 |                       |
| <input type="checkbox"/> cuci mobil tipe 1 | Abdrego Wijaya | H 0001 XD | Transfer       |                | sudah dibayar | IDR 50,000.00 |                       |
| <input type="checkbox"/> cuci mobil tipe 1 | Abdrego Wijaya | H 0001 SD | Cash           |                |               | IDR 50,000.00 |                       |
| <input type="checkbox"/> cuci motor tipe 2 | Vania Felicia  | H 0001 OD | Transfer       |                |               | IDR 15,000.00 |                       |

Showing 1 to 6 of 6 results Per page: 10

Figure 3.12 Menu Transactions

The Vehicles menu is shown in Figure 3.13. The menu displays detailed data on registered customer vehicles. Admin can add customer vehicles.

Vehicles > List

**Vehicles** Add vehicle

Search

| Customer name                              | License plate | Vehicle type | Vehicle image | Vehicle brand | Vehicle model  | Created at        |
|--------------------------------------------|---------------|--------------|---------------|---------------|----------------|-------------------|
| <input type="checkbox"/> Abdrego Wijaya    | H 0001 XD     | Mobil        |               | risian        | Sedan          | 22 May 2024-00:51 |
| <input type="checkbox"/> Abdrego Wijaya    | H 0001 SD     | Mobil        |               | honda         | Pickup         | 22 May 2024-00:53 |
| <input type="checkbox"/> Vania Felicia     | H 0001 OD     | Motor        |               | honda         | Matik          | 02 Jun 2024-19:14 |
| <input type="checkbox"/> Kevin Hartono     | H 0001 QX     | Mobil        |               | honda         | Sedan          | 03 Jun 2024-07:02 |
| <input type="checkbox"/> Jessica Oktaviana | H 0001 JK     | Mobil        |               | honda         | Starline Wagon | 03 Jun 2024-10:54 |
| <input type="checkbox"/> Jessica Oktaviana | H 0001 KL     | Motor        |               | yamaha        | Matik          | 05 Jun 2024-03:12 |

Showing 1 to 6 of 6 results Per page: 10

Figure 3.13 Menu Vehicles

### Owner Panel

The owner dashboard is shown in figures 3.14, 3.15, and 3.16. The menu displays the number of Users, Customers, Services, Transactions, and Vehicles in numbers. There are also 3 charts to display the number of Transactions, Customers and Vehicles that can be set the time recorded. In addition, there is a customer ranking to find out the order of customers who most often to the least often use services.

Logout Dashboard Customers Users

**Dashboard**

|                                             |                                        |                                  |
|---------------------------------------------|----------------------------------------|----------------------------------|
| Users<br>3<br>Registered Users              | Customers<br>4<br>Registered Customers | Services<br>6<br>Active Services |
| Transactions<br>8<br>Completed Transactions | Vehicles<br>6<br>Active Vehicles       |                                  |

Figure 3.14 Number Data on Owner's Dashboard

There are also 3 charts to display the number of Transactions, Customers and Vehicles that can be set to record time.

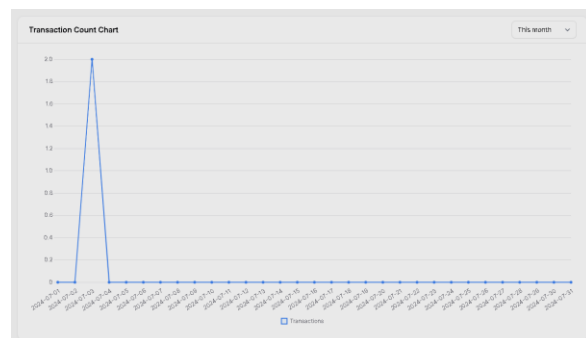


Figure 3.15 Chart Transaction di Dashboard Owner

In addition, there is a customer ranking to determine the order of customers who most often to the least often use the service.

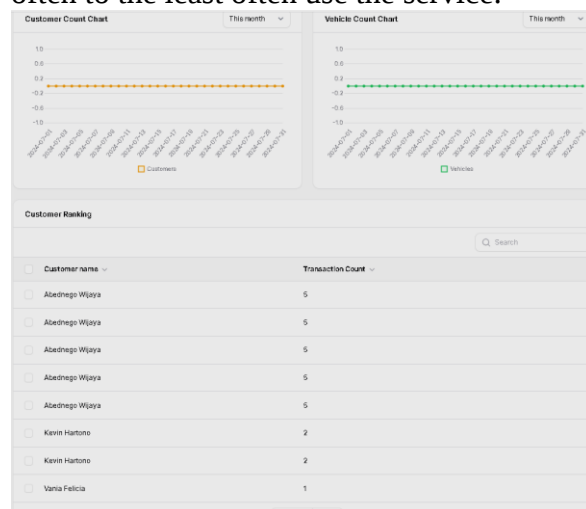


Figure 3.16 Chart Customer dan Vehicle with Customer Ranking di Dashboard Owner

The Customer menu is shown in Figure 3.17. The menu displays registered customer data.

Customers > List

**Customers**

Search

| Name                                       | Phone number   | Address           | Created at        |
|--------------------------------------------|----------------|-------------------|-------------------|
| <input type="checkbox"/> Abdrego Wijaya    | 08123289023    | Jl. Maju Jaya     | 22 May 2024-00:50 |
| <input type="checkbox"/> Vania Felicia     | 085432923929   | Jl. Suara Merdeka | 22 May 2024-00:52 |
| <input type="checkbox"/> Kevin Hartono     | 08735476588966 | Jl. Pahlawan      | 03 Jun 2024-06:42 |
| <input type="checkbox"/> Jessica Oktaviana | 085756377953   | Jl. Inewi Bumi    | 03 Jun 2024-07:03 |

Showing 1 to 4 of 4 results Per page: 10

Figure 3.17 Menu Customers in Panel Owner

The Users menu is shown in Figure 3.18. The menu displays account details that can be used to access the system.

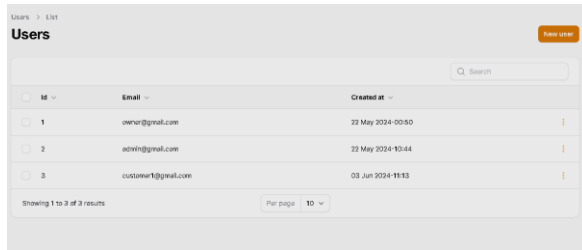


Figure 3.18 Menu Users in Panel Owner

### Technical Testing

Technical testing is done with the Black Box method. The Black Box testing method is a system testing method carried out by the user to find out whether the system is working properly as expected or not, without having to know the details of the code of the system being tried [15].

| No | Features            | Expected Output                                                                                                        | Validity (Y/N) | % Score |
|----|---------------------|------------------------------------------------------------------------------------------------------------------------|----------------|---------|
| 1  | Login               | Admin and Owner can log in to the system                                                                               | Y              | 100     |
| 2  | Dashboard           | Admin can view transactions and customers charts with time filters by day, week, month, 3 months, 6 months, and 1 year | Y              | 100     |
|    |                     | Owner can view details and charts of Users, Customer Services, Transactions, and Vehicles                              | Y              | 100     |
| 3  | Manage customers    | Admin and Owner can manage the Customers menu                                                                          | Y              | 100     |
| 4  | Manage services     | Admin and Owner can manage the Services menu                                                                           | Y              | 100     |
| 5  | Manage transactions | Admin and Owner can manage the Transactions menu                                                                       | Y              | 100     |

Table 3.1 Technical Testing Results

### Business Testing

Interviews were conducted to find out the impact on the company after this cashier information system was implemented. The author also made direct observations to the admin and owner to find out the response to this system. When making observations, the author found out that before using this system the company only used paper media to record every existing transaction. In other words, the process of recording transactions is done manually using paper and pen, so it is considered ineffective and less secure in recording and reviewing income. From the admin's point of view, it can be seen that with this system, the recording becomes more structured and neat. Meanwhile, from the

owner's point of view, it can be seen that with this system, the recording becomes better in bookkeeping or documenting transactions so that the income can be known. The use of this system began to be used gradually replacing the old way of using paper manually. Adaptation is also needed for both admins and owners to be able to operate this system properly and correctly.

## IV. CONCLUSION

Interviews were conducted to find out the impact on the company after this cashier information system was implemented. The author also made direct observations to the admin and owner to find out the response to this system. When making observations, the author found out that before using this system the company only used paper media to record every existing transaction. In other words, the process of recording transactions is done manually using paper and pen, so it is considered ineffective and less secure in recording and reviewing income. From the admin's point of view, it can be seen that with this system, the recording becomes more structured and neater. Meanwhile, from the owner's point of view, it can be seen that with this system, the recording becomes better in bookkeeping or documenting transactions so that the income can be known. The use of this system began to be used gradually replacing the old way of using paper manually. Adaptation is also needed for both admins and owners to be able to operate this system properly and correctly.

## REFERENCES

- [1] E. Triandini, S. Jayanatha, A. Indrawan, G. Werla Putra, and B. Iswara, "Metode Systematic Literature Review untuk Identifikasi Platform dan Metode Pengembangan Sistem Informasi di Indonesia," *Indones. J. Inf. Syst.*, vol. 1, no. 2, p. 63, 2019, doi: 10.24002/ijis.v1i2.1916.
- [2] M. D. Irawan, "Design Of Monitoring Information System Stock Items At Partner Stores In Web-Based Semarang," *J. Bus. Technol.*, vol. 2, no.

- 2, pp. 57–63, 2022, doi: 10.24167/jbt.v2i2.4394.
- [3] Agus Cahyo Nugroho, “Sistem Informasi Manajemen Ekstrakurikuler Berbasis Web Menggunakan Metode Rapid Application Development,” *Teknika*, vol. 10, no. 3, pp. 199–205, 2021, doi: 10.34148/teknika.v10i3.407.
- [4] et al Brad Williams, Justin Tadlock, “What the rest api is,” in *Professional WordPress® Plugin Development*, 2020, pp. 279–313.
- [5] Diodora Yessayabella and Yohanna Adys, “Implementasi Sistem Informasi Manajemen Kasir Berbasis Aplikasi Moka Pos (Point of Sales) Pada Kafe X Tahun 2022,” *Jamanta J. Mhs. Akunt. Unita*, vol. 1, no. 2, pp. 54–71, 2022, doi: 10.36563/jamanta\_unita.v1i2.476.
- [6] V. B. Anwari, F. Faras, and Samsinar, “Implementasi Sistem Informasi Kasir Pada Rakab Mercon Berbasis Web,” *Semin. Nas. Inov. Teknol.*, pp. 1–8, 2020.
- [7] QRIS, “QRIS (Quick Response Code Indonesian Standard), Satu QR Code untuk Semua Payment,” 2023. <https://gris.online/homepage/> (accessed Sep. 19, 2023).
- [8] E. T. Kurniawati, I. Zuhroh, and N. Malik, “Literasi dan Edukasi Pembayaran Non Tunai Melalui Aplikasi QR Code Indonesian Standard (QRIS) Pada Kelompok Milenial,” *Stud. Kasus Inov. Ekon.*, vol. 05, no. 01, pp. 23–30, 2021.
- [9] Jijon Raphita Sagala, “Model Rapid Application Development (Rad) Dalam Pengembangan Sistem Informasi Penjadwalanbelajar Mengajar,” *J. Mantik Penusa*, vol. 2, no. 1, p. 88, 2021.
- [10] M. P. Tatroe Kevin, *Programming PHP: Creating Dynamic Web Pages - Kevin Tatroe, Peter MacIntyre - Google Books*, vol. 2, 2022.
- [11] D. Purnama Sari and R. Wijanarko, “Implementasi Framework Laravel pada Sistem Informasi Penyewaan Kamera (Studi Kasus di Rumah Kamera Semarang),” *J. Inform. dan Rekayasa Perangkat Lunak*, vol. 2, no. 1, p. 32, 2020, doi: 10.36499/jinrpl.v2i1.3190.
- [12] N. C. S. Muljono, D. Gunadi, and A. C. Nugroho, “Rancang Bangun Website Pemesanan Makanan Kedai Twins Menggunakan Laravel PHP Framework,” *Praxis (Bern. 1994).*, vol. 3, no. 1, p. 47, 2020, doi: 10.24167/praxis.v3i1.2818.
- [13] D. Harrin, Z. Fietje, Ryan Scherler, and R. Chandler, “Filament,” 2023. <https://github.com/filamentphp/filament/tree/3.x> (accessed Jul. 17, 2024).
- [14] Y. D. Wijaya, “Penerapan Metode Rapid Application Development (Rad) Dalam Pengembangan Sistem Informasi Data Toko,” *J. SITECH Sist. Inf. dan Teknol.*, vol. 3, no. 2, pp. 95–102, 2021, doi: 10.24176/sitech.v3i2.5141.
- [15] K. Anita, A. D. Wahyudi, and E. R. Susanto, “Aplikasi Lowongan Pekerjaan Berbasis Web Pada Smk Cahaya Kartika,” *J. Teknol. dan Sist. Inf.*, vol. 1, no. 1, pp. 75–80, 2020, doi: 10.33365/jtsi.v1i1.213.