Implementation and Analysis of Service-Based Integrated Platform for Research Center Using Thomas Erl Method

Muhammad Adriyan Putra1, Rahmat Yasirandi, Dita Oktaria

School of Computing, Informatics Study Program, Telkom University, Bandung, Indonesia
Email: 1adriyanputra@student.telkomuniversity.ac.id, 2batanganhitam@telkomuniversity.ac.id, 3dioktaria@telkomuniversity.ac.id
Correspondence Author Email: adriyanputra@student.telkomuniversity.ac.id

Abstract—The process of developing a platform is a process that consists of various stages. Still, until now, there has been no awareness in utilizing available data and information sources to serve as the basis for developing or creating a platform that can improve the quality of an integrated system. For this reason, a Platform as a Service (PaaS) architecture was built that provides application development and deployment services to process data and information obtained from relevant agencies at Telkom University to monitor the performance achievements of units within Telkom University, especially the Human Centric Engineering Research Center (HUMIC RC). This requires a service-oriented system platform design that can integrate various diverse information systems into a single unit. In this study, the Service Oriented Architecture (SOA) method from Thomas Erl was used to design a service-oriented system platform, where every business process is built in the form of services. REST Service Modeling Process technology is used in modeling services, which produces an integrated service design. In making the platform architecture this time, the API gateway is one of the middleware used on a microservice-based platform. The service-oriented platform that has been built can assist the Institutional Planning, Development, Control or Perencanaan, Pengembangan, Pengendalian Institusi (P3I) unit of Telkom University in monitoring the performance achievements of units within Telkom University so that there is no duplication of data, facilitates the storage of evidence of accomplishments, and reduces the stage and time of the process of searching for proof of performance achievements.

Keywords: Cloud Computing; Platform as a Service; Service Oriented Architecture (SOA); REST Service Modeling Process; API

1. INTRODUCTION

Cloud computing is a computing model in which processing, storage, networking, and software resources are made available as services on the internet. Cloud computing also has several services, such as Infrastructure as a Service (IaaS), which provides storage services, Platform as a Service (PaaS), which provides platform services for creating applications, and Software as a Service (SaaS) which offers ready-made applications to clients [1].

Cloud computing can also be applied in various ways, one example is the monitoring of unit performance at Telkom University. Telkom University has many research center units, one of which is the Research Centers for HUMIC Engineering. Every three months, monitoring of unit performance achievements within Telkom University is carried out. This is where many aspects are assessed, and these aspects involve many related units, such as the directorate of finance, directorate of cooperation, faculties, and other associated units. However, of the many associated units, no service-oriented system platform can integrate various diverse information systems to connect and provide good service to every user who uses the platform.

Using PaaS, the system provider makes most of the choices determining how the application infrastructure operates, such as the type of OS used, APIs, programming languages, and management capabilities [2]. With the existence of a service-based platform using the PaaS concept, which aims to provide cloud computing-based data storage and processing services that can be accessed by the Institutional Planning, Development, Control or Perencanaan, Pengembangan, Pengendalian Institusi (P3I) unit of Telkom University and every unit within Telkom University, then a more efficient, low-budget data storage container can be made, and every service provided can run well. With the creation of a service-based platform, it is hoped that in the future, both P3I Telkom University and each unit within Telkom University will be able to improve the quality and the performance of each unit in the Telkom University environment.

In this study, the Service Oriented Architecture (SOA) method from Thomas Erl was used in designing a service-based system platform. The SOA method is a solution to design data integration from several different systems [3]. SOA is a technology architecture-based in software development [3]. Implementing SOA on various information systems will run well and be able to provide significant benefits if it is supported by the existence of regulations in the SOA implementation process [4].

Analysis of information system integration with SOA has been done in previous studies. One of them is In 2015 [5]. This study implements the Service Oriented Architecture (SOA) method using web service technology to implement a system based on data integration. Making Web Services utilizing the SOA method, which entails breaking down service packages into manageable components that can be updated on a regular basis. This SOA method will also be capable of integrating a variety of other packaged services that have been developed [6]. The application of SOA using web service technology is carried out by connecting the Ministry of Religion Data Collection application and several PDTA applications. Several agencies have benefited from the trial of SOA-based applications, including the PDTA data collection process that can be carried out at any time without being tied to a specific time.
In the following research [7], an architectural design and prototype of an integrated system with the SOA approach and MVC method was built and applied to the Research Center for the Development of Science and Technology, the Indonesian Institute of Sciences. The design in this study is an architecture with an SOA approach implemented using REST technology which produces 4 (four) services, namely personnel services, asset services, inventory services, and financial services. In addition to the results of the SOA architecture design, this research also produced an integrated system prototype as a dashboard for web-based services.

In another research on SOA was conducted in 2019 [8]. In this research, a platform was built to improve the quality of an integrated system. For this reason, a Platform as a Service (PaaS) architecture was created which provides application development and deployment services to process data and information obtained from practicum activities during the lecture period based on cloud computing using the Service Oriented Architecture (SOA) method. In making the platform architecture this time, the API gateway is one of the middleware used on a microservice-based platform. The results of the implementation and analysis were carried out to prove that the architecture using the API gateway as the middleware built can be considered to develop a laboratory service system for Telkom University practicum.

Then, as for research [9], an information system integration design was built using the Service Oriented Architecture (SOA) methodology. Where every business process is made in the form of services. REST Service Modeling Process technology is used in modeling services, which produces an integrated service design. The built services are then tested for priorities using a quality attribute scenario based on the characteristics of the ISO 25010 model, namely compatibility, maintainability, reliability, and performance with expert respondents in the SOA field. The test results show that all scenarios are included in the tolerable criteria, which means that all methods can be applied.

Another research in 2015 [10]. The study's findings include android-based applications that use web services to construct Service Oriented Architecture to combine academic information systems, library information systems, and personnel information systems, making data and information sharing easier and faster.

In further research [11]. It has built an integrated car rental information system using Service Oriented Architecture (SOA) by implementing Web Service. With an integrated car rental information system, it is hoped that car rentals can exchange car inventory data and information on problematic borrowers. In order for the car rental procedure to become faster, more effective, and efficient, it must be able to aid and facilitate borrowers in finding a car.

Further research on SOA [12], an e-Commerce service was built using the SOA method. In this research, the leading service is designed and implemented in e-Commerce services using web services. The conclusion of this research is the design and implementation of e-Commerce for organizations whose business processes continue to change and develop following the precise needs of using analysis and design based on Service Oriented Architecture using Web Services.

2. RESEARCH METHODOLOGY

2.1 System Design

The research workflow shown in Figure 1 is a research stage that can be interpreted as a series of processes from start to finish. The first process is business process analysis based on each activity of the HUMIC research center. Next, the SOA design is modeled using REST Modeling Process technology. The next process is service development using the Express JS framework. After the service is built, testing is carried out in various types of tests. The test results are then analyzed according to the parameters that have been determined until the conclusion stage is completed.

![Figure 1. Research Workflow](image)

2.2 Business Process Analysis

The development of this service-based platform is based on every business process in the HUMIC Research Center. Each of these business processes is grouped from each activity of the HUMIC Research Center. In this study, we collected data and information by interviewing resource persons to obtain the business processes of each HUMIC Research Center activity. The business processes produced are Financial Business Processes, Research Business Processes, Cooperation Business Processes, Human Resource (HR) Management Business Processes, and Performance Business Processes.

2.3 Service Oriented Architecture Design

Service-based platforms are integrated using the SOA method. Each business process is built-in services that the requestor can later access. In modeling services on SOA, the REST Service Modeling Process technology is described in Figure 2 as follows [13]:

Muhammad Adriyan Putra, Copyright © 2022, MIB, Page 828
Submitted: 24/01/2022; Accepted: 04/03/2022; Published: 25/04/2022
2.3.1 Decompose Business Process

This stage aims to divide an enormous responsibility into several parts for easy completion [13]. Therefore, decomposition is carried out at this stage to describe the business process into detailed entities to form service candidates [13]. The business processes and the formation of other service candidates are shown in Table 1.

| Business Process | Decompose Business Process | Service Candidate |
|------------------|-----------------------------|-------------------|
| Registration     | Create an account on the system by adding a username and password. Data entry. | Registration Service |
| Login            | Enter the username and password. Login verification. | Login Service |
| Financial        | Submit a monthly budget plan. Hold a conference. Hold workshops. Set income from RC. Set outcome from RC. Measuring the income and expenses of the RC. | Financial Service |
| Research         | Formation of the research team Paper is submitted to the Journal. Paper is submitted to the conference. Paper accepted in the journal. Paper accepted at the conference. | Research Service |
| Cooperation      | Cooperation Program with Industry / companies / government / universities. Implemented Cooperation Program. Publication of Tel-U image enhancement program through the RC. | Cooperation Service |
| HR Management    | Doing recruitment. Manage members’ data. | HR Management Service |
| Performance      | Review of management contract achievements. Provide a final assessment of whether the unit has complied with all management contracts. | Performance Service |

2.3.2 Filter out unsuitable actions

At this stage, the business process entities are separated, which currently, the process is still done manually [13]. In this study, there are business process entities whose operations are still carried out manually at an institution, namely forming a research team, conducting recruitment, reviewing the achievements of management contracts, providing a final assessment of whether the unit has fulfilled all management contracts.
2.3.3 Define Entity Service Candidates

At this stage, a separation is made between services that are agnostic and non-agnostic [13]. Agnostic services are services that can help other business processes, while non-agnostic services are services that have specific goals [13]. Agnostic services in this study are grouped to form entity service candidates, as shown in Figure 3.

Figure 3. Entity Service Candidates

2.3.4 Identify Process-Specific Logic

At this stage, identifying the service that will start a system is carried out [13]. In this research, the service candidate who starts this information system is the service candidate Monitor Unit with the start entity, as shown in Figure 4. The service candidate Monitor Unit will be encapsulated into the service layer task.

Figure 4. Service Candidate Monitor Unit

2.3.5 Define Microservice Candidates

At this stage, non-agnostic services are defined as microservice candidates because they have a specific purpose, namely to perform a logical verification process. The identified non-agnostic service is Service Login, as shown in Figure 5.

Figure 5. Microservice Candidate
2.3.6 Identify Resource

At this stage, it is identified how services relate to resources [13]. The resource is a data store identified using a slash (“/”). Slash (“/”) is used to determine the service part associated with the resource. Table 2 is the relationship between the service and the resources contained in the Monitor Unit.

**Table 2. Service Relationship with Resource on RC Monitoring**

| Service                      | Resource                  |
|------------------------------|---------------------------|
| Registration Service         | /Registration/           |
| Login Service                | /Login/                  |
| Financial Service            | /Financial/              |
| Cooperation Service          | /Cooperation/            |
| HR Management Service        | /HRManagement/           |
| Research Service             | /Research/               |
| Performance Service          | /Performance/            |

2.3.7 Associate Service Capabilities with Resource and Methods

After determining the resource, the next step is to associate each action on the business process entity with the HTTP method [13]. The HTTP methods used in this study include GET, POST, PUT, and DELETE, listed in Table 3 below.

**Table 3. Associate Service Capabilities with Resource and Methods**

| Service                      | Resource                  | Resource and Method                                                                 |
|------------------------------|---------------------------|--------------------------------------------------------------------------------------|
| Registration Service         | /Registrasi/              | Create an account on the system by adding a username and password (POST + /Registration/CreateAccount) |
|                              |                           | Data entry (POST + /Registration/EnterRegistrationData)                             |
|                              |                           | Enter the username and password (GET + /Login/)                                      |
|                              |                           | Login verification (GET + /Login/LoginVerification)                                  |
|                              |                           | Submit a monthly budget plan (POST + /Financial/SubmitBudgetPlan)                    |
|                              |                           | Hold a conference (POST + /Financial/Conference)                                     |
| Login Service                | /Login/                  | Hold workshop (POST + /Financial/Workshop)                                          |
|                              |                           | Set income from RC (POST + /Financial/IncomeRC)                                     |
|                              |                           | Set outcome from RC (POST + /Financial/OutcomeRC)                                   |
| Financial Service            | /Financial/              | Measuring the income and expenses of the RC (GET + /Financial/MeasureIncomeOutcome)  |
|                              |                           | Formation of the research team (POST + /Research/FormationResearchTeam)              |
|                              |                           | Paper is submitted to the Journal (POST + /Research/SubmitPaperJournal)              |
|                              |                           | Paper is submitted to the conference (POST + /Research/SubmitPaperConference)        |
|                              |                           | Paper accepted in the journal (GET + /Research/PaperAcceptedJournal)                 |
|                              |                           | Paper accepted at the conference (GET + /Research/PaperAcceptedConference)            |
| Research Service             | /Research/               | Cooperation Program with Industry / companies / government / universities (POST + /Cooperation/CooperationProgram) |
|                              |                           | Implemented Cooperation Program (GET + /Cooperation/ImplementedCooperationProgram)    |
|                              |                           | Publication of Tel-U image enhancement program through the RC (POST + /Cooperation/PublicationEnhancementProgram) |
| Cooperation Service          | /Cooperation/            | Doing recruitment (POST + /HRManagement/Recruit)                                     |
| HR Management Service        | /HRManagement/           | Manage members data (GET + /HRManagement/ManageData)                                 |
| Performance Service          | /Performance/            | Review of management contract achievements (POST + /Performance/ReviewAchievements)  |
|                              |                           | Provide a final assessment of whether the unit has complied with all management contracts (POST + /Performance/FinalAssessment) |

2.3.8 Identify Candidate Service Compositions

The service candidates are grouped into three layers at this stage, namely the task service layer, microservice layer, and entity service layer [13]. In Figure 6, the service candidate in the service layer task will run an information
system. Service candidates in the microservice layer contain non-agnostic services with a specific purpose. The service candidate in the entity service layer is an agnostic service.

3. RESULT AND DISCUSSION

3.1 Service Development

At this stage, the service candidate that has been modeled is built with REST technology using the Express JS framework. Express JS provides all the building blocks needed to create web applications and microservices [14]. Each service on the platform already has its API, registered, and can be searched or found, as shown in Figure 7. The author uses Postman as a design for every existing API to register and use it.

3.2 Test Scenario

In this research, testing is carried out so that the functionality of each service on the system can run well and can be used. The tests carried out are Compatibility Testing, Performance of Scalability Testing, Connectivity Testing, and Reliability Testing [15]. The first test is Compatibility Testing, this test focuses on the possible level of service from the platform that can be accepted by various types of programming languages used by users. The testing procedure will be tried through several programming languages used to access services on the platform [16]. The second test is Performance Testing, this test aims to see the system's resilience by testing network performance in data distribution [16]. The third test is Connectivity Testing, this test seeks to determine how the response is received by the user if there is a problem with the connectivity platform used [17]. The last test is Reliability Testing, this test aims to find out each service can be used under any circumstances [17]. The scenario of the test can be seen in Table 4 below.
### Table 4. Test Scenario

| Test Type         | Precondition       | Criteria                                                                 | Test Procedure                                                                 | Expectation                                                                 |
|------------------|--------------------|--------------------------------------------------------------------------|--------------------------------------------------------------------------------|----------------------------------------------------------------------------|
| Compatibility    | Server status is running | The user requests services according to the programming language         | Each service server will make a user service request                           | Service requests can be accessed through several programming languages     |
| Performance       | Server status is running | See the value of Memory and CPU usage                                     | Seeing service ability in providing required reactions when the platform is running | Service provides a fast response when retrieving and sending data.          |
| Connectivity      | Server status is running | The connection on the server is turned off                                | Server that was previously running directly shut down                          | All existing services cannot be accessed                                    |
| Reliability       | Server status is running | The service is running, and the system is restarting/reloading           | When the user is accessing the platform, the system restarts/reloads           | The system can still be used when the server restarts/reloads               |

### 3.3 Test Result

In this test, we carried out several stages of testing, which will be explained as follows:

a. **Compatibility Testing**
   
   The test results show that platforms can be accessed with several different programming languages. It can be seen in Figure 8 on the SOA proposal architecture using the JS framework on the back-end and can use various programming languages on the front-end.

![Figure 8. Results of Compatibility Testing](image8)

b. **Performance Testing**
   
   The results of the Performance testing test are divided into two results in the form of memory and CPU usage. From these two results, it can be seen that the memory usage generated when the user accesses the server is relatively small, and the platform can still be accessed commonly. As shown in Figure 9, the CPU usage generated by each user is 0.20% of the CPU usage value, this shows that the service provides a fast response when retrieving data.

![Figure 9. Results of Performance Testing](image9)

c. **Connectivity Testing**
   
   According to the testing scenario in Table 4, the expectation is that each service available on the platform cannot be executed. When the server is running according to Figure 10, and the platform created can be...
accessed in Figure 11, then the server that initially lights up is shut down suddenly in Figure 12. With the turn off the server on the platform created, making each service that is running into irreversible and the platform cannot be used as in Figure 13. The platform built is monitored using PM2 monitoring tools for the following testing.

![Figure 10. Test Results when Server Status is Running](image1)

![Figure 11. Test Results when the Platform Can Be Accessed](image2)

![Figure 12. Test Results when the Server is Shut Down](image3)

![Figure 13. Test Results when the Platform Can't Be Accessed](image4)
d. Reliability Testing

According to the test scenario in Table 5, the expectation is that every service available on the platform can still be run even if it is restarted or reloaded on the server. According to Figure 14, the running server will be restarted or reloaded when running. Restarting or reloading the platform server that was created will not significantly affect the platform on the platform being accessed by the user, as shown in Figure 14.

![Figure 14. Platform test results when the server is restarting/reloading](image)

### 3.4 Analysis Of Test Result

The test results show that all tested scenarios meet all aspects of assessing quality attributes, so this service-based platform is excellent and feasible to use in building a unit monitoring service system at Telkom University.

On this service-based system platform, it can be seen that each existing service already uses its API, where the API Gateway serves as middleware in directing user needs to each current service so that communication between the user and each service provided can run well. According to what the user wants. API Gateway is responsible for determining each user request route on each service, composition, and translation of each protocol. All service requests from users must go through the API Gateway first, then be routed according to the appropriate service. With the API Gateway as middleware, every existing service is integrated with each other. The user does not have to know how the communication between each service is, the user only needs to choose what service is required so that the API Gateway will create a communication route for each service so that the user gets the desired result.

### 4. CONCLUSION

In this study, we designed a service-based system platform with the SOA methodology, where every business process is built in the form of services. REST Service Modeling Process technology is used in modeling services, which produces an integrated service design. The services that are made are then tested using quality attribute scenarios, namely compatibility, performance scalability, connectivity, and reliability. Based on the results of this study, there is a simplification of the process to provide data and information to the user. So that the monitoring and evaluation process can be done quickly. Timeliness in providing data and information is increasing. Furthermore, the proposed service uses SOA Architecture to ensure data and information availability. This study's use of SOA makes it simple to integrate systems, data, and business processes.

### REFERENCES

[1] I. Gusti Ngurah Wina Dharma, I. Made Sukarsa, and N. Putu Sutramiani, “Rancang Bangun Sistem E-Commerce Marketplace Gypsum Berbasis Cloud Computing,” *J. Ilm. Merpati (Menara Penelit. Akad. Teknol. Informasi)*, vol. 7, no. 1, p. 37, 2019, doi: 10.24843/jim.2019.v07.i01.p05.
B. Kahanwal and T. P. Singh, “The Distributed Computing Paradigms: P2P, Grid, Cluster, Cloud, and Jungle,” Int. J. Latest Res. Sci. Technol. ISSN, vol. 1, pp. 183–187, [Online]. Available: http://www.mkjournals.com/ijlorst.htm.

J. B. Wijanarko and R. Afwani, “Integrasi Sistem Manajemen Tata Usaha Dengan Metode Service Oriented Architecture (SOA) Pada SMP Negeri 10 Mataram,” J. CoreIT J. Has. Penelit. Ilmu Komput. dan Teknol. Inf., vol. 1, no. 1, pp. 18–16, 2015.

R. Asmara, J. A. Nur Hasim, and A. P. Utama, “Integrasi E-Government Kabupaten Sidoarjo dengan Service Oriented Architecture (SOA),” INOVTEK Polbeng - Seri Inform., vol. 5, no. 1, p. 16, Jun. 2020, doi: 10.35314/ISL.V5I1.1094.

W. Warkim and D. I. Sensuse, “Model Integrasi Sistem dengan Pendekatan Metode Service Oriented Architecture dan Model View Controller pada Pusat Penelitian Perkembangan Iptek Lembaga Ilmu Pengetahuan Indonesia,” J. Tek. Inform. dan Sist. Inf., 2017, doi: 10.28932/jutisi.v3i1.572.

Joel Andrew M K Ginting, “Implementasi dan Analisis API Gateway sebagai Middleware Pada Platform as a Service (Studi Kasus: Sistem Layanan Laboratorium Praktikum),” 2019.

A. Firman Daru and A. Nugroho, “Hybrid Mobile Application Dengan Metode Service Oriented Architecture,” 2019. [Online]. Available: www.tracer.usm.ac.id.

R. Hilmawan and I. Iskandar, “Aplikasi pendataan PDTA se-kota pekanbaru berbasis service oriented architecture menggunakan teknologi web service,” J. CoreIT J. Has. Penelit. Ilmu Komput. dan Teknol. Inf., vol. 1, no. 1, pp. 10–16, 2015.

A. Tunggul, S. Christanto, and R. Kurniawati, “Penerapan Service Oriented Architecture Menggunakan Web Service Pada Aplikasi Perpustakaan Berbasis Android 7.5,” 2015.

A. Kurniawan, A. A. Nugroho, and S. Mulyono, “Sistem Informasi Rental Mobil Terintegrasi Menggunakan Service Oriented Architecture,” TRANSISTOR Elektro dan Inform., vol. 2, no. 2, pp. 134–142, 2018, [Online]. Available: http://jurnal.unissula.ac.id/index.php/El/article/view/3053/2216.

W. Warkim and D. I. Sensuse, “Model Integrasi Sistem dengan Pendekatan Metode Service Oriented Architecture dan Model View Controller pada Pusat Penelitian Perkembangan Iptek Lembaga Ilmu Pengetahuan Indonesia,” J. Tek. Inform. dan Sist. Inf., 2017, doi: 10.28932/jutisi.v3i1.572.