Container Infrastructure on Laravel to Improve
Online Shop

Sucianna Ghadati Rabiha 1, Achmad Fadhitya Muharram 2, Fahmi Anwar Kusuma 2, Hendrawan Sulistyanto 2, Eduard Pangestu Wonohardjo 2, Emny Harna Yossy 2,*

* Corespondence Author:  e-mail: emny.yossy@binus.ac.id

1 Information Systems Department; Binus Online Learning; Bina Nusantara University; Jl. Kyai H. Syahdan No.9, Jakarta, Indonesia; (021) 534 5830; e-mail: sucianna.rabiha@binus.ac.id
2 Computer Science Department; Binus Online Learning; Bina Nusantara University; Jl. Kyai H. Syahdan No.9, Jakarta, Indonesia; (021) 534 5830; e-mail: achmad.muharram@binus.ac.id, fahmi.kusuma@binus.ac.id, hendrawan.sulistyanto@binus.ac.id, eduard.wonohardjo@binus.ac.id, emny.yossy@binus.ac.id

Submitted : 31/07/2023
Revised : 06/08/2023
Accepted : 13/08/2023
Published : 30/09/2023

Abstract

A laundry business is a business that operates laundry services or similar and is often used by the public. The development of this business is quite fast, as many consider renting laundry to be more convenient. This rapid increase is inversely proportional to the evolution of companies in using technology as a means of digitizing their businesses. The digitization process can make it easier for business owners to manage, monitor, promote, and receive business reviews. One attempt to overcome this is to develop a web-based laundry management and publishing application using the Laravel framework and container infrastructure, with the aim of making business management and advertising easier for laundry entrepreneurs and customers to choose from to facilitate washing area. This study uses data collection techniques in the form of literature reviews, questionnaires and uses technologies to facilitate the development process such as the Laravel framework, MySQL, Google Maps API, and container and waterfall infrastructure as application development methods. The test technique used is black box. The result of the research is a web-based laundry application that conforms to the eight golden rules, has been tested by laundry owners and customers, and can achieve the research goals.

Keywords: laundry, waterfall, laravel, container, black box testing

1. Introduction

In the business world, the positive effects of information technology are not only felt in large companies. Small and medium-sized businesses managed through the use of information technology will be able to assist business actors in carrying out all their business operations. As well as being used for transaction management, promoting the business, tracking the place of business to providing a valuation of the business. In addition, having an information system can reduce human error (Adenowo & Adenowo, 2013). The application of this business strategy can be used by business people in the service sector, such as B. laundries, are used. A laundry business is a business that operates laundry services or similar and is often used by the general public. Due to the
limited time they have, they have to represent the needs of laundry in this business (Firmansyah Adiputra, 2015).

The development of the laundry business in terms of sales and the opening of new businesses must be accompanied by proper management and publication/advertising. Kotler states that promotional efforts are marketing efforts that offer various short-term intense efforts to create a desire to try or buy a product or service (Kotler, 2000). A company management that still works according to the manual method will certainly make it difficult for entrepreneurs to monitor the development of their company. The need for service reviews is also very important for customers, especially to determine whether or not they use the service. If reviews are still done manually, it becomes difficult to attract the attention of new customers.

Several previous authors have made several products and research; examples of products published are the D-Laundry and Justclean applications. This application can connect customers and laundry businesses based on mobile and web applications for customers (D-Laundry, 2023).

In addition, the study with the background of someone who needs certain laundry services and has difficulty obtaining information on the existence of laundry services around him because the search process still relies on information from people around him, newspapers, and the internet. This research aims to create a mobile-based marketplace to help bring together many laundry service providers and consumers. The expected results of the application are in the form of an application that can store service provider and customer data, the application can display the location of the nearest laundry, the user can order services in the application, the user can confirm payments, the user gets notifications when the order is complete, and the user can view order history. (Angrijaya, 2019).

The following research aims to build a website-based laundry application to facilitate laundry businesses in managing their laundry transactions. The result to be achieved by this research is the existence of a website-based laundry management system or application. The suggestion for developing this research is to carry out a maintenance process at a later date to add more up-to-date features to compete with other laundry services and always back up data so as not to experience problems that affect the performance of the website (Hendrawan et al., 2020).
Based on the above case, the author took the initiative to create a web-based laundry management and publishing application using Laravel framework and container infrastructure, with the aim to enable laundry entrepreneurs to manage and publish/promote their business and their customers make it easier to find the nearest and best laundry. In this study, the authors used data collection techniques in the form of literature reviews and questionnaires. This application also uses multiple technologies to simplify the build and development process, such as Laravel framework, MySQL, Google Map API, and Container Infrastructure, and uses waterfall as the application development method. The testing technique used by the author is the black box testing technique.

2. Research Method

There are several related studies that we tried to investigate and analyze, namely Making a mobile-based Online Laundry Service Marketplace. The self-test of this system function found that all functions work well, and direct tests by 20 respondents concluded that the laundry service marketplace application has managed to provide users of suitable laundry services and future works such as providing chat and developing of applications on the IOS platform and including maps if you indicate the location of the laundry during registration (Angrijayya, 2019). Laundry management system design with the waterfall method using observation, interviews and analysis as data collection method, waterfall as application development method, black box testing as application testing method has research results in the form of database implementation using Mysql. (Syaputra et al., 2021).

Website-based laundry management system application that uses observation and interviews as data collection methods, extreme programming as application development methods, black box testing as application testing methods, and has research results in the form of laundry applications that can be used to manage customer data, laundry packages, record laundry transaction data, create Annual reports and can facilitate employees to digitize their business and have future work in the form of adding an automatic wash load calculation function, a payment amount function and a discount function for regular customers (Hendrawan et al., 2020).

Building an Android-based on-demand laundry ordering marketplace application using questionnaire as data collection method, prototype as application development method, black box testing as application testing method has research results in the form of test results from 9 (nine) questions of the questionnaire to 10 (ten) It can be concluded that more than 77.33% of the respondents answered that the Android-based...
on-demand laundry ordering marketplace application is considered good and has future tasks such as adding features to other alternative payment methods. Methods to make it easier for customers to pay for services (Tan, 2019).

The design of the information system for the entry and exit of goods in laundries, using literature and observation as a data collection method, waterfall as an application development method, black box testing as an application testing method, has research results in the form of application advantages. Storage of data already has a database. Implementation of the Waterfall method and data validation before data entry and application deficiencies such as lack of data backup on the server, application is only for one administrator, data is not well coordinated and storage is easily attacked by viruses because it is too sensitive (Amelyah, 2019).

Based on the research above, we use the waterfall method because the authors believe that this method can ease the implementation process. Apart from that, technologically it can be seen that the use of the MySQL (Kromann, 2018) database and web-based development or the use of HTML and CSS (Sari et al., 2022) and the Laravel framework (Stauffer, 2019) is quite widespread, this is also done by the author, but the author uses additional auxiliary tools, namely the Laravel framework to simplify and speed up the process.

Figure 1 describes the phases of the research carried out. The first phase of research is the study of literature. In this phase, the author collects related research and applications to find information relevant to the problem under study. The next phase is data collection. In this phase, the author collects data using a digital questionnaire aimed at laundry customers and laundry business owners. Then the application development phase, in this phase the author uses the waterfall method. This method consists of 5 phases for system development, namely requirements analysis, design, implementation, test and operation and maintenance (Adenowo & Adenowo, 2013). Then perform the results analysis phase. At this stage, the author analyzes the results of the research using the principles of the Eight Golden Rules (Wong, 2020) and the blackbox testing method (Jaya, 2018). The final stage is conclusion. At this stage, the author draws conclusions about the results of the research based on the objectives and the formulation of the problem.
To solve the problems described earlier in Introduction, we provide a solution by creating a web-based publishing/promotion and management application for laundry using Laravel framework and container infrastructure. This system aims to make it easier for laundry entrepreneurs to digitally develop their business. On the other hand, it makes it easier for potential customers to find and use laundry services. The solution was obtained after we carried out an analysis by comparing similar applications and comparing the results of the questionnaires distributed to laundry owners and customers. This questionnaire aims to obtain data on the needs of users and owners of laundry facilities when using our application.

In a comparative analysis of similar applications, we found several functions that were not too complete to meet the solutions. Based on the data collection through the questionnaire received, there were then several features required by users and laundry owners. By comparing similar applications and the questionnaire, a web-based laundry publishing and management application that uses the Laravel framework and container infrastructure must have several key features that can support the application's usability.

The function that we offer as a suggestion to solve the above problem is the function of finding laundries based on distance to make it easier for customers to find the nearest laundromat, the function of detailed laundry information about the services offered, view, the function of viewing reviews and ratings of laundries provided by other users to help customers find the best laundry, functions provide reviews and ratings of laundries based on the services provided, functions check entered or orders the work process. There are 3 categories including incoming orders, processing and done but delivered, features to manage income and expenses data to make it easier for laundry owners to find them in the form of reports, features to manage customer data for laundries and features to manage employee data for optimization employee administration for additions, reductions, or remuneration of employees.
3. Results and Analysis

3.1. Data Collection

The data collection process was carried out by distributing questionnaires aimed at the general public to collect data to serve as a reference for functions developed for the public user side or laundry customers, and a questionnaire also aimed at laundry owners, which has been used as a reference for features used by laundry owners. The total questionnaires completed were 16 questionnaires from the laundry owner and 57 questionnaires from the general public.

3.2. Software Description

The system created is a laundry market application to collect data from each laundry that has registered with the system. The system can be used by two parties, either the laundry customer or the laundry owner. The features offered to customers and laundry owners are different. Features for customers include viewing a list of all laundries, viewing details of the laundry along with a list of each package provided, and customers can also view and submit reviews for each laundry. In addition to functions for laundry owners in the form of financial management for their own laundry, every laundry can also view customer ratings and evaluate these ratings.

Web-based software designed to run in a browser so that users do not have to install it and this software can be opened from a computer or a user's smartphone. The software uses a container infrastructure to facilitate the scaling process if necessary to increase the scalability of the software. Using the container infrastructure, there is not much configuration to do when scaling.

3.3. Software Design

3.3.1. Use Case Diagram

The use case diagram design in our system has four actors and 34 use cases. The actors in the diagram consist of Administrator, Owner, Employees and Customers. Each actor has a different use case, e.g., B. Administrator has logged out, approved, and denied verification and registration of the user account. The owner must opt-out, manage service records, employees, customers, income, expenses, account registration, and change account profiles. Employees can log out, manage data, services, customers, income, and expenses, and see ratings. While customers need to view public reviews, laundromat information, list laundries, review orders, contact stores, and leave reviews. Our chart also has use cases with additional use cases that actors can run, e.g., B. Use cases that manage service data. Actors who have access to these use cases have the right to view, add, change, and delete service data. This
also applies to the use case of managing customer and employee data. In addition, there are several additional use cases, e.g., B. managing expense categories and changing store opening and closing dates.

3.3.2. Activity Diagram

The activity diagram design we have created has been created for each actor and is described based on the functions that can be used by each actor, for actors in the form of owners of laundry activity diagrams that are described are login, logout functions, Account management data, manage service data, manage data input, manage data output, manage employee data, manage customer data. For those who have the admin role, we describe the activity diagram on the Reject New User, Accept New User, Reject Rating, Accept Rating functions. And the last role is a public user or public user. The functions described in the activity diagram are Leave Reviews, Review Orders, Contact Laundries, View Lists of Laundries, View Laundries Details, View Reviews.

3.3.3. Class Diagram

In the class diagram we created, there are several classes including MapService, DashboardController, ExpenseController, EmployeeController, ServiceController, AdminController, Expense, Customer, OrderDetail, User, CategoriesController, Service, Order, ExpenseCategory, ReviewController, Review, and WorkTimes. The use of this class diagram is not only used to represent existing classes in the application but is also used to describe the dependencies and relationships of each class, namely DashboardController has a relationship to Expense, OrderDetail, Order and Review classes. The AdminController class has a relationship to the User, Order, and Review classes. The ServiceController class has a relationship to the Service and OrderDetail classes, the ExpenseController class has a relationship to the Expense and ExpenseCategory classes. The CategoriesController class has a relationship to the ExpenseCategory class. And the ReviewController class has a relation to the Review class.

3.3.4. Sequence Diagram

We created a sequence diagram to represent the flow that starts from the user side until it is processed by the system and then the result is received from the user, thus showing the flow of data provided and processed in the system. The sequence diagrams we create include sequence diagrams for the login and logout process, the process of checking laundry orders, the process of finding the closest and best laundry, and management processes that can be used by laundry users. In addition, we also
describe a sequence diagram to show the approval process for registering laundries in the system and approving customer reviews for specific laundries.

3.3.5. Architecture Diagram

Fig. 2 is an architecture diagram that illustrates the application we developed by implementing containers in which multiple containers are created so that load balancing is shared using a load balancer server and application downtime is minimized.

![Architecture Diagram](image)

Source: Research Result (2023)

Figure 2. Architecture Diagram

3.4. Software Design

Below is an explanation and display of each page in the application (see the result in figure 3):

1. Login page display: The login view is used to enter the application. This view is accessible to employees and business owners. This feature does not apply if the user is a customer, as customers can access their features without having to enter the application.

2. Displaying the Owner Dashboard Page: The owner dashboard to monitor deal progress. This view shows some data like income, expenses, total orders, ratings, and preferred services.

3. Income Page Display: The display is used for business income management. This page is accessible to owners and employees.

4. Expense Page Display: The display is used for business expense management. This page is accessible to owners and employees.
5. Public home page display: The feature is useful to display a list of laundries registered in the application, which can be used by users acting as customers.

6. View order check page: This function is useful to monitor laundry orders; in this function, customers can know their orders in 3 stages, namely incoming, processed, finished, and already picked up.

Source: Research Result (2023)

Figure 3. Design Result

3.5. System Evaluation Result

While developing a web-based laundry management and publishing application using the Laravel framework and container infrastructure, we use the eight golden rules principles as a fundamental reference for good application design. The results of the user interface evaluation are as follows: a) Strive for Consistency: App design implements consistent colors, fonts, layout, and button placement. The primary colors
used in this application are blue, green, and white. Meanwhile, we use the same font on each application page, and the placement and buttons in our application are in the same location. b) Seek Universal Usability: Applied for novice users. When entering an email to log in, an example email template can use to enter the application. In addition, the application has provided buttons and writing that are large enough so that users of all ages can see them. For example, the homepage has buttons and descriptions for features that are large enough and clear. c) Offer Informative Feedback: Feedback on every action that the user performs, such as in the display of a dialog box to display detailed order information in the application. d) Design Dialogs to Yield Closure: When saving the data, a dialog box will appear in the display containing information that the data has added. e) Prevent Errors: This application provides a message when the user wants to enter data in a view. For example, when a user wants to add income data, explanatory information is provided for each field. f) Permit Easy Reversal of Actions: In this application, if there is an error in entering data. A dialog box will appear to confirm whether the data will be changed. g) Keep Users in Control: This application has a dashboard page that displays a navigation page that provides information so that users can use the desired features according to their wishes. H) Reduce Short-term Memory Load: In the application, a list box will select the service models available at the laundry shop.

Based on the evaluation of the user interface that has been carried out, it is concluded that the Web-Based Laundry Shop Management and Publication Application Using the Laravel Framework and Container Infrastructure meets the requirements needed as a basic reference in considering a good application design according to the Eight Golden Rules principles.

The evaluation of this system takes the form of tests related to application functions developed on the basis of the test cases, together with the results of the tests carried out. Functional tests are carried out by 2 parties, namely the person who owns the laundry and also the party of public users or customers of the laundry. Functions to support user requirements from the perspective of users of laundry services can be met, namely, with the function of searching for the nearest and best laundry, a function of checking the status of laundry, and a function of displaying details of each laundry. Functions to support user needs on the part of the laundry owner can be fulfilled, namely by performing a laundry business management function by the laundry owner, such as: Financial management, customer management, employee management, service management. In this way, from the results of the evaluation performed with the
user, both the laundry owner user and the public user as the laundry service user, it can be determined that the application provided meets the user's needs and can help users.

4. Conclusion

Based on the results of our application development, the application we created uses the Laravel framework and implements a container infrastructure. Using this container infrastructure also improves application performance when processing many requests coming to the server. In addition, it can also be concluded that our application conforms to the principles we use, such as the eight golden rules. These include consistency of appropriate design, information that makes it easier for new users to use the site, the ability to display trigger information as user feedback, displaying messages after the user has taken an action on the web, displaying explanatory Information in fields, display of the cancel function of an action, navigation that appears when the user wants to perform an action and lastly shows the list box menu on a view.

In addition, our application has also been tested directly on the user according to the black box method, so that the user can directly check whether the developed functions meet the user's needs. Suggestions for further system development are to develop applications for the Android or iOS platforms and provide relevant additional functions to make them more complete.

Acknowledgements

Thank you to Bina Nusantara University who has imparted valuable knowledge on research.

Author Contributions

The authors took the initiative to create a web-based laundry management and publishing application using Laravel framework and container infrastructure.

Conflicts of Interest

The author declare no conflict of interest.

References


