Journal of Technology and Computer (JOTECHCOM)
Vol. 3, No. 2, May 2026, pp. 109~119
ISSN: 3048-0477 (Media Online)

Android-Based Bus Scheduling and Ticket Reservation System Using Round Robin

Muhamad Hafiz[1], Baginda Harahap[2], Dinur Syahputra[3]
[1]Program Studi Teknologi Informasi, Fakultas Teknologi, Universitas Battuta
[2,3]Program Studi Informatika, Fakultas Teknologi, Universitas Battuta
[1]hafizgedoy47@gmail.com, [2]profesionalbaginda@gmail.com, [3]dinsyahui12@gmail.com

Article Info

Article history:

Received March 29, 2026
Revised May 05, 2026
Accepted May 22, 2026

Keywords:

Round Robin algorithm
Bus scheduling
Bus ticket reservation
Android application
Information system

ABSTRACT

Bus departure scheduling that is still carried out manually at PT Eldivo Tunas Arta often causes delays in information delivery and difficulties in managing operational schedules. This condition can affect the quality of service provided to customers and reduce the efficiency of company management. This study aims to design and implement an Android-based bus scheduling and ticket reservation system by applying the Round Robin algorithm. The Round Robin algorithm was chosen because it is capable of distributing departure schedules fairly and evenly among available bus fleets. The system development method used in this study is the System Development Life Cycle (SDLC), which consists of several stages including requirement analysis, system design, implementation, and testing. The results of this study show that the developed system can facilitate customers in accessing departure schedule information, booking bus tickets, and making digital payments more efficiently. In addition, from the management perspective, the system can reduce errors in schedule recording and improve operational efficiency. Therefore, the implementation of the Round Robin algorithm in the bus scheduling and ticket reservation system is proven to be effective in improving service quality and operational management of the company.

Corresponding Author:

Muhammad Hafiz
Universitas Battuta
Email: hafizgedoy47@gmail.com

1. INTRODUCTION

The rapid development of information technology has encouraged various sectors to adopt digital services, including the transportation sector. Digitalization allows service management processes to become faster, more accurate, and more efficient. In bus transportation services, the management of departure schedules and passenger queue systems is an important aspect that influences service quality, punctuality, and customer satisfaction. Ineffective scheduling and lack of accessible information often cause congestion at terminals, long waiting times, and dissatisfaction among passengers.

PT. Eldivo Tunas Arta, a land transportation company that provides regular and charter bus services, still faces several problems related to scheduling and queue management. Currently, information regarding bus departure schedules is not always available in real time, causing uncertainty for passengers. As a result, passengers often experience long waiting times and overcrowding at terminals. In addition, the charter bus reservation system is still conducted conventionally, where customers must visit the company office directly or contact administrative staff via telephone to obtain information about routes, fleet availability, and service prices. This process is considered inefficient and may reduce customer interest in using the service.

Another issue arises from the internal operational system of the company. The scheduling of bus fleets is still carried out manually using traditional bookkeeping methods. Such a system is prone to recording errors, data inconsistencies, and difficulties in managing fleet rotation fairly. Consequently, some buses may remain idle for extended periods while others experience excessive operational workloads. These problems indicate the need for a more structured, automated, and efficient scheduling system.

Several previous studies have explored the application of the Round Robin algorithm in scheduling systems. Research conducted by Wahyu Nur Cholifah (2022) entitled "Sistem Penjadwalan Bus di Terminal Jatijajar Depok Menggunakan Algoritma Round Robin" showed that manual scheduling systems often lead to passenger congestion and inefficiencies; the implementation of the Round Robin algorithm in the study successfully created a more organized scheduling system because each bus fleet received an equal and fair departure time allocation [1]. Similarly, research by Yonalisa S. Pane (2024) titled "Sistem Penjadwalan Kegiatan Terminal di Pelabuhan dengan Menggunakan Metode Round Robin Berbasis Web" demonstrated that the algorithm can effectively distribute terminal activity schedules fairly and efficiently within a web-based management system [2].

Other studies have also highlighted the effectiveness of the Round Robin algorithm in various scheduling contexts. Research by Excalanta Rully (2012) emphasized that the lack of structured scheduling systems in bus transportation services can reduce service efficiency and customer satisfaction [3]. In addition, studies conducted by Tri Dharma Putra (2021) in CPU scheduling [4], Sofiansyah Fadli (2020) in event organizer scheduling systems [5], and Arif Maulana Komaruddin (2019) in web server load balancing [6] demonstrate that the Round Robin method is capable of distributing workloads evenly and efficiently across different systems. These studies confirm that the algorithm is widely applicable for solving scheduling problems that require fairness and balanced resource allocation.

Based on these problems and the potential solutions identified in previous research, this study proposes the implementation of the Round Robin algorithm in a bus scheduling and ticket reservation system for PT. Eldivo Tunas Arta. The proposed system is developed as an Android-based application that allows passengers to access departure schedules, view queue positions, and make reservations online. From the management perspective, the system is expected to improve operational efficiency, optimize fleet rotation, and provide a more structured and transparent scheduling process. Therefore, this research is conducted under the title "Implementation of the Round Robin Algorithm for Bus Scheduling and Ticket Reservation System at Eldivo Based on Android."

2. METHOD

Round Robin (RR) is a process scheduling algorithm that uses a preemptive strategy, where each process is allocated a fixed amount of CPU time called a time quantum. This algorithm is widely used in CPU scheduling along with other methods such as FCFS (First Come First Served), SJF (Shortest Job First), and Priority Scheduling (Komaruddin et al., 2019).

In the Round Robin method, all processes are considered equally important and are given the same time quantum to execute. If a process has not finished when its time quantum expires, the CPU will switch to the next process in the queue. This mechanism ensures fairness because each process receives equal CPU time.

The Round Robin scheduling process can be summarized as follows:

1. Maintain a queue of ready processes according to their arrival order.
2. Execute the process at the front of the queue.
3. If the process finishes before the time quantum expires, it is removed from the queue.
4. If the time quantum expires and the process is not finished, the process is moved to the end of the queue.

The effectiveness of the Round Robin algorithm is influenced by the size of the time quantum. If the time quantum is too large, the scheduling mechanism tends to behave similarly to the First Come First Serve (FCFS) method. Conversely, if the time quantum is too small, the system will perform more frequent context switching, which can increase overhead and reduce overall efficiency (Sofiansyah Fadli & Maulana Ashari, 2020).

Round Robin is a preemptive scheduling algorithm that allocates an equal time slice, known as a time quantum, to each process in the queue. Each process is executed in turn, and when its allocated time expires or the process finishes earlier, the scheduler moves to the next process. This mechanism ensures fairness because every process receives the same opportunity to use the CPU without priority over others.

The Round Robin scheduling algorithm can be implemented through the following steps:

1. Maintain a list of runnable processes according to their order of arrival.
2. Select the process at the front of the queue for execution.
3. If the process completes before the time quantum expires, it is removed from the queue.

4. If the time quantum expires before the process finishes, the running process is moved to the end of the ready queue and the next process at the front of the queue is executed (Wahyu Nur Cholifah, 2022).

The flowchart of the Round Robin method is shown as follows :

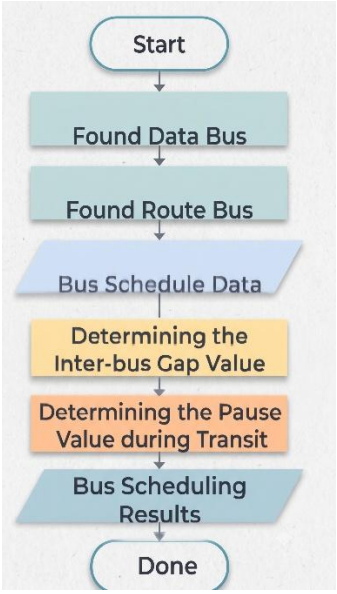


Figure 1. Flowchart Method Round Robbin

2.1. Round Robin Calculation

In the calculation of the Round Robin algorithm for the Android-Based Bus Scheduling and Ticket Reservation System for Eldivo, the following data describes the bus departure scheduling using the Round Robin method.

Table 1. Bus Route Data of PT. Eldivo Tunas Arta

No	Origin	Transit	Destination	Transit Processing	Description	City	Estimated Time from Previous City
1	Medan	Pekanbaru	Jambi	Start	Medan	-	
				Transit	Pekanbaru	1 hour 0 minutes	
				End	Jambi	2 hours 0 minutes	
2	Jambi	Pekanbaru	Medan	Start	Jambi	-	
				Transit	Pekanbaru	10 hours 40 minutes	
				End	Medan	12 hours 50 minutes	
3	Medan	Dumai	Bukit Tinggi	Start	Medan	-	
				Transit	Dumai	1 hour 0 minutes	
				End	Bukit Tinggi	1 hour 0 minutes	
4	Bukit Tinggi	Dumai	Medan	Start	Bukit Tinggi	-	
				Transit	Dumai	5 hours 0 minutes	
				End	Medan	5 hours 0 minutes	
5	Siantar	Bagan Batu	Surabaya	Start	Siantar	-	
				Transit	Bagan Batu	7 hours 10 minutes	
				End	Surabaya	45 hours 0 minutes	

Bus Departure Schedule Based on Route
The following schedule represents the bus departure route for PT. Eldivo Tunas Arta:

Route: Medan – Pekanbaru – Jambi
Departure Time : 18:00
Arrival Time : 06:00

Inter-Departure Interval : 60 Minutes
Transit Waiting Time : 60 Minutes
Time Quantum (Customer Service Duration) : 60 Minutes
Total Working Hours per Day :
10 Hours = 600 Minutes
Based on these parameters, the bus departure data can be described as follows.

Table 2. Departure Time and Completion Time Data

Route Name	Departure Time	Number of Buses	Burst Time (Conpletion Time)
Medan – Pekanbaru – Jambi	14 June 2025 10:25 (WD = 0)	5	5 × 60 Minutes = 300 Minutes

Total Burst Time (Total completion duration of all activities):
300 Minutes
Round Robin Method

Table 3. Round Robin Scheduling

Bus Name	Burst Time	Start Time	Finish Time	Remaining Time
Bus A	300 minutes	17:15	18:15	240 minutes
Bus B	240 minutes	08:00	09:00	180 minutes
Bus C	180 minutes	10:00	11:00	120 minutes
Bus D	120 minutes	12:00	13:00	60 minutes
Bus E	60 minutes	14:00	15:00	0 minutes

Global system design uses the UML modeling language, which consists of Use Case Diagrams and Sequence Diagrams.

2.2. Data Base

The next step is to design the table structure for the database system to be created. The following is a draft of the table structure:

Administrator Structure

The user table is used to store data. Details of this table structure can be seen in Table 4 below:

Table 4. Administrator Table Design

No.	Field Name	Data Type	Null	Key
1	id_administrator	int (11)	No	Primary Key
2	nama	varchar(50)	No	-
3	username	varchar(30)	No	-
4	password	varchar(30)	No	-

Bus Table Structure

The bus table is used to store data. Details of this table structure can be seen in Table 5 below:

Table 5. Bus Table Design

No.	Field Name	Data Type	Null	Key
1	no_plat	varchar	No	Primary Key
2	merk_type	varchar	No	-
3	kapasistas	int	No	-
4	kelas	varchar	No	-
5	fasilitas	text	No	-
6	foto	text	No	-

Booking Details Table Structure

The booking details table is used to store data. A more detailed explanation of this table structure can be seen in Table 6 below:

Table 6. Booking Detail Table Design

No.	Field Name	Data Type	Null	Key
1	id_penumpang	int	No	Primary Key
2	no_booking	varchar	No	Foreign Key
3	nama_penumpang	varchar	No	-
4	kursi	int	No	-
5	status	varchar	No	-

Schedule Table Structure

The schedule table is used to store departure and arrival schedule data. The detailed structure of this table is presented in Table 7 below.

Table 7. Schedule Table Design

No.	Field Name	Data Type	Null	Key
1	id_jadwal	int	No	Primary Key
2	no_plat	varchar	No	-
3	hari_berangkat	int	No	-
4	jam_berangkat	varchar	No	-
5	hari_tiba	int	No	-
6	jam_tiba	varchar	No	-
7	id_tiket	int	No	-
8	id_rute	int	No	-

Confirmation Table Structure

The confirmation table is used to store payment confirmation data. The detailed structure of this table is shown in Table 8.

Table 8. Confirmation Table Design

No.	Field Name	Data Type	Null	Key
1	id_konfirmasi	int	No	Primary Key
2	tanggal	datetime	No	-
3	no_booking	varchar	No	-
4	id_pengguna	int	No	-
5	foto	text	No	-

User Table Structure

The user table is used to store user account information. The detailed structure of this table is presented in Table 9.

Table 9. User Table Design

No.	Field Name	Data Type	Null	Key
1	id_pengguna	int	No	Primary Key
2	nama_pengguna	varchar(50)	No	-
3	handphone	varchar(20)	No	-
4	email	varchar(50)	No	-
5	password	varchar(15)	No	-

City Table Structure

The city table is used to store city data. The detailed structure of this table is presented in Table 10.

Table 10. City Table Design

No.	Field Name	Data Type	Null	Key
1	id_kota	int	No	Primary Key
2	nama	varchar	No	-
3	alamat	text	No	-

Booking Table Structure

The booking table is used to store ticket reservation data. The detailed structure of this table is presented in Table 11.

Table 11. Booking Table Design

No.	Field Name	Data Type	Null	Key
1	no_booking	varchar	No	Primary Key
2	tanggal_booking	datetime	No	-
3	id_pengguna	int(11)	No	-
4	pelanggan	varchar	No	-
5	tanggal	date	No	-
6	id_rute	int	No	-
7	id_jadwal	int	No	-
8	jumlah_seat	int	No	-
9	total_harga	decimal	No	-
10	status	varchar	No	-

Booking Table Structure

The booking table is used to store ticket reservation data. The detailed structure of this table is presented in Table 12.

Table 12. Booking Table Design

No.	Field Name	Data Type	Null	Key
1	id_rute	int	No	Primary Key

2	dari	varchar	No	-
3	tujuan	varchar	No	-

Bus Route Table Structure

The bus route table is used to store the relationship between buses and routes. The detailed structure of this table is presented in Table 13.

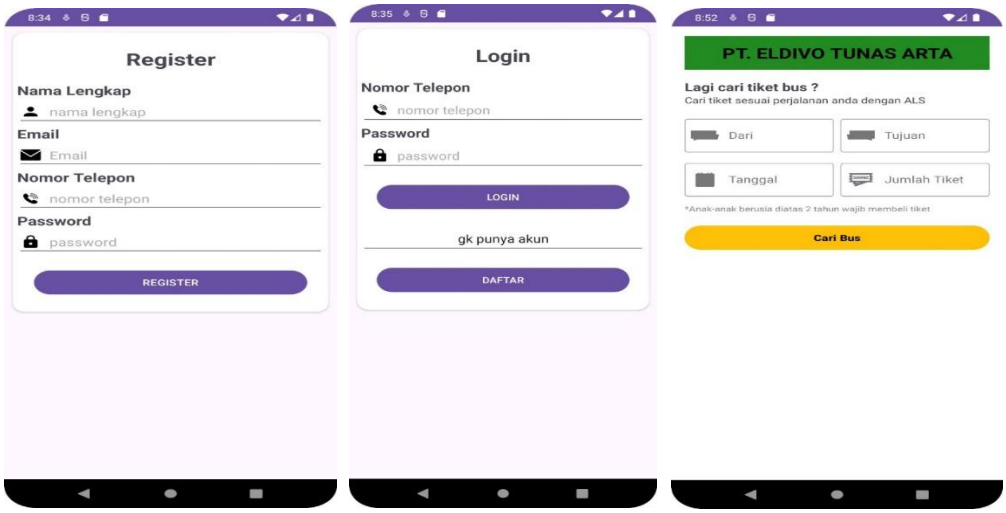
Table 13. Bus Route Table Design

No.	Field Name	Data Type	Null	Key
1	id_rute_bus	int	No	Primary Key
2	no_plat	varchar	No	-
3	id_rute	int	No	-

2.3. Design

Application menu display for customers

1. Registration Menu Display : The Registration Menu Display is the first screen that appears when the program is run. It functions as a form to fill in the customer's personal data.
2. Login Display : The Login Display is the first screen that appears when the program is run. It functions as a form to input the admin username and password.
3. Main Menu Data Form Display : The Main Menu Data Form Display is a screen that shows the main menu data and functions to identify and display the main menu data.
4. Order Data Form Display : This display shows order data and functions to identify and display order information.
5. Schedule Data Display : This display shows schedule data and functions to identify and display schedule information.
6. Seat Selection Display : This display shows seat selection data and functions to identify and display seat selection information.
7. Order Management Display : This display is an order management form that functions to input order management data.
8. Payment Display : This display is a payment form that functions to input payment data.
9. Order Detail Information Display :This display is an order detail information form that functions to input order information data.



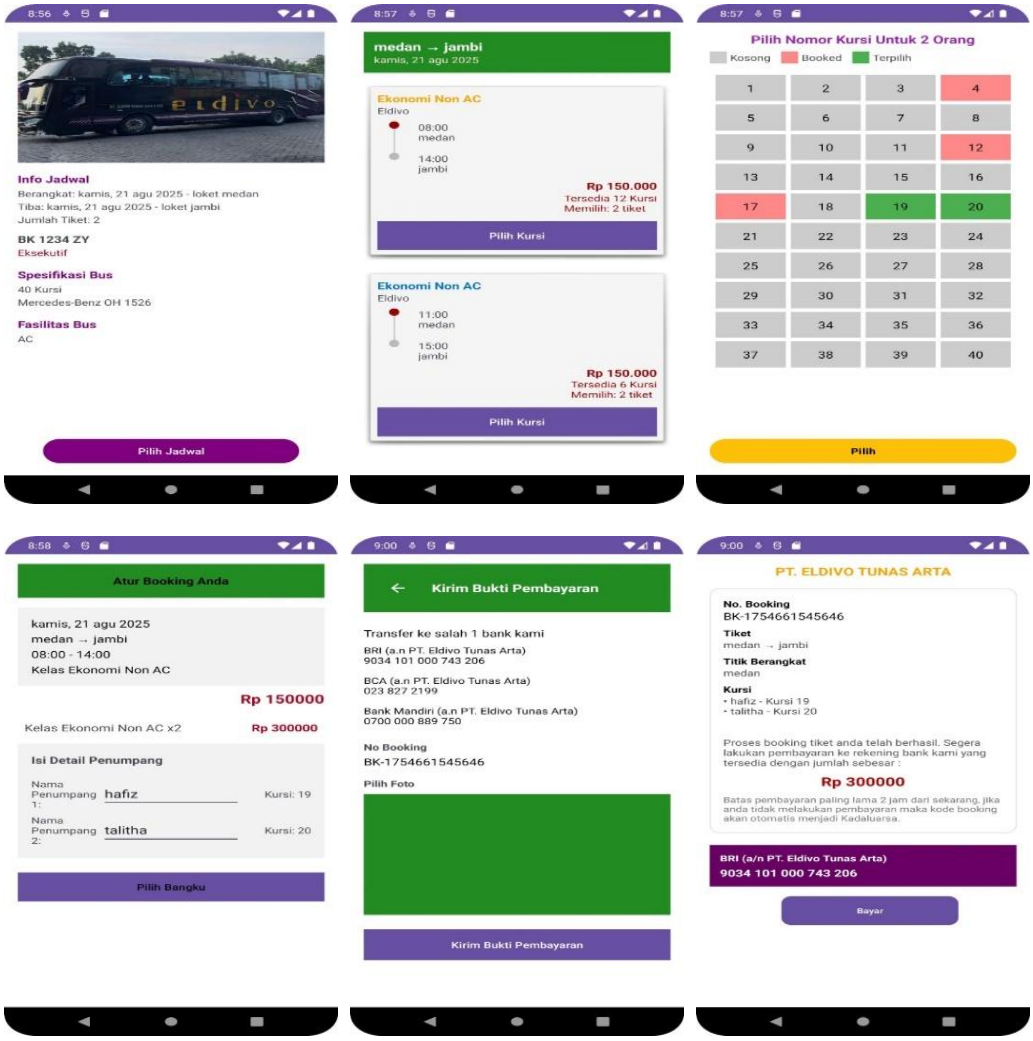


Figure 2. Menu display for customers

Application Menu Display for Admin

1. Login Menu Display : The Login Display is the first screen that appears when the program is run. It functions as a form for entering the admin username and password.
2. Bus Data Display : This form displays bus data options. When a bus data option is selected, the program will display the bus data.
3. Bus Route Form Display : This form displays bus route data options. When a bus route is selected, the program will display the bus route data.
4. City Form Display : This form displays city data options. When a city input is selected, the program will display the city data.
5. Create Bus Schedule Form Display : This form displays options for creating a bus schedule. When the create schedule option is selected, the program will display the bus schedule data.
6. Order Form Display : This form displays order data options. When an order data option is selected, the program will display the order data.
7. Payment Data Form Display : This form displays payment data options. When a payment data option is selected, the program will display the payment data.

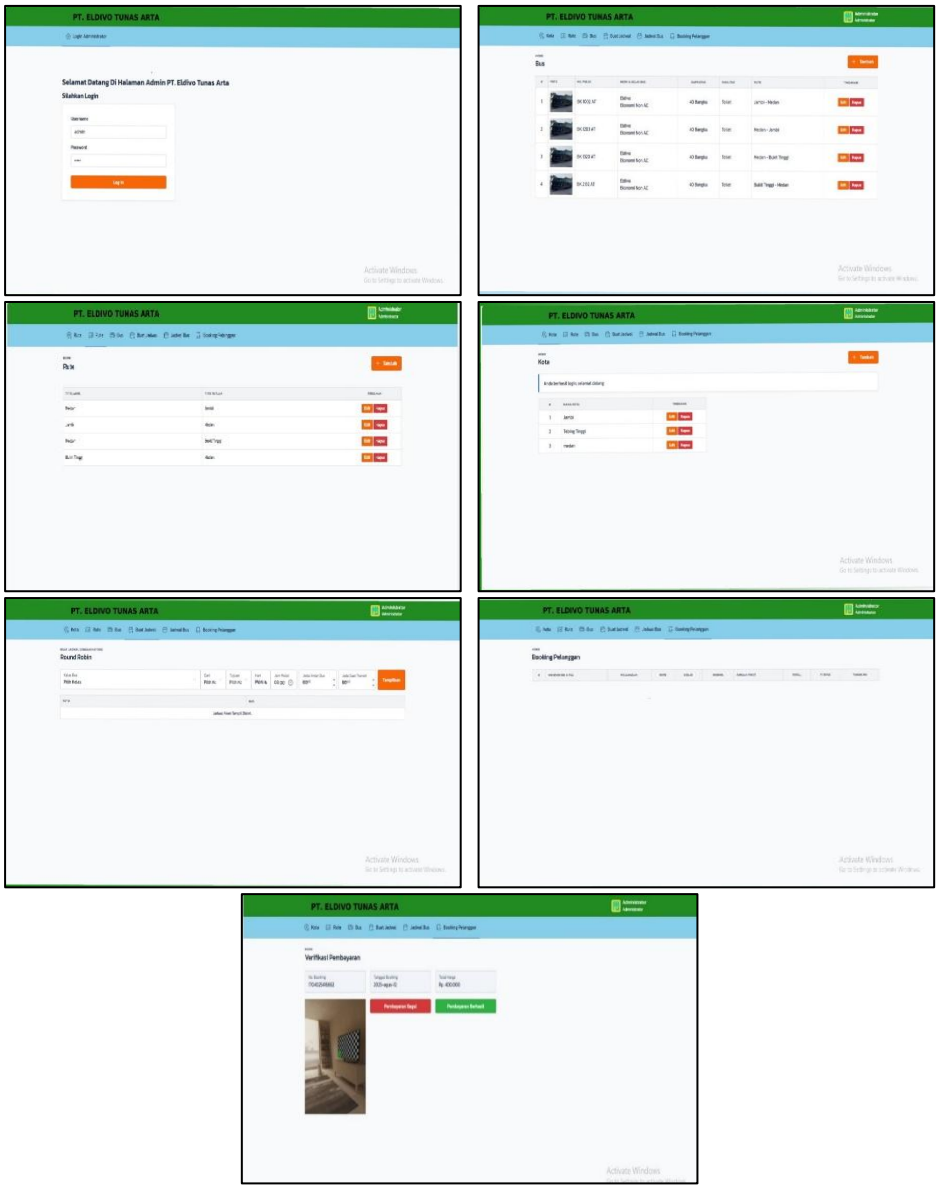


Figure 3. Application menu display for admin

2.4. System and Program Testing

System testing aims to ensure that the system is in a ready-to-use condition. The instrument used to carry out this testing is:

Test Case (Normal Data)

Table 14. Login System Testing

No.	Data Input	Expected Result	Observation	Conclusion
1	Username: User Password: User Click the login button	The form displays access to the User section as the main page of the accounting system data center	Able to enter the User main page	[✓] Accepted [ ] Rejected

Test Case (Invalid Data)

No.	Data Input	Expected Result	Observation	Conclusion
1	Username: User Password: User Click the login button	Cannot log in to the user page and an error message appears	Redirected to the error message page	[✓] Accepted [ ] Rejected

Test Case (Normal Data)

Table 15. Bus Data System Testing

No.	Data Input	Expected Result	Observation	Conclusion
1	Edit data	The bus data to be edited in the database, click save, then the data on the server database will be updated	The bus data in the database changes after clicking save	[✓] Accepted [ ] Rejected
2	Delete data	The bus data is deleted from the database by clicking delete, then the data on the server database will be removed	The bus data is successfully deleted from the database	[✓] Accepted [ ] Rejected

Test Case (Invalid Data)

No.	Data Input	Expected Result	Observation	Conclusion
1	Input data that does not match the data type	An error message appears indicating incorrect data input	A message appears stating that the data input is incomplete	[✓] Accepted [ ] Rejected

Test Case (Normal Data)

Table 16. City Data System Testing

No.	Data Input	Expected Result	Observation	Conclusion
1	Edit data	City data in the database is updated after clicking save	City data in the database changes successfully	[✓] Accepted [ ] Rejected
2	Delete data	City data is deleted from the database after clicking delete	City data is successfully deleted from the database	[✓] Accepted [ ] Rejected

Test Case (Invalid Data)

No.	Data Input	Expected Result	Observation	Conclusion
1	Input data that does not match the data type	An error message appears indicating incorrect data input	A message appears stating that the data input is incomplete	[✓] Accepted [ ] Rejected

Test Case (Normal Data)

Table 17. Bus Schedule System Testing

No.	Data Input	Expected Result	Observation	Conclusion
1	Edit data	Bus schedule data in the database will be updated after clicking save	Bus schedule data changes successfully	[✓] Accepted [ ] Rejected
2	Delete data	Bus schedule data is deleted from the database after clicking delete	Bus schedule data is successfully deleted	[✓] Accepted [ ] Rejected

Test Case (Invalid Data)

No.	Data Input	Expected Result	Observation	Conclusion
1	Input data that does not match the data type	An error message appears indicating incorrect data input	A message appears stating that the data input is incomplete	[✓] Accepted [ ] Rejected

Test Case (Normal Data)

Table 18. Payment Data System Testing

No.	Data Input	Expected Result	Observation	Conclusion
1	Upload	Payment data is entered into the database by clicking add, and the data is stored on the server database	Payment data is successfully stored in the database	[✓] Accepted [ ] Rejected

Test Case (Invalid Data)

No.	Data Input	Expected Result	Observation	Conclusion
1	Input data that does not match the data type	An error message appears indicating incorrect data input	A message appears stating that the data input is incomplete	[✓] Accepted [ ] Rejected

Test Case (Normal Data)

Table 19. Route Data System Testing

No.	Data Input	Expected Result	Observation	Conclusion
1	Add data	Route data is added to the database after clicking add	Route data is successfully stored in the database	[✓] Accepted [ ] Rejected

2	Edit data	Route data in the database is updated after clicking save	Route data changes successfully	[✓] Accepted [ ] Rejected
3	Delete data	Route data is deleted from the database after clicking delete	Route data is successfully deleted	[✓] Accepted [ ] Rejected

Test Case (Invalid Data)

No.	Data Input	Expected Result	Observation	Conclusion
1	Input data that does not match the data type	An error message appears indicating incorrect data input	A message appears stating that the data input is incomplete	[✓] Accepted [ ] Rejected

3. RESULTS AND DISCUSSION

3.1. System Implementation and Functionality (Results)

The developed application successfully addresses the challenges of manual ticketing by providing the public with real-time access to bus ticket availability and schedules at PT Eldivo Tunas Arta. Through this platform, users can seamlessly perform self-service ticket reservations.

Architecturally, the system was fully modeled using Unified Modeling Language (UML), which includes use case, class, activity, and sequence diagrams to ensure structured logic and scalability. The front-end application was built on the Android platform to maximize mobile accessibility, while a robust MySQL database handles backend data storage securely.

Currently, the transactional workflow relies on a semi-automated verification model. After selecting a schedule and securing a seat, customers complete their payment via bank transfer and upload their proof of transfer through the app. The administrative team at PT Eldivo Tunas Arta then verifies the transaction manually before the system formally confirms and allocates the reserved seat to the passenger.

3.2. Framework Analysis and Future Adjustments (Discussion)

While the integration of UML modeling and Android-MySQL architecture provides a stable and operational foundation for PT Eldivo Tunas Arta, the current transaction and communication workflows present clear opportunities for optimization.

First, the manual verification of bank transfers introduces a operational bottleneck, especially during peak travel seasons. Transitioning from manual uploads to integrated electronic payment gateways—such as e-banking, virtual accounts, or credit cards—would fully automate the validation process, reducing human error and significantly accelerating booking confirmation times.

Second, the user experience can be further enhanced by addressing the lack of proactive communication channels. Incorporating automated notification features via SMS, email, or push notifications is a crucial next step. This addition would not only remind passengers of upcoming departure schedules but also provide real-time updates regarding sudden delays or route changes, thereby increasing the overall reliability of the service.

4. CONCLUSION

The rapid development of information technology has encouraged the transportation sector to adopt digital systems in order to improve service efficiency and accessibility. This research developed an Android-based bus scheduling and ticket reservation system for PT. Eldivo Tunas Arta to overcome the limitations of manual scheduling and ticket booking processes.

The system was designed using UML modeling, including use case diagrams, class diagrams, activity diagrams, and sequence diagrams. The implementation of the Round Robin algorithm helps manage the rotation of bus fleets more fairly and systematically, so that the scheduling process becomes more efficient and well organized.

The application was implemented using the Android platform with a MySQL database for data storage, while the payment process is carried out through bank transfer with proof of payment verification. The results show that the system can improve operational efficiency and provide easier access for customers to obtain schedule information and book bus tickets.

REFERENCES

[1] S. Mardiyati and W. N. Cholifah, “Sistem penjadwalan bus di Terminal Jatijajar Depok menggunakan algoritma Round Robin: Artikel ilmiah,” J. Fasilkom, vol. 12, no. 1, pp. 48–55, May 2022.

[2] Y. S. Pane, “Sistem penjadwalan kegiatan terminal di pelabuhan dengan menggunakan metode Round Robin berbasis web,” J. Rekayasa Sist. (JUREKSI), vol. 2, no. 2, pp. 617–628, Jan. 2024.
[3] R. Excalanta, Y. Nataliani, and Y. R. Beeh, “Perancangan sistem informasi penjadwalan bus dengan metode Round Robin,” Universitas Kristen Satya Wacana, Salatiga, Indonesia, Tech. Rep., 2012.
[4] T. D. Putra and R. Purnomo, “Analisis algoritma round robin pada penjadwalan CPU,” J. Ilm. Teknol. Inform. Asia, vol. 15, no. 2, p. 85, 2021.
[5] S. Fadli, M. Ashari, and K. Imtihan, “Sistem penjadwalan event organizer dengan metode Round Robin (RR),” J. Manaj. Inform. dan Sist. Inform., vol. 3, no. 2, pp. 100–107, Jul. 2020.
[6] A. M. Komaruddin and P. Rispian, “Load balancing dengan metode Round Robin untuk pembagian beban kerja web server,” J. Siliwangi Seri Sains dan Teknol., vol. 5, no. 2, pp. 47–50, Dec. 2019.