

Analisa Kebutuhan dan Perancangan Sistem Informasi Produksi Dinas Pternakan dan Kesehatan Hewan Kabupaten Malang Berbasis Teknologi *Service Oriented Architecture*

Agum Septian Gumelar¹, Mochammad Chandra Saputra², Niken Hendrakusma Wardani³

Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Brawijaya

Email : ¹ggagoem94@gmail.com, ²andra@ub.ac.id , ³niken.tif@gmail.com.

Abstrak

Perancangan sistem merupakan suatu tahap yang menjadi dasar pengembangan sistem. Dinas Peternakan dan Kesehatan Hewan Kabupaten Malang merupakan salah satu instansi pemerintahan yang menjadi sumber informasi yang mencakup segala bidang peternakan. Bidang produksi merupakan salah satu bidang penting yang memiliki informasi tentang perkembangan ternak, pakan, dan budidaya yang ada di kabupaten malang. Kegiatan pelaporan menjadi salah satu kegiatan inti dalam bidang ini. Dalam pelaporan sering terjadi kerancuan data dan ketidak efisienan waktu karena harus melakukan rekap data dua kali. *Sharing* data juga sering menjadi kendala yang mengakibatkan informasi yang kurang *up to date*. Dengan adanya sebuah sistem yang berbasiskan SOA, maka dapat memaksimalkan kegiatan pelaporan di bidang produksi ini. Sehingga sebuah penelitian dilakukan untuk menganalisis kebutuhan, melakukan perancangan sistem, memastikan konsistensi, dan memastikan memenuhi beberapa kriteria SOA. Penelitian ini diawali dengan proses pengumpulan data menggunakan teknik observasi, wawancara, dan studi pustaka untuk mendapatkan data awal mengenai kebutuhan pengguna. Langkah selanjutnya adalah proses analisis dan perancangan menggunakan metodologi *Ripple* yang akan menghasilkan definisi kebutuhan dan artefak perancangan, kemudian dilakukan proses pengujian perancangan dengan menggunakn *Consistency Analysis* untuk mengetahui konsistensi dari perancangan dan *Service Litmus Test* untuk mengetahui apakah perancangan sudah memenuhi beberapa kriteria SOA. Pengujian konsistensi pendefinisian kebutuhan akan menghasilkan sebuah nilai RCI (*Requirement Consistency Index*) dan memenuhi persamaan pada *business alignment* serta *reusable* pada *litmus test*. Analisis kebutuhan menghasilkan *system use case* sebagai definisi dari kebutuhan fungsional sistem dan *supplementary requirements* sebagai definisi dari kebutuhan non-fungsional sistem. Perancangan sistem menghasilkan artefak perancangan, meliputi: *user interface sketch*, *communication diagram*, *class diagram*, *sequence diagram*, dan *database schema*. Pengujian konsistensi pendefinisian kebutuhan menghasilkan nilai RCI sebesar 100% yang berarti pendefinisian kebutuhan sistem sudah 100% konsisten. Pengujian *litmus test* menghasilkan perancangan sudah memenuhi *business alignment* dan sudah memenuhi kriteria *reusable*

Kata kunci: pelaporan, SOA, *Ripple*, *Consistency Analysis*, *Service Litmus Test*

Abstract

A design system is a fundamental phase of system development. Department of Animal Husbandry and Animal Health Malang Regency is one of the government agencies are a source of information that covers all areas of the animal husbandry. Production department is one important area that has information about the development of livestock, feed and cultivation in poor districts. Event reporting to be one of the core activities in this department. Often a confusion in the reporting data and inefficiencies of time having to do a recap of the data twice. Sharing data is also often a constraint resulting in less up to date information. Given an SOA-based system, it can maximize the activity of reporting in this production field. Thus study was conducted to analyze the requirements, designing the system, ensure consistency, and ensure it meets several criteria SOA. This study begins with data collection process using observation, interview, and literature study to obtain preliminary data of the user's needs. The next step is to do an analysis and design using Ripple methodology to produce requirements definition and artifacts design, then do the testing process design by using Consistency Analysis to determine the consistency of the design and the Service Litmus Test to determine if the design

had to meet several criteria SOA. Testing consistency defining requirements will result in a value of RCI (Requirement Consistency Index) and satisfies the equation on business alignment and reusable on a litmus test. Requirement analysis generates use case as the definition of the functional system requirements and supplementary requirements as the definition of non-functional system requirements. Designing the system generates artifact design, include: user interface sketch, communication diagram, class diagram, sequence diagram, and database schema Testing consistency defining requirements resulted in RCI value of 100%, which means defining the needs of the system is 100% consistent. Testing litmus test resulted in the design meets the business alignment and meets the criteria reusable.

Keywords: reporting, SOA, Ripple, Consistency Analysis, Service Litmus Test

1. PENDAHULUAN

Dalam rangka pelaksanaan Peraturan Pemerintah Nomor 8 Tahun 2008 tentang Tahapan, Tata Cara Penyusunan, Pengendalian dan Evaluasi Pelaksanaan Perencanaan Pembangunan Daerah, dipandang perlu suatu sistem informasi monitoring laporan dalam upaya meningkatkan kualitas operasional suatu program dan berkontribusi penting dalam memandu pembuat kebijakan di seluruh strata organisasi pemerintah.

Saat ini di dinas peternakan dan kesehatan hewan dalam hal monitoring masih tergolong kurang cepat dan tepat. Salah satu bidang yang memiliki masalah serupa adalah bidang produksi. Pada bidang ini terdapat tiga sub bidang yaitu sub bidang pembibitan ternak, budidaya ternak, dan pakan ternak. Disetiap sub bidang memiliki kendala yang hampir sama yaitu mengenai dokumen laporan yang digunakan untuk bahan rapat, dimana ada beberapa dokumen yang masih manual atau *hardcopy* meskipun ada juga yang sudah tersimpan dalam bentuk *digital* atau *softcopy*. Dokumen *hardcopy* ini dikumpulkan dalam media penyimpanan dimana setiap sub bidang memiliki media penyimpanan sendiri-sendiri. Salah satu sub bidang yaitu bidang pembibitan ternak dimana terdapat dokumen laporan yang bermacam-macam, salah satunya adalah laporan yang berasal dari petugas inseminasi buatan. Laporan dari petugas IB dikumpulkan secara manual atau harus mendatangi dinas.

Selain itu *sharing data* antar bidang sering terkendala karena adanya *human error*, dimana saat suatu dokumen dibutuhkan, dokumen itu terdapat pada komputer pegawai A maka hanya pegawai A yang mengetahui letak dokumen tadi. Untuk *sharing data* di tingkat dinas, ada beberapa yang masih manual dan ada juga yang sudah menggunakan teknologi email dan fax.

Dilihat dari segi efisiensi waktu,

produktivitas kerja,serta biaya tentu akan lebih efisien jika dibuat sebuah sistem yang dapat melakukan rekap data secara otomatis, dapat menampung laporan dari petugas IB di lapangan tanpa harus datang ke dinas, serta menyediakan layanan *sharing data* baik internal maupun eksternal. Selain itu dengan adanya sistem ini dapat mempercepat proses pencarian data, meminimalisir adanya kesalahan data dan duplikasi data. Sistem ini juga memberikan keuntungan waktu pada instansi dan juga petugas lapangan. Petugas lapangan bisa melaporkan kegiatan melalui *device* yang mereka miliki.

Perancangan sistem ini berbasiskan SOA, dengan berbasis SOA model arsitektur perangkat lunak memiliki komponen-komponen yang bersifat *loose coupling* dan *high cohesion*. Menurut Permana (2010) Aplikasi pada SOA dibangun berdasarkan *service*, yang merupakan implementasi dari proses bisnis yang ada dan dapat diakses oleh klien dari berbagai aplikasi. Untuk berkomunikasi antar satu sistem informasi dengan sistem informasi lainnya membutuhkan penyesuaian dimana penyesuaian ini mengharuskan kedua sistem membuka detail sistem yang memiliki resiko keamanan data. Sedangkan jika membangun sistem baru akan memerlukan waktu dan biaya yang banyak. SOA menjadi solusi dari masalah tersebut (Sari, 2010).

Dalam penelitian ini yang pertama dilakukan adalah mngambil data primer dan sekunder yang kemudian dijadikan acuan untuk dasar *requirement*. Selanjutnya melakukan perancangan dan pembuatan *web service*. Kemudian peneliti menguji rancangan dengan *consistency analysis* dan *serice litmus test* serta NuSOAP.

2. LANDASAN KEPUSTAKAAN

2.1. Konsep Dasar Service Oriented Architecture

Menurut Michael Rosen (2008) SOA merupakan sebuah gaya arsitektur yang diperuntukkan dalam pembangunan solusi *enterprise*, nilai *real* dari SOA muncul saat *reusable service* terkombinasi.

Menurut Guoling Liu (2010) SOA merupakan sebuah pendekatan arsitektur atau gaya untuk membangun perangkat lunak yang kompleks dari satu set universal yang saling berhubungan dan blok bangunan yang saling berintegrasi, yang disebut layanan.

SOA sebagai bentuk arsitektur teknologi yang menganut prinsip-prinsip berorientasi layanan. SOA adalah sebuah gaya arsitektural yang memodularisasi sistem informasi menjadi *services*. Konsep berorientasi layanan ini adalah melakukan pendekatan dengan membagi masalah besar menjadi sekumpulan *Service* kecil yang bertujuan untuk menyelesaikan permasalahan tertentu (Erl, 2009). Tabel 2.1 berikut menunjukkan beberapa karakteristik SOA menurut Erl (2009).

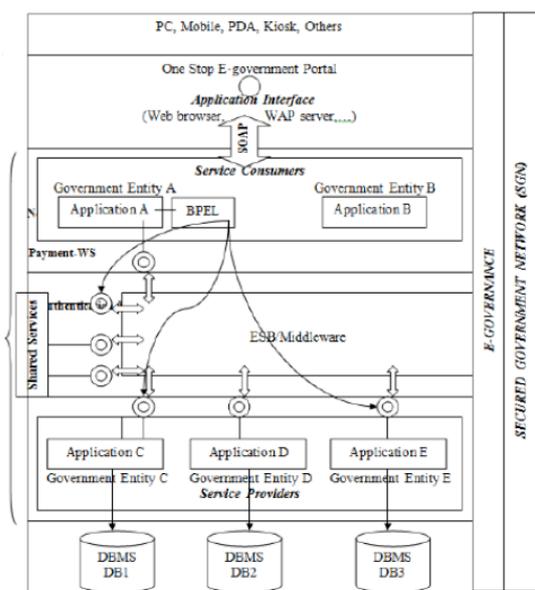
Tabel 2.1 Karakteristik SOA

NO	Karakteristik	Penjelasan
1	Loose coupling	<i>Loose coupling</i> , yaitu bahwa <i>services</i> tersebut mempertahankan sebuah hubungan yang meminimalisasi ketergantungan dan mereka hanya perlu menjaga kesadaran antar satu sama lain.
2	Interoperability	Interoperability, yaitu bahwa <i>service</i> bisa digunakan disegala device Pc, mobile, PDA dan sebagainya.
3	Service contract	<i>Service contract</i> , <i>services</i> melekat dan taat pada sebuah kesepakatan komunikasi, yang didefinisikan secara kolektif oleh satu atau lebih deskripsi <i>service</i> dan dokumen yang berhubungan.
4	Autonomy	<i>Autonomy</i> , bahwa <i>services</i> mempunyai kendali berdasarkan logika yang dienkapsulasi.

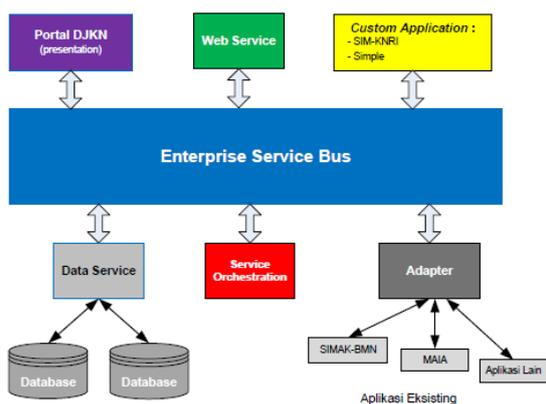
5	Abstraction	<i>Abstraction</i> , di luar apa yang dideskripsikan pada <i>service contract</i> , <i>services</i> menyembunyikan logika dari dunia luar.
6	Reusability	<i>Reusability</i> , logika terbagi menjadi <i>services</i> dengan tujuan untuk digunakan kembali.
7	Composability	<i>Composability</i> , kumpulan dari <i>services</i> dapat dikoordinasikan dan dihimpun untuk membentuk <i>services</i> yang berbeda.
8	Statelessness	<i>Statelessness</i> , <i>services</i> meminimalisasi sifat berpegang teguh pada informasi tertentu untuk sebuah aktivitas.
9	Discoverability	<i>Discoverability</i> , <i>services</i> dirancang dengan sifat yang deskriptif sehingga mereka dapat ditemukan dan ditentukan dengan menggunakan mekanisme penemuan yang ada.

2.2. Arsitektur SOA

Architectural context adalah tahap awal dari SOA, di mana organisasi sedang mengembangkan referensi arsitektur (Rosen, 2008). Menurut Zakaria I. Saleh, Rand A. Obeidat dan Yaser Khamayseh (2013) menerangkan terdapat empat layer dalam membangun arsitektur SOA yaitu client layer, presentation layer, Application layer, dan Data layer. Menurut Juli Sapta (2013) ESB merupakan sebuah penghubung (*message broker*) diantara *service* yang ada dalam sistem. Gambar 2.1 dan gambar 2.2 merupakan referensi arsitektur berbasis SOA.



Gambar 2.1 Referensi arsitektur SOA



Gambar 2.2 Referensi arsitektur SOA

2.3. Konsep Dasar Metodologi Ripple

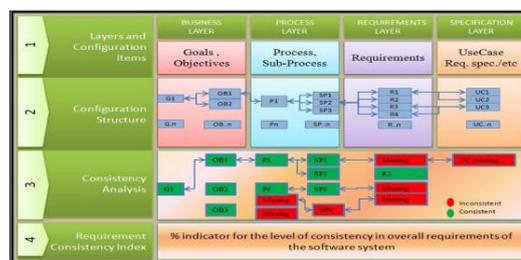
Metodologi Ripple adalah suatu metodologi yang diturunkan dari metodologi RUP (*Rational Unified Process*) dan lebih sederhana daripada metodologi RUP (O’Docherty, 2005). Pembangunan sebuah system dengan menggunakan metodologi Ripple terdiri dari 9 fase, yaitu: *genesis phase, requirements phase, analysis phase, design phase, class specification phase, implementation phase, testing phase, deployment phase, dan maintenance phase*. Hasil yang diperoleh, meliputi: definisi kebutuhan sistem, artefak perancangan sistem, *source code* sistem, hasil pengujian sistem, dan perawatan sistem.

2.4. Web Service Description Language

Web Service Description Language (WSDL), sebuah bahasa yang digunakan oleh service provider untuk menggambarkan service yang diciptakannya dalam konteks interoperabilitas (Sapta, 2013). Definisi *service* WSDL menyediakan dokumentasi untuk sistem terdistribusi dan bertindak sebagai petunjuk untuk mengotomatiskan detail yang dilibatkan di dalam komunikasi aplikasi (Christensen dalam Suyanto 2007). Sistem lain saling berhubungan dengan *Web service* di dalam cara yang ditentukan oleh deskripsinya yang menggunakan pesan SOAP, secara khas disampaikan menggunakan HTTP dengan *XML serialization*, bersama dengan standar lain yang terkait dengan *web* (Booth dalam Suyanto 2007). *Extensible Markup Language (XML)* adalah bagian dari SGML. Tujuannya untuk memungkinkan SGML umum untuk dilayani, diterima, dan diproses pada *Web* yang sekarang dengan HTML (Suyanto, 2007). *Bind* merupakan proses pengurutan data dalam WSDL (Gudgin, 2003).

2.5. Konsep Dasar Consistency Analysis: Requirements Configuration Structure

Requirements Configuration Structure adalah sebuah pendekatan untuk melakukan consistency analysis pada suatu perancangan perangkat lunak dengan memanfaatkan hubungan struktural antar-elemen perancangan, terutama berfokus pada konsistensi pendefinisian kebutuhan. Gambar 2.3 adalah gambaran lengkap kerangka kerja yang digunakan untuk membantu melakukan analisis konsistensi menggunakan Requirements Configuration Structure.



Gambar 2.3 Requirement Configuration Structure Framework

Sumber: Nistala dan Kumari (2013)

Requirement Consistency Index berfungsi untuk melakukan perhitungan terhadap persentase konsistensi dalam sebuah pendefinisian kebutuhan. Perhitungan dilakukan dengan persamaan 2.1.

$$RCI = \frac{A}{B+C} \tag{2.1}$$

Keterangan

- A: jumlah elemen kebutuhan yang konsisten
- B: jumlah total elemen kebutuhan
- C: jumlah elemen kebutuhan yang terdefinisi secara tidak benar

2.6. Konsep Dasar Service Litmus Test

Service litmus test merupakan sebuah test pada kandidat service dengan menerapkan kriteria kriteria khusus yang ada di dalam litmus test sendiri (Bieberstein, 2008). Menurut Kannan (2011) kriteria yang akan dinilai dengan litmus test adalah sebagai berikut:

1. Business Alignment : service yang dirancang harus sama dengan bisnis yang ada
2. Service are loosely coupled : service yang ada harus memiliki tingkat coupling yang rendah
3. Service are reusable : service yang dirancang dapat digunakan dalam lingkungan yang lain dengan persyaratan yang sama
4. Service are autonomous : service secara otomatis
5. Service are stateless : service dirancang dengan semua parameter yang diperlukan.

Berikut persamaan yang ada dalam *service litmus test* menurut Kannan (2011):

1. Bussines Alignment : dalam business alignment terdapat tiga persamaan yakni service consumer usage pattern, service producer usage pattern, dan consumer storage pattern. Yang kemudian semua di jelaskan pada persamaan 2.2 sampai 2.4.

$$SCUP_{si,si+1} = \left\{ \begin{array}{l} N(C_i) = N(C_{i+1}), N(B_i) = N(B_{i+1}) \\ C_i = C_{i+1} \\ B_i = B_{i+1} \end{array} \right\} \tag{2.2}$$

$$SPUP_{si,si+1} = \left\{ \begin{array}{l} N(C_i) = N(C_{i+1}), N(E_i) = N(E_{i+1}) \\ C_i = C_{i+1} \\ E_i = E_{i+1} \end{array} \right\} \tag{2.3}$$

$$CSP_{si,si+1} = \left\{ \begin{array}{l} C_i \neq C_{i+1} \\ E_i = E_{i+1} \\ B_i = B_{i+1} \end{array} \right\} \tag{2.4}$$

2. Berdasarkan gambar persamaan 2.5 dapat dijelaskan bahwa perancangan dapat dikatakan *reusable* jika terdapat dokumen lengkap mengenai perancangan yang

dapat dimengerti sehingga memungkinkan untuk digunakan kembali.

$$EMI = \left\{ \begin{array}{l} 1 \text{ Documents are available} \\ 0 \text{ Documents are not available} \end{array} \right\} \tag{2.5}$$

2.7. Hubungan SOA dan Object Oriented

Menurut Khanam (2010) bahwa pendekatan *object oriented* (OO) merupakan sebuah *wrapper* / bungkus untuk merepresentasikan SOA. Dapat dicontohkan bahwa sebuah SOA yang menggunakan teknologi *web service* sebagai perantara interaksi dengan *user*, harus memiliki *interface* sendiri atau *SOA interface*. Hal ini bisa di selesaikan dengan membuat *wrapper* yang langsung berinteraksi dengan *user*, pembuatannya dapat dilakukan dengan pendekatan *object oriented* (Khanam, 2010).

3. PENGUMPULAN DAN PENGOLAHAN DATA

Diagram alir tahap penelitian yang akan dilakukan dalam skripsi ini terdapat pada Gambar 3.1.



Gambar 3.1 Diagram alir tahap penelitian

3.1. Studi Literatur

Langkah ini termasuk dalam perencanaan dalam penelitian mengenai pendekatan *service oriented architecture* dalam analisis dan perancangan sistem. Diantaranya, mempelajari literature yang akan digunakan dalam penelitian. Dalam melakukan studi literatur, peneliti diharapkan mampu memahami konsep pendekatan *service oriented architecture* dalam analisis dan perancangan sebuah sistem.

3.2. Pengumpulan Data

3.2.1. Observasi

Observasi yang dilakukan pada penelitian ini adalah observasi non-perilaku, bertujuan untuk mengetahui prosedur pengelolaan proposal kegiatan Program Mahasiswa Wirausaha.

3.2.2. Wawancara

Wawancara yang dilakukan dalam penelitian ini adalah wawancara personal, bertujuan untuk mendapatkan kebutuhan dari pengguna dan melakukan verifikasi terhadap pendefinisian kebutuhan yang dilakukan oleh peneliti.

3.3. Analisis dan Perancangan Perangkat Lunak

Fase ini diawali dengan analisa hasil wawancara yang kemudian dilanjutkan dengan arsitektur sistem yang dihasilkan berdasarkan referensi. Kemudian melakukan perancangan dengan metode *Ripple* yang menghasilkan *use case diagram*, *class diagram*, *sequence diagram* dan *database schema* serta memetakannya kedalam arsitektur. Kemudian membuat rancangan WSDL.

3.4. Pengujian Perancangan Perangkat Lunak

Dalam fase ini akan mendefinisikan kebutuhan dan melakukan perancangan. Definisi kebutuhan dan metode perancangan dilakukan dengan menggunakan metode ripple, termasuk didalamnya *use case diagram*, *class diagram*, dan *sequence diagram*. Setelah itu memilih reference arsitektur SOA untuk pengembangan arsitektur sistem yang akan dibuat.

4. ANALISA DAN PERANCANGAN

4.1. Analisis Hasil Wawancara

Tujuan wawancara adalah untuk mengetahui requirement dari dinas peternakan dan kesehatan hewan kabupaten malang. Kriteria dapat diaplikasikan dengan rancangan adalah kegiatan non teknis seperti pelaporan. Tabel 4.1 akan menjelaskan mengenai hasil analisis.

Tabel 4.1 Analisa Hasil Wawancara

No	Kegiatan	Sistem	Business use case list
1	Inseminasi Buatan merupakan kegiatan melakukan kawin suntik pada ternak sapi milik peternak yang dilakukan oleh inseminator	Tidak, karena kegiatan inseminasi merupakan kegiatan teknis.	-
2	Pelaporan oleh inseminator Pelaporan dilakukan dengan cara menyimpan dalam bentuk soft copy maupun hard copy kemudian di serahkan kepada dinas setiap bulannya	Ya, karena kegiatan pelaporan merupakan kegiatan non teknis.	IB1
3	Perekapan laporan oleh pegawai seksi pembibitan Perekapan laporan dilakukan oleh pegawai seksi pembibitan sebelum dilakukan disposisi kepada atasan	Ya, karena kegiatan perekapan laporan merupakan kegiatan non teknis.	IB2
4	Penyuluhan tentang pakan ternak Penyuluhan dilakukan pada peternak dan para produsen pakan. Dimana penyuluhan ini dilakukan setiap 3 bulan sekali	Tidak, karena kegiatan penyuluhan merupakan kegiatan teknis.	-
5	Melaporkan laporan hasil luas area tanaman pakan serta persebaran produksi pakan Kegiatan yang dilakukan pegawai seksi pakan setiap 3 bulan sekali yaitu membuat laporan persebaran produsen pakan dan luas area tanaman pakan	Ya, karena kegiatan pelaporan merupakan kegiatan non teknis.	PN1
6	Perekapan laporan oleh pegawai seksi pakan Perekapan dilakukan oleh pegawai seksi pakan dalam 3 bulan sekali sebelum dilakukan disposisis keepada atasan	Ya, karena kegiatan perekapan laporan merupakan kegiatan non teknis.	PN2
7	Perekaman/record ternak budidaya serta melaporkan hasil rekam	Ya, karena <i>record</i> merupakan	BD1

	Perekaman dilakukan dengan cara pegawai seksi budidaya datang dan meninjau lokasi yang selanjutnya melakukan pendataan mengenai peternak yang akan menerima bantuan ternak	kegiatan non teknis.	
8	Perekapan laporan oleh pegawai seksi budidaya Perekapan laporan budidaya dilakukan setiap tahun nya dikarenakan ada laporan perkembangan ternak selain itu juga terdapat laporan kegiatan dimana laporannya berisi daerah mana saja yang menerima bantuan ternak, berapa banya ternak dan siapa saja yang memperoleh bantuan	Ya, karena kegiatan pelaporan merupakan kegiatan non teknis.	BD2
9	Kirim laporan (IB,pakan, dan budidaya) kepada jenjang lebih lanjut. Pengiriman laporan biasanya dilakukan dengan cara menyerahkan dokumen hard copy kepada pihak kabupaten yang selanjutnya akan dilanjutkan ke pada provinsi.	Ya, karena kegiatan distribusi merupakan kegiatan non teknis.	IB3,PN3 dan BD3

4.2. Arsitektur Sistem

Fase ini akan menampilkan hasil arsitektur yang telah dibuat berdasarkan referensi yang ada. Gambar 4.1 adalah arsitektur sistem informasi yang dibuat.

4.3. Fase Genesis

Fase ini bertujuan untuk mendapatkan kebutuhan dari pengguna, yaitu kegiatan pelaporan pada bidang produksi yang meliputi pembibitan ternak, pakan , dan budidaya.

4.4. Fase Business Requirement

Fase ini bertujuan untuk menggambarkan kondisi/proses yang saat ini berjalan.

4.4.1. Business Activity Diagram

Gambar 4.2 adalah *business activity diagram* untuk proses “Petugas Melaporkan kegiatan IB kepada Dinas Peternakan dan Kesehatan Hewan Kabupaten Malang”.

4.5. Fase System Requirement

Fase ini bertujuan untuk mendefinisikan kebutuhan dari segi sistem yang berdasar pada kebutuhan pengguna.

4.5.1. User Interface Sketch

Gambar 4.3 sampai gambar 4.4 adalah *user interface sketch* yang berkaitan dengan proses “petugas lapangan melakukan *insert* laporan”.

4.5.2. Use Case Design

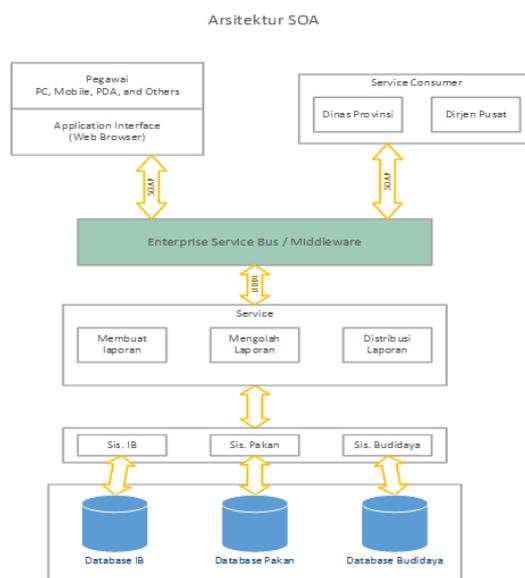
Gambar 4.5 adalah *use case design* yang merupakan gambaran kebutuhan fungsional dari sistem.

4.6. Fase Analysis

Fase ini bertujuan untuk membuat gambaran perancangan awal, disesuaikan dengan kebutuhan fungsional sistem yang telah didefinisikan.

4.6.1. Communication Diagram

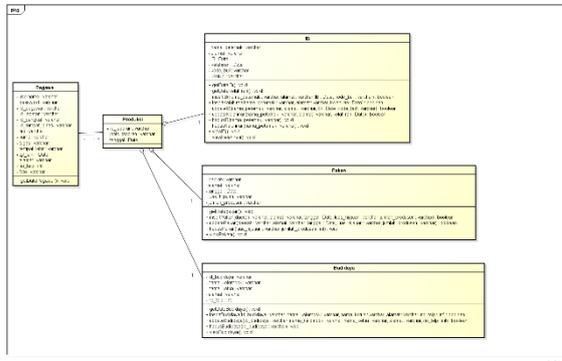
Gambar 4.6 adalah *communication diagram* yang menggambarkan hubungan antar kelas serta pesan yang disampaikan dalam sebuah proses.



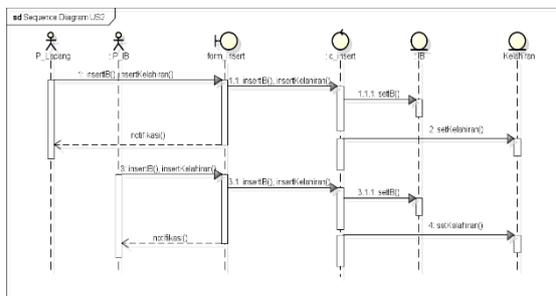
Gambar 4.1 Arsitektur Sistem Informasi Berbasis SOA

4.7.2. Sequence Diagram

Gambar 4.8 adalah *sequence diagram* yang menggambarkan bagaimana sebuah objek berinteraksi melalui pesan sekuensi.



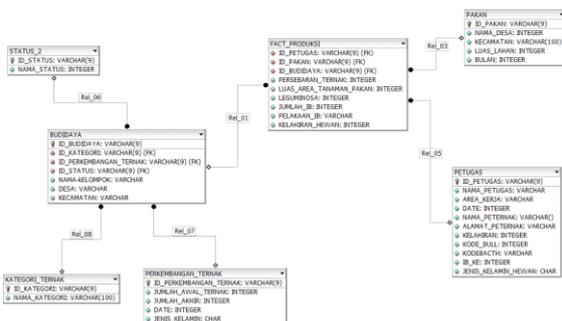
Gambar 4.7 Class diagram



Gambar 4.8 Sequence diagram

4.7.3. Database Schema

Gambar 4.9 adalah *database schema* yang menggambarkan kerangka untuk menyusun *database*.



Gambar 4.9 Database schema

5. PENGUJIAN

5.1. Consistency Analysis: Requirements Configuration Structure

Framework requirements configuration structure digunakan untuk melakukan pengujian konsistensi perancangan yang fokus pada

konsistensi definisi kebutuhan. *Framework* ini mempunyai 4 komponen kerangka kerja yaitu *layers and configuration items, configuration structure, consistency analysis, dan requirement consistency index*. Selanjutnya dilakukan perhitungan persentase konsistensi dari pendefinisian kebutuhan sebagai keluaran dari proses.

Gambar 5.1 adalah proses analisis terhadap 4 pendefinisian kebutuhan.

Setelah dilakukan analisis terhadap 4 pendefinisian kebutuhan, maka diperoleh hasil sebagai berikut:

- i) Jumlah total elemen kebutuhan, selanjutnya disebut dengan variabel B memiliki nilai 39 (terdiri dari: 9 *input business layer, 9 input process layer, 7 input requirements layer, dan 14 input specification layer*).
- ii) Jumlah elemen kebutuhan yang terdefinisi secara tidak benar, selanjutnya disebut dengan variabel C memiliki nilai 0.
- iii) Jumlah elemen kebutuhan yang konsisten, selanjutnya disebut dengan variabel A memiliki nilai 39.

Setelah diketahui nilai-nilai untuk variabel A, B, dan C, maka didapatkan persentase konsistensi dari pendefinisian kebutuhan sistem informasi produksi Dinas Peternakan dan Kesehatan Kabupaten Malang dengan perhitungan sebagai berikut:

$$\begin{aligned}
 RCI &= \frac{A}{B+C} \times 100\% \\
 &= \frac{39}{39+0} \times 100\% \\
 &= \frac{39}{39} \times 100\% \\
 &= 100\%
 \end{aligned}$$

Berdasarkan hasil perhitungan diatas, diperoleh nilai 100% untuk persentase konsistensi dari pendefinisian kebutuhan sistem informasi produksi Dinas Peternakan dan Kesehatan Hewan Kabupaten Malang. Hal ini mengandung pengertian bahwa masing-masing elemen definisi kebutuhan dari keempat *layer* sudah saling berhubungan, tidak ada elemen definisi kebutuhan yang tidak berdasar pada tujuan pengguna (*business layer*), serta pendefinisian kebutuhan pada *process layer, requirements layer, dan specification layer* sudah konsisten.

5.2. Service Litmus Test

Pengujian service litmus test ini dilakukan untuk mengetahui apakah perancangan yang sudah dibuat sudah memenuhi kriteria SOA. Dalam pengujian ini ada 5 tahap yaitu mulai menentukan business alignment kemudian melakukan pengujian untuk mengetahui perancangan loosely coupling, reusable, autonomous, dan stateless atau tidak. Dalam menentukan dan menguji rancangan menggunakan diagram UML.

Langkah awal yang dilakukan dalam *litmus test* ini adalah menguji *business alignment* dengan cara menguji *sequence diagram* yang sudah ada dengan persamaan 2.2. Kemudian diperoleh hasil dari *service litmus test* seperti pada tabel 5.1. Kemudian dilakukan *test* untuk mengetahui apakah perancangan yang sudah dibuat memenuhi syarat *reusable* atau belum. Hasil test dapat dilihat pada tabel 5.1.

Tabel 5.1 Hasil *Service Litmus Test*

Pengujian	Hasil	
Business Alignment	SCUP	Ya
	SPUP	Ya
	CSP	Tidak
Reusable	Ya	

Penjelasan lengkap hasil *service litmus test* pada tabel 5.1 adalah sebagai berikut:

1. Business Alignment

Terdapat tiga pemetaan berdasarkan persamaan rumus yaitu ada service consumer usage pattern, service producer usage pattern dan consumer storage pattern. Berikut hasil dari setiap persamaan.

- a. Untuk SCUP (service consumer usage pattern) diperoleh hasil controller dan boundary yang sama-sama satu di setiap sequence yang di uji. Itu memiliki arti bahwa jika boundary antara satu dan lainnya sama maka konsumennya sama dan jika controllernya sama maka memiliki tujuan bisnis yang konsisten
- b. Kemudian untuk SPUP (service producer usage pattern) diperoleh hasil disetiap proses pengolahan laporan class data antara satu proses dan lainnya sama di setiap bidangnya dan memiliki controller yang sama di setiap bidangnya. Dengan ini dapat diartikan bahwa setiap proses di setiap bidang menggunakan data bisnis yang sama serta memiliki logika bisnis yang sama.

- c. Kemudian untuk CSP (consumer storage pattern) diperoleh hasil yang kurang maksimal dikarenakan pada bidang IB class data dan boundary tidak sama oleh Karena itu dapat diartikan bahwa pada bidang ini tidak dapat diakses melalui controller yang berbeda untuk menyimpan dan memperoleh informasi objek. Sedangkan untuk dua bidang yang lain sudah memenuhi syarat dengan class data dan boundary yang sama, maka dapat diartikan proses pada kedua bidang dapat diakses melalui controller yang berbeda untuk menyimpan dan memperoleh data atau informasi objek

2. Reusable

Untuk pengujian ini perancangan yang dibuat sudah tergolong reusable Karena sudah memenuhi persyaratan yaitu memiliki dokumen yang lengkap serta jelas mengenai rancangan yang sudah dibuat. Dokumen ini terletak pada bab 4

6. KESIMPULAN

1. Analisa hasil wawancara menghasilkan kegiatan yang kemudian di kelompokkan ke dalam 2 kelompok, dapat di implementasikan ke dalam sistem dan tidak. Kriteria dapat di implementasikan adalah kegiatan non teknis. Didapatkan 7 kegiatan yang dapat di implementasikan dan 2 kegiatan yang tidak. Kemudian dari hasil tersebut didapatkan kebutuhan fungsional dan non fungsional yaitu 14 kebutuhan fungsional (sistem dapat melakukan proses *login*, *logout*, *insert* laporan IB, *update* laporan IB, hapus laporan IB, *insert* laporan Pakan, *update* laporan Pakan, hapus laporan Pakan, *insert* laporan Budidaya, *update* laporan Budidaya, hapus laporan Budidaya, *view* laporan IB, *view* laporan Pakan, dan *view* laporan Budidaya) dan 7 kebutuhan non fungsional (*availability*, *interoperability*, *ergonomy*, *memory*, *response time*, *security*).
2. Perancangan dilakukan dengan metode *Ripple*, dimana metode ini merupakan metode perancangan berbasis OO (*object oriented*). *Object oriented* sendiri merupakan teknologi *wrapper* untuk SOA. Kemudian dari perancangan dihasilkan *use case diagram* (14 *use case* dan 8 *actor*), *class diagram* (5 *class*, 33 *attribut* dan 30

- operation), 17 *sequence diagram* (*sequence diagram* US1, US2, US3, US4, US5, US5_peg, US6, US7, US8, US9, US9_peg, US10, US11, US12, US13, US13_peg dan US14) dan *physical data model*. Selain itu didapatkan hasil pemetaan kedalam arsitektur SOA sebagai berikut: pada *service layer* berisi *sequence diagram* (US2, US3, US4, US6, US7, US8, US10, US11 dan US12), *class diagram* (*class* IB, Pakan dan Budidaya) dan *physical data model* (tabel *Fact_Produksi*, Budidaya dan Pakan). Kemudian pada *data layer* hanya berisi *physical data model*. Untuk *application layer* berisi *sequence diagram* (US5, US5_peg, US9, US9_peg, US13 dan US13_peg), *class diagram* (*class* Pegawai dan Produksi) dan *physical data model* (tabel *Fact_Produksi* dan Petugas).
3. Melakukan pengujian *consistency analysis* dan didapatkan hasil jumlah variabel B 39 (9 *input business layer*, 9 *input process layer*, 7 *input requirements layer*, dan 14 *input specification layer*), jumlah variable C 0 dan jumlahan variable A 39, yang kemudian dilakukan penghitungan RCI dan mendapatkan hasil 100% yang berarti perancangan yang dibuat sudah konsisten.
 4. Fase *Business alignment* pada *service litmus test* mendapatkan hasil untuk tiap tiap persamaan sebagai berikut : SCUP(*service consumer usage pattern*) dengan hasil tercapai, SPUP(*service provider usage pattern*) dengan hasil tercapai dan CSP(*consumer storage pattern*) dengan hasil tidak tercapai. Pada CSP, tidak tercapai Karena salah satu komponen uji tidak sesuai dengan persamaan yang ada. Artinya perancangan kurang sesuai dengan proses bisnis karena salah satu persamaan yaitu CSP(*consumer storage pattern*) tidak tercapai secara maksimal.
 5. Fase *reusable* pada *service litmus test* menghasilkan perancangan sudah memenuhi kriteria *reuseable*. Hasil ini diperoleh karena terdapat dokumen lengkap berisikan diagram desain yang menjelaskan fungsi *service*. Dapat disimpulkan bahwa perancangan memenuhi salah satu kriteria SOA yaitu *reusable*.

DAFTAR PUSTAKA

- Bieberstein, Nobert., 2008. Executing SOA: A Practical Guide for the Service-Oriented Architect. Indiana: IBM Press.
- Dinas Peternakan dan Kesehatan Hewan Kabupaten Malang, 2011. <<http://peternakan.malangkab.go.id>> [diakses pada 12 April 2016].
- Erl, Thomas., 2005. SOA Design Pattern. 1st ed. Prentice Hall.
- Kannan, K. Bhamidipaty, A. dan Narendra, N.C., 2011. Design Time Validation of Service Orientation Principles using Design Diagrams. Bangalore: IBM Research India.
- Sapta, July. dan Hantana, Putra., 2013. Pendekatan Service Oriented Architecture (SOA) Pada Pelaksanaan E-Government Kementrian Hukum dan HAM RI. Volume 2. Denpasar: JANAPATI.
- Suyanto, Asep., 2007. Web Service. Jurnal Komputer Nasional.