Design and Development of the "Campoga" Sports Field Reservation System Based on a Website

Nicolas Siregar[a1], Dwi Putra Githa[a2], Gusti Made Arya Sasmita[a3]
[a]Program Studi Teknologi Informasi, Fakultas Teknik, Universitas Udayana,
Bukit Jimbaran, Bali, Indonesia
e-mail: [1]nicholregar@gmail.com, [2]dwiputragitha@gmail.com, [3]aryasasmita@unud.ac.id

Abstrak

Pengembangan sistem informasi pemesanan lapangan olahraga berbasis web ini bertujuan untuk mempercepat proses reservasi dan pengelolaan data reservasi lapangan. Sistem ini diharapkan dapat mengelola data reservasi dengan lebih efisien dan tepat. Penelitian ini bertujuan untuk memahami bagaimana merancang dan membangun sistem informasi reservasi lapangan olahraga berbasis web, yang dikenal dengan nama Campoga. Metode yang digunakan dalam pengembangan sistem ini adalah metode Waterfall. Hasil dari penelitian ini adalah terciptanya Sistem Informasi Pemesanan Lapangan Olahraga berbasis website, yang dibangun menggunakan Laravel, database MySQL, dan Visual Studio Code sebagai editor pengujian.

Kata kunci: *lapangan olahraga, pemesanan, rancang bangun, sistem informasi*

Abstract

The development of this web-based sports field reservation information system aims to accelerate the reservation process and management of field reservation data. This system is expected to manage reservation data more efficiently and accurately. The purpose of this study is to understand how to design and build a web-based sports field reservation information system, known as Campoga. The method used in the development of this system is the Waterfall method. The result of this study is the creation of a web-based sports field reservation information system, built using Laravel, MySQL database, and Visual Studio Code as the testing editor.

Keywords : *design, information systems, orders, sports fields*

1. Introduction

The Sports Field Booking Website is a platform designed to assist the public in finding information about and booking sports fields, while also helping field rental owners manage field usage information. The public often faces difficulties in finding available sports fields that meet their desired criteria. The goal of this website is to provide a solution for searching and booking sports fields. To achieve this, an observation of several sports field rental places, a survey of sports field users, and a literature review on booking systems were conducted as the basis for developing this website. Users can select their preferred sports field and then make an online reservation. The booking will be received by the sports field rental owners, making the transaction process more effective and efficient.

A previous study [1] served as a guide for the researchers in designing and developing this information system, aiming to provide convenience for renters when booking futsal field schedules without the need to visit the futsal venue directly. This application is developed using the Waterfall model, a linear design approach. It is implemented in PHP, with MySQL for data storage, and employs black-box testing as the method for evaluation. This application can be used by three types of users: admin, staff, and renters. Admins can manage field data, rental prices, renters, bookings, payment confirmations, and static pages. Renters can book fields through the website, check field availability, and confirm their payments. With the existence of this sports field rental information system, the public is expected to be able to choose and book their desired sports field online. The bookings will be received by the sports field rental owners, making the transaction process more effective and efficient.

2. Research Methodology

The analysis flow in the Sports Field Rental Information System research is designed using the waterfall method. The Waterfall method represents a structured and step-by-step approach to software development.

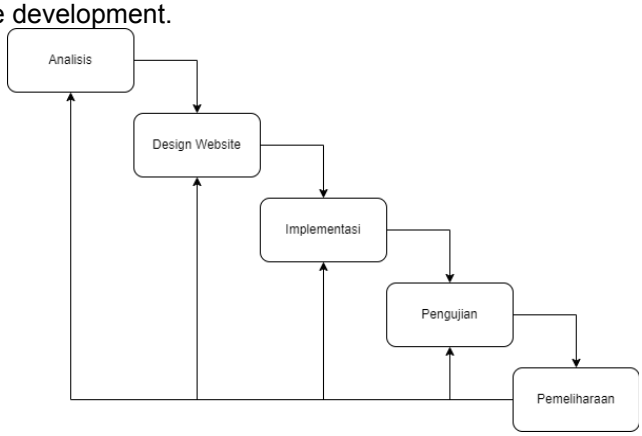

Figure 1 Waterfall Method

Figure 1 illustrates the analysis flow explained by the waterfall method. The design process begins with an analysis and research phase for the website to be created. Then, the system design is realized, depicting the website that will be developed. The next step is the implementation phase, or the creation of the website. Once the implementation is complete, testing is conducted using the black-box testing method. The design process is then concluded with the maintenance phase to ensure the website system functions properly at all times[2].

2.1 The data collection methods

The data collection methods used in the research for the development of the Sports Field Rental Information System "Campoga" Based on a Website include Observation, Literature Review, and Interviews. Data collection was carried out through surveys conducted on-site or by directly observing the rental process occurring at the sports fields. Data was also gathered from relevant journals or references that support the development of the report. Additional data collection was conducted through face-to-face interactions and direct question-and-answer sessions between the researchers and the informants [3].

2.2 Implementation

In this phase, the development of the Campoga information system application, based on a website, was carried out using the Laravel framework. The process began with the installation of Laravel and the creation of a MySQL database to store information. The application was coded using PHP, with Visual Studio Code as the editor. The result is an application that allows users to efficiently book sports fields via the web platform.

2.3 Testing

In this phase, the system undergoes testing to ensure its performance. The testing is performed using the black-box method, which includes both scenario testing and user acceptance testing. Scenario testing evaluates various features of the system, while user acceptance testing determines if the system meets the expectations and needs of the users [4]. The testing phase is conducted to verify that the information system functions properly and aligns with user requirements.

3. Literature Review

The literature review discusses theories related to the writing of research reports.

3.1 Website

The web has evolved through three major stages or generations: Web 1.0, Web 2.0, and Web 3.0 (also known as the Semantic Web). Web 1.0 emerged first and had a profound impact on how industries and media functioned during that era. It marked the beginning of the internet, focusing primarily on static, read-only content. Compared to today’s web, Web 1.0 was

less interactive because users could only read information on the website. In Web 2.0, users could drag & drop, chat, use voice features, and enjoy auto-completion on desktop. Web 3.0 or the semantic web allows users to search, access, and share information from various software. Therefore, Web 3.0 is considered more efficient than its predecessors, as users can access information from more than just desktop devices. The web is widely used, especially by companies, as a key information technology strategy for various reasons [5].

3.2 Hypertext Transfer Protocol (HTTP)

HTTP (HyperText Transfer Protocol) is the protocol that outlines the rules for web browsers to request and retrieve documents, and for web servers to deliver those documents. It serves as the communication standard between web servers and clients, enabling the transfer of data over the web. HTTP was first used in 1990 with version 0.9. Version 0.9 was a protocol that sent documents without filtering based on document type. Later, version 1.0 was developed to send documents based on their type. In 1999, version 1.1 was developed to accommodate caching, proxies, and persistent connections [6].

3.3 HTML

HTML is the language used to create a website. Web users can open and read collections of documents that consist of information from their own computer or a remote computer. Users can access other pages on a website by clicking on specific sections of the webpage. The process starts with a web client request, which is received and processed by the web server, and then transparently distributed back to the web client [7].

3.4 Hypertext Preprocessor (PHP)

Hypertext Preprocessor (PHP) is a programming language that runs on a web server and functions to process data on the server. Data sent by the client user is processed and stored in the web server's database and can be displayed again when accessed. To run PHP program codes, the files must be uploaded to the server. Uploading is the process of transferring data or files from the client’s computer to the web server [8].

3.5 Laravel

Laravel is a widely used PHP framework for developing web applications, ranging from small-scale to large-scale projects. Known for its strong performance, rich features, and scalability, it is favored by developers across the globe. The framework follows the MVC (Model-View-Controller) architecture, which separates the application's data, business logic, and presentation layer. This structure not only simplifies learning but also accelerates the development of web application prototypes. Additionally, Laravel comes with built-in functionalities like authentication, mailing, routing, session management, and task scheduling. Developers appreciate Laravel for its flexibility, enabling them to customize project structures to fit specific web application requirements [9].

3.6 Waterfall Method

The Waterfall method, also known as the classical life cycle, is formally called the "Linear Sequential Model." It describes a structured, step-by-step approach to software development. The process begins with the specification of user requirements and then moves through stages such as planning, design, development, and implementation. After the system is delivered to users, ongoing support is provided for the entire software product. This method emphasizes a clear and linear progression through each phase of development.[10].

3.7 Black-Box Testing Method

Black-box testing is a software testing technique that evaluates an application's functionality without considering its internal workings or requiring access to its code or programming. This approach focuses on ensuring the application meets its specifications and requirements. Test cases are generated based on external descriptions such as specifications, requirements, and design documents. While primarily functional, black-box testing can also include non-functional tests. Test designers select both valid and invalid inputs and check if the correct outputs are produced, all without knowledge of the system's internal structure. This method can be applied at any stage of the software testing process [4].

3.8 User Acceptance Testing (UAT)

User Acceptance Testing (UAT) is the final testing phase in the product development process to ensure that the developed system meets user requirements. UAT is typically carried out by the client or end-user. The key purpose of UAT is to prevent errors, ensure the system operates according to the required functionality, and confirm that the final system works properly and has been implemented effectively [11].

4. Results and Discussion

This discussion will explain the results of the system design and program implementation. The feature design will be explained through a feature design scheme, the system design will be illustrated using a context diagram, and the database design will be explained using a conceptual data model (CDM).

4.1 Feature Design

Feature design outlines the key features to be included in the information system, describing their functionality and how they will meet user needs. The planned feature design for the development of the "Campoga" Sports Field Rental Information System based on a website, here is the image in question.

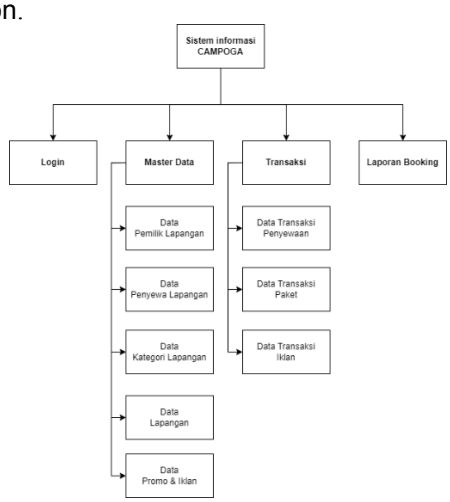

Figure 2. Feature design

Figure 3.4 is the feature design diagram for the research on the development of the "Campoga" Sports Field Rental Information System based on a website. The explanation of the Feature Design above is as follows.

a. Login: This feature is used by Admin or Users to log into the Campoga website..
b. Field Owner Data: This is a list of field owners who have registered and made payments for their accounts.
c. Field Tenant Data: This is a list of field renters who have registered on the Campoga website.
d. Field Category Data: This feature is used by the system admin to add field categories to the Campoga website.
e. Field Data: This feature is used by field owners to add their fields to the system. Information to be inputted includes Field Category, Facilities, Schedule, and Rental Price.
f. Promo and Ad Data: This feature is used by field owners to add desired promotions and advertisements.
g. Rental Transaction Data: This feature is used to view and confirm transactions made by users.
h. Package Transaction Data: This feature is used for package transactions made by field owners, which are then confirmed by the system admin.
i. Ad Transaction Data: This feature is used for ad transactions made by field owners, which are then confirmed by the system admin.
j. Booking Report: This feature can be used by the Admin and field owners to view and confirm completed transactions.

4.2 Context Diagram

The context diagram is designed to show the overall boundaries of a system in relation to its surrounding environment. Below is the context diagram for the research on the "Campoga" Sports Field Rental Information System.

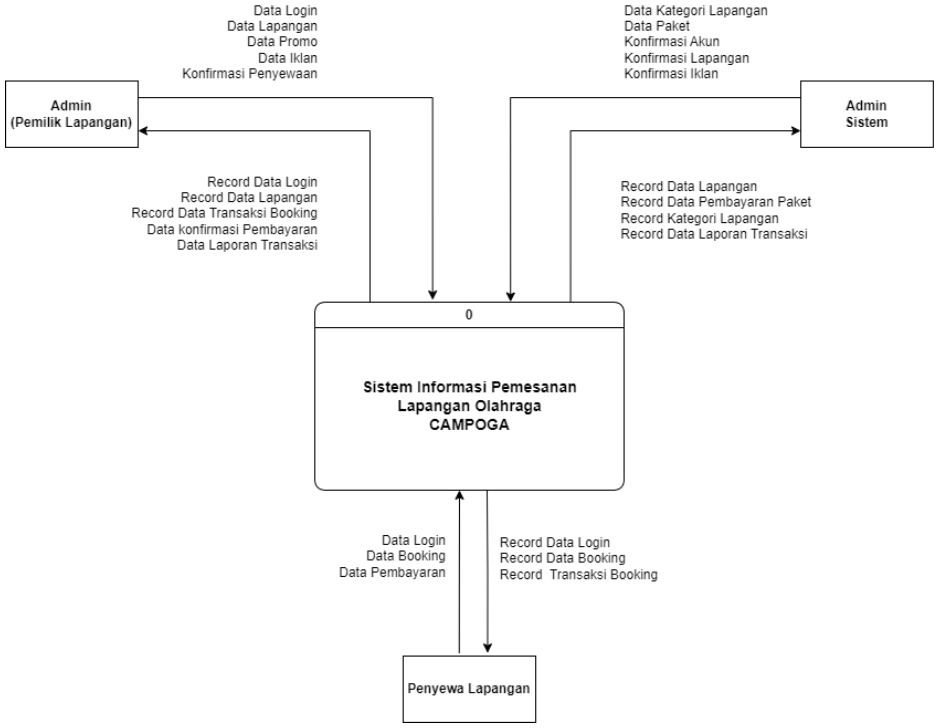

Figure 3. Context Diagram

Figure 3 is the context diagram for the "Campoga" Sports Field Rental Information System research. In this context diagram, there are 3 entities: the system admin, field admin, and renter. Further explanation of this context diagram is as follows.

a. System Admin can input and generate Account Data, Field Data, Payment Data, Field Master Package Data, and Field Category Data. The system admin can also access User Master Data, Master Field List Data, Transaction Report Data, and User Report Data.
b. Field Admin can input and generate Login Data, Field Data, Schedule Data, Booking Transaction Data, and Payment Confirmation Data. The field admin can also access Transaction Report Data.
c. Field Renters can input and generate Login Data, Booking Data, and Payment Data. Field renters can also access Booking Transaction Data.

4.3 Physical Data Model (PDM)

Physical Data Model (PDM) is a data model that contains tables for each entity along with the relationships between the tables. The PDM serves to store data related to various elements within the website[12]. Below is the PDM of the Campoga information system..

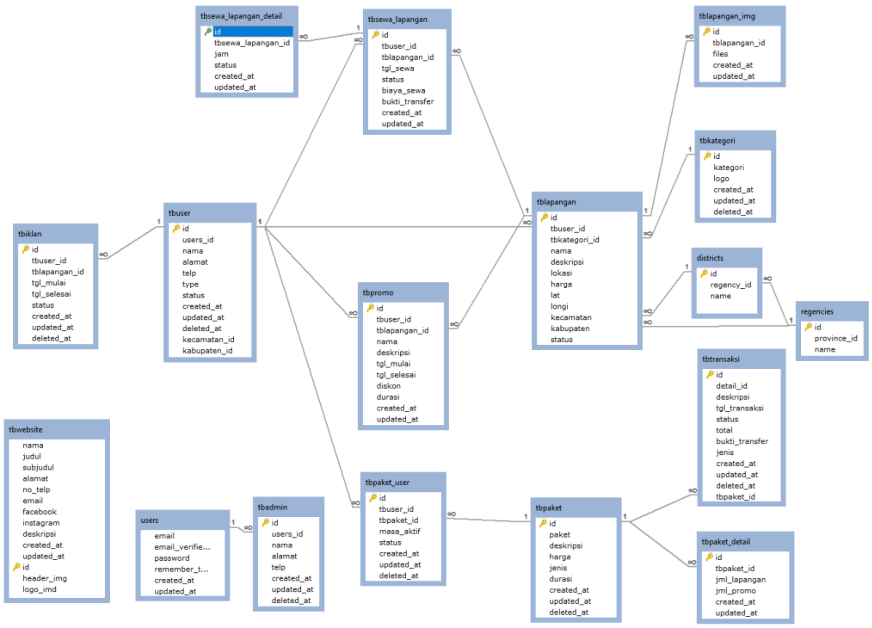

Figure 4. Physical Data model (PDM)

Figure 4 shows the the Campoga Physical Data Model, which includes seventeen tables. Twelve of these tables are interconnected, two pairs of tables are related, and one table stands alone. This PDM serves as a representation of the data flow within the system.

4.4 System Implementation

This system is developed using the Laravel framework to build a sports field rental website. Laravel is used to build the user interface with Blade features and manage the back-end components, such as views, routing, and controllers. The programming language used is PHP, while the data is stored using MySQL as the Database Management System.

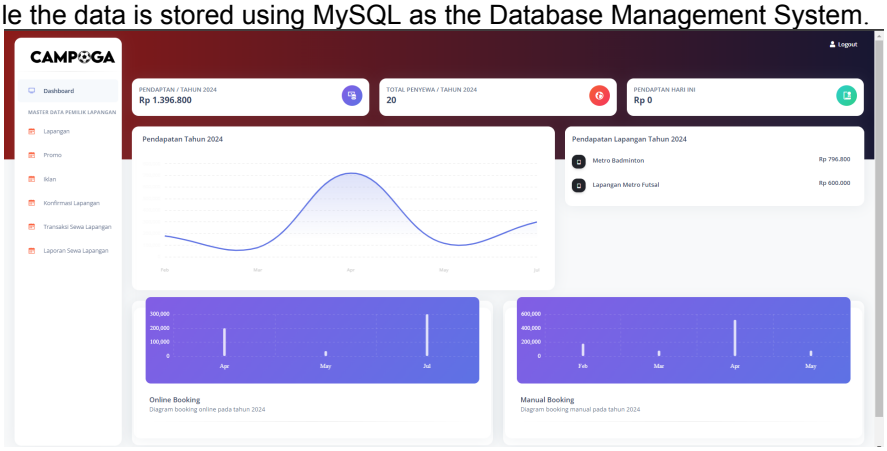

Figure 5. the Dashboard page for Field Owners

Figure 5 shows the dashboard page for Field Owners. On this page, there are several statistics such as annual revenue, total renters, today's revenue, and revenue charts. The sidebar menu contains features that can be used by field owners, including field management, promotions, advertisements, field confirmations, field rental transactions, and rental reports.

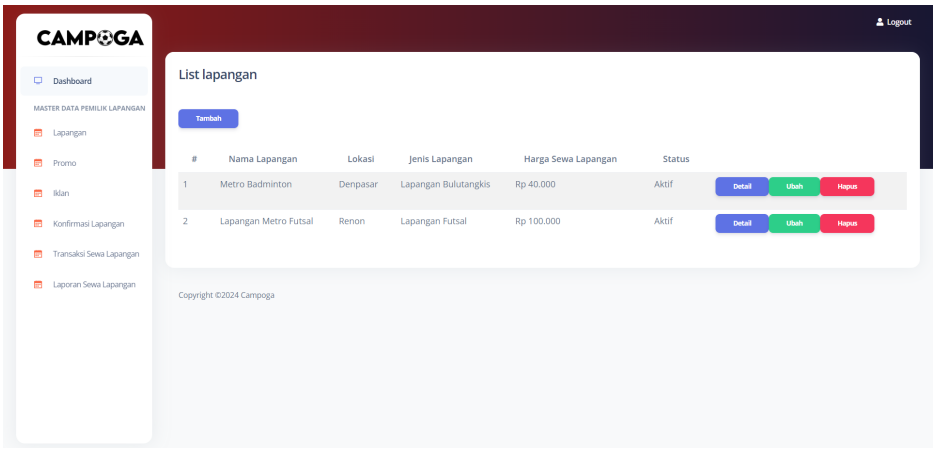

Figure 6. page listing all fields owned

Figure 6 displays the page that lists all the fields owned by the field owner. On this page, the owner can view details, add, edit, and delete field data. Any addition or modification of field details made by the owner must be confirmed by the Campoga website admin.

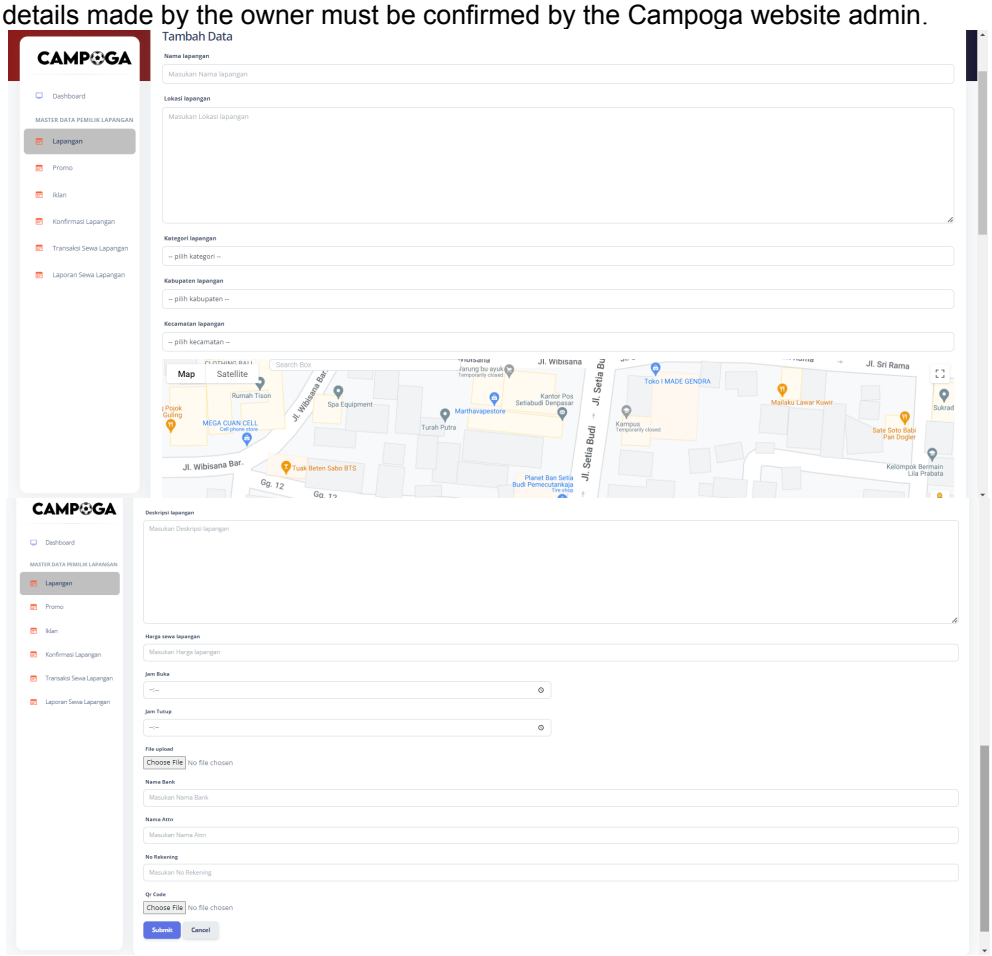

Figure 7. the feature to add a field

Figure 7 shows the feature to add a field by the field owner. On this page, the admin must input data into the available fields, as shown in the image, and then click "submit" to save the data that has been created.

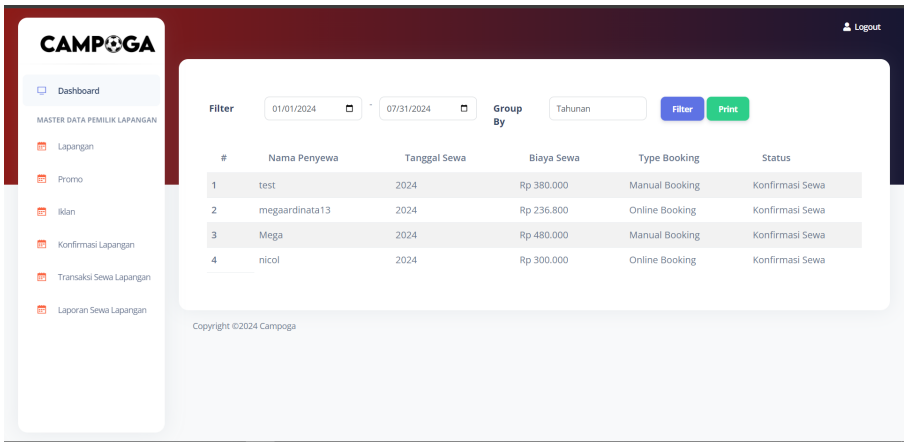

Figure 8. the field rental report menu

Figure 8 shows the field rental report menu. On this page, field owners can view a report in table format containing completed transactions. Field owners can filter the report by day, month, and year using the filter button. They can also print the transaction report by clicking the "print" button.

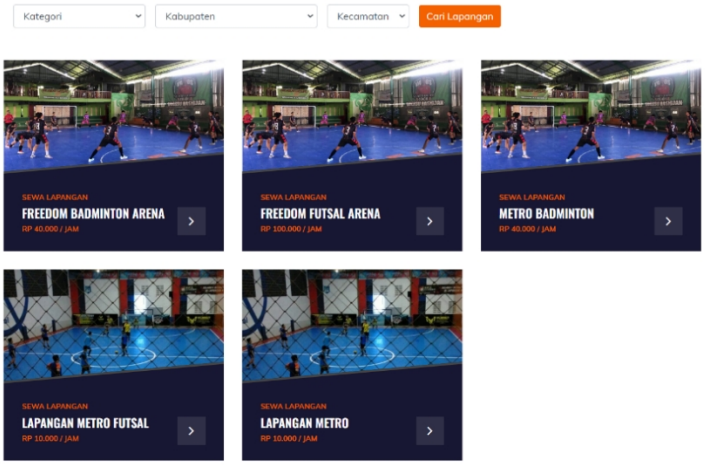

Figure 9. searching for a field

Figure 9 shows the initial view for users when searching for a field. At the top of the website, there is a filter to assist users in selecting a field based on its category or location.

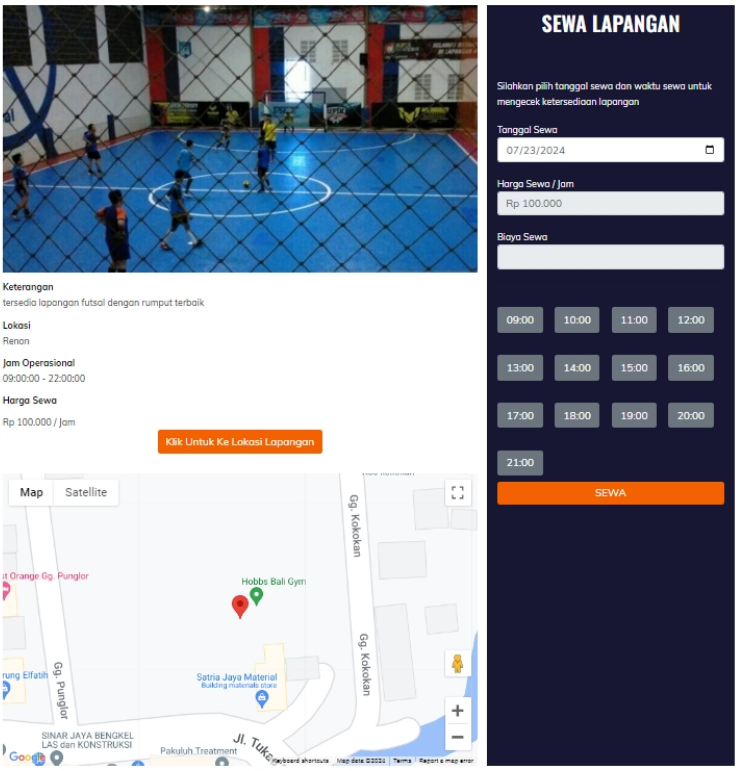

Figure 10. the user selects the rental time and date

Figure 10 shows the page where the user selects the rental time and date. There is a map to guide users to the selected field's location by clicking "click to go to the field location." Once the desired time and date are selected, the renter can click the "rent" button.

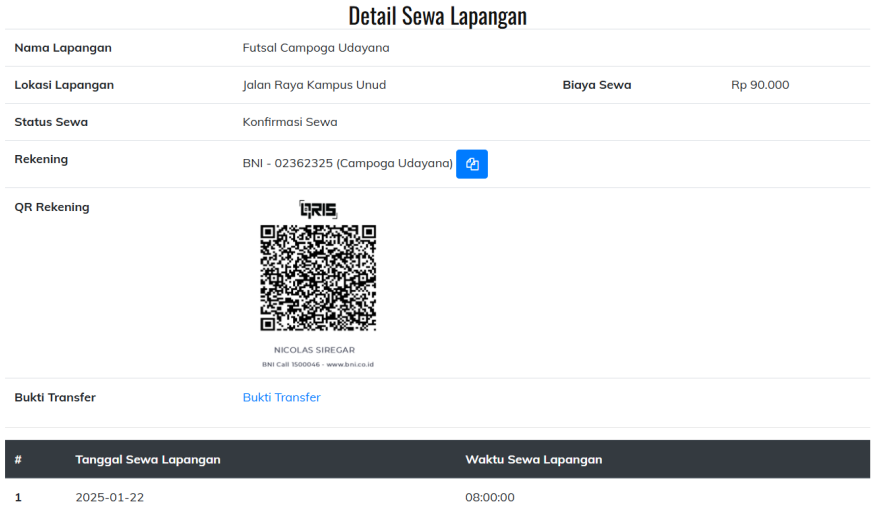

Figure 11. the renter has made the payment

Figure 11 shows the page when the renter has made the payment, which has been confirmed by the field owner. The rental status will change to "rental confirmed," indicating that the field rental transaction has been confirmed by the field owner.

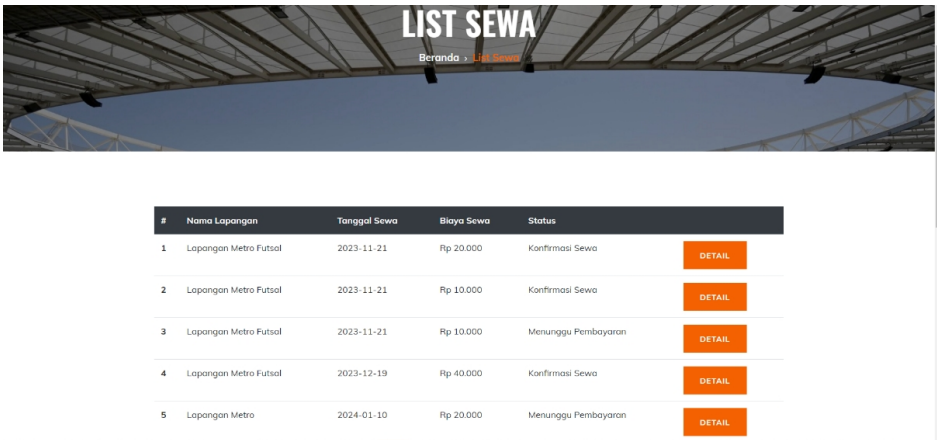

Figure 12. listing rental transactions

Figure 12 shows the page listing rental transactions that have been made by the renter. This page contains the transaction history, with a "details" button that shows the details of the transactions made by the field renter.

4.5 System Testing

Black box testing is used to evaluate the functionality of the website. For the testing scenarios of the Campoga website, tests were conducted by visiting four different field locations and distributing questionnaires to four field owners and 15 field renters. Below are the results of the Campoga website testing.

Tabel 1 Results of the Black Box Testing method

Test Scenario	Expected Result	Test Result	Conclusion
For Customers Website			
Logging in with the condition that all fields are filled in and the correct password is entered.	The system grants login access and directs the user to the user dashboard page.	The system successfully logged in and directed the user to the dashboard page.	Success
Registering with the condition that all fields are filled and meet the required criteria.	The system grants registration access and directs the user to the user dashboard.	The system successfully registered and directed the user to the user dashboard.	Success
Searching for a field based on name, category, or location.	The system will search for the field based on the name, category, or location provided.	The system successfully performed the search as requested by the user.	Success
Making a reservation for a field and selecting an available time slot.	The system will process the booking according to the user's choice and then redirect to the reservation details page.	The system successfully accepted the reservation and directed the user to the reservation details page.	Success
The user did not make a payment within the given time limit.	The system will cancel the reservation if the user does not complete the payment within the specified time.	The system successfully canceled the reservation due to non-payment within the time limit.	Success
Checking the reservation status using email for manual booking.	The system will send the reservation status via email.	The system successfully sent the reservation status via email.	Success
For Field Owners' Website			
Logging in with the	The system grants login	The system successfully	Success

condition that all fields are filled in and the correct password is entered.	access and directs the user to the field owner's dashboard page.	logged in and directed the user to the field owner's dashboard page.	
Selecting an available subscription package.	The system accepts the chosen subscription package and directs to the registration page.	The system successfully accepted the chosen subscription package and directed to the registration page.	Success
Adding field, advertisement, and promo data with all required fields filled in.	The system grants access to add fields, advertisements, and promotions.	The system successfully added the field, advertisement, and promo data with all required fields filled in.	Success
Adding a manual order by entering the renter's data, ensuring all fields are filled.	The system will accept the order with the renter's data entered in the available fields.	The system successfully added the order with the renter's data as entered.	Success
Printing the displayed report.	The system accepts the command to print the report in PDF format.	The system successfully printed the report in PDF format.	Success
For Admin Website			
Logging in with the condition that all fields are filled in and the correct password is entered.	The system grants login access and directs the user to the admin website dashboard.	The system successfully logged in and directed the user to the admin website dashboard.	Success
Adding field categories and packages for the fields.	The system accepts the input for adding field categories and packages.	The system successfully added the field categories and packages as entered.	Success
Confirming the registration of fields, packages, and advertisements added by field owners.	The system grants confirmation access and activates the fields, packages, and advertisements.	The system successfully confirmed the registration and activated the fields, packages, and advertisements.	Success

Table 1 displays the results of the Black Box Testing method. Based on the results for users, field owners, and the admin website, all the links were able to correctly direct users to the appropriate pages. Additionally, all buttons on the website worked as intended. All test cases were successful without any significant issues, indicating that the system meets the expected functionality.

5. Conclusion

The testing of the Campoga sports field reservation system was carried out manually using black-box testing along with User Acceptance Testing (UAT). These methods were employed to gather results from two types of tests: scenario testing and system testing on the Campoga website. The results from the scenario testing showed that all scenarios were executed successfully as expected by the researcher. The available buttons correctly redirected users to the intended pages.

The results of the User Acceptance Testing (UAT) were divided into two groups: field owners and renters. Both field owners and renters received excellent scores. Field owners achieved a score of 88.75% for the interface aspect and 88.6% for the usefulness aspect. Renters, on the other hand, received a score of 90.65% for the interface aspect and 87.2% for the usefulness aspect. Based on these results, the system is considered successful and meets the needs of its users.

References

[1] H. Purwanto, F. A. Nugraha, M. R. Prayogha, and R. M. Syahputra, "Sistem Informasi Penyewaan Lapangan Futsal Berbasis Web," *J. Inform. dan Rekayasa Perangkat Lunak*, vol. 3, no. 2, p. 100, 2021, doi: 10.36499/jinrpl.v3i2.4499.

[2] M. Susilo, "Rancang Bangun Website Toko Online Menggunakan Metode Waterfall," *InfoTekJar (Jurnal Nas. Inform. dan Teknol. Jaringan)*, vol. 2, no. 2, pp. 98–105, 2018, doi: 10.30743/infotekjar.v2i2.171.

[3] V. Y. P. Ardhana, "Perancangan Sistem Informasi Rekam Medis Puskesmas Berbasis UML," *SainsTech Innov. J.*, vol. 4, no. 1, pp. 97–104, 2021, doi: 10.37824/sij.v4i1.2021.302.

[4] D. Febiharsa, I. M. Sudana, and N. Hudallah, "Uji Fungsionalitas (Blackbox Testing) Sistem Informasi Lembaga Sertifikasi Profesi (SILSP) Batik dengan AppPerfect Web Test dan Uji Pengguna," *Joined J. (Journal Informatics Educ.*, vol. 1, no. 2, p. 117, 2018, doi: 10.31331/joined.v1i2.752.

[5] D. R. Gemina, *Perancangan User Interface Situs Web E-Letter UIN Jakarta Menggunakan Metode Five Planes*, vol. 2, no. 1. 2020.

[6] A. S. Nurjaman and V. Yasin, "Konsep Desain Aplikasi Sistem Manajemen Kepegawaian Berbasis Web Pada Pt. Bintang Komunikasi Utama (Application design concept of web-based staffing management system at PT Bintang Komunikasi Utama)," *J. Inf. Syst. Informatics Comput.*, vol. 4, no. 2, p. 143, 2020, doi: 10.52362/jisicom.v4i2.363.

[7] S. Mariko, "Aplikasi website berbasis HTML dan JavaScript untuk menyelesaikan fungsi integral pada mata kuliah kalkulus," *J. Inov. Teknol. Pendidik.*, vol. 6, no. 1, pp. 80–91, 2019, doi: 10.21831/jitp.v6i1.22280.

[8] A. Mubarak, "Rancang Bangun Aplikasi Web Sekolah Menggunakan Uml (Unified Modeling Language) Dan Bahasa Pemrograman Php (Php Hypertext Preprocessor) Berorientasi Objek," *JIKO (Jurnal Inform. dan Komputer)*, vol. 2, no. 1, pp. 19–25, 2019, doi: 10.33387/jiko.v2i1.1052.

[9] D. Mediana and A. I. Nurhidayat, "Rancang Bangun Aplikasi Helpdesk (A-Desk) Berbasis Web Menggunakan Framework Laravel (Studi Kasus di PDAM Surya Sembada Kota Surabaya)," *J. Manaj. Inform.*, vol. 8, no. 2, pp. 75–81, 2018, [Online]. Available: http://ejournal.ukrida.ac.id/ojs/index.php/TIK/article/view/1495/1617.

[10] A. A. Wahid, "Analisis Metode Waterfall Untuk Pengembangan Sistem Informasi," *J. Ilmu-ilmu Inform. dan Manaj. STMIK*, no. November, pp. 1–5, 2020.

[11] A. A. Surya and I. Haromain, "Rancang Bangun Website Lelang Mobil menggunakan Framework Codeigniter 3 pada PT.ABC," *J. Teknol. Terpadu*, vol. 9, no. 2, pp. 133–142, 2023, doi: 10.54914/jtt.v9i2.1031.

[12] I. R. Mukhlis and R. Santoso, "Perancangan Basis Data Perpustakaan Universitas Menggunakan MySQL dengan Physical Data Model dan Entity Relationship Diagram," *J. Technol. Informatics*, vol. 4, no. 2, pp. 81–87, 2023, doi: 10.37802/joti.v4i2.330.