SIBOX Smart Loker System with Dynamic Systems Development Method

Daffa Praramadhana[1], Akmal Fauzan Suranta[1], Iqbal Amrulloh[1], Meida Cahyo Untoro[1], Mugi Praseptiawan[1]
[1]Teknik Informatika, Institut Teknologi Sumatera
cahyo.untoro@if.itera.ac.id*

ARTICLE INFO

Article history:

Received April 2023
Received in revised form
Accepted April 2023
Available online Mei 2023

ABSTRACT

At this time logistics companies have a very important role in everyday life, especially in package delivery. The high level of online shopping is one of the reasons why the role of logistics is very important in our daily lives. Thus, seeing the common problems that occur in the conventional logistics delivery process, we came up with the idea to create an integrated smart locker to be one of the replacements for existing logistics outlets, we hope that with this application we can reduce the company's logistics expenses used in procuring logistics outlets and make it easier for couriers to work in the package delivery process. We created this application using a microservices architecture with the SDLC method used is Agile Dynamic System Development (DSDM). React JS framework as an interface and Express js and Laravel as an application that works behind the interface. The idea that we initiated we raised as a project within the company with the client from the company. In the end, the smart locker program made using the DSDM method has been completed and is ready to be implemented in the company and the client.

Keywords: Backend, Frontend, Microservice, Peripherals.

Abstrak

Pada saat ini perusahaan logistik memiliki peran yang sangat penting dalam kehidupan sehari-hari khususnya dalam pengiriman paket. Tingginya tingkat belanja online menjadi salah satu alasan mengapa peran logistik sangat penting dalam kehidupan kita sehari-hari. Maka dari itu, melihat permasalahan umum yang terjadi pada proses pengiriman logistik konvensional, maka muncullah ide untuk membuat smart locker terintegrasi menjadi salah satu pengganti outlet logistik yang sudah ada, kami berharap dengan adanya aplikasi ini dapat mengurangi logistik perusahaan pengeluaran yang digunakan dalam pengadaan logistik di outlet dan memudahkan kurir bekerja dalam proses pengiriman paket. Aplikasi ini kami buat menggunakan arsitektur microservices dengan metode SDLC yang digunakan adalah Agile Dynamic System Development (DSDM). Bereaksi kerangka kerja JS sebagai antarmuka dan Express js dan Laravel sebagai aplikasi yang bekerja di belakang antarmuka. Ide yang kami gagas kami angkat sebagai proyek di dalam perusahaan dengan klien dari perusahaan tersebut. Pada akhirnya program smart locker yang dibuat dengan metode DSDM telah selesai dan siap untuk diimplementasikan di perusahaan dan client.

Kata Kunci: Backend, Frontend, Microservice, Peripherals.

1. Introduction

Lockers is still part of the facility in various aspects such as offices, education, banking, workplaces and other areas that require a place to store goods. Lockers are places for storing goods which are usually used in tourist attractions, libraries, sports venues, or other public places. The function of a locker as a storage area should have a high level of security because what is stored in it are valuables. The security of a locker is very dependent on the door lock. So far, lockers are used using conventional safety locks made of metal. The use of locks like this besides looking old-fashioned in its use is also not effective to ensure the security of goods in the locker. Lockers have become storage areas with various locking mechanisms. The locking mechanism on the lockers has specially evolved with the induction of smart technology [1].

Lockers that can provide services such as safekeeping of goods, sending and receiving packages, filling in merchant top-ups, etc. where there are many definitions for a device that can be said to be smart. But in this article the smart locker in question is a locker that can provide services for sending and receiving packages or goods.

One of the facts is the frequent occurrence of theft and loss of goods at locker rental or borrowing places. Thieves can easily open locker locks using a wire or other imitation keys. In addition, conventional keys are easily duplicated, damaged, and there is even the possibility of being lost or forgotten [2]. Lock locker doors Many lockers rental service providers replace their locker locks with combination padlocks as a solution to make the lockers in their places more secure Using a combination padlock does not necessarily prevent theft of items stored in the locker. twirling the code on the padlock until finally getting the appropriate code. Many people send packages, they will definitely go to the logistics outlet, but this is less efficient because users at the logistics outlet still need human labour to process the packages to be sent, and this eats up quite a lot of the logistics company's budget and the user/sender does not get certainty as to what time the package will be processed by logistics, then when the package is sent to the destination address the courier will deliver it to the destination address, but what if someone's package is not at his house and especially if the user's house is in an apartment. It would be very inconvenient for the courier to take care of this.

Currently there is a startup company engaged in the field of smart lockers for sending and receiving goods called pop-box where several of these lockers are placed in several apartments for collection and delivery of packages [3]. In previous research, a book-lending management system was created using lockers in the web-based Muria Kudus University library using the code-igniter framework [4] followed by research conducted [5] that produced a package custody system by ship using a bootstrap framework using the SDLC Rapid Application Development (RAD) method. Microservices is one type of software architecture where a software is broken down and divided into smaller components and have their respective functions. In the microservices architecture, each function has its own role. Changing one microservice (module/function) does not necessarily change other parts, thereby increasing the scalability of an organization's application systems [6].

By looking at some of the previous research and business processes of a company and then the results of the research and business processes can be used as a common thread to produce solutions to problems in sending and receiving packages, the author has the idea to create a smart locker system for sending and receiving, the next package there is also a maintenance system as well. This system can be implemented in a Kiosk because Kiosk is provided at the author's research location.

2. LITERATURE REVIEW

2.1. Electronic Based Government System

It is a system issued by the Presidential Regulation of the Republic of Indonesia No. 95 of 2018. Where government administration utilizes information and communication technology to provide services to SPBE Users. In implementing SPBE there are several principles that must be implemented, namely: Effectiveness, integration, Continuity, Efficiency, Accountability, Interoperability and Security.
Flowchart is a description of a sequence of procedures in a program in the form of sequential steps displayed graphically. Flow chart "flowchart" is a chart "chart" that describes a flow "flow" in the procedures of a system or program logically. A flowchart is a flowchart that shows a flow in a system procedure or program based on logic which is usually used to help analyze programs and programmers in solving a problem into smaller elements that can help analyze other alternatives in operation.

User Interface Design is an application or website that will be made to look important in visualization which will be a picture for users to interact with the application. The goal is to design a good and effective interface for the application system which means it is ready to be used by the user and can provide results according to user needs. User Experience Design is a process of improving usability and experience in user interaction with products aimed at increasing user satisfaction with the product. The purpose of using UX Design is to make the products made easier for users to use.

2.2. Prototype

Prototype is the stage for implementing ideas that have been made into a product or application to produce a usage scenario according to user needs. Wireframe Low-Fidelity is a basic design which contains content that is simple to start a design. Low-Fidelity wireframes are generally defined by basic elements such as text, buttons, navigation, pictures, etc. Which is useful for communicating, determining very basic functions and the layout of a basic element.

High-Fidelity Wireframe is an advanced design where a detailed version of a simple framework will be created, such as elements added to the framework, namely colors, icons, images, etc. Wireframe Hi-Fi defines an interaction between elements and a visual hierarchy of each page. Hi-Fi is used sequentially to show how certain functions work and can describe the flow of applications / products that can be completed with the correct flow.

2.3. Evaluation

At this test stage an experiment will be carried out with the "user" or users. At this stage the test must be carried out because it will get a "feedback" or experience and input from users to improve the product or application that is implemented.

Usability According to ISO 9421 – 11 (1998) is the extent to which a product can be used by users to achieve certain goals efficiently, effectively and users feel comfortable in the context of its use. In general, the notion of usability is an attribute of quality that is used to evaluate how easy the interface is to use.

3. RESEARCH METHOD

This smart locker system using the Dynamic Systems Development Method (DSDM) (Figure 1). DSDM because this method is an Agile method that is applied in companies where students practice, and this method leads to customer centricity because in making software the role of the client is very much needed. In previous research conducted by [7] using the DSDM method produced a web-based information system for payroll systems with the results of the system being successfully implemented with payroll monitoring and easy processing of salary data for employees in the company where the research was conducted [8]. The DSDM method consists of five phases, namely Feasibility Study, Business Study, Functional Model Iteration, Design [9] and Build Iteration, Implementation [10].

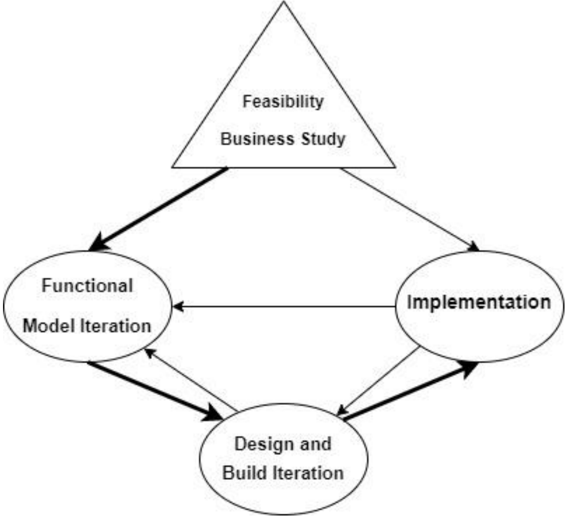


Figure 1 Research Method

Feasibility Study is the phase of selecting the most appropriate solution in software development from the tech stack and software development methodology system. Business Study is a phase of business process analysis that will be applied to software starting from designing the Business Process Model and comparative studies to competitors. Functional Model Iteration is the prototype and ERD design phase so that in this phase there will be quite intense communication between the client and the development team. From this communication that discusses the prototype design, an agreement will be generated that will be applied from the prototype to the programming stage which will be implemented/used by the client later.

Design and Build Iteration is the programming phase of what has been designed, but it is also possible that there will be a design improvement stage due to adjusting requests from the client so that the design here is also slightly modified but programming will still be carried out following the updated design. The output of this stage is a program that has been tested so that at this stage development and testing will be carried out in one phase.

Implementation is the phase of using the program that has been built by the development team so that in this phase the user will implement the program to solve problems based on the problems that have been given at the initial stage. The software architecture used is microservices. Microservices is an architecture that uses separate systems such as frontend and backend with different servers, so each server or service has its own responsibilities. The services that are managed are more measurable and flexible and also more specific in developing the system [11].

In previous research conducted by [12] in building a web-based marketplace with a microservices architecture, it has been successfully implemented with a percentage of 100% successful testing and obtaining optimal results in terms of program appearance, user convenience, and system performance [13]. Then research conducted by Yuri Chandra Tri Putra, et al in building an Islamic religious teaching web using a microservice architecture was successfully implemented and the website was successfully built more efficiently and reduced server workload [14]. Host, backend, frontend, and peripherals stand alone with different servers or ports (**Figure 2**)

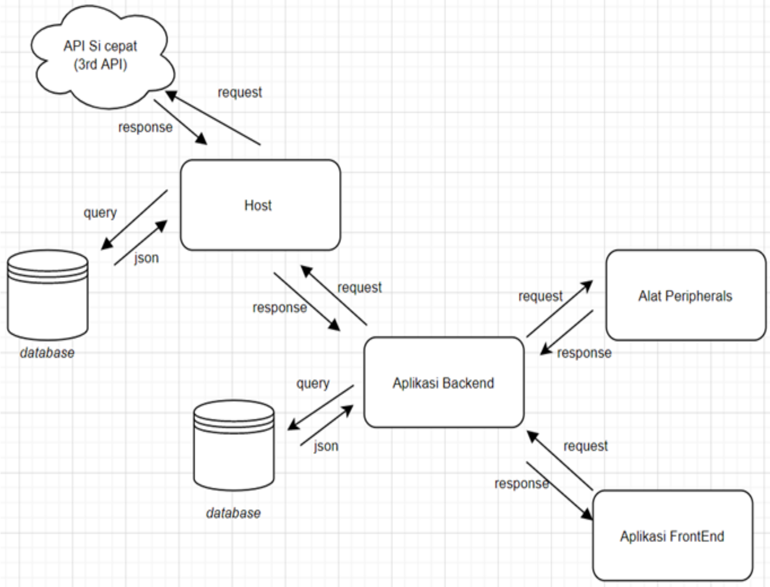


Figure 2 System Block Diagrams

4. RESULTS AND DISCUSSION

The results of this study are a program that is embedded in a kiosk that is integrated with lockers. Then before entering the implementation stage, the black box testing method is used to adjust the client's wishes.
Feasibility Study and Business Study, this phase a solution is created to the problem given by the client through functional requirements and the solution is represented first by the Business Process Model using a User Flow Diagram (Figure 4). Then there is also a use case diagram that has been designed as follows. Functional Model Iteration. In this phase, Entity Relationship Diagram (ERD) and product prototypes are

produced. For ERD it can be seen on the ERD link and the prototype can be seen on the prototype (Figure 3).

Design and Build Iteration, in this phase the code and several design revisions are carried out, the code here is carried out in parallel on the frontend, backend, and peripherals so that the frontend produces a UI that will be interacted with by the user, backend and peripherals produce an endpoint that will be used by the frontend for carry out the business processes that have been defined in the User Flow Diagram. Frontend In the front end we divide into several steps, namely sending customers, picking up customers, and operator menus Figure 5.

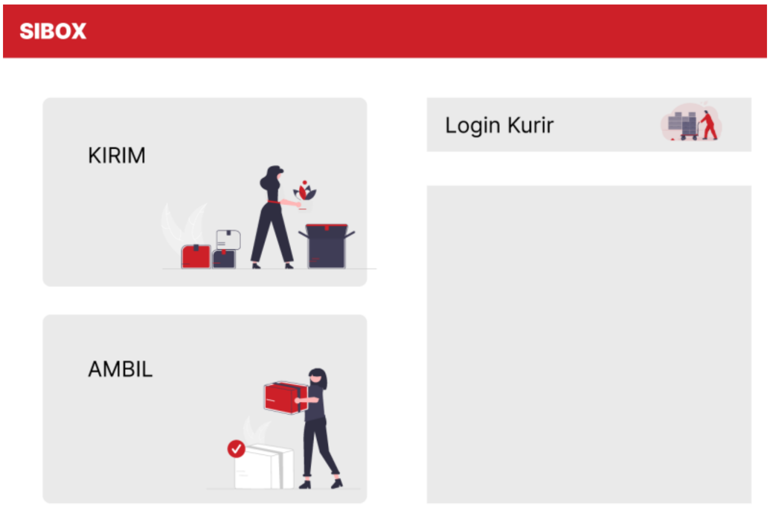
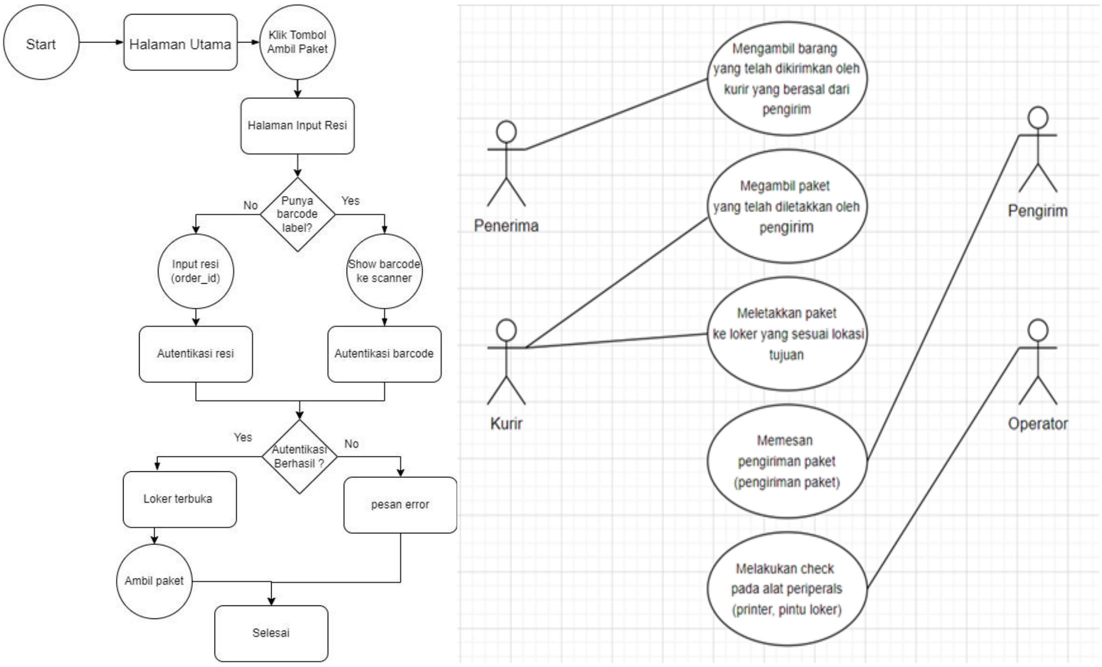
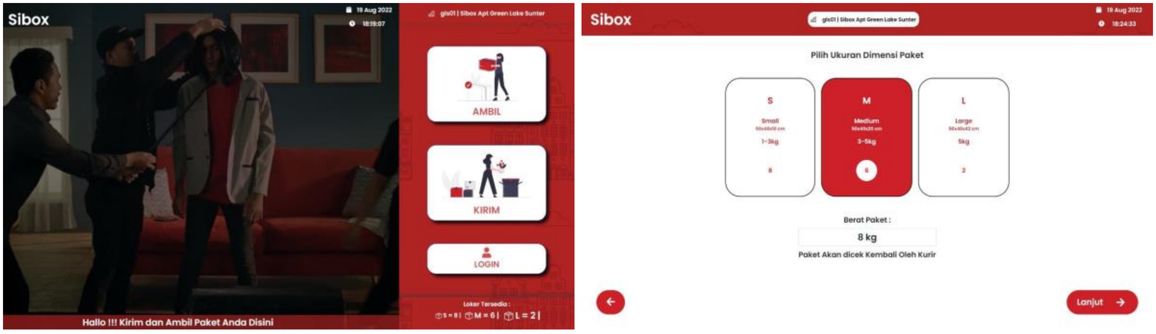


Figure 3 Sibox device display prototype


Start
Halaman Utama
Klik Tombol Ambil Paket
Halaman Input Resi
Punya barcode label?
No
Yes
Input resi (order_id)
Show barcode ke scanner
Autentikasi resi
Autentikasi barcode
Autentikasi Berhasil ?
Yes
No
Loker terbuka
pesan error
Ambil paket
Selesai
Mengambil barang yang telah dikirimkan oleh kurir yang berasal dari pengirim
Megambil paket yang telah diletakkan oleh pengirim
Meletakkan paket ke loker yang sesuai lokasi tujuan
Memesan pengiriman paket (pengiriman paket)
Melakukan check pada alat periperals (printer, pintu loker)
Penerima
Kurir
Pengirim
Operator

(a) (b)

Figure 4 Flowchart Diagram, b. Use case Diagram

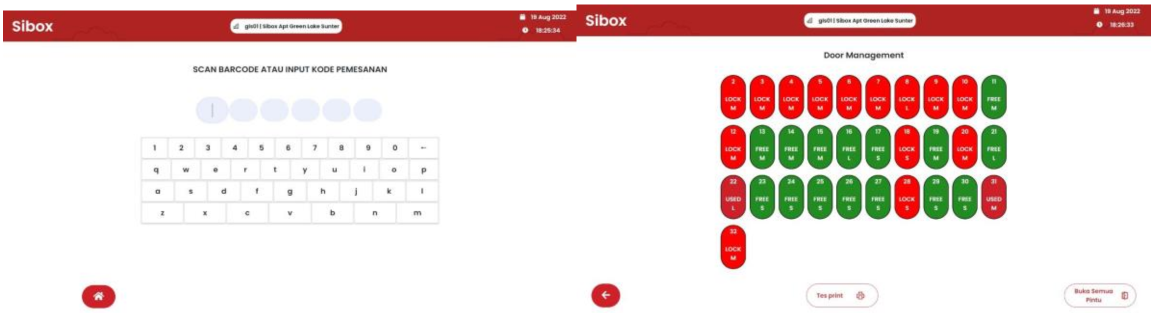

Sibox
19 Aug 2022
18:19:07
gls01 | Sibox Apt Green Lake Sunter
AMBIL
KIRIM
LOGIN
Loker Tersedia :
S = 8 | M = 6 | L = 2 |
Hallo !!! Kirim dan Ambil Paket Anda Disini
Sibox
gls01 | Sibox Apt Green Lake Sunter
19 Aug 2022
18:24:33
Pilih Ukuran Dimensi Paket
S
Small
1-3kg
8
M
Medium
3-5kg
6
L
Large
5kg
2
Berat Paket :
8 kg
Paket Akan dicek Kembali Oleh Kurir
Lanjut

(a) (b)

Figure 5 a. Landing Page for Each User, b. The process of ordering package delivery

Process the customer arrives with the package to be sent and follows the steps in the application, such as filling in the sender's data, recipient's data along with the destination address, dimensions and weight of the package, and type of delivery. customers before being able to put the package into the locker will be asked to pay, in this case you use payment via QRIS.

Customer Pick Up, at this stage the customer wants to pick up the package that has been sent to Sibox, in this process the customer has brought the order code that he got immediately after the package was sent. The order code here functions to open an existing locker to pick up the package. Operator Menu This menu is a special menu used for operators. This menu functions to check the existing doors, in this menu you can also see which doors are being used. there is also a button to check peripherals to open all doors in one click of a button and check print in one click of a button Figure 6.



(a) (b)

Figure 6 a. Collection Code, b. Locker Maintenance Menu

Testing method, for program testing, the author uses black box testing with the lead engineer at the company where the research takes place. The following are the results of the black box testing that has been carried out. Implementation, in the end the program created by the author can be completed by installing the program into the KiosK and it is accepted with a sign that the client agrees and will discuss further regarding the smart locker program and the company agrees to complete this project. For program testing, the authors used black box testing with the lead engineer at the company where the research took place [15]. The following are the results of the black box testing that has been carried out (Table 1).

Table 1 Black Box Testing

Code	Scenario	Status
Code-01	The recipient managed to pick up the goods at the locker.	Success
Code-02	The courier picks up the package using a 6-digit pick-up code.	Success
Code-03	Courier puts the parcel into the appropriate destination locker.	Success
Code-04	The sender places an order for package delivery	Success
Code-05	Operators log in	Success
Code-06	The operator checks the status of each door in the locker (starting from opening, deactivating the door) after logging in.	success

Code-07	The host synchronizes data each locker	Success
Code-08	Local Server does not synchronize if the host error.	Success
Code-09	Every transaction in the locker will be stored by the locker program	Success
Code-10	The frontend will give an error if the local server has an error	Success

The results of testing the system using the System Usability Scale (SUS) method according to 10 respondents are as follows (Cahyo Untoro, 2020). Based on the survey and score calculation that has been carried out, an SUS score of 7.76.

5. Conclusion

From the creation of this smart locker system, it can be concluded that the SDLC with agile techniques using the Dynamic System Development (DSDM) method was successfully implemented and the program was finally completed. The program can solve problems that occur on the client by using the DSDM method. It is proven by carrying out the steps in this method, the results of the project are successful, marked by the client having carried out a demonstration and the company giving a sign of success. The smart locker system used for sending and receiving packages and maintenance of integrated tools in the locker has been completed by inserting the software into the KiosK in the locker and is ready to be implemented further and get a value of SUS 7.76.

Table 2 System Usability Scale Testing

R	Q 1	Q 2	Q 3	Q 4	Q 5	Total	Average
1	4	5	5	5	4	23	7.6
2	5	5	5	4	5	24	8
3	4	5	5	4	3	21	7.2
4	4	5	5	5	4	23	7.6
5	5	5	5	4	5	24	8.4
6	4	5	5	5	4	23	7.6
7	5	5	5	4	5	24	8
8	4	5	5	5	4	23	8
9	5	5	5	4	5	24	8.4
10	4	5	5	4	3	21	6.8
			Average				7.76

DAFTAR PUSTAKA

[1] F. Suryatini, R. R. Firdaus, S. Rodiah, and A. H. Mardiansyah, “RANCANG BANGUN SISTEM PENGUNCIAN LOKER OTOMATIS MENGGUNAKAN GOOGLE ASSISTANT DAN E-KTP BERBASIS INTERNET OF THINGS Jurusan Teknik Otomasi Manufaktur dan Mekatronika , Politeknik Manufaktur Bandung , Bandung Abstrak,” pp. 168–173, 2021.

[2] I. Komang, “Rancang Bangun Sistem Pengunci Loker Otomatis Dengan Kendali Akses Menggunakan Rfid Dan Sim 800L,” *J. Ilm. Mhs. Kendali dan List.*, vol. 1, no. 1, pp. 33–41, 2020, doi: 10.33365/jimel.v1i1.187.

[3] F. F. Luthfi and D. M. Midyanti, “Sistem Keamanan pada Loker Berbasis Internet of Things,” vol. 07, no. 03, pp. 200–206, 2022.

[4] PIPIN FIBRIAWAN, “SISTEM PENGELOLAAN PEMINJAMAN LOKER DI PERUSTAKAAN UNIVERSITAS MURIA KUDUS BERBASIS WEB,” Universitas Muria Kudus, 2015.

[5] L. Suryani and R. F. Waliulu, “PENITIPAN PAKET BERBASIS WEBSITE MENGGUNAKAN FRAMEWORK BOOTSTRAP (Studi Kasus Anbiya Jastip) DESIGN AND IMPLEMENTATION OF A WEBSITE-BASED PACKAGE INFORMATION SYSTEM USING BOOTSTRAP FRAMEWORK (Case Study of Anbiya Jastip),” vol. 7, no. 2, 2021.

[6] A. Purwanto *et al.*, “SISTEM INFORMASI DASHBOARD DIGITAL BADAN PUSAT STATISTIK,” vol. 2, no. 1, pp. 1–9, 2023.

[7] W. Nyunando and D. Nasien, “Implementasi Agile Dynamic System Development Method

Berbasis Web Pada Sistem Penggajian," *J. Mhs. Apl. Teknol. Komput. dan Inf.*, vol. 2, no. 1, pp. 33–38, 2020.

[8] M. Cahyo Untoro, "Implementation of User-Centered Design in Online Seminar Application," *Pixel J. Ilm. Komput. Graf.*, vol. 13, no. 1, pp. 1–8, 2020, doi: 10.51903/pixel.v13i1.152.

[9] T. Tumini and S. Sugiyanti, "Penerapan Dynamic System Development Method Pada Sistem Monitoring Status Gizi Balita," *Informatics Digit. Expert*, vol. 2, no. 1, pp. 7–13, 2020, doi: 10.36423/ide.v2i1.426.

[10] D. Ayu, N. Wulandari, M. Dika Atthariq, W. D. Nanda, and L. Yusuf, "IMPLEMENTASI DYNAMIC SYSTEM DEVELOPMENT METHOD (DSDM) PADA SISTEM INFORMASI MANAJEMEN BENGKEL MOBIL BERBASIS WEB," *Sist. Inf. |*, vol. 8, no. 1, pp. 10–17, 2021.

[11] A. Sinambela, Ernawati, and F. Farady Coastera, "Implementasi Arsitektur Microservices Pada Rancang Bangun Aplikasi Marketplace Berbasis Web," *J. Rekursif*, vol. 9, no. 1, pp. 1–13, 2021.

[12] S. Li *et al.*, "Understanding and addressing quality attributes of microservices architecture: A Systematic literature review," *Inf. Softw. Technol.*, vol. 131, p. 106449, 2021, doi: 10.1016/j.infsof.2020.106449.

[13] M. Innes *et al.*, "A Differentiable Programming System to Bridge Machine Learning and Scientific Computing," 2019, [Online]. Available: http://arxiv.org/abs/1907.07587

[14] Yuri Chandra Tri Putra, Thomas Adi Purnomo Sidi, and Joseph Eric Samodra, "Implementasi Arsitektur Microservice pada Aplikasi Web Pengajaran Agama Islam Home Pesantren," *J. Inform. Atma Jogja*, vol. 1, no. 1, pp. 88–97, 2020.

[15] G. Todingrante, Sudirman, and Afifah, "IMPLEMENTASI BLACK BOX TESTING PADA WEBSITE," *J. ILMU Komput. KHARISMA.TECH*, vol. 17, no. 01, pp. 135–148, 2022, doi: 10.55645.