

Website-Based Ticket Ordering Application Design Using Laravel Framework

Case Study: Bedegung Waterfall Tour, Muara Enim Regency

Lia Nur Liana¹, Wawan Gunawan^{2,*}

^{1,2} Information Systems, Raden Intan State Islamic University of Lampung, Indonesia
^{1,2} Jl. Lieutenant Colonel H. Endro Suratmin, Sukarama, Lampung, Indonesia

lianurlia120601@gmail.com
wawan.gunawan@radenintan.ac.id

Received: July 4th, 2025; Revised: November 11th, 2025; Accepted: November 25th, 2025.

Abstract

Bedegung Waterfall is a well-known natural tourism destination located in Bedegung Village, Muara Enim Regency, offering beautiful scenery and a prominent waterfall. Despite its popularity, the ticket booking process is still conducted manually, resulting in long queues, slow transactions, service delays, and traffic congestion around the tourist area, which reduces visitor comfort and management efficiency. To address these issues, this study proposes developing a web-based ticket-booking application to streamline transactions and enhance tourism service quality. The system was designed and developed using the Extreme Programming (XP) software development method to ensure flexibility, rapid development, and effective collaboration. The application was implemented using the Laravel framework, PHP as the programming language, and MySQL as the database management system, due to its reliability in handling large volumes of data. System design was supported by Flowcharts, Use Case Diagrams, Class Diagrams, Activity Diagrams, and Entity-Relationship Diagrams (ERDs) to clearly describe system workflows and data relationships. The objective of this research is to design, implement, and evaluate a web-based ticket booking system that improves transaction efficiency, reduces queues, and enhances visitor convenience. The system supports three user roles: Super Admin, Tourist Staff, and Member, each with distinct access rights and functionalities. System testing was conducted using the Black-Box Testing method to evaluate functional suitability against user requirements. A total of 37 testing scenarios were executed across all user roles, and the results showed that all system functions operated as expected, achieving a 100% testing success rate, indicating that the developed application is effective and feasible for implementation at Bedegung Waterfall.

Keywords: *Ticket Booking, Website, Laravel Framework, Extreme Programming (XP)*

This is an open access article under the CC BY-SA license.



Corresponding Author:

*Wawan Gunawan

Information Systems, Raden Intan State Islamic University of Lampung, Indonesia
Jl. Lieutenant Colonel H. Endro Suratmin, Sukarama, Lampung, Indonesia
Email: wawan.gunawan@radenintan.ac.id

I. INTRODUCTION

The tourism sector in Indonesia has experienced significant growth, including natural attractions such as the Bedegung Waterfall in Muara Enim Regency, South Sumatra. As a popular tourist destination, Bedegung Waterfall attracts numerous visitors annually. However, the manual ticket-booking system currently in place poses several operational challenges. Long queues, inaccurate record-keeping, and service delays are the primary issues faced by management. These problems not only reduce visitor comfort but also negatively impact the efficiency of tourism management.

To address these issues, a solution is needed that can modernize the ticket booking system, making it more efficient and user-friendly [1]. One approach is to develop a web-based ticket booking application using the Laravel framework [2, 3, 4, 5]. Laravel was chosen for its support for structured, secure, and efficient web application development. Additionally, the Extreme Programming (XP) method was applied during the development process to enhance flexibility and collaboration between developers and stakeholders [6, 7, 8]. The system was tested using black-box testing to ensure all features function as expected. This study aims to design and implement a web-based ticket-booking application to enhance the efficiency of the ticket-booking process at Bedegung Waterfall. The application is expected to address existing problems and enhance the quality of service and the visitor experience.

II. RESEARCH METHOD

A. Research methods

The research method used in this study is Extreme Programming (XP), an agile software development approach that emphasizes flexibility, iterative development, continuous testing, and close collaboration between developers and stakeholders to deliver high-quality software that meets user requirements [9, 10, 8].

B. Method of collecting data

Data are essential to research quality and outcomes. There are three data collection methods included, among others, as follows:

1. Observation

Observation is a data collection method that is carried out by visiting directly to the object to be studied. To obtain real data, we conducted direct observations at the Bedegung Muara Enim waterfall tourist attraction.

2. Interviews

Interviews are a method of collecting data in which questions and answers are exchanged directly with the respondent. By holding direct discussions with Mrs. Else Yusnani as head of the Bedegung Waterfall Tourism UPTD to find out what is needed in designing and making this system. The author also posed questions to tourism officers and received answers regarding aspects of the tourist ticket sales system, of which the author was previously unaware. The author also conducted direct interviews with visitors to the Bedegung Muara Enim Waterfall tourist attraction regarding the effectiveness of purchasing tourist tickets manually.

3. Literature Study

In conducting a literature study, the author complements the gaps in data obtained from field studies. Data is collected from print and electronic media sources and can be used as a reference for discussing problems.

4. Documentation

Documentation of data collection methods with evidence of activities.

C. System development methods

To guide the system development in Figure 1, the Extreme Programming (XP) software development model was applied. XP is a suitable method for small-scale, less complex projects, enabling iterative, flexible development. The main stages in the XP process implemented in this research are:

- 1) Planning

In this stage, the business context, system requirements, and project goals were identified. Key outputs, including feature lists, user roles (super admin, staff, and members), and the application timeline, were de-

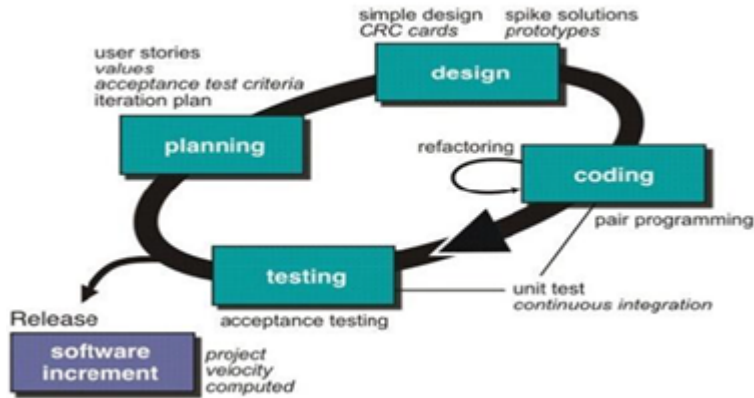


Fig. 1. Stages of Extreme Programming [6]

terminated. A needs analysis was also conducted to outline both functional and non-functional requirements for the system.

2) Design

The system was designed using the Model-View-Controller (MVC) architecture with Laravel as the framework. Various diagram types, including Flowcharts [2, 11], Use Case Diagrams [3], and Entity-Relationship Diagrams (ERDs) [12, 11, 13], were created to illustrate system processes and data structures. In the use case diagram in Figure 2 and in Figure 3, there are three actors: visitors, admin, and tourist officers. A use case is a description of a system function from the view/perspective of a system user in .

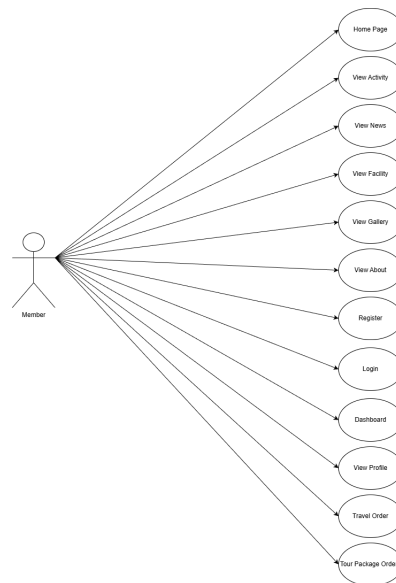


Fig. 2. Use Case Diagram Member Waterfall Tourism Website Application

1. User Admin and Tourist Staff

The Admin is the primary actor in the system and has the highest level of access rights. The Admin is responsible for managing, updating, and uploading all content displayed on the website to ensure users receive accurate, up-to-date information. These responsibilities include managing user accounts, tourism content, galleries, sliders, categories, facilities, tour packages, booking data, and system settings. In this role, as shown in Figure 3, the Admin ensures that all system functionality operates properly and that the information presented to users is reliable and well-maintained.

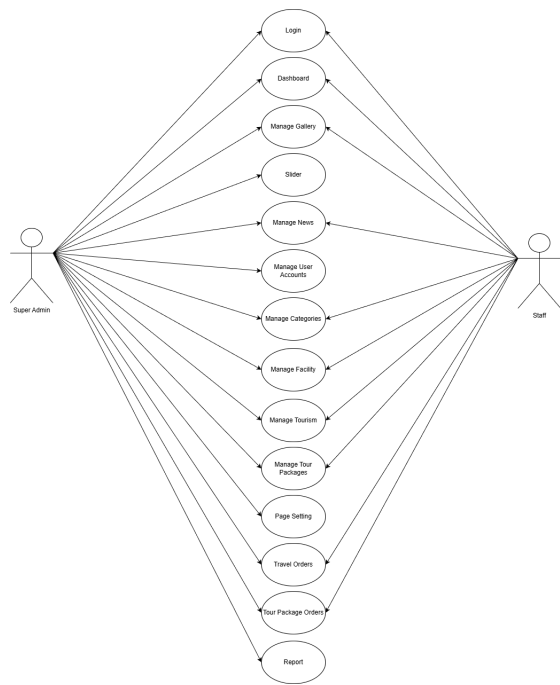


Fig. 3. Use Case Diagram Member Waterfall Tourism Website Application

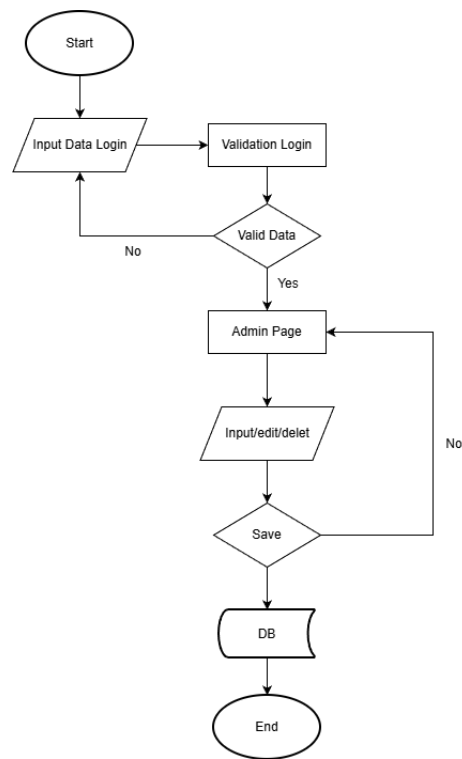


Fig. 4. Flowchart for Admin and Tourist Staff

Figure 4 illustrates the system workflow for Admin and Tourist users. The process begins with the login stage, where users must enter valid credentials to access the system. After successful authentication, users are directed to the dashboard according to their respective access rights. Admin users can perform full data management activities, including adding, editing, and deleting system data, while Tourist Staff have limited access based on their assigned permissions. Both roles can manage tourism-related information such as galleries, categories, facilities, tours, tour packages, and booking data. The flowchart clearly shows decision points, process flows, and system responses, ensuring structured and efficient system operation.

2. User Flowchart

Figure 5 presents the flowchart for system users (Members). The process starts with user authentication through the login or registration feature. Once logged in, users can access various system features, including viewing tourism information, browsing galleries, checking tour packages, and making ticket or tour package reservations. Users can also manage their personal account information and submit messages through the contact form. This flowchart demonstrates in Figure 5 a simple and user-friendly interaction flow, emphasizing ease of use and efficient transaction processing.

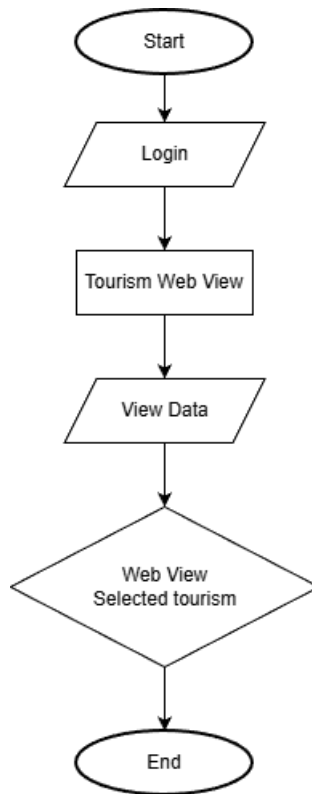


Fig. 5. User Flowchart

3. Entity Relationship Diagrams (ERD) Entity-Relationship Diagrams are tools used to create databases and illustrate database workflows. In ERD, there are 3 basic elements in Figure 6: entities, attributes, and relationships [12, 11, 13, 14, 10].

3) Coding

The coding phase involves implementing the system's functionality in accordance with the planned requirements and design. Laravel and PHP were used to develop the back end, and MySQL was selected as the database management system.

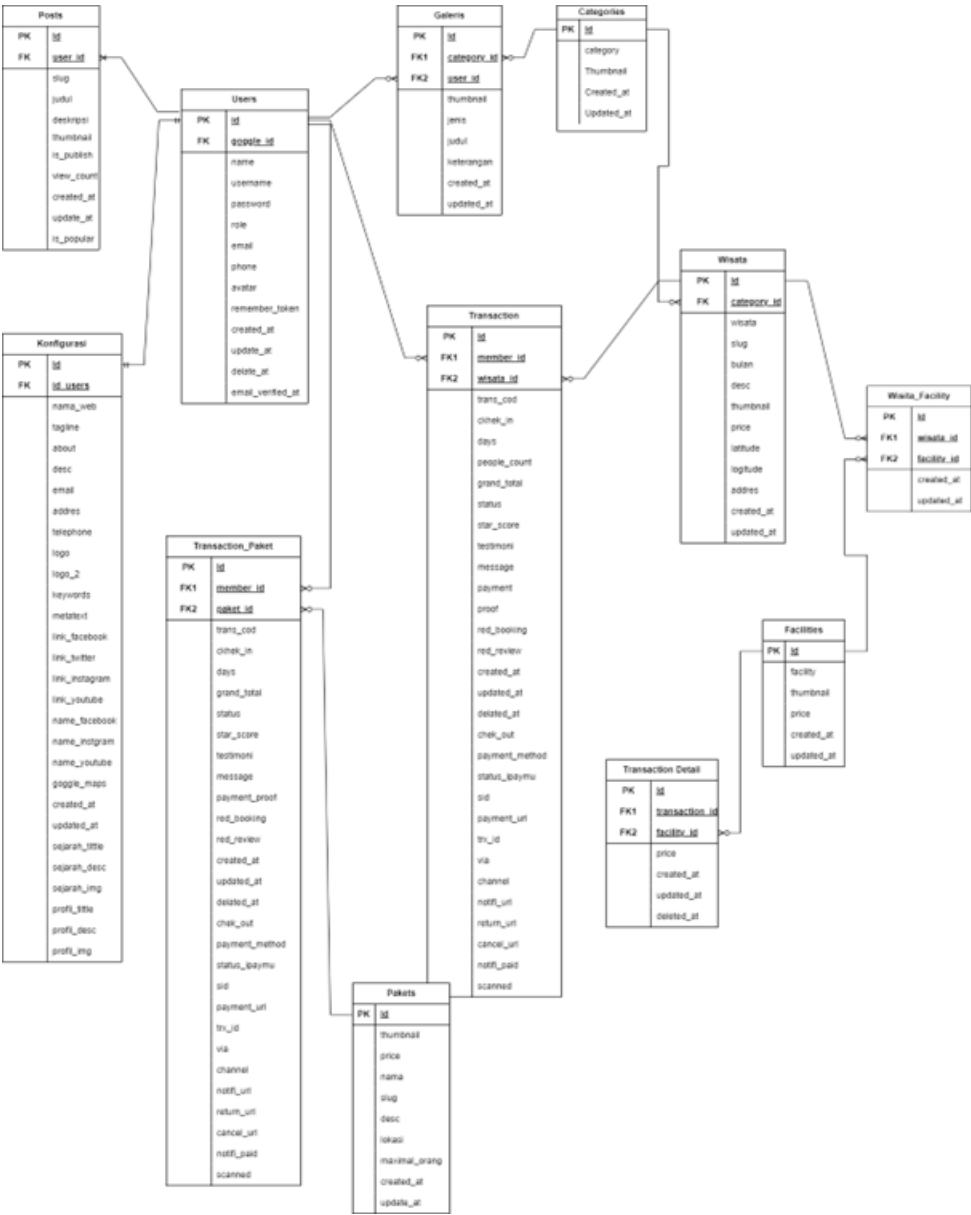


Fig. 6. Entity Relationship Diagram

4) Testing

Black-box testing was employed to ensure that the system met the functional requirements. This type of testing focuses on the system's outputs, ensuring that all user interactions (e.g., booking, payment processing) function as expected, without examining the underlying code [15, 2, 4].

III. RESULTS AND DISCUSSION

A. Results

After conducting the analysis and design, the author will develop a web-based ticket-booking program informed by the analysis results and the implemented plans. The results of this research are a website-based ticket-booking application built on the Bedegung Waterfall Tourism Laravel framework, featuring several menus for visitor, tourist officer, and admin pages. As for the visitor page, there are several menus including the log in menu, Register, homepage, profile, tour orders, tourist location package orders, then on the tour officer page, there are several menus including log in, homepage, gallery, news, categories, facilities, account, tour bookings, ticket bookings. Next on the admin page, there are menus including log in, tour data, add tour, tour details, data officer, add officer, officer details, add ticket, ticket details, purchase, purchase details, and data report.

B. Discussion

The page displays for visitors, tourist officers, and admin on the website-based ticket booking application at Bedegung Muara Enim Waterfall Tourism, as follows :

1) Dashboard Menu Admin Page

In Figure 7 the Super Admin Dashboard serves as the main control panel for system administration. This page provides an overview of system activities, including user management, tourism data, booking statistics, and system notifications. From the dashboard menu, the Super Admin can easily navigate to other system modules, including gallery management, tour categories, facilities, tour packages, bookings, and reports. The dashboard is designed to present concise, relevant information to support effective monitoring and decision-making.

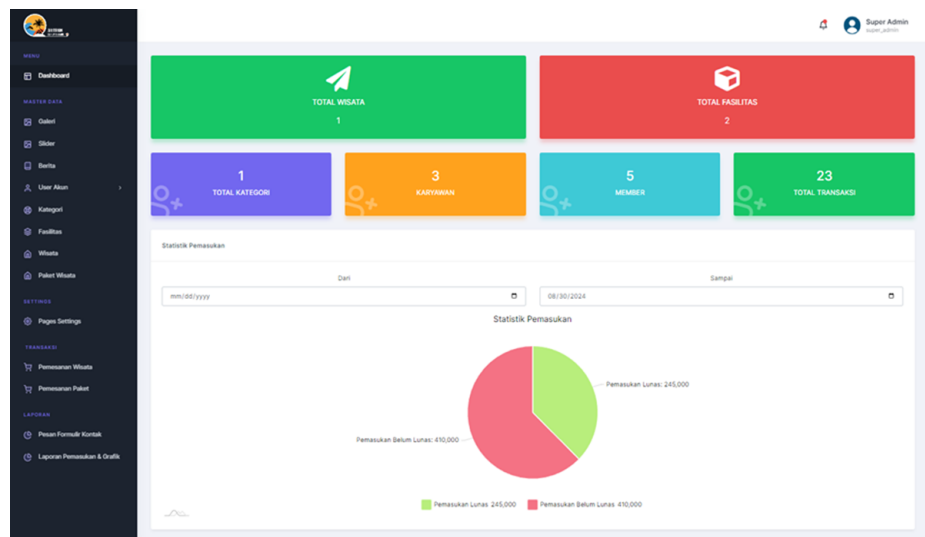


Fig. 7. Super Admin Dashboard

2) Staff Page

1. Tour List Display

In Figure 8, the Tour List Display page allows staff to view and manage all available tour data in the system. This page presents a list of tourist destinations, along with relevant information, including tour names, prices, and availability. Staff members can perform management actions, including

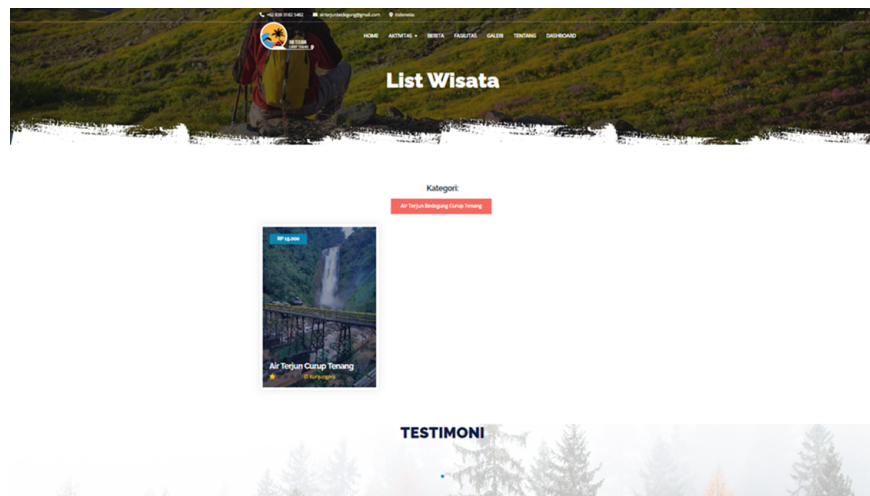


Fig. 8. Tour List Display

adding, editing, and deleting tour data, ensuring that the information presented to visitors remains accurate and up to date.

2. Added Tourist Menu

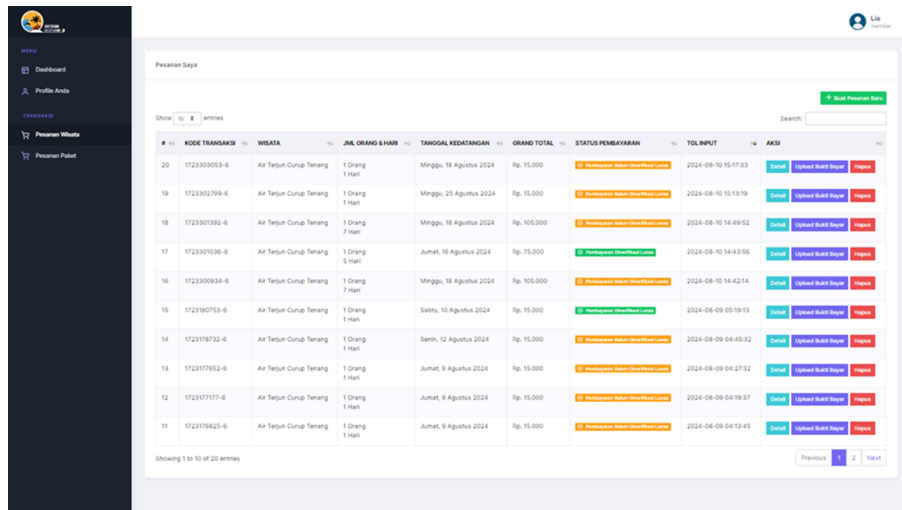
In Figure 9 the Added Tourist Menu page is used by staff to input new tourist destination data into the system. This page provides a form that enables staff to enter detailed information, including tour name, description, ticket price, images, and available facilities. The form is designed to be user-friendly, enabling staff to efficiently add new tourism content to the system.

Fig. 9. Added Tourist Menu

3) Visitor Page

1. Tour Order Display

In Figure 10 tour Order Display page enables visitors to view details of their tour ticket orders. This page presents booking information, including the order number, selected tour, booking date, ticket quantity, total cost, and order status. Through this menu, users can monitor the status of their reservations and verify that their booking details are correct.

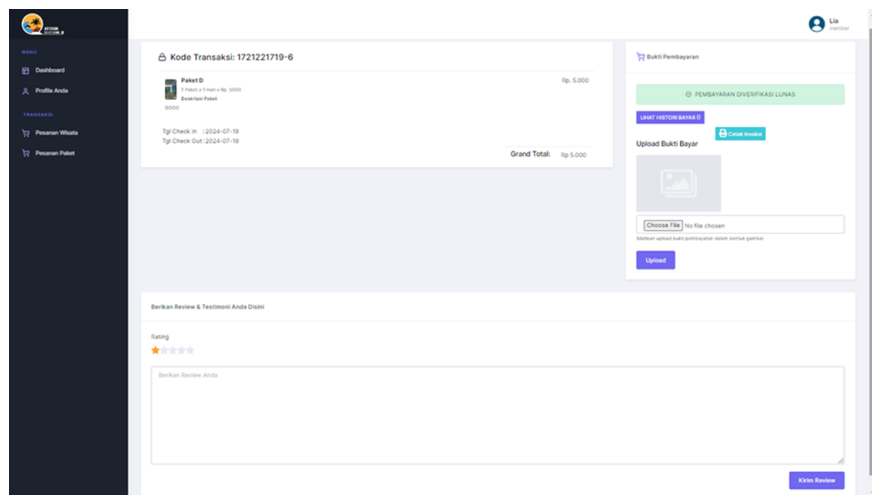


#	KODE TRANSAKSI	WISATA	JAL, ORANG & HARI	TANGGAL KEDATANGAN	GRAND TOTAL	STATUS PEMBAYARAN	TOL INPUT	Aksi
20	172393053-6	Air Terjun Curup Tenang	1 Orang 1 hari	Minggu, 18 Agustus 2024	Rp. 15.000	Pembayaran Belum Diverifikasi Lunas	2024-08-10 15:17:33	Detail Upload Bukti Bayar Hapus
19	17239302799-6	Air Terjun Curup Tenang	1 Orang 1 hari	Minggu, 25 Agustus 2024	Rp. 15.000	Pembayaran Belum Diverifikasi Lunas	2024-08-10 15:13:19	Detail Upload Bukti Bayar Hapus
18	1723930392-6	Air Terjun Curup Tenang	1 Orang 7 hari	Minggu, 18 Agustus 2024	Rp. 105.000	Pembayaran Belum Diverifikasi Lunas	2024-08-10 14:49:52	Detail Upload Bukti Bayar Hapus
17	172393036-6	Air Terjun Curup Tenang	1 Orang 5 hari	Jumat, 16 Agustus 2024	Rp. 75.000	Pembayaran Telah Diverifikasi Lunas	2024-08-10 14:43:56	Detail Upload Bukti Bayar Hapus
16	17239300934-6	Air Terjun Curup Tenang	1 Orang 7 hari	Minggu, 18 Agustus 2024	Rp. 105.000	Pembayaran Belum Diverifikasi Lunas	2024-08-10 14:42:14	Detail Upload Bukti Bayar Hapus
15	1723930753-6	Air Terjun Curup Tenang	1 Orang 1 hari	Sabtu, 10 Agustus 2024	Rp. 15.000	Pembayaran Telah Diverifikasi Lunas	2024-08-09 05:19:13	Detail Upload Bukti Bayar Hapus
14	1723930732-6	Air Terjun Curup Tenang	1 Orang 1 hari	Senin, 12 Agustus 2024	Rp. 15.000	Pembayaran Belum Diverifikasi Lunas	2024-08-09 04:45:52	Detail Upload Bukti Bayar Hapus
13	1723930732-6	Air Terjun Curup Tenang	1 Orang 1 hari	Jumat, 9 Agustus 2024	Rp. 15.000	Pembayaran Belum Diverifikasi Lunas	2024-08-09 04:27:52	Detail Upload Bukti Bayar Hapus
12	1723930737-6	Air Terjun Curup Tenang	1 Orang 1 hari	Jumat, 9 Agustus 2024	Rp. 15.000	Pembayaran Belum Diverifikasi Lunas	2024-08-09 04:19:37	Detail Upload Bukti Bayar Hapus
11	1723930825-6	Air Terjun Curup Tenang	1 Orang 1 hari	Jumat, 9 Agustus 2024	Rp. 15.000	Pembayaran Belum Diverifikasi Lunas	2024-08-09 04:13:45	Detail Upload Bukti Bayar Hapus

Fig. 10. Tour Order Display

2. Tour Package Booking Menu Display

In Figure 11, the Tour Package Booking Menu Display allows visitors to make reservations for tour packages offered by the tourism management. This page provides detailed information about available tour packages, including package descriptions, pricing, and included facilities. Visitors can select a desired package, complete the booking form, and proceed with the reservation process in a simple and efficient manner.



Kode Transaksi: 172121719-6

QR Code: 172121719-6

Grand Total: Rp. 5.000

Bukti Pembayaran

PEMBAYARAN DIVERIFIKASI LUNAS

Upload Bukti Bayar

Choose File No file chosen

Upload

Berikan Review & Testimoni Anda Disini

Rating

Berikan Review Anda

Kirim Review

Fig. 11. Tour Package Booking Menu Display

3. Print Invoice Menu Display

In Figure 12, the Print Invoice Menu Display allows visitors to generate and print an invoice for their tour or tour package bookings. This invoice contains essential transaction details, including the booking ID, visitor information, tour details, payment amount, and transaction date. This feature enables visitors to maintain a formal record of their transactions and supports transparency in the booking process.

4) System Testing

In Table I, Black Box Testing is a software testing method that focuses on evaluating the external functionality of a system without examining its internal structure or source code. This testing approach in Table II is based on system requirements specifications and assesses whether each function performs as

WISATA AIR TERJUN BEDEGUNG CURUP TENANG KODE TRANSAKSI: 172301036-6

1. DATA IDENTITAS PEMESAN

Nama Lengkap : Lia
 Alamat Email : lianurliana2022@gmail.com
 Nomor HP :

2. DATA PESANAN

Wisata : Air Terjun Curup Tenang
 Jumlah Orang : 1
 Tanggal Kadarangan : Jumat, 18 Agustus 2024
 Jumlah Hari Kunjungan : 5 Hari

3. DATA PEMBAYARAN

Status Pembayaran : ☒ LUNAS ☐ BELUM LUNAS
 Metode Pembayaran : ☐ Manual ☒ Payment Gateway
 Jumlah Pembayaran : Rp 75.000
 Pajak : Rp 0
 Jumlah Pembayaran : Rp 75.000



Fig. 12. Tour Invoice Menu Display

expected from the user's perspective. In this research, Black Box Testing was applied in Table III to verify that all system features—including dashboards, data management, booking processes, and invoice generation—functioned correctly and met user requirements.

Based on the Black Box testing results shown in Table I Black Box Testing Design (Super Admin), it can be seen that all features tested at the Super Admin level function according to the expected scenarios and results. Testing was conducted on 14 main menus: Dashboard, Gallery, Sliders, User Account, Category, News, Facility, Tour, Tour Packages, Page Settings, Tour Booking, Tour Package Booking, Contact Form Message, and Order Report. Each test scenario yielded valid results, indicating that the system can display pages, process data, and perform the functions of adding, modifying, deleting, and managing data properly. With 14 test scenarios, all successful, the success rate of Black Box testing at the Super Admin level reached 100%, indicating that all administrative functions of the system comply with functional requirements.

TABLE I. BLACK BOX TESTING DESIGN (SUPER ADMIN)

Menu	Scenario	Testing Steps	Expected Results	Results Obtained	Ob-
Dashboard	Admin can access the dashboard	Login as admin > click the dashboard menu	Admin dashboard is displayed	Valid	
Gallery	Admin can manage travel image gallery	Login as admin > click gallery menu	Gallery can add, edit and delete data	Valid	
Sliders	Admin can manage slider display	Login as admin > click the slider menu	Slider can add, edit and delete images	Valid	
User Account	Admin can manage all user accounts	Login as admin > click on the user account menu	User accounts can add, edit and delete user accounts	Valid	
Category	Admin can manage tour categories	Login as admin > click on the category menu	Categories can add, edit and delete categories	Valid	
News	Admin can manage news	Login as admin > click on the news menu	News can add, edit and delete news	Valid	
Facility	Admin can manage facility information	Login as admin > click on the facilities menu	Facilities can add, edit and delete facilities	Valid	
Tour	Admin can manage the list of tour tickets	Login as admin > click on the travel menu	The tour menu can add, delete and edit tour tickets	Valid	

Menu	Scenario	Testing Steps	Expected Results	Results Obtained	Ob-
Tour Packages	Admin can manage tour package tickets	Login as admin > click on the tour package menu	The tour package menu can add, edit and delete tour package tickets	Valid	
Page Settings	Admin can manage page settings	Login as admin > click the page setting menu	The page setting menu can manage page settings	Valid	
Tour Booking	Admin can view and manage tour ticket reservations	Login as admin > click on tour booking	The list of tour reservations is displayed and can be processed	Valid	
Tour Package Booking	Admin can view and manage tour package bookings	Login as admin > click on booking tour packages	The list of package orders is displayed and can be processed	Valid	
Contact Form Message	Admin can view all messages from contact form	Login as admin > click on the contact form message menu	A list of all messages from the contact form is displayed	Valid	
Order Report	Admin can view all order reports	Login as admin > click on the order report menu	The order report is displayed	Valid	

In Table II Black Box Testing (Employees), testing focused on functions accessible to users with the Employees role. There were 11 test scenarios covering Dashboard access, Slider display on the main page, and management of Gallery, User Account, Category, News, Facility, Tour, Tour Packages, Tour Booking, and Tour Package Booking. The test results showed that all menus could be accessed and executed according to the specified scenarios, with no errors found in the process or system display. All tests obtained Valid results, indicating that access rights and functionality for Employees had been implemented properly. Thus, of the 11 test scenarios conducted, the success rate of Black Box testing at the Employees level was 100%.

TABLE II. BLACK BOX TESTING (TOURIST STAFF)

Menu	Scenario	Testing Steps	Expected Results	Results Obtained	Ob-
Dashboard	Tourist Staff can access the dashboard	Login as tourist staff > click dashboard menu	Tourist staff dashboard is displayed	Valid	
Sliders	Tourist Staff can view the travel promotion slider on the home menu	Access without login > click home menu	Promotional slider is displayed on the home page	Valid	
Gallery	Tourist Staff can manage a gallery of travel images	Login as tourist staff > click gallery menu	Gallery can add, edit and delete data	Valid	
User Account	Tourist staff can access their account information	Login as tourist staff > click User Account menu	The account information form is displayed	Valid	
Category	Tourist Staff can manage travel categories	Login as tourist staff > click on category menu	Categories can add, edit and delete categories	Valid	
News	Tourist Staff can manage news	Login as tourist staff > click on news menu	News can add, edit and delete news	Valid	
Facility	Tourist Staff can manage facility information	Login as Tourist Staff > click on the facilities menu	Facilities can add, edit and delete facilities	Valid	

Menu	Scenario	Testing Steps	Expected Results	Results Obtained	Ob-
Tour	Tourist staff can manage the list of tour tickets	Login as tourist staff > click on the travel menu	The tour menu can add, delete and edit tour tickets.	Valid	
Tour Packages	Tourist staff can manage tour package tickets	Login as tourist staff > click on the tour package menu	The tour package menu can add, edit and delete tour package tickets	Valid	
Tour Booking	Tourist Staff can view and manage tour ticket reservations	Login as tourist staff > click on the tour booking menu	The list of tour reservations is displayed and can be processed	Valid	
Tour Package Booking	Tourist Staff can view and manage tour package bookings	Login as tourist staff > click on the tour package menu	The list of package orders is displayed and can be processed	Valid	

The test results listed in Table III Black Box Testing (Member) show that all features intended for Members can be used according to user needs. Testing was conducted on 12 scenarios, including Dashboard access; Gallery and Slider displays; User Account access and updates; Category, News, Facility, and Tour displays; Tour Package displays; and the processes of ordering tour tickets, ordering tour packages, and sending messages via the Contact Form. All test scenarios resulted in a Valid status, indicating that the system successfully displayed information and processed Members' input. Based on these results, the success rate of Black Box testing at the Member level was 100%.

TABLE III. BLACK BOX TESTING (MEMBER)

Menu	Scenario	Testing Steps	Expected Results	Results Obtained	Ob-
Dashboard	Member can access the dashboard	Login as a member > click the dashboard menu	Member dashboard is displayed	Valid	
Gallery	Members can view a gallery of tourist images	Login as a member > click the gallery menu	A gallery of tourist images is displayed	Valid	
Home (slider)	Members can view the image slider on the home menu	Access without login > click home menu	Image slider is displayed on the home page	Valid	
User Account	Members can access and update account information	Login as a member > click the user account menu	The account information form is displayed and can be updated	Valid	
Category	Members can view tourism categories	Login as a member > click the category menu	List of travel categories is displayed	Valid	
News	Members can read the latest news	Login as a member > click on the news menu	List of latest news is displayed	Valid	
Facility	Members can view facility information	Login as a member > click on the facilities menu	Tourist facility information is displayed	Valid	
Tour	Members can view tour tickets	Login as a member > click on the travel menu	List of tourist tickets is displayed	Valid	

Menu	Scenario	Testing Steps	Expected Results	Results Obtained	Ob-
Tour Pack-ages	Members can view tour packages	Login as a member > click on the tour package menu	List of tour pack-ages is displayed	Valid	
Tour Book-ing	Members can make tour ticket reservations	Login as a member > click on the tour booking menu	The tour ticket booking form is displayed and can be processed	Valid	
Package Or-dering	Members can make reservations for tour packages	Login as a member > click on the pack-age ordering menu	The tour package ticket booking form is displayed and can be pro-cessed	Valid	
Contact Form Mes-sage	Members can send messages via the contact form	Login as a member > click on the con-tact form message menu	Message sent suc-cessfully	Valid	

IV. CONCLUSION

This research has developed a web-based ticket-booking application for the Bedegung Waterfall in Muara Enim Regency using the Laravel framework. The developed application aims to simplify and automate the ticket-booking process, thereby improving service efficiency in tourism management and providing greater convenience for visitors. The system was implemented using PHP as the programming language and MySQL as the database management system. It supports multiple user roles, including super admin, staff, and members, each with different access rights and functionalities. The main features of the system include online ticket and tour package booking, user account management, content management, transaction processing, and integration with an online payment gateway to support cashless transactions. System testing was conducted using the black-box testing method to evaluate functional correctness from the user's perspective. The test results showed that all features tested across all user roles performed as expected. All test cases produced valid results, indicating a 100% success rate in functional testing. This confirms that the system meets the specified functional requirements and operates reliably, with no functional errors. In conclusion, the developed system is expected to provide significant benefits for tourism management by improving operational efficiency, data management, and service quality. Additionally, it enhances visitor satisfaction by offering a practical, efficient, and accessible platform for online ticket booking.

REFERENCES

- [1] G. Vial, "Understanding digital transformation," *MIS Quarterly*, vol. 43, no. 1, pp. 223–251, 2019.
- [2] P. Isaias and T. Issa, "Digital transformation in web-based information systems," *Journal of Web Engineering*, vol. 20, no. 5, pp. 1341–1360, 2021.
- [3] F. Almeida, "Web applications development: Trends and challenges," *Future Internet*, vol. 12, no. 7, p. 115, 2020.
- [4] A. Susanto and Meiryani, "Information system development in the digital era," *Journal of Information Systems Engineering*, vol. 6, no. 2, pp. 85–95, 2022.
- [5] I. Pappas and A. Woodside, "Digital platforms and information systems," *Information Systems Frontiers*, vol. 23, no. 4, pp. 815–828, 2021.
- [6] K. Beck and C. Andres, "Extreme programming explained: Embrace change in modern software development," *IEEE Software*, vol. 37, no. 3, pp. 40–47, 2020.
- [7] A. Mishra and D. Dubey, "A comparative study of agile software development methodologies," *International Journal of Software Engineering*, vol. 14, no. 2, pp. 45–55, 2021.

- [8] N. Salleh and E. Mendes, “Agile practices in software engineering: A systematic review,” *Empirical Software Engineering*, vol. 26, no. 1, pp. 1–38, 2021.
- [9] S. Alsaqqa and S. Sawalha, “Agile software development: Methodologies and trends,” *International Journal of Interactive Mobile Technologies*, vol. 13, no. 11, pp. 246–255, 2019.
- [10] R. S. Pressman, “Software engineering practices in modern information systems,” *Journal of Systems and Software*, vol. 174, p. 110872, 2021.
- [11] P. Mohagheghi and V. Dehlen, “Where is the proof? a review of experiences from applying uml,” *Software and Systems Modeling*, vol. 18, no. 3, pp. 2191–2213, 2019.
- [12] G. Booch, J. Rumbaugh, and I. Jacobson, “Unified modeling language (uml): Current trends,” *IEEE Computer*, vol. 52, no. 5, pp. 72–80, 2019.
- [13] M. Staron, “Model-based software development with uml,” *Software Quality Journal*, vol. 29, no. 4, pp. 1137–1156, 2021.
- [14] C. Coronel and S. Morris, “Database design and the entity-relationship model,” *ACM SIGMOD Record*, vol. 49, no. 1, pp. 38–44, 2020.
- [15] T. Connolly, “Data modeling and database design: An overview,” *Information and Software Technology*, vol. 117, p. 106194, 2020.