Implementation of *Cloud Computing on Website-* Based Car Rental System at Dampit Trans Solo

Tegar Danardana Lokananta [1], Ihsan Cahyo Utomo [2]
Faculty of Communication and Informatics,Information Technology
Muhammadiyah University of Surakarta
Central Java, Indonesia
e-mail: [1] tegardanardana991@gmail.com, [2] icu886@ums.ac.id

Revised: March, 08, 2025; Accepted: May, 04,2025

Abstract

Website applications must have a scalable, flexible, and reliable server infrastructure, in order to facilitate system development from design to distribution to users. With its various superior services, such as Google App Engine (GAE), Google Cloud SQL, and Google Cloud Storage, Google Cloud is one of the cloud computing service providers that has succeeded in providing solutions to these infrastructure needs. The case study presented in this study is the development of a website-based car rental information and management system at the Dampit Trans Solo company, with a server infrastructure that implements cloud computing. This system was developed to provide solutions for the company and Dampit Trans customers, by simplifying the company's business processes and providing a cost-effective website infrastructure. System quality testing was carried out using three methods, namely black box testing, System Usability Scale (SUS), and access speed evaluation. From the black box test, it was shown that the system was appropriate and had no errors in terms of functionality. The SUS test produced an average value of 80.04 which interpreted the system as being able to be used in real cases with a "good" assessment range. In addition, the system has excellent website access speed performance after being tested using Gtmetrix and Google PageSpeed Insights, which displays an average time to load a website page of 1.8 seconds with a grade of A, and website performance with a grade of 95.

Keywords : *cloud computing, google cloud, car rental, information systems, website*

Abstract

A website application must have a scalable, flexible, and reliable server infrastructure to facilitate the development of a system from design to distribution to users. With its various outstanding services, such as Google App Engine (GAE), Google Cloud SQL, and Google Cloud Storage, Google Cloud has become one of the cloud computing service providers that successfully provides solutions to these infrastructure needs. The case study presented in this research is the development of a website-based car rental information and management system at Dampit Trans Solo company, using cloud computing infrastructure. This system was developed to provide solutions for Dampit Trans company and customers by simplifying the company's business processes and providing a cost-effective website infrastructure. System quality testing is conducted using three methods: black box testing, System Usability Scale (SUS), and access speed evaluation. The black box testing showed that the system was functional and error-free. The SUS testing resulted in an average score of 80.04, interpreting the system as usable in real-world scenarios with a "good" rating range. Furthermore, the system demonstrated excellent website access speed performance after testing using Gtmetrix and Google PageSpeed Insights, with an average page load time of 1.8 seconds and an A grade, and a website performance score of 95.

Keywords *: cloud computing, car rental, Google Cloud, information system, website.*

1. Introduction

The use of computers in a company for data processing is often referred to as "information systems" or "information technology", and has been widely used for commercial-scale data processing since the 1960s, soon after the introduction of mainframe computers. The paradigm of business programming has undergone a major shift due to the advancement of new technologies. Since the introduction of personal computers in the 1980s, the client-server architecture has significantly replaced the giant mainframes that were previously used. In the 2000s, web-based commercial systems and end-user-focused e-commerce sites replaced the client-server architecture. This happened along with the increasing awareness of how important the internet was at that time [1], [2] .

The results of technological advances in today's life require every individual to have digital skills as a primary competency, and transform towards digitalization in all aspects of life, including business. One of these transformations is the implementation of a website-based information system as a means to facilitate various jobs, due to its advantages in providing information worldwide and flexible in its application. Information Systems are a combination of people, hardware, and services that prioritize service quality [3] .

Dampit Trans Solo located in Jagir, Gentan Sub-district, Baki District, Sukoharjo Regency, Central Java is a business that provides transportation and four-wheeled vehicle rental services, and has been established since the 2010s. During its operation, Dampit Trans Solo still uses a conventional system to receive and record orders from its service users, namely by ordering via the WhatsApp communication service. The method applied also gradually became a problem, when the intensity of the number of orders increased significantly, especially during national holidays. Service providers are overwhelmed in responding to orders, and there are often mis-data collection and scheduling of rental periods. From these obstacles, a system is needed that can provide solutions in the form of effectiveness and efficiency for tenants and service providers by utilizing technological developments that benefit business actors.

Research needs to identify the problems raised, whether there are solutions offered and developed from previous research [4] . Looking at the existing facts, there are actually many car rental systems that have been developed with various technologies. Research [5] produced a car rental system website that was built to provide clearer and more accurate information to customers about car rental products, but did not provide details about the features for ordering cars. The system was developed with PHP, the Yii framework, and MySQL, then concluded that the program developed using this technology provided a relatively low percentage in terms of ease of use, appropriate information, and ease of remembering. The same thing was done in research [6]–[8] , where the car rental system was developed on various platforms including websites and Android mobiles, with almost the same technology, namely PHP, the CodeIgniter framework, and MySQL. The results and objectives of various previous studies are more or less the same, namely a website that can be used to rent cars, and provide information related to service providers to prospective customers.

Cloud Computing is defined as a secure, always available, and on-demand computing access to a set of resources that can be easily configured and managed through an interface [9] . Cloud computing has also become a popular choice when it comes to computing used to design and manage website infrastructure, with one important factor that differentiates it is its flexibility compared to other distributed computing. This flexibility allows the allocation and distribution of resources to be done automatically and dynamically. Google Cloud is one of Google's cloud computing service providers that offers various solutions for creating and managing the infrastructure of various types of digital products, one of which is a website [9], [10] .

Based on the description above, this study will develop a website-based car rental information and management system at the Dampit Trans Solo company, with a server infrastructure that implements cloud computing. This website system will be built with the Javascript programming language, the React JS and Node JS frameworks, Git, and Platform as a Service (PaaS) services from Google Cloud, namely Google App Engine (GAE) and Google Cloud SQL. Javascript was chosen as the main programming language because it has several advantages such as a simple structure, compatible with various other programming languages, and has a very large community [11] . The results of this study are expected to provide solutions to the company's previous ineffective business processes, as well as the benefits of flexibility and scalability of server operational costs from Google Cloud to the needs of the website system infrastructure being built.

2. Research methods

The method used in developing a website-based car rental system in Dampit Trans Solo uses the System Development Life Cycle (SDLC) method with the waterfall approach model method. The waterfall method is a sequential development model where the process control of each step is sequential like a waterfall [12] . Each step in the waterfall method must be completed before continuing to the next phases in a structured and consistent manner [13] . The use of the SDLC method in this study was chosen because it has a cost-effective and time-efficient nature for developing small-scale software systems. The waterfall method has five sequential flows, including: system requirements analysis, system design, implementation, system testing, and maintenance. The detailed stages of the waterfall method can be seen in Figure 1.

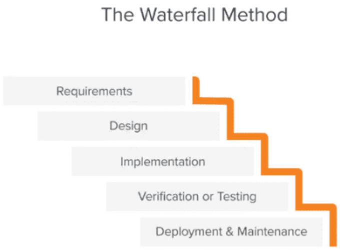


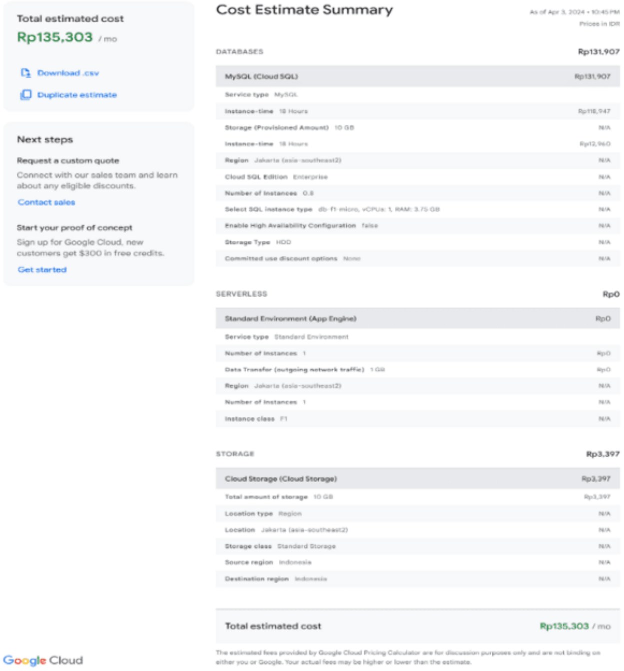
Figure 1. *Waterfall* Method **Stages**

2.1. Needs Analysis

Needs analysis is the initial stage of the application of the waterfall method in this study. In this phase, the design begins with analyzing the needs of the system that will be applied in the form of a website by going through several stages [5] . The first stage of analysis, observations were made on the management and business transactions that occurred at the Dampit Trans Solo company. After the observation, interviews were conducted with sources, especially the owner and several customers of the Dampit Trans Solo service to find out what was needed so that the system could be adjusted and on target. The results of the analysis stages through observation and interviews were realized in the design of use case diagrams, activity diagrams, and entity relationship diagrams.

The second stage of analysis is the analysis of cloud infrastructure needs that are in accordance with the needs of the system being built. Platform as a Service (PaaS) services are services that are suitable for use by the system because PaaS provides a flexible, scalable, easy-to-manage cloud platform environment that can be applied both in development, testing, and production deployment environments (What Is PaaS? | Google Cloud, nd). Google Cloud is one of the largest cloud platforms in the world that guarantees the satisfaction of developers in overcoming cloud infrastructure problems and needs with comprehensive solutions, one of which includes offering modern, secure, smart, and innovative cloud services (Why Google Cloud | Google Cloud, nd).

The final analysis stage is to determine what type of Google Cloud service will be implemented in the final result of the car rental website at the Dampit Trans Solo company. The Google Cloud services used are Google App Engine (GAE) as a fully managed server infrastructure, Google Cloud SQL as a MySQL relational database, and Google Cloud Storage as a system file storage. The three services are designed into a Cloud Architecture Diagram to facilitate understanding of the flow of the three services. The estimated cost of using the three services can be seen in Figure 2.



Total estimated cost
Rp135,303 / mo

Download .csv
Duplicate estimate

Next steps

Request a custom quote
Connect with our sales team and learn about any eligible discounts.
Contact sales

Start your proof of concept
Sign up for Google Cloud, new customers get $300 in free credits.
Get started

Google Cloud

Cost Estimate Summary

As of Apr 3, 2024 • 10:45 PM
Prices in IDR

DATABASES	Rp131,907
MySQL (Cloud SQL)	Rp131,907
Service type MySQL	
Instance-time 18 Hours	Rp118,947
Storage (Provisioned Amount) 10 GB	N/A
Instance-time 18 Hours	Rp12,960
Region Jakarta (asia-southeast2)	N/A
Cloud SQL Edition Enterprise	N/A
Number of Instances 0.8	N/A
Select SQL instance type db-f1-micro, vCPUs: 1, RAM: 3.75 GB	N/A
Enable High Availability Configuration false	N/A
Storage Type HDD	N/A
Committed use discount options None	N/A

SERVERLESS	Rp0
Standard Environment (App Engine)	Rp0
Service type Standard Environment	
Number of Instances 1	Rp0
Data Transfer (outgoing network traffic) 1 GB	Rp0
Region Jakarta (asia-southeast2)	N/A
Number of Instances 1	N/A
Instance class F1	N/A

STORAGE	Rp3,397
Cloud Storage (Cloud Storage)	Rp3,397
Total amount of storage 10 GB	Rp3,397
Location type Region	N/A
Location Jakarta (asia-southeast2)	N/A
Storage class Standard Storage	N/A
Source region Indonesia	N/A
Destination region Indonesia	N/A

Total estimated cost Rp135,303 / mo

The estimated fees provided by Google Cloud Pricing Calculator are for discussion purposes only and are not binding on either you or Google. Your actual fees may be higher or lower than the estimate.

Figure 2 . *Google Cloud Pricing Calculator*

2.2. System Design

1. At the system design stage, the needs analysis that has been done in the previous stage will be developed into a reference for making the implemented design. The resulting design applies the *Unified Modeling Language* (UML) method, which is a standard language for visualization, design, and documentation of systems, which can be briefly referred to as *a blueprint* of a software before it is made [14] . The UML method has several types of diagrams in its application, in this study the development will use *Use Case Diagram, Activity Diagram, Entity Relationship Diagram* (ERD), and *Cloud Architecture Diagram* .

a. *Use Case Diagram*

Detail *Use Case Diagram* is shown in Figure 3 where the system has two actors, namely the admin actor and the user actor. Each actor has different access rights, such as the user actor only has the right to access registration, login/logout, order *dashboard* , information on *the homepage* , and orders. While the admin actor has more access rights towards management of user orders.

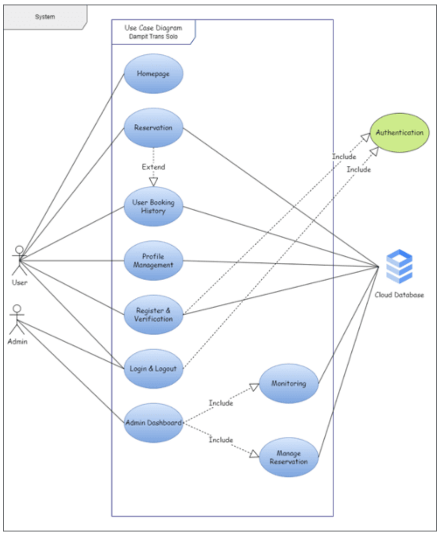


Figure 3. *Use Case* **Diagram**

b. *Activity Diagram*

Activity Diagram is used to create a visualization of the flow of the system process, both from the user and system side so that it is easy to understand by developers [15] . Below Figure 4, an activity diagram image owned by the user and admin on the system being studied will be attached.

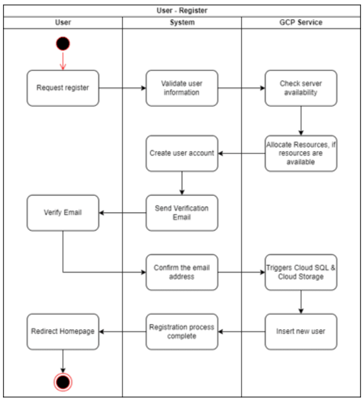
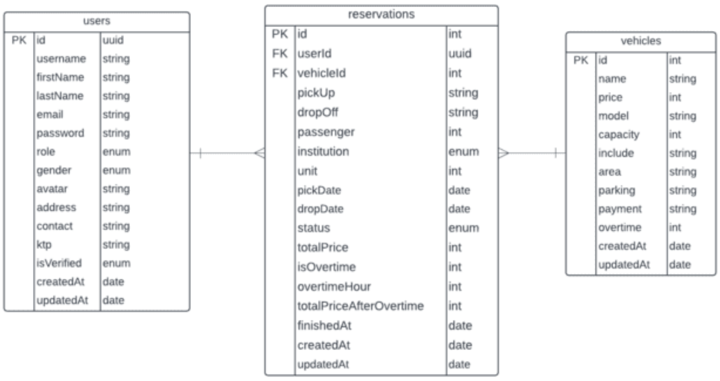

User - Register
User
System
GCP Service
Request register
Validate user information
Check server availability
Allocate Resources, if resources are available
Create user account
Send Verification Email
Verify Email
Confirm the email address
Triggers Cloud SQL & Cloud Storage
Insert new user
Registration process complete
Redirect Homepage

Figure 4 . *Activity Diagram* User Register

c. *Entity Relationship Diagram* (ERD)

ERD is a form of network modeling that uses data structures in abstract form [16] . ERD is used in building a database design that is easy to understand because of the symbols with certain meanings, one of which is knowing the relationship between two or more tables in the system being built. Figure 5 below is a display of the ERD design in the research that was designed.



users		
PK	id	uuid
	username	string
	firstName	string
	lastName	string
	email	string
	password	string
	role	enum
	gender	enum
	avatar	string
	address	string
	contact	string
	ktp	string
	isVerified	enum
	createdAt	date
	updatedAt	date

reservations		
PK	id	int
FK	userId	uuid
FK	vehicleId	int
	pickUp	string
	dropOff	string
	passenger	int
	institution	enum
	unit	int
	pickDate	date
	dropDate	date
	status	enum
	totalPrice	int
	isOvertime	int
	overtimeHour	int
	totalPriceAfterOvertime	int
	finishedAt	date
	createdAt	date
	updatedAt	date

vehicles		
PK	id	int
	name	string
	price	int
	model	string
	capacity	int
	include	string
	area	string
	parking	string
	payment	string
	overtime	int
	createdAt	date
	updatedAt	date

Figure 5 . *Entity Relationship Diagrams*

d. *Cloud Architecture Diagram*

Cloud Architecture Diagram is a visual representation of architecture or applications hosted on GCP [17] . This diagram is used to provide an overview of the GCP service components used in the infrastructure of a cloud-based project. This diagram will make it easier for architects and development teams to understand, communicate, and plan best practice solutions that are appropriate for certain cases. In Figure 6, a cloud architecture diagram containing the GCP services used in this study is attached.

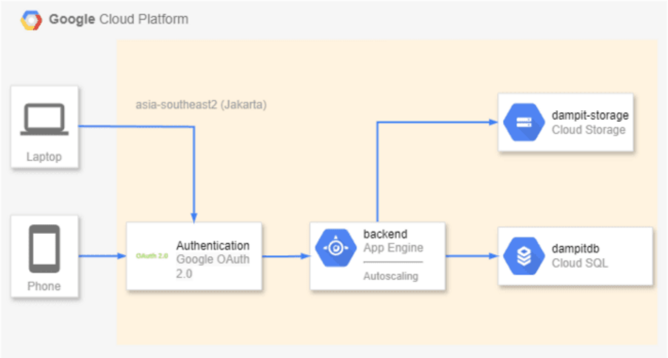


Figure 6 . *Architecture Cloud Diagram*

2.3 *Implementation*

Implementation is the stage of implementing the system design that has been designed, into a program code that can run according to its function. The design of this *website -based car rental system* uses several technologies that will be used, namely React JS, Node JS, Express, MySQL, Git, and *cloud computing.* For the implementation of *cloud computing* , Google Cloud services will be used, including: Google App Engine, *Google Cloud Storage* , and *Google Cloud SQL* . Based on the related technology documentation, here is a brief explanation of the technology that will be used in this system:

a. React JS is a Javascript *library* for designing interactive and responsive user interfaces on websites. React's main feature is the use of the *Virtual DOM concept* and components, which allow developers to divide the user *interface* into smaller, more manageable parts.

b. Node JS is a Javascript *runtime environment* that runs outside the browser, allowing developers to run Javascript code on *the server side.* Node JS has an event *-driven architecture* that allows for effective and efficient processing of multiple connections simultaneously. This makes Node JS a popular technology choice in developing network applications, web *servers* , and various other applications that require *real-time* and large-scale data processing.

c. MySQL is a popular *open source Relational Database Management System* (RDBMS) . It is used as a medium for storing, managing, and accessing data in the form of structured tables. In addition, the use of *Structured Query Language* (SQL) makes it easy to develop web and business applications for retrieving, updating, deleting, and inserting data into the database.

d. Express is a lightweight and flexible web application *framework based on Node JS. Express provides features such as routing,* HTTP *request handling, and middleware* to make it easy for developers to create RESTful APIs for web applications quickly and minimally.

e. Git is *a version control system* (VCS) used to manage changes in the development program code of a project system or application. Git allows users to track code changes, manage code versions efficiently, and collaborate on work with other developers on the same project.

f. *Google App Engine* (GAE) is one of GCP's *Platform as a Service* (PaaS) services that provides development and *deployment environment management* for *website applications* and other software. With GAE, developers can develop applications without thinking about the physical or virtual *server infrastructure* behind it because App Engine is a *serverless service* . GAE has many advantages, such as automation for flexible and fully controlled application scalability, as well as the implementation of *pay as you go* or pay according to service usage.

g. *Google Cloud Storage* is a secure and scalable cloud storage service that allows users to store and manage various types of data, including images, videos, documents, and the like. It also supports geographic data replication, making data quickly and reliably accessible from multiple locations around the world.

h. *Google Cloud SQL* is a GCP service for SQL database management. This service can be fully managed by users, and supports various types of data such as MySQL, PostgreSQL, and SQL *server* . This service simplifies database development, management, and scalability, by providing automation of maintenance, monitoring, and data archiving. With Cloud SQL, users can easily deploy *their* databases in the Google *cloud* , making them fast, secure, and ready to use for any need.

2.4 *Testing*

testing stage is a system testing stage that focuses on analyzing the results of the overall system test that has been integrated according to the components that have been created previously [18] . The *testing stage* applies *black box testing* , *System Usability Scale* (SUS) testing, and access speed evaluation. *Black box testing* is carried out to test the feasibility of the entire system, through observation of the results of the execution and functionality of the system [19] . *Black box* was chosen based on the circumstances of the owner of the Dampit Trans Solo company who did not understand the structure and flow of technology in the development process, which supports the function of *black box testing* in testing the behavior of the subject of the analysis of the requirements specification without having to know the program code of the system used [20] . *System Usability Scale (SUS)* testing is a test that uses a questionnaire to measure the perception of usability created by John Brooke in 1986. SUS is used to evaluate and find out how much achievement is in running the system [21] . Access speed evaluation is applied to test the performance of the system application built and *servers* from various categories, and the assessment depends on *the platform* used.

2.5 Deployment & *Maintenance*

maintenance phase is the final stage and is the production phase of the *website -based car rental system* for Dampit Trans Solo. The system consisting of *front end* and *back end server programs* will be *deployed* and *hosted* using a *domain name.* www.dampittrans.my.id to be accessed via *browser* . Furthermore, the system will be maintained periodically and updated service costs and features to suit customers and company needs in the future.

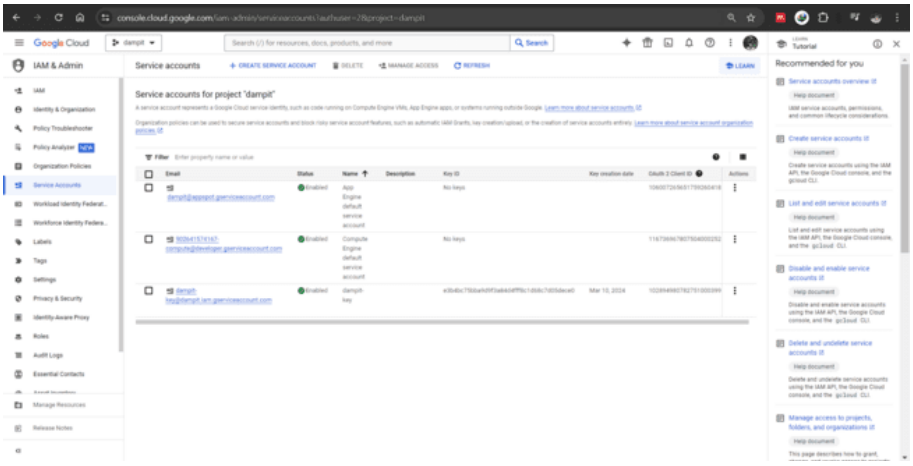
3. RESULTS AND DISCUSSION

The result of this research is a car rental *website* in Dampit Trans Solo company whose system infrastructure implements Google Cloud services. This system is divided into two main views based on its role, namely the view for users and the view for admins which are determined from the results of account authentication when logging in to *the website* . The existence of this system is able to simplify the ordering process, and vehicle reservation management at Dampit Trans Solo which was initially still done manually. The results of the application of *Cloud Computing in creating a website* -based car rental system at Dampit Trans Solo are as follows.

3.1 Results

3.1.1 *Identity and Access Management* (IAM)

Identity and Access Management (IAM) is a Google Cloud service used to manage all permissions and policies of *cloud projects* . The Dampit Trans Solo project implements *the principle of least privilege* , where the policy of all services is only held by the *owner role,* while programs are only given limited access to certain services in the form of *service accounts.* The display of the IAM page can be seen in Figure 7.


console.cloud.google.com/iam-admin/serviceaccounts?authuser=2&project=dampit
Google Cloud
dampit
Search (/) for resources, docs, products, and more
Search
IAM & Admin
Service accounts
+ CREATE SERVICE ACCOUNT
DELETE
MANAGE ACCESS
REFRESH
LEARN
IAM
Identity & Organization
Policy Troubleshooter
Policy Analyzer NEW
Organization Policies
Service Accounts
Workload Identity Federat...
Workforce Identity Federa...
Labels
Tags
Settings
Privacy & Security
Identity-Aware Proxy
Roles
Audit Logs
Essential Contacts
Manage Resources
Release Notes
Tutorial
Recommended for you
Service accounts overview
Help document
Create service accounts
List and edit service accounts
Disable and enable service accounts
Delete and undelete service accounts
Manage access to projects, folders, and organizations
Filter Enter property name or value

Service accounts for project "dampit"

A service account represents a Google Cloud service identity, such as code running on Compute Engine VMs, App Engine apps, or systems running outside Google. Learn more about service accounts

Organization policies can be used to secure service accounts and block risky service account features, such as automatic IAM Grants, key creation/upload, or the creation of service accounts entirely. Learn more about service account organization policies

Email	Status	Name ↑	Description	Key ID	Key creation date	OAuth 2 Client ID	Actions
dampit@appspot.gserviceaccount.com	Enabled	App Engine default service account		No keys		[illegible]	⋮
902641574167-compute@developer.gserviceaccount.com	Enabled	Compute Engine default service account		No keys		[illegible]	⋮
dampit-key@dampit.iam.gserviceaccount.com	Enabled	dampit-key		[illegible]	Mar 10, 2024	[illegible]	⋮

IAM service accounts, permissions, and common lifecycle considerations.

Create service accounts using the IAM API, the Google Cloud console, and the gcloud CLI.

List and edit service accounts using the IAM API, the Google Cloud console, and the gcloud CLI.

Disable and enable service accounts using the IAM API, the Google Cloud console, and the gcloud CLI.

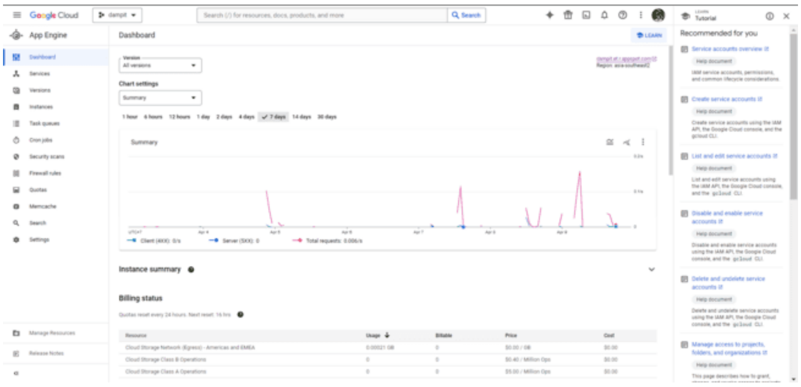
Delete and undelete service accounts using the IAM API, the Google Cloud console, and the gcloud CLI.

This page describes how to grant,

Figure 7 . *Identity Access Management* (IAM)

3.1.2 *Google App Engine* (GAE)

Google App Engine is a PaaS service from Google Cloud that is used in the *back-end* program *deployment process of* the Dampit Trans Solo car rental website. GAE was chosen because it has advantages that are very helpful for developers and companies, one of which is a service that only charges according to *the resources* used, even the service can be free (no charge) as long as *the server* runs without exceeding the GAE *free tier quota* . The following is a display of the GAE page in Figure 8.



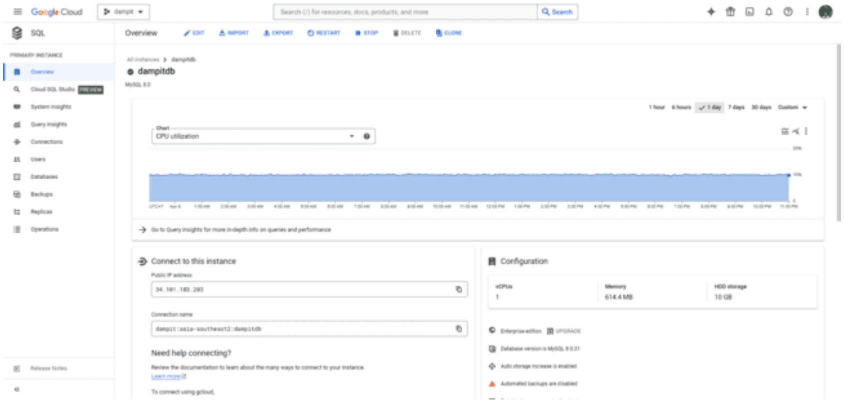
Quotas reset every 24 hours. Next reset: 16 hrs

Resource	Usage ↓	Billable	Price	Cost
Cloud Storage Network (Egress) - Americas and EMEA	0.00021 GB	0	$0.00 / GB	$0.00
Cloud Storage Class B Operations	0	0	$0.40 / Million Ops	$0.00
Cloud Storage Class A Operations	0	0	$5.00 / Million Ops	$0.00

Figure 8 . *Google App Engine* (GAE)

3.1.3 *Google Cloud SQL*

Google Cloud SQL is used as a place to store and manage the Dampit Trans Solo *website database* . One of the main advantages *of Google Cloud SQL* besides managing *relational* databases is its ability to perform automatic backups if a disaster occurs on *the server. on-premise,* making it one of the solutions to the world's major *on-premise* database *server problems* . This service is used in the system as a container for the SQL Dampit database. The *Google Cloud SQL display* can be seen in Figure 9.



All instances > dampitdb

dampitdb

MySQL 8.0

Go to Query insights for more in-depth info on queries and performance

Connect to this instance

Public IP address: 34.101.183.283

Connection name: dampit:asia-southeast2:dampitdb

Need help connecting?

Review the documentation to learn about the many ways to connect to your instance.
Learn more

To connect using gcloud,

Configuration

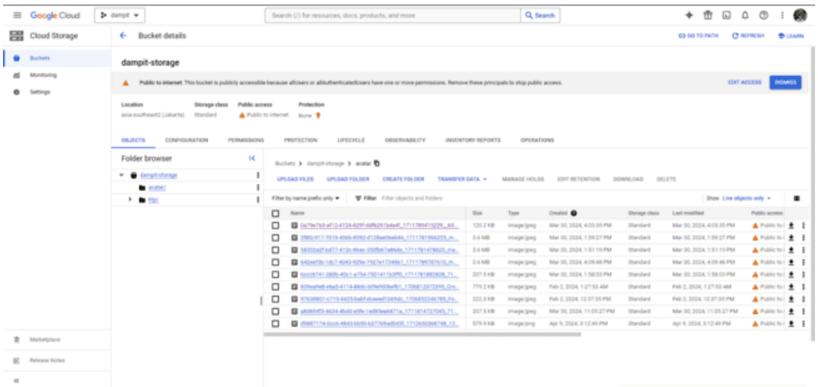
vCPUs	Memory	HDD storage
1	614.4 MB	10 GB

- Enterprise edition
- Database version is MySQL 8.0.31
- Auto storage increase is enabled
- Automated backups are disabled

Figure 9 . *Google Cloud SQL*

3.1.4 *Google Cloud Storage*

Google Cloud Storage is a Google Cloud service that is specifically used as a storage container for various types of files. One of the advantages of *Google Cloud Storage,* besides its ease in integrating services with programs, is its ability to categorize storage based on how often files are used. This service is used by the system as storage for dynamic files such as system *logs* , user avatars, and user ID cards. *Service accounts* are used in this service as additional security and confidentiality of file access. The *Google Cloud Storage display* can be seen in Figure 10.



dampit-storage

Public to internet This bucket is publicly accessible because allUsers or allAuthenticatedUsers have one or more permissions. Remove these principals to stop public access.

Location	Storage class	Public access	Protection
asia-southeast2 (Jakarta)	Standard	Public to internet	None

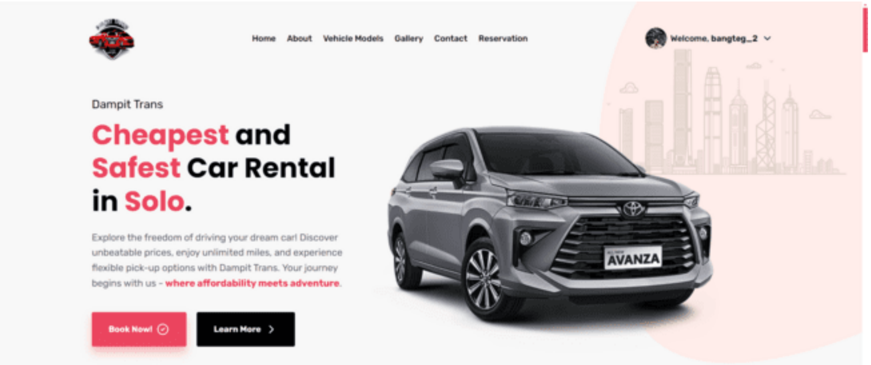
Buckets > dampit-storage > avatar

Name	Size	Type	Created	Storage class	Last modified	Public access
[illegible]	120.2 KB	image/jpeg	Mar 30, 2024, 4:03:35 PM	Standard	Mar 30, 2024, 4:03:35 PM	Public to i
[illegible]	3.6 MB	image/jpeg	Mar 30, 2024, 1:59:27 PM	Standard	Mar 30, 2024, 1:59:27 PM	Public to i
[illegible]	3.6 MB	image/jpeg	Mar 30, 2024, 1:51:19 PM	Standard	Mar 30, 2024, 1:51:19 PM	Public to i
[illegible]	3.6 MB	image/jpeg	Mar 30, 2024, 4:09:48 PM	Standard	Mar 30, 2024, 4:09:48 PM	Public to i
[illegible]	207.5 KB	image/jpeg	Mar 30, 2024, 1:58:03 PM	Standard	Mar 30, 2024, 1:58:03 PM	Public to i
[illegible]	779.2 KB	image/jpeg	Feb 2, 2024, 1:27:53 AM	Standard	Feb 2, 2024, 1:27:53 AM	Public to i
[illegible]	222.3 KB	image/jpeg	Feb 2, 2024, 12:37:35 PM	Standard	Feb 2, 2024, 12:37:35 PM	Public to i
[illegible]	207.5 KB	image/jpeg	Mar 30, 2024, 11:05:27 PM	Standard	Mar 30, 2024, 11:05:27 PM	Public to i
[illegible]	579.9 KB	image/png	Apr 9, 2024, 3:12:49 PM	Standard	Apr 9, 2024, 3:12:49 PM	Public to i

Figure 10 . *Google Cloud Storage*

3.1.5 Home Page

The main page is the *default display of the website* that can be accessed by everyone with or without logging in first. This page contains all information related to the company, services, and contacts of the Dampit Trans Solo company can be found on this display. On the main page there are also several navigation buttons that can be used to start the registration process, and login for those who have not logged in. The display of the main page can be seen in Figure 11.



Dampit Trans

Cheapest and Safest Car Rental in Solo.

Explore the freedom of driving your dream car! Discover unbeatable prices, enjoy unlimited miles, and experience flexible pick-up options with Dampit Trans. Your journey begins with us - where affordability meets adventure.

Figure 11 . Main Page

Register Page

register page is used for users who do not have an account registered in the system *database* . On this page, users are required to complete personal information such as *username* , full name, email, and *password* as shown in Figure 12. After successful registration, the user will receive a verification email sent to the registrant's email as shown in Figure 13.

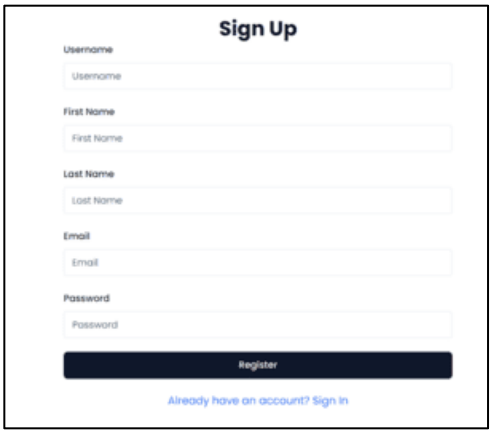
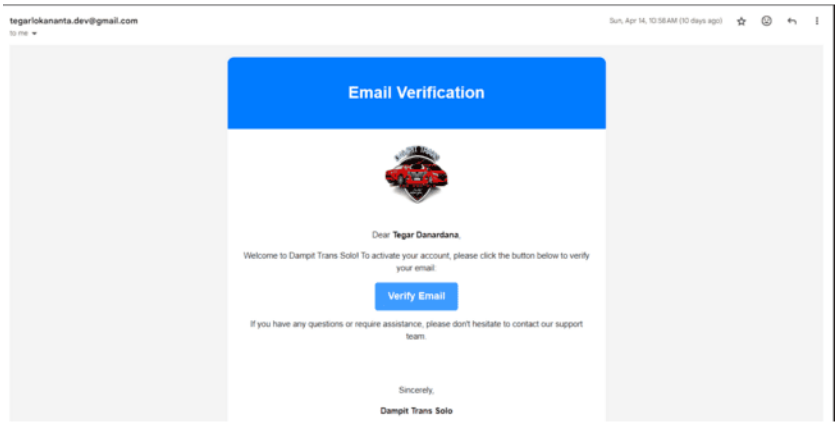


Figure 12 . Register Page


tegarlokananta.dev@gmail.com
to me
Sun, Apr 14, 10:58 AM (10 days ago)
Email Verification
Verify Email

Dear **Tegar Danardana**,

Welcome to Dampit Trans Solo! To activate your account, please click the button below to verify your email:

If you have any questions or require assistance, please don't hesitate to contact our support team.

Sincerely,

Dampit Trans Solo

Figure 13 . Account Verification Email

3.1.7 Login Page

The login page is a page used by the system for the user and admin authentication process before being directed to a special page for each account. The login process is done by entering the email and *password* that has been registered, or by selecting a Gmail account that has been linked to the browser via the Google OAuth feature. The appearance of the login page can be seen in Figure 14.

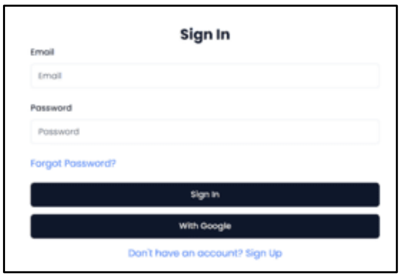
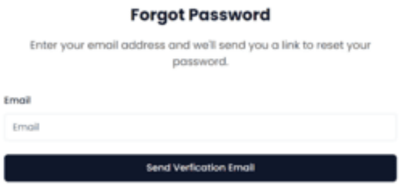


Figure 14 . Login Page

Password Page

The forgot *password page* is a page accessed by users who forget *the password of* the account registered during the login process. This page will ask the user to enter the registered email as in Figure

15, and then the system will send an email containing a link to confirm the *password change* which can be seen in Figure 16.



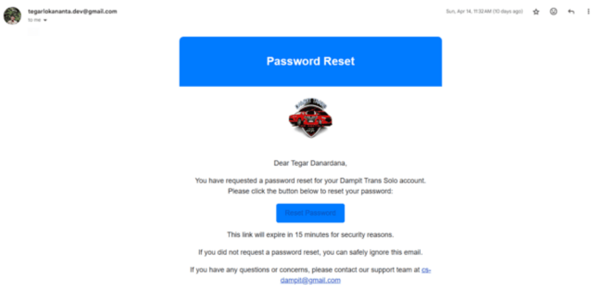
Forgot Password

Enter your email address and we'll send you a link to reset your password.

Email

Email

Figure 15 . Forgot *Password* Page



tegarlokananta.dev@gmail.com
to me

Sun, Apr 14, 11:32 AM (10 days ago)

Password Reset

Dear Tegar Danardana,

You have requested a password reset for your Dampit Trans Solo account.
Please click the button below to reset your password:

This link will expire in 15 minutes for security reasons.

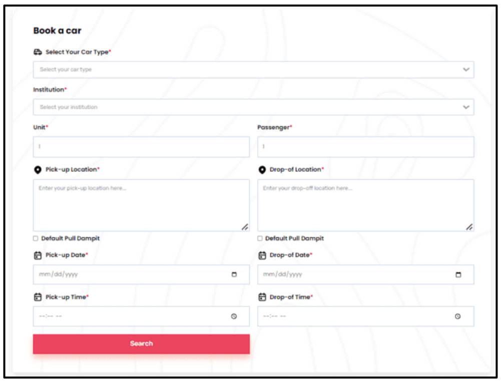
If you did not request a password reset, you can safely ignore this email.

If you have any questions or concerns, please contact our support team at cs-dampit@gmail.com

Figure 16 . Forgot *Password* Confirmation Email

3.1.9 User Reservation Page

The user reservation page is a page that can be accessed from the main page by users who have logged in, with a page display that can be seen in Figure 17. On this page, users will be asked to write important information needed for data collection, such as: vehicle type, departure and return locations, number of passengers, number of vehicles rented, and rental dates. The system will receive reservation data and then the admin will receive an email notification of the order that has just been received. The notification email can be seen in Figure 18.



Book a car

Select Your Car Type*

Select your car type

Institution*

Select your institution

Unit*

1

Passenger*

1

Pick-up Location*

Enter your pick-up location here...

□ Default Pull Dampit

Drop-of Location*

Enter your drop-off location here...

□ Default Pull Dampit

Pick-up Date*

mm/dd/yyyy

Drop-of Date*

mm/dd/yyyy

Pick-up Time*

--:-- --

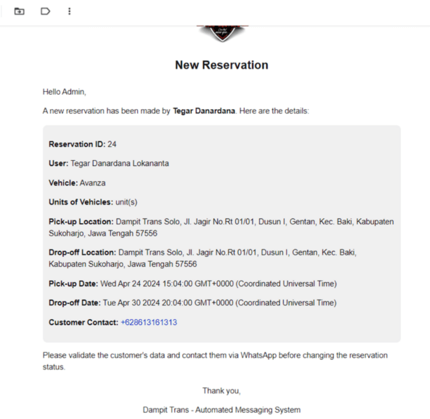
Drop-of Time*

--:-- --

Figure 17 . User Reservation Page

This figure 17 displays the interface of the reservation page accessible to users after logging into the website. On this page, users are prompted to fill in key information required to make a reservation. These include selecting the vehicle type, specifying the departure and return locations, number of passengers, number of vehicles to rent, and the rental period (start and end dates). Once the form is submitted, the system processes the reservation and sends a notification to the admin for further handling.

This figure 18 shows the automated email notification sent to the admin when a new reservation is made by a user. The email contains essential reservation details that allow the admin to review and respond accordingly. This automation helps streamline the reservation workflow and reduces the chance of missed or delayed order processing.



New Reservation

Hello Admin,

A new reservation has been made by **Tegar Danardana**. Here are the details:

Reservation ID: 24

User: Tegar Danardana Lokananta

Vehicle: Avanza

Units of Vehicles: unit(s)

Pick-up Location: Dampit Trans Solo, Jl. Jagir No.Rt 01/01, Dusun I, Gentan, Kec. Baki, Kabupaten Sukoharjo, Jawa Tengah 57556

Drop-off Location: Dampit Trans Solo, Jl. Jagir No.Rt 01/01, Dusun I, Gentan, Kec. Baki, Kabupaten Sukoharjo, Jawa Tengah 57556

Pick-up Date: Wed Apr 24 2024 15:04:00 GMT+0000 (Coordinated Universal Time)

Drop-off Date: Tue Apr 30 2024 20:04:00 GMT+0000 (Coordinated Universal Time)

Customer Contact: +628613161313

Please validate the customer's data and contact them via WhatsApp before changing the reservation status.

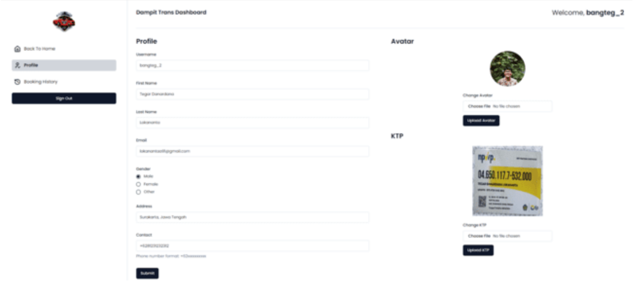
Thank you,

Dampit Trans - Automated Messaging System

Figure 18 . Order Notification Email For Admin

3.1.10 User Profile Management Page

The user profile management page is a page used by users to complete or update user information on the system. The data that can be set on this page is the personal data of the account owner, which is used for vehicle reservation data, such as: *avatar,* full name, address, gender, contact, and identity card (KTP). The appearance of the user profile management page can be seen in Figure 19.



Dampit Trans Dashboard

Welcome, **bangteg_2**

Profile

Username: bangteg_2

First Name: Tegar Danardana

Last Name: Lokananta

Email: [illegible]

Gender: ● Male ○ Female ○ Other

Address: Surakarta, Jawa Tengah

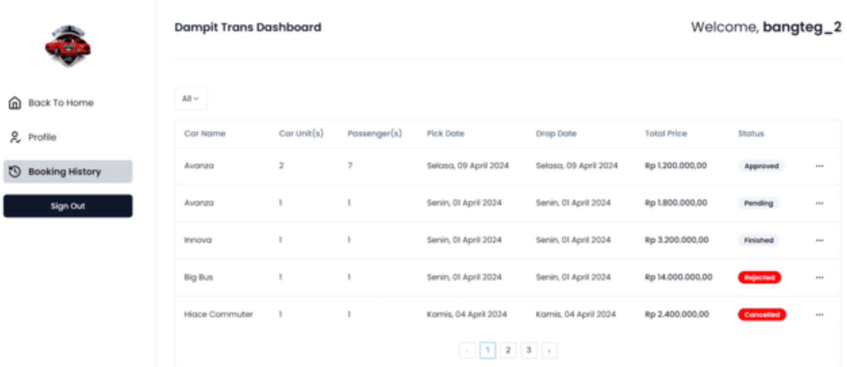
Contact: [illegible]

Phone number format: +62xxxxxxxxx

Figure 19 . User Profile Management Page

3.1.11 User Reservation History Page

The user reservation history page is a page that contains a collection of reservations that are being and have been made by the user. On this page, users can see detailed information from the data entered when making a reservation, the reservation status, the fees to be paid, and information about the vehicle being reserved. The appearance of the user reservation history page can be seen in Figure 20.



Dampit Trans Dashboard

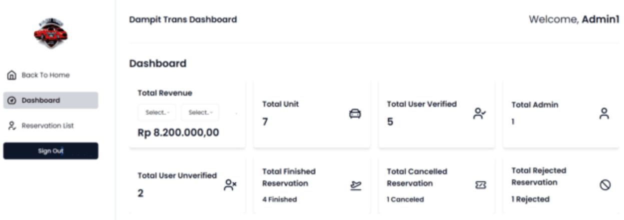
Welcome, **bangteg_2**

Car Name	Car Unit(s)	Passenger(s)	Pick Date	Drop Date	Total Price	Status	
Avanza	2	7	Selasa, 09 April 2024	Selasa, 09 April 2024	Rp 1.200.000,00	Approved	···
Avanza	1	1	Senin, 01 April 2024	Senin, 01 April 2024	Rp 1.800.000,00	Pending	···
Innova	1	1	Senin, 01 April 2024	Senin, 01 April 2024	Rp 3.200.000,00	Finished	···
Big Bus	1	1	Senin, 01 April 2024	Senin, 01 April 2024	Rp 14.000.000,00	Rejected	···
Hiace Commuter	1	1	Kamis, 04 April 2024	Kamis, 04 April 2024	Rp 2.400.000,00	Cancelled	···

Figure 20 . User Reservation History Page

Monitoring Page Admin *Dashboard*

Monitoring page admin *dashboard* is a page that can be displayed after the admin logs in. This page contains information such as total income, number of users, number of vehicles, number of reservations, and other information that is intended to make it easier for the admin to observe important information from the system. *Monitoring page display* The admin *dashboard* can be seen in Figure 21.



Dampit Trans Dashboard

Welcome, Admin1

Dashboard

Total Revenue: Rp 8.200.000,00

Total Unit: 7

Total User Verified: 5

Total Admin: 1

Total User Unverified: 2

Total Finished Reservation: 4 Finished

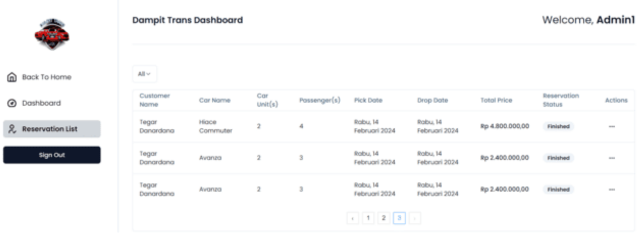
Total Cancelled Reservation: 1 Canceled

Total Rejected Reservation: 1 Rejected

Figure 21 . *Dashboard* page Admin *Monitoring*

3.1.13 Admin Reservation Management Page

The admin reservation management page is a page used by the admin to manage all reservations made by users, the appearance of this page can be seen in Figure 22. On this page, the admin will be shown the entire list of user reservations, the admin can also filter the data displayed based on its status . The admin can also see detailed information about the reservations made, and then change the reservation status on the reservation details page as seen in Figure 23. If the admin has changed the reservation status, the user who made the reservation will receive an email notification from the admin based on the status given as in Figure 24.

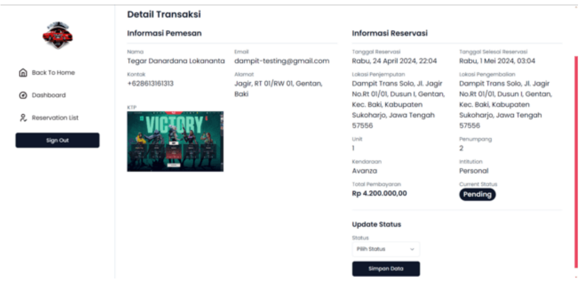


Dampit Trans Dashboard

Welcome, Admin1

Customer Name	Car Name	Car Unit(s)	Passenger(s)	Pick Date	Drop Date	Total Price	Reservation Status	Actions
Tegar Danardana	Hiace Commuter	2	4	Rabu, 14 Februari 2024	Rabu, 14 Februari 2024	Rp 4.800.000,00	Finished	...
Tegar Danardana	Avanza	2	3	Rabu, 14 Februari 2024	Rabu, 14 Februari 2024	Rp 2.400.000,00	Finished	...
Tegar Danardana	Avanza	2	3	Rabu, 14 Februari 2024	Rabu, 14 Februari 2024	Rp 2.400.000,00	Finished	...

Figure 22 . Admin Reservation Management Page



Detail Transaksi

Informasi Pemesan

Nama: Tegar Danardana Lokananta

Email: dampit-testing@gmail.com

Kontak: +628613161313

Alamat: Jagir, RT 01/RW 01, Gentan, Baki

KTP

Informasi Reservasi

Tanggal Reservasi: Rabu, 24 April 2024, 22:04

Tanggal Selesai Reservasi: Rabu, 1 Mei 2024, 03:04

Lokasi Penjemputan: Dampit Trans Solo, Jl. Jagir No.Rt 01/01, Dusun I, Gentan, Kec. Baki, Kabupaten Sukoharjo, Jawa Tengah 57556

Lokasi Pengembalian: Dampit Trans Solo, Jl. Jagir No.Rt 01/01, Dusun I, Gentan, Kec. Baki, Kabupaten Sukoharjo, Jawa Tengah 57556

Unit: 1

Penumpang: 2

Kendaraan: Avanza

Intitution: Personal

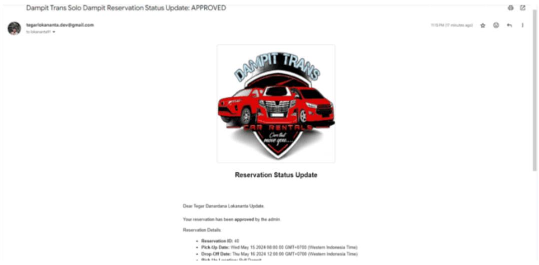
Total Pembayaran: Rp 4.200.000,00

Current Status: Pending

Update Status

Status

Figure 23 . Reservation Details Page



Dampit Trans Solo Dampit Reservation Status Update: APPROVED

tegarlokananta.dev@gmail.com
to lokananta91

7:15 PM (17 minutes ago)

Reservation Status Update

Dear Tegar Danardana Lokananta Update,

Your reservation has been **approved** by the admin.

Reservation Details:

- **Reservation ID:** 40
- **Pick-Up Date:** Wed May 15 2024 08:00:00 GMT+0700 (Western Indonesia Time)
- **Drop-Off Date:** Thu May 16 2024 12:00:00 GMT+0700 (Western Indonesia Time)
- **Pick-Up Location:** Bull Dampit

Figure 24 . Reservation Status Email Has Been Updated For User

3.2 Testing

3.2.1 *Black Box Testing*

2. *Black Box* testing is conducted to test the entire process of the *website -based car rental system* , which aims to ensure that the system runs and functions properly and in accordance with the initial design. The results of the *Black Box testing* are documented in Table 1 which shows that the system built can run properly and according to error.

Table 1. Black Box Test Results

Part	Testing	3. Results
Home Page	Menu *About*	4. Valid
	Vehicle Models Menu	5. Valid
	Dampit *Gallery Menu*	6. Valid
	Contact Menu	7. Valid
	8. Whatsapp Button	9. Valid
10. Register	11. Filling in Registration Data	12. Valid
	13. Sending Account Verification Email	14. Valid
	15. Account Verification	16. Valid
	17. Hash Password	18. Valid
19. Login (user)	20. Regular Login	21. Valid
	22. Login Using Google	23. Valid
	24. Role Validation	25. Valid
26. Forgot *the password*	27. Sending Confirmation Email	28. Valid
	29. Updating *Password*	30. Valid
31. Login (admin)	32. Regular Login	33. Valid
	34. Login Using Google	35. Valid
	36. *Role* Validation	37. Valid
38. Profile Management (user)	39. Adding Data	40. Valid
	41. Updating Data	42. Valid
	43. Adding Avatar	44. Valid
	45. Adding KTP	46. Valid
47. Reservation (user)	48. Entering Reservation Data	49. Valid
	50. Make a Reservation	51. Valid
	52. Sending Notification Email to Admin	53. Valid
54. Reservation History (user)	55. Showing Reservation List	56. Valid
	57. Filter Displayed Data	58. Valid
	59. View Reservation Details	60. Valid
	61. Cancel Reservation	62. Valid
	63. Sending Notification Email to Admin	64. Valid
65. *Dashboard* (admin)	66. Displaying *Monitoring Data*	67. Valid
	68. Data Filters	69. Valid
70. Reservation Management (admin)	71. Displaying All User Reservation List	72. Valid
	73. Filter Displayed Data	74. Valid
	75. View Reservation Details	76. Valid
	77. Make a Reservation Status Update	78. Valid
	79. Sending Reservation Status Notification Emails to Users	80. Valid
81. Other	82. Error Handling	83. Valid
	84. Data Pagination	85. Valid
	86. Responsive View	87. Valid

88.

3.2.2 System Usability Scale (SUS) – Update Abstract, closing

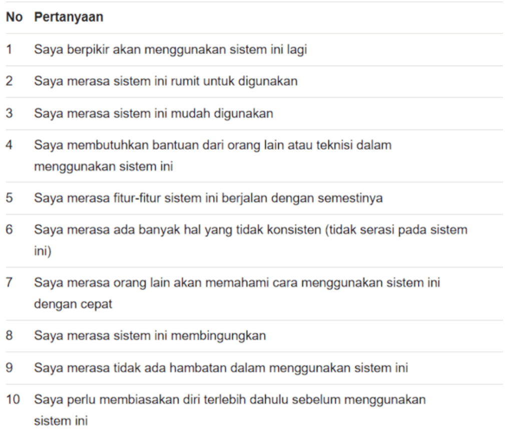
89. SUS testing is applied to identify *the usability* or usefulness of the Dampit Trans Solo car rental system seen from the user's perspective, with an assessment based on the average value range as in Table 2. A total of 57 respondents filled out a questionnaire consisting of ten (10) questions and five (5)

answer choices with a *Likert scale* ranging from "Strongly disagree" to "Strongly Agree" [22] . The respondents who filled out the questionnaire were divided into various backgrounds, including Dampit Trans Solo employees, students, employees, and pupils. The list of questions used in the SUS testing can be seen in Figure 25.

Table 2. SUS **assessment**

Category	Average Value Range	Interpretation
Very bad	0 – 25	Can not be used
Bad	25 – 50	In dire need of improvement
Currently	51 – 67	Marginal, needs some improvement
Good	68 – 80	Acceptable
Very good	81 - 100	Very satisfactory

90.



No	Pertanyaan
1	Saya berpikir akan menggunakan sistem ini lagi
2	Saya merasa sistem ini rumit untuk digunakan
3	Saya merasa sistem ini mudah digunakan
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi pada sistem ini)
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat
8	Saya merasa sistem ini membingungkan
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini

91.

Figure 25 . SUS Question List

92. The results of the SUS test can be seen in Table 3, which when calculated using the SUS assessment method, gets a score of 80.04. The calculation method for the SUS assessment method for each respondent is as follows: the answer value for odd questions is minus one (1), plus each even question calculated using the number five (5) minus the answer value. The total SUS value for each respondent is then multiplied by 2.5 and the average is calculated to produce a final value in the form of an average SUS value. Based on the SUS assessment table, a value of 80.04 is categorized as a "Good" result, so it can be interpreted that the system is acceptable to users, and assesses their overall experience as quite positive.

Table 3. SUS Questionnaire **Results**

Respondents	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	SUS Value
1	5	1	5	1	5	1	5	1	5	1	100
2	5	2	5	1	5	1	5	1	5	2	95
3	3	2	3	4	3	3	2	3	4	4	47.5
4	5	2	4	3	4	2	4	1	4	4	72.5
5	4	2	4	3	5	2	4	1	5	3	77.5
6	5	1	5	3	5	1	5	1	5	1	95
7	5	2	4	2	5	2	4	2	4	2	80
8	4	1	4	2	4	2	3	2	4	3	72.5
9	3	4	3	3	3	4	3	3	4	4	45
10	5	1	5	5	5	1	4	1	5	1	87.5

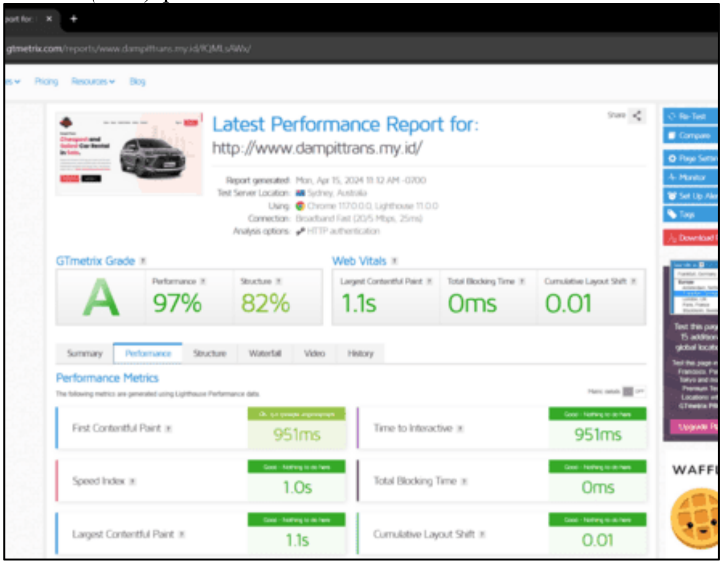
Respondents	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	SUS Value
11	5	1	5	1	5	1	5	1	5	1	100
12	5	1	5	1	5	1	5	1	5	1	100
13	5	5	5	1	5	1	5	1	5	5	80
14	5	1	5	1	5	1	5	1	5	1	100
15	5	1	5	1	5	1	5	1	5	1	100
16	5	5	5	5	5	1	5	1	5	3	75
17	5	1	5	1	5	2	5	1	4	2	92.5
18	5	1	5	1	5	1	5	1	5	1	100
19	5	1	5	2	5	2	5	1	4	2	90
20	5	1	5	1	5	1	5	1	5	1	100
21	5	2	5	2	4	1	5	1	4	1	90
22	5	1	5	1	5	1	5	1	5	1	100
23	4	1	5	1	3	2	5	1	3	1	85
24	4	2	4	3	4	2	3	2	4	2	70
25	4	2	4	4	5	2	5	2	5	3	75
26	5	1	5	1	5	1	5	1	5	1	100
27	5	1	5	1	5	1	5	1	5	1	100
28	3	1	3	4	4	4	4	1	3	4	57.5
29	5	1	5	1	5	1	5	1	5	1	100
30	4	2	4	2	4	2	4	2	4	2	75
31	5	1	5	1	5	1	5	1	5	1	100
32	3	3	3	4	4	3	4	3	3	4	50
33	5	1	5	1	5	1	5	1	4	2	95
34	4	2	4	3	4	2	4	2	4	4	67.5
35	5	1	5	2	5	1	5	1	5	2	95
36	4	1	4	3	5	1	5	1	5	1	90
37	4	2	4	2	5	1	3	3	4	5	67.5
38	3	2	3	1	3	4	2	1	2	3	55
39	5	1	5	1	5	2	5	1	5	1	97.5
40	4	2	5	4	4	2	4	2	4	2	72.5
41	3	2	4	1	3	4	4	3	4	4	60
42	4	1	5	2	5	3	4	1	5	3	82.5
43	4	2	5	1	5	2	5	2	4	3	82.5

Respondents	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	SUS Value
44	4	2	3	5	5	4	4	1	5	5	60
45	5	2	4	1	4	2	4	2	4	2	80
46	4	2	5	4	4	3	3	2	4	4	62.5
47	3	2	3	3	4	3	4	3	3	4	55
48	5	5	5	2	5	1	5	1	5	3	82.5
49	5	1	5	1	5	1	5	1	5	2	97.5
50	5	1	5	2	5	1	5	1	5	5	87.5
51	5	1	5	2	5	2	5	2	5	3	87.5
52	4	1	4	2	4	2	4	2	4	3	75
53	3	3	3	4	3	3	2	4	2	4	37.5
54	5	3	5	3	4	3	4	3	4	4	65
55	3	3	3	4	3	3	3	4	3	4	42.5
56	5	1	5	1	5	1	5	1	5	1	100
57	4	3	3	3	3	3	3	3	3	3	52.5
Average value											80.04

93.

3.2.3 Access Speed Evaluation

94. The evaluation of access speed is applied to determine the speed of *website access* from the user or *client side.* Based on the test conducted using the GTmetrix service in Figure 26, it is known that the average *loading speed of the Dampit Trans Solo car rental website* page is 1.8 seconds with *website performance* falling into category A. This speed can be said to be in accordance with the standards of Nielsen which states that the *loading time* or waiting time of a *website* is said to be good if it is no more than 10 seconds [23] . The Dampit Trans Solo system access speed test was also conducted using *Google PageSpeed Insights* as seen in Figure 27, it is stated that the *website performance value* reaches 95, system accessibility reaches 80, *best practices reach a value of 96, and the search engine optimization (SEO)* performance value reaches 82.

95. 

gtmetrix.com/reports/www.dampittrans.my.id/lQMLsAWx/

Latest Performance Report for:
http://www.dampittrans.my.id/

Report generated: Mon, Apr 15, 2024 11:12 AM -0700
Test Server Location: Sydney, Australia
Using: Chrome 117.0.0.0, Lighthouse 11.0.0
Connection: Broadband Fast (20/5 Mbps, 25ms)
Analysis options: HTTP authentication

GTmetrix Grade: A — Performance 97% — Structure 82%

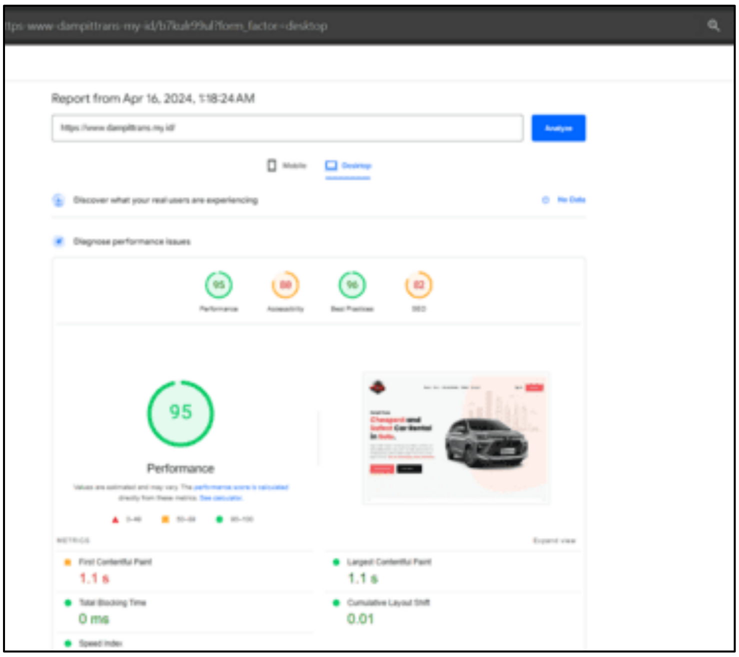
Web Vitals: Largest Contentful Paint 1.1s — Total Blocking Time 0ms — Cumulative Layout Shift 0.01

Performance Metrics

The following metrics are generated using Lighthouse Performance data.

First Contentful Paint	951ms	Time to Interactive	951ms
Speed Index	1.0s	Total Blocking Time	0ms
Largest Contentful Paint	1.1s	Cumulative Layout Shift	0.01

Figure 26 . GTmetrix Test Results


ttps www dampittrans my id/b7kuk99ul?form_factor=desktop

Report from Apr 16, 2024, 1:18:24 AM

https://www.dampittrans.my.id/

Analyze

Mobile Desktop

Discover what your real users are experiencing

No Data

Diagnose performance issues

95 Performance

96 Best Practices

82 SEO

95

Performance

METRICS

Expand view

First Contentful Paint

1.1 s

Largest Contentful Paint

1.1 s

Total Blocking Time

0 ms

Cumulative Layout Shift

0.01

Speed Index

Figure 27 . *Google PageSpeed Insights* Test Results

96. 4. Conclusion

Implementation of *cloud computing* on a *website- based car rental system* at Dampit Trans Solo using the SDLC *waterfall model method* is capable of... produce a satisfying and highly reliable web application. The *website -based car rental system* that was created has a prominent advantage in terms of functionality, proven by *Black Box testing* on all features and the results are "valid" or error-free. In terms of its level of usability in real cases, the system is measured using the *System Usability Scale* (SUS) test, resulting in a value of 80.04 with a "Good" assessment range for the system according to 57 respondents. The use of Google Cloud as a *cloud computing service platform* also greatly facilitates the design of a secure, scalable, fast, and relatively cost-effective *cloud infrastructure* . With such a design, the Dampit Trans Solo car rental system access speed performance is very good, where based on GTmetrix and Google *PageSpeed Insights* the average time needed for users to access the page is only 1.8 seconds, and *the website performance* reaches a value of 95.

Overall, the implementation of *cloud computing* greatly helps developers in building application systems with excellent performance and can be relied on in real business processes. In the future, the system can be further optimized in the use of Google Cloud services implemented with programs, in order to improve *website speed performance* , maximize security and infrastructure *monitoring* , and adjust the service *budget* to the needs of real company cases. The system also needs to be improved in terms of *user experience* (UX) by adjusting the needs of *real case* admins and users, so that the potential of all features can be maximized.

References

[1] L. Haji, O. Ahmed, S. Zeebaree, H. Dino, R. Zebari, and H. Shukur, “Impact of Cloud Computing and Internet of Things on the Future Internet,” *Technol. Reports Kansai Univ.* , vol. 62, no. 05, pp. 2179–2190, 2020.

[2] HS Malallah, R. Qashi, LM Abdulrahman, MA Omer, and AA Yazdeen, “Performance Analysis of Enterprise Cloud Computing: A Review,” *J. Appl. Sci. Technol. Trends* , vol. 4, no. 01, pp. 01–12, 2023, doi: 10.38094/jastt401139.

[3] F. Thung and IS Suwardi, "Blood parasite identification using feature based recognition," in *Proceedings of the 2011 International Conference on Electrical Engineering and Informatics* , 2011, pp. 1–4. doi: 10.1109/ICEEI.2011.6021590.

[4] D. Gunawan, IC Utomo, F. Yasin, A. Irsyadi, and DA Puspa, “Implementation of MERN Stack in the Development of New Student Admissions System,” vol. 11, no. 2, pp. 102–110, 2023.

[5] GJTRI Pradana, “Design and Construction of a Car Rental System at Garuda Car Rental Based on a Website Using the Waterfall Method,” 2019, [Online]. Available:

http://repository.unmuhjember.ac.id/id/eprint/7135

[6] N. Hidayati, "Design and Construction of a Car Rental Information System Using the Waterfall Model," *J. Comput. Inform. (J-SAKTI*, vol. 5, no. 2, pp. 1019–1033, 2021.

[7] A. Pendri, CP Trisia, RA Fandi, RA Mahroja, TT Mannan, and Y. Heriana, "Designing a Web-based Car Rental System Using Agile Method," *Pros. Semin. Nas. Technol. Inf. and Business* , pp. 686–697, 2022.

[8] MH Romadhon, Y. Yudhistira, and M. Mukrodin, "Android-Based Car Rental Information System and Website Using Codeigniter Framework 3 Case Study: CV Kopja Mandiri," *J. Sist. Inf. and Technol. Perad.* , vol. 2, no. 1, pp. 30–36, 2021.

[9] P. Wankhede, M. Talati, and R. Chinchamalatpure, "Comparative Study of Cloud Platforms - Microsoft Azure, Google Cloud Platform and Amazon Ec2," *J. Res. Eng. Appl. Sci.* , vol. 05, no. 02, pp. 60–64, 2020, doi: 10.46565/jreas.2020.v05i02.004.

[10] A. Elanda and K. Suhada, "Jurnal Indonesia: Informatics and Communication Management VPS AND MULTI-REGION LOAD BALANCER DESIGN FOR PT SCHLEMMER AUTOMOTIVE INDONESIA WEBSITE USING Jurnal Indonesia: Informatics and Communication Management," vol. 4, no. 3, pp. 1274–1283, 2023.

[11] Ariata, "What is JavaScript? How is it Different from Java?," *Hostinger Tutorial* , 2023. https://www.hostinger.co.id/tutorial/apa-itu-javascript (accessed Sep. 30, 2023).

[12] E. Crookshanks, "Development Methodologies and SDLC BT - Practical Enterprise Software Development Techniques: Tools and Techniques for Large Scale Solutions," E. Crookshanks, Ed. Berkeley, CA: Apress, 2015, pp. 37–59. doi: 10.1007/978-1-4842-0620-1_4.

[13] AF Diansyah *et al.* , "Comparative Analysis of Software Development Lifecycle Methods in Software Development : A Systematic Literature Review," vol. 4, no. 2, pp. 97–106, 2023, doi: 10.25008/ijadis.v4i2.1295.

[14] F. Ciccozzi, I. Malavolta, and B. Selic, "Execution of UML models: a systematic review of research and practice," *Softw. Syst. Model.* , vol. 18, no. 3, pp. 2313–2360, 2019, doi: 10.1007/s10270-018-0675-4.

[15] Ergashef, "Design of Automated Enterprise Information Systems Using UML," *Methodical Res. J.* , vol. 3, no. 12, pp. 25–31, 2022.

[16] R. Umar, AF Sudrajat, U. Ahmad, and D. Yogyakarta, "Implementation of Cloud Computing in Web-Based Dieng Homestay Reservation System," *J. Sist. Inf.* , vol. 5341, no. October, p. 2, 2018.

[17] JT Huang, LY Chang, and HC Lin, "Implementation of IoT, Wearable Devices, Google Assistant and Google Cloud Platform for Elderly Home Care System," *Int. Conf. Inf. Commun. Technol. Aging Well e-Health, ICT4AWE - Proc.* , vol. 2021-April, no. Ict4awe, pp. 203–212, 2021, doi: 10.5220/0010473102030212.

[18] A. Daffa' Tsaqif and IC Utomo, "Web-Based Event Ticket Sales Application Using React Js Library," *J. Comput. and Tech. Inform.* , vol. 1, no. 1, pp. 83–100, 2023, [Online]. Available: https://doi.org/10.54082/kontak.12

[19] WN Fathoni and M. Maryam, "DESIGN OF WEB-BASED CERTIFICATE SERVICE INFORMATION SYSTEM (Case Study: Dawungan Village, Masaran District, Sragen Regency)," *J. Educ. and Technol. Indones.* , vol. 1, no. 5, pp. 199–208, 2021, doi: 10.52436/1.jpti.42.

[20] AC Praniffa, A. Syahri, F. Sandes, U. Fariha, QA Giansyah, and ML Hamzah, "Black Box and White Box Testing of Web-Based Parking Information System ," *J. Test. and Implementation Sist. Inf.* , vol. 1, no. 1, pp. 1–16, 2023.

[21] AW Soejono, A. Setyanto, and AF Sofyan, "Evaluation of UNRIYO Website Usability Using System Usability Scale (Case Study: UNRIYO Website)," *Respati* , vol. 13, no. 1, pp. 29–37, 2018, doi: 10.35842/jtir.v13i1.213.

[22] GS Mahendra and IKA Asmarajaya, "Evaluation Using Black Box Testing and System Usability Scale in the Kidung Sekar Madya Application," *Sinkron* , vol. 7, no. 4, pp. 2292–2302, 2022, doi: 10.33395/sinkron.v7i4.11755.

[23] S. Hamid, NZ Bawany, and K. Zahoor, "Assessing ecommerce websites: Usability and accessibility study," *2020 Int. Conf. Adv. Comput. Sci. Inf. Syst. ICACSIS 2020* , no. November, pp. 199–204, 2020, doi: 10.1109/ICACSIS51025.2020.9263162.