

**INTEGRASI SISTEM INFORMASI AKUNTANSI PADA
ENTEPRISE RESOURCE PLANNING PONDOK
PESANTREN TIPE D MENGGUNAKAN
*SERVICE ORIENTED
ARCHITECTURE***

SKRIPSI

Oleh :
AZIZ FAJAR
NIM. 12650027



**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2016**

**INTEGRASI SISTEM INFORMASI AKUNTANSI PADA
ENTEPRISE RESOURCE PLANNING PONDOK
PESANTREN TIPE D MENGGUNAKAN
SERVICE ORIENTED
ARCHITECTURE**

SKRIPSI

Oleh:
AZIZ FAJAR
NIM. 12650027



**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2016**

**INTEGRASI SISTEM INFORMASI AKUNTANSI PADA *ENTERPRISE
RESOURCE PLANNING* PONDOK PESANTREN TIPE D
MENGUNAKAN *SERVICE ORIENTED
ARCHITECTURE***

SKRIPSI

Diajukan kepada:

Universitas Islam Negeri Maulana Malik Ibrahim Malang

Untuk memenuhi Salah Satu Persyaratan dalam

Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh :

AZIZ FAJAR

NIM. 12650027

JURUSAN TEKNIK INFORMATIKA

FAKULTAS SAINS DAN TEKNOLOGI

UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM

MALANG

2016

LEMBAR PERSETUJUAN

**INTEGRASI SISTEM INFORMASI AKUNTANSI PADA *ENTERPRISE*
RESOURCE PLANNING PONDOK PESANTREN TIPE D
MENGUNAKAN *SERVICE ORIENTED*
*ARCHITECTURE***

SKRIPSI

**Oleh:
Aziz Fajar
NIM. 12650027**

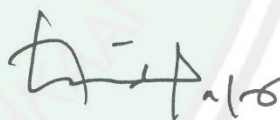
Telah Diperiksa dan Disetujui untuk Diuji:
Tanggal, 03 Januari 2017

Dosen Pembimbing I



M. Ainul Yaqin, M.Kom
NIP: 19761013 200604 1 004

Dosen Pembimbing II



Linda Salma Angreani, M.T
NIP: 19770803 200912 2 005

Mengetahui,
Ketua Jurusan Teknik Informatika



Dr. Cahyo Crysdian
NIP: 19740424 200901 1 008

LEMBAR PENGESAHAN





INTEGRASI SISTEM INFORMASI AKUNTANSI PADA *ENTERPRISE RESOURCE PLANNING* PONDOK PESANTREN TIPE D MENGGUNAKAN *SERVICE ORIENTED ARCHITECTURE*

SKRIPSI

Oleh:
Aziz Fajar
NIM. 12650027


Telah Dipertahankan di Depan Dewan Penguji Skripsi dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)

Tanggal, 3 Januari 2017

| Susunan Dewan Penguji | | Tanda Tangan |
|-----------------------|---|--|
| Penguji Utama | : <u>Syahiduz Zaman, M.Kom</u> NIP: 19700502 200501 1 005 |  (.....) |
| Ketua Penguji | : <u>Fatchurrochman, M.Kom</u> NIP: 19700731 200501 1 001 |  (.....) |
| Sekretaris Penguji | : <u>M. Ainul Yaqin, M.Kom</u> NIP: 19761013 200604 1 004 |  (.....) |
| Anggota Penguji | : <u>Linda Salma Angreani, M.T.</u> NIP: 19770803 200912 2 005 |  (.....) |

Mengesahkan,
 Ketua Jurusan Teknik Informatika




Dr. Cahyo Crysdiyan
 NIP: 19740424 200901 1 008

HALAMAN PERNYATAAN ORISINALITAS

Saya yang bertanda tangan dibawah ini:

Nama : Aziz Fajar
NIM : 12650027
Jurusan : Teknik Informatika
Fakultas : Sains dan Teknologi

Menyatakan dengan sebenarnya bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka. Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi sesuai dengan peraturan yang berlaku.

Malang, 3 Januari 2017

Yang membuat pernyataan



Aziz Fajar

NIM. 12650027

MOTTO

Jangan menjelaskan tentang dirimu kepada siapapun. Karena yang menyukaimu tidak butuh itu, dan yang membencimu tidak percaya itu. – Ali Bin Abi Thalib

That which does not kill us makes us stronger. – Friedrich Nietzsche



HALAMAN PERSEMBAHAN

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Kupersembahkan sebuah

karya sederhana ini untuk orang-orang yang

paling kusayangi, kubanggakan, dan selalu

memberikan semangat untukku

Seluruh keluarga besarku

khususnya Ayah Farid dan Ibu Wiwik yang tercinta

yang selalu ikhlas mendoakan putra-putrinya

Saudara-saudariku, Mbak lis dan Dek Arief

Teman-Teman di Grup Sosialita dan Kontrakan Plosok

Yang membuat setiap hari tidak pernah sepi

Dan terutama Pipit, Om Fajarivan dan Eko

yang telah membantu terselesaikannya skripsi ini

Semoga Allah SWT melindungi dan menjaga

mereka dalam naungannya..

Aamiin

KATA PENGANTAR

Assalamu'alaikum Wr. Wb.

Segala puji bagi Allah SWT tuhan semesta alam, karena atas segala rahmat dan karunia-Nya sehingga penulis mampu menyelesaikan skripsi dengan judul “Integrasi Sistem Informasi Akuntansi pada *Enterprise Resource Planning* Pondok Pesantren Tipe D Menggunakan *Service Oriented Architecture*” dengan baik dan lancar. Shalawat serta salam selalu tercurah kepada tauladan terbaik Nabi Muhammad SAW yang telah membimbing umatnya dari zaman kebodohan menuju Islam yang *rahmatan lil alamiin*.

Dalam menyelesaikan skripsi ini, banyak pihak yang telah memberikan bantuan baik secara moril, nasihat dan semangat, maupun materiil. Atas segala bantuan yang telah diberikan, penulis ingin menyampaikan doa dan ucapan terimakasih yang sedalam-dalamnya kepada.

1. Prof. Dr. H. Mudjia Rahardjo, M.Si., selaku Rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Muhammad Ainul Yaqin, M.Kom selaku Pembimbing I yang telah meluangkan waktu untuk membimbing, memotivasi, mengarahkan, dan memberi masukan kepada penulis dalam pengerjaan skripsi ini hingga akhir.
3. Linda Salma Angreani, M.T selaku Pembimbing II yang telah meluangkan waktu, membimbing, dan mengarahkan penulis dalam pengerjaan skripsi ini hingga akhir.
4. Syahiduz Zaman, M.Kom selaku Penguji I saya yang telah membimbing dan memberikan masukan pada skripsi ini.
5. Fatchurrochman, M.Kom selaku Penguji II yang juga telah membimbing dan memberikan masukan pada skripsi ini.

6. Dr. Cahyo Crysdiان selaku Ketua Jurusan Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.
7. Almarhumah Ibu Ratna Puspita Ellyani yang selama pengerjaan skripsi ini menjabat sebagai penanggungjawab Gedung Pusat Informasi yang telah memberikan izinnya pada tim kami untuk menggunakan ruang pusat informasi sebagai tempat diskusi dan penelitian.
8. Segenap Dosen Teknik Informatika yang telah memberikan bimbingan keilmuan kepada penulis selama masa studi.
9. Teman-teman seperjuangan Teknik Informatika angkatan 2012.
10. Tim Skripsi Sukses yang telah berjuang bersama dan banyak memberikan bantuan kepada peneliti.

Berbagai kekurangan dan kesalahan mungkin pembaca temukan dalam penulisan skripsi ini, untuk itu penulis menerima segala kritik dan saran yang membangun dari pembaca sekalian. Semoga apa yang menjadi kekurangan bisa disempurnakan oleh peneliti selanjutnya dan semoga karya ini senantiasa dapat memberi manfaat. Amin. *Wassalamualaikum Wr.Wb*

Malang, 3 Januari 2017

Penulis

DAFTAR ISI

| | |
|--|------------------------------|
| HALAMAN JUDUL | i |
| HALAMAN PENGAJUAN | ii |
| HALAMAN PERSETUJUAN..... | Error! Bookmark not defined. |
| HALAMAN PENGESAHAN | Error! Bookmark not defined. |
| HALAMAN PERNYATAAN ORISINALITAS.... | Error! Bookmark not defined. |
| MOTTO | vi |
| HALAMAN PERSEMBAHAN..... | vii |
| KATA PENGANTAR..... | viii |
| DAFTAR ISI..... | x |
| DAFTAR GAMBAR..... | xii |
| DAFTAR TABEL..... | xiv |
| ABSTRAK | xv |
| ABSTRACT | xvi |
| المخلص..... | xvii |
| BAB I PENDAHULUAN | 1 |
| 1.1 Latar Belakang..... | 1 |
| 1.2 Rumusan Masalah | 4 |
| 1.3 Hipotesis | 4 |
| 1.4 Tujuan Penelitian..... | 4 |
| 1.5 Manfaat Penelitian..... | 4 |
| 1.6 Batasan Masalah..... | 5 |
| 1.7 Sistematika Penulisan..... | 5 |
| BAB II KAJIAN PUSTAKA | 7 |
| 2.1 <i>Enterprise Resource Planning</i> (ERP)..... | 7 |
| 2.2 Akuntansi..... | 16 |
| 2.2.1 Akuntansi Pendidikan | 17 |
| 2.2.2 Sistem Informasi Akuntansi | 20 |
| 2.3 <i>Business Process Modelling</i> (BPM)..... | 23 |
| 2.4 SOA | 27 |
| 2.5 Web Service..... | 29 |
| BAB III ANALISIS DAN DESAIN SISTEM | 33 |
| 3.1 Analisis Sistem | 33 |
| 3.1.1 Gambaran Umum..... | 33 |

| | | |
|---|--|-----------|
| 3.1.2 | Sumber Data..... | 34 |
| 3.1.3 | Analisis Proses Bisnis | 34 |
| 3.2 | Desain Sistem | 45 |
| 3.2.1 | Pemodelan Sistem Informasi Akuntansi | 45 |
| 3.2.2 | Penerapan SOA | 52 |
| 3.2.3 | Pemodelan <i>Service</i> | 54 |
| 3.3 | Prosedur Penelitian..... | 57 |
| BAB IV HASIL DAN PEMBAHASAN..... | | 59 |
| 4.1 | Sistem Informasi Akuntansi | 59 |
| 4.2 | Konfigurasi <i>Service</i> | 60 |
| 4.2.1 | Konfigurasi <i>Service</i> dengan nuSoap | 60 |
| 4.2.2 | Konfigurasi <i>Service</i> pada ESB | 63 |
| 4.2.3 | Pengujian ESB | 70 |
| 4.3 | Komunikasi Antar <i>Service</i> | 71 |
| BAB V PENUTUP..... | | 86 |
| 5.1 | Kesimpulan..... | 86 |
| 5.2 | Saran..... | 86 |
| DAFTAR PUSTAKA..... | | 88 |
| LAMPIRAN..... | | 90 |

DAFTAR GAMBAR

| | |
|--|----|
| Gambar 2.1 Arsitektur ERP | 8 |
| Gambar 2.2 Sistem Global ERP | 9 |
| Gambar 2.3 Notasi <i>Event</i> | 25 |
| Gambar 2.4 Notasi <i>Activity</i> | 26 |
| Gambar 2.5 Notasi <i>Gateway</i> | 26 |
| Gambar 2.6 Notasi Penghubung | 27 |
| Gambar 2.7 Arsitektur <i>Web Service</i> | 30 |
| Gambar 2.8 Blok Bangunan <i>Web Service</i> | 32 |
| Gambar 3. 1 <i>Context Diagram</i> | 45 |
| Gambar 3.2 DFD Level 1 | 46 |
| Gambar 3.3 <i>Entity Relationship Diagram</i> | 48 |
| Gambar 3.4 Pemodelan SI Akuntansi dengan Sistem Lainnya..... | 49 |
| Gambar 3.5 Desain <i>Ouput</i> Daftar Akun..... | 50 |
| Gambar 3.6 Desain <i>Output</i> Jurnal..... | 50 |
| Gambar 3.7 Desain <i>Output</i> Buku Besar..... | 51 |
| Gambar 3.8 Desain <i>Output</i> Neraca | 51 |
| Gambar 3.9 Desain <i>Input</i> Akun Baru..... | 52 |
| Gambar 3.10 Arsitektur SOA..... | 54 |
| Gambar 3.11 Model Integrasi Sistem Informasi Akuntansi ke <i>Service</i> Keuangan..... | 55 |
| Gambar 3.12 Model Integrasi Sistem Informasi Akuntansi ke <i>Service</i> BPM..... | 56 |
| Gambar 3.13 Model Integrasi Sistem Informasi Keuangan ke <i>Service</i> Akuntansi..... | 56 |
| Gambar 3.14 Pengambilan Data SI Akuntansi Melalui ESB | 57 |
| Gambar 3. 15 Mindmap Penelitian | 57 |
| Gambar 4.1 Halaman Daftar Akun | 59 |
| Gambar 4.2 Halaman Jurnal..... | 60 |
| Gambar 4.3 Registrasi <i>Service</i> | 62 |
| Gambar 4.4 WSDL <i>Service</i> Tambah Akun..... | 63 |
| Gambar 4.5 Konfigurasi ESB Langkah ke-1 | 64 |
| Gambar 4.6 Konfigurasi ESB Langkah ke-2 | 64 |
| Gambar 4.7 Konfigurasi ESB Langkah ke-3 | 65 |
| Gambar 4.8 Konfigurasi ESB Langkah ke-4 | 65 |
| Gambar 4.9 Konfigurasi ESB Langkah ke-5 | 66 |
| Gambar 4.10 Konfigurasi ESB Langkah ke-6 | 66 |
| Gambar 4.11 Konfigurasi ESB Langkah ke-7 | 67 |
| Gambar 4.12 Konfigurasi ESB Langkah ke-8 | 67 |
| Gambar 4.13 Konfigurasi ESB Langkah ke-9 | 68 |
| Gambar 4.14 Konfigurasi ESB Langkah ke-10 | 68 |
| Gambar 4.15 Konfigurasi ESB Langkah ke-11 | 69 |
| Gambar 4.16 Konfigurasi ESB Langkah ke-12 | 70 |
| Gambar 4.17 Pengujian ESB..... | 71 |

| | |
|--|----|
| Gambar 4.18 Tampilan <i>Request</i> Departemen | 72 |
| Gambar 4.19 <i>Request</i> Departemen..... | 72 |
| Gambar 4.20 WSDL Departemen..... | 73 |
| Gambar 4.21 Tampilan <i>Form</i> pencarian | 73 |
| Gambar 4.22 Tampilan Data Transaksi Keuangan | 74 |
| Gambar 4.23 <i>Request</i> Transaksi..... | 74 |
| Gambar 4.24 WSDL Transaksi | 75 |
| Gambar 4.25 <i>Source Code</i> Jurnal..... | 76 |
| Gambar 4.26 <i>Request</i> Kode Akun..... | 76 |
| Gambar 4.27 WSDL Kode Akun | 77 |
| Gambar 4.28 <i>Request</i> Nama Akun..... | 77 |
| Gambar 4.29 WSDL Nama Akun | 77 |
| Gambar 4.30 Gambar Jurnal Akuntansi..... | 78 |
| Gambar 4.31 <i>Request</i> Tambah Jurnal | 78 |
| Gambar 4.32 WSDL Tambah jurnal | 78 |
| Gambar 4.33 <i>Form</i> Buku Besar | 79 |
| Gambar 4.34 <i>Souce Code</i> Buku Besar | 79 |
| Gambar 4.35 <i>Request</i> Jurnal | 80 |
| Gambar 4.36 WSDL Jurnal..... | 80 |
| Gambar 4.37 Tampilan Buku Besar..... | 81 |
| Gambar 4.38 <i>Request</i> Tambah Neraca..... | 81 |
| Gambar 4.39 WSDL Tambah Neraca | 82 |
| Gambar 4.40 <i>Form</i> Neraca..... | 82 |
| Gambar 4.41 <i>Source Code</i> Neraca | 82 |
| Gambar 4.42 <i>Request</i> Neraca..... | 83 |
| Gambar 4.43 WSDL Neraca | 83 |
| Gambar 4.44 <i>Tampilan</i> Neraca | 84 |

DAFTAR TABEL

| | |
|---|----|
| Tabel 3.1 Analisis Proses Bisnis | 36 |
| Tabel 3.2 Penentuan Akun | 37 |
| Tabel 3.3 Pembuatan Jurnal | 37 |
| Tabel 3.4 Penentuan Buku Besar | 37 |
| Tabel 3.5 Pembuatan Neraca..... | 37 |
| Tabel 3.6 Analisis Kebutuhan Fungsional | 38 |
| Tabel 3.7 Identifikasi dan Analisis Kebutuhan Non-Fungsional..... | 39 |
| Tabel 3.8 Identifikasi <i>Output</i> | 43 |
| Tabel 3.9 Identifikasi Input..... | 44 |
| Tabel 3.10 Tabel Akun..... | 46 |
| Tabel 3.11 Tabel Jenis Transaksi | 47 |
| Tabel 3.12 Tabel Jurnal..... | 47 |
| Tabel 3.13 Tabel Neraca | 47 |
| Tabel 3.14 Data Service Akuntansi Keuangan | 55 |
| Tabel 4.1 Detail Registrasi Service..... | 62 |

ABSTRAK

Fajar, Aziz. 2016. Sistem Informasi Akuntansi Berbasis *Service Oriented Architecture* Pada Pondok Pesantren Untuk Integrasi dengan Sistem *Enterprise Resource Planning*. Skripsi. Jurusan Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang.
Pembimbing: (I) M. Ainul Yaqin, M.Kom dan (II) Linda Salma Angreani, M.T

Pondok pesantren sebagai sebuah organisasi pasti memiliki visi, misi, dan tujuan yang ingin dicapai. Untuk mencapai visi dan misinya tersebut, maka diperlukan suatu sistem yang baik sehingga pelaksanaan kegiatan yang ada dalam pondok tersebut dapat berjalan dengan baik. Salah satu aspek yang harus dikelola dalam sebuah organisasi tentu adalah aspek keuangan yang memiliki dampak besar pada keberlangsungan organisasi tersebut. Akuntansi sebagai salah satu bagian dalam pengelolaan keuangan suatu organisasi memiliki peran yang sangat krusial.

Penelitian ini membuat sistem informasi akuntansi pada pondok pesantren yang terintegrasi dengan sistem ERP. Integrasi tersebut menggunakan *web service* yang menggunakan arsitektur *Service Oriented Architecture* (SOA). Arsitektur SOA sangat tepat digunakan untuk sistem informasi yang tergabung dalam ERP. Hal ini ditunjukkan dengan lancarnya komunikasi antara service suatu sistem informasi dengan service lainnya. Pada SOA terdapat ESB yang berperan sebagai broker.

Kata Kunci: Akuntansi, ERP, SOA.

ABSTRACT

Fajar, Aziz. 2016. Integration of Accounting Information System to Enterprise Resource Planning of Type D Boarding School Using Service Oriented Architecture. Thesis. Informatic Engineering Department, Faculty of Science and Technology. Islamic State University of Maulana Malik Ibrahim Malang.

Supervisor: (I) M. Ainul Yaqin, M.Kom dan (II) Linda Salma Angreani, M.T

Islamic Boarding School as an organization must have vision, mission, and purpose which they want to achieve. To achieve their vision and mission, they need a decent system so the activity inside the boarding school can be successfully done. One of the aspects that must be managed inside an organization is financial aspect which have big impact to the organization continuity. Accounting as a part of financial management of an organization have a crucial role.

This research's purpose is to create an accounting information system which integrated with ERP system for boarding school. The integration done by applying web service and Service Oriented Architecture (SOA). SOA is a great choice for the information system integrated with ERP indicated by successful communication of information system's services with other services. In this research ESB takes role as the broker of those services.

Keywords: Accounting, ERP, SOA.

المخلص

فجر، عزيز. 2016. نظم المعلومات المحاسبية بخدمة المنحى العمارة بناء على مدرسة داخلية للتكامل مع نظام تخطيط موارد المؤسسات

الأطروحة قسم المعلوماتية، كلية العلوم والتكنولوجيا، الجامعة الحكومية الإسلامية مولانا مالك إبراهيم مالانج.

المشرف: الأول محمد عين اليقين، الماجستير في الحاسوب والثاني ليندا سلمى أنغريني، الماجستير في تكنولوجيا

تجب للمدرسة الداخلية كمنظمة أن تكون رؤية ورسالة وأهداف أن يتحقق. لتحقيق رؤية ورسالة، تحتاج إلى نظام جيد لتنفيذ الأنشطة في المدرسة أن تعمل بشكل جيد. أحد الجوانب التي يجب أن تدار داخل المدرسة هو الجانب المالي لديها تأثير كبير على استدامة المؤسسة. المحاسبة هي جزء من الإدارة المالية للمنظمة دورا حاسما للغاية.

وتهدف هذه الدراسة إلى جعل نظم المعلومات المحاسبية في مدرسة داخلية التي تتكامل مع نظام تخطيط موارد المؤسسات. العمارة التكامل باستخدام خدمات الويب العمارة الخدمية (SOA). الخدمية العمارة هي مناسبة جدا لاستخدامها في نظم المعلومات المدرجة في نظام تخطيط موارد المؤسسات. ويتجلى ذلك من خلال نجاح التواصل بين خدمة نظام المعلومات مع الخدمات الأخرى. في هذه الدراسة، وتستخدم ESB التي تقوم بدور وسيط الخدمات.

كلمات البحث: المحاسبة، تخطيط موارد المؤسسات، الخدمية.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pondok pesantren adalah suatu lembaga pendidikan yang bertujuan untuk mendalami ilmu agama Islam dan mengamalkannya sebagai pedoman hidup keseharian (Suyono & Wahyuni, 2014). Namun, dalam perkembangannya pondok pesantren saat ini tidak hanya mengajarkan wawasan keagamaan saja, ilmu-ilmu umum juga mulai diajarkan pada pondok-pondok pesantren saat ini.

Pondok pesantren dalam hal ini dapat diasumsikan sebagai sebuah organisasi. Sebagai sebuah organisasi, pasti memiliki visi, misi, dan tujuan yang ingin dicapai. Sedangkan, untuk mencapai visi dan misinya tersebut, maka diperlukan suatu sistem yang baik sehingga pelaksanaan kegiatan yang ada dalam pondok tersebut dapat berjalan dengan baik. Terutama untuk sebuah pondok pesantren yang besar, memerlukan pengelolaan yang baik dalam seluruh aspek yang berjalan dalam pondok tersebut.

Salah satu aspek yang harus dikelola dalam sebuah organisasi tentu adalah aspek keuangan yang memiliki dampak besar pada keberlangsungan organisasi tersebut. Akuntansi sebagai salah satu bagian dalam pengelolaan keuangan suatu organisasi memiliki peran yang sangat krusial. Akuntansi digunakan untuk menyajikan informasi/data keuangan bermanfaat yang bisa dijadikan sebagai *input* atau bahan pertimbangan dalam mengambil keputusan bisnis. Agar bisa bermanfaat sebagai *input* dalam pengambilan keputusan bisnis, maka informasi keuangan harus bersifat akurat, relevan, dan mudah dipahami oleh pihak-pihak yang memerlukan (pengguna laporan keuangan). Untuk memahami laporan

keuangan dengan baik, maka perlu laporan yang sistematis, logis, dan mudah dianalisa.

Banyak sekali aplikasi akuntansi yang beredar saat ini. Salah satunya adalah Wave Accounting. Kelebihan Wave Accounting adalah tidak berbayar, meskipun tidak semua fitur diberikan secara gratis. Kelebihan lainnya adalah bisa digunakan multi bisnis, *unlimited invoice*, mudah digunakan, dan hemat waktu. Aplikasi akuntansi lainnya yang banyak juga digunakan adalah Zipbooks. Zipbooks adalah *software* akuntansi yang gratis serta memungkinkan untuk membuat laporan akuntansi dan pengkategorian pengeluaran. Kelebihan lain dari Zipbooks adalah sinkronisasi dengan rekening bank pengguna.

Namun, aplikasi-aplikasi akuntansi ini bersifat *stand alone* atau belum terintegrasi dengan sistem informasi yang lain. Untuk tercapainya pengelolaan akuntansi yang baik bagi pondok pesantren, maka diperlukan adanya sistem informasi akuntansi yang dapat terintegrasi dengan sistem informasi yang lain. Dengan adanya *Enterprise Resource Planning* (ERP), diharapkan proses akuntansi untuk pondok pesantren dapat berjalan dengan efektif dan efisien. Sehingga, kesalahan-kesalahan yang umum dilakukan dapat berkurang jika dibandingkan dengan sistem yang konvensional. Untuk membangun sistem akuntansi terintegrasi diperlukan beberapa data dari sistem lain. Untuk mengintegrasikan seluruh sistem yang dibutuhkan diperlukan sebuah arsitektur yang dapat memberikan layanan komunikasi seluruh pada sistem pondok pesantren. Salah satu arsitektur yang telah banyak digunakan adalah *Service Oriented Architecture* (SOA). Salah satu kekurangan dari SOA adalah biaya

investasi yang tinggi akan tetapi dengan arsitektur tersebut maka *service-service* yang telah dibuat akan bersifat *reusable*.

Agama Islam mengajarkan pentingnya menulis semua utang dan piutang seperti terdapat pada Surat Al Baqarah Ayat 282 yang berbunyi:

يَا أَيُّهَا الَّذِينَ آمَنُوا إِذَا تَدَايَنْتُمْ بِدَيْنٍ إِلَىٰ أَجَلٍ مُّسَمًّى فَاكْتُبُوهُ ۚ وَلْيَكْتُب بَيْنَكُمْ كَاتِبٌ بِالْعَدْلِ ۚ وَلَا يَأْبَ كَاتِبٌ أَنْ يَكْتُبَ كَمَا عَلَّمَهُ اللَّهُ ۚ فَلْيَكْتُبْ وَلْيُمْلِلِ الَّذِي عَلَيْهِ الْحَقُّ وَلْيَتَّقِ اللَّهَ رَبَّهُ وَلَا يَخْسِ مِنْهُ شَيْئًا ۚ فَإِنْ كَانَ الَّذِي عَلَيْهِ الْحَقُّ سَفِيهًا أَوْ ضَعِيفًا أَوْ لَا يَسْتَطِيعُ أَنْ يُمِلَّ هُوَ فَلْيُمْلِلْ وَلِيُّهُ بِالْعَدْلِ ۚ وَاسْتَشْهِدُوا شَهِيدَيْنِ مِنْ رِجَالِكُمْ ۚ فَإِنْ لَمْ يَكُونَا رَجُلَيْنِ فَرَجُلٌ وَامْرَأَتَانِ مِمَّنْ تَرْضَوْنَ مِنَ الشُّهَدَاءِ أَنْ تَضِلَّ إِحْدَاهُمَا فَتُذَكَّرَ إِحْدَاهُمَا الْأُخْرَىٰ ۚ وَلَا يَأْبَ الشُّهَدَاءُ إِذَا مَا دُعُوا ۚ وَلَا تَسْأَمُوا أَنْ تَكْتُبُوهُ صَغِيرًا أَوْ كَبِيرًا إِلَىٰ أَجَلِهِ ۚ ذَلِكُمْ أَقْسَطُ عِنْدَ اللَّهِ وَأَقْوَمٌ لِلشَّهَادَةِ وَأَدْنَىٰ أَلَّا تَرْتَابُوا ۚ إِلَّا أَنْ تَكُونَ تِجَارَةً حَاضِرَةً تُدِيرُونَهَا بَيْنَكُمْ فَلَيْسَ عَلَيْكُمْ جُنَاحٌ أَلَّا تَكْتُبُوهَا ۚ وَأَشْهِدُوا إِذَا تَبَايَعْتُمْ ۚ وَلَا يُضَارَّ كَاتِبٌ وَلَا شَهِيدٌ ۚ وَإِنْ تَفَعَّلُوا فَإِنَّهُ فَسُوقٌ بِكُمْ ۚ وَانقُوا اللَّهَ ۚ وَيَعْلَمُكُمْ اللَّهُ ۚ وَاللَّهُ بِكُلِّ شَيْءٍ عَلِيمٌ

Artinya:

Hai orang-orang yang beriman, apabila kamu bermu'amalah tidak secara tunai untuk waktu yang ditentukan, hendaklah kamu menuliskannya. Dan hendaklah seorang penulis di antara kamu menuliskannya dengan benar. Dan janganlah penulis enggan menuliskannya sebagaimana Allah mengajarkannya, meka hendaklah ia menulis, dan hendaklah orang yang berhutang itu mengimlakkan (apa yang akan ditulis itu), dan hendaklah ia bertakwa kepada Allah Tuhannya, dan janganlah ia mengurangi sedikitpun daripada hutangnya. Jika yang berhutang itu orang yang lemah akalnya atau lemah (keadaannya) atau dia sendiri tidak mampu mengimlakkan, maka hendaklah walinya mengimlakkan dengan jujur. Dan persaksikanlah dengan dua orang saksi dari orang-orang lelaki (di antaramu). Jika tak ada dua oang lelaki, maka (boleh) seorang lelaki dan dua orang perempuan dari saksi-saksi yang kamu ridhai, supaya jika seorang lupa maka yang seorang mengingatkannya. Janganlah saksi-saksi itu enggan (memberi keterangan) apabila mereka dipanggil; dan janganlah kamu jemu menulis hutang itu, baik kecil maupun besar sampai batas waktu membayarnya. Yang demikian itu, lebih adil di sisi Allah dan lebih menguatkan persaksian dan lebih dekat kepada tidak (menimbulkan) keraguanmu. (Tulislah mu'amalahmu itu), kecuali jika mu'amalah itu perdagangan tunai yang kamu jalankan di antara kamu, maka tidak ada dosa bagi kamu, (jika) kamu tidak menulisnya. Dan persaksikanlah apabila kamu berjual beli; dan janganlah penulis dan saksi saling sulit menyulitkan. Jika kamu lakukan (yang demikian), maka sesungguhnya hal itu adalah suatu kefasikan pada dirimu. Dan bertakwalah kepada Allah; Allah mengajarmu; dan Allah Maha Mengetahui segala sesuatu.

Kandungan makna yang tersirat dalam surat Al-Baqarah ayat 282 adalah mengenai pentingnya pencatatan dalam sebuah transaksi baik hutang maupun

dagang. Hal tersebut dilakukan demi mendapatkan kepercayaan dan mengatasi kelupaan manusia terhadap kewajibannya dalam keuangan.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas dapat ditarik rumusan masalah sebagai berikut:

1. Bagaimana mengintegrasikan sistem informasi akuntansi dengan ERP Pondok Pesantren?
2. Bagaimana mengotomasikan akun yang digunakan pada saat transaksi?

1.3 Hipotesis

Hipotesis dari rumusan masalah di atas adalah sebagai berikut.

1. Untuk mengintegrasikan sistem informasi perencanaan produksi dengan sistem lainnya yang berbasis ERP dapat menggunakan *web service* dengan *Service Oriented Architecture*.
2. Akun akan diberikan kepada tiap transaksi yg umum terjadi pada *enterprise* dan transaksi tersebut akan otomatis memiliki akun debit dan kredit tanpa *input user*.

1.4 Tujuan Penelitian

Tujuan dari penelitian yang akan dilakukan adalah sebagai berikut.

1. Mengintegrasikan sistem informasi akuntansi dengan ERP.
2. Mengotomasikan akun yang digunakan pada saat transaksi.

1.5 Manfaat Penelitian

Manfaat dari penelitian yang akan dilakukan adalah sebagai berikut.

Dengan adanya penelitian ini diharapkan dapat memberikan manfaat, diantaranya sebagai berikut.

1. Meningkatkan akurasi dalam penentuan akun dalam proses akuntansi.
2. Mengurangi biaya pengembangan dan pemeliharaan *software*.

1.6 Batasan Masalah

Untuk menghindari meluasnya permasalahan yang ada, serta keterbatasan ilmu dan kemampuan yang dimiliki peneliti maka batasan masalah pada penelitian ini adalah sistem akuntansi pendidikan yang akan dibangun hanya meliputi proses akuntansi pada pondok pesantren tipe D.

1.7 Sistematika Penulisan

Penulisan laporan skripsi ini tersusun dalam lima bab dengan sistematika penulisan sebagai berikut.

BAB I PENDAHULUAN

Bab ini merupakan pendahuluan yang berisi tentang latar belakang masalah, rumusan masalah, batasan masalah, tujuan, manfaat, dan sistematika penulisan skripsi.

BAB II KAJIAN PUSTAKA

Bab ini menjelaskan tentang informasi dan dasar-dasar teori yang akan digunakan sebagai penunjang untuk penyusunan tugas akhir ini.

BAB III ANALISIS DAN DESAIN SISTEM

Bab ini menjelaskan tentang analisa dan desain sistem untuk membuat sistem manajemen meliputi identifikasi kebutuhan dalam pembuatan sistem dan langkah-langkah pembuatan sistem.

BAB IV HASIL DAN PEMBAHASAN

Bab ini berisi tentang penjelasan sistem yang telah dibangun dan hasil uji coba yang telah dilakukan. Penjelasan sistem dan hasil uji coba sistem akan mendapatkan hasil sehingga dapat ditarik sebuah kesimpulan.

BAB V PENUTUP

Bab ini berisi tentang kesimpulan berdasarkan pembahasan sistem, tujuan, dan manfaat untuk pondok pesantren dan saran yang dapat bermanfaat untuk pengembangan aplikasi sistem.



BAB II

KAJIAN PUSTAKA

2.1 *Enterprise Resource Planning (ERP)*

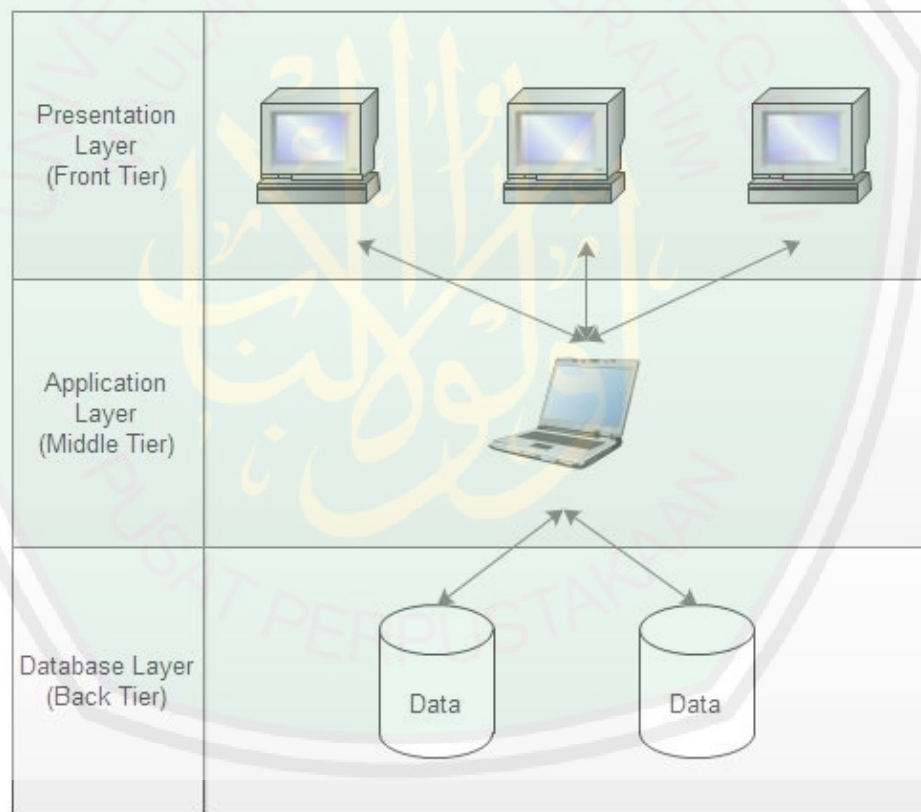
Enterprise Resource Planning adalah suatu *cross-functional* atau sistem informasi yang diperuntukkan bagi perusahaan manufaktur maupun jasa guna mengintegrasikan dan mengotomasikan proses bisnis di dalam pabrik, logistik, distribusi, akuntansi, keuangan, dan sumber daya manusia. Implementasi ERP merupakan investasi dan juga tulang punggung perusahaan guna meningkatkan efisiensi kinerja serta mengembangkan bisnis. Pada prinsipnya dengan sistem ERP, sebuah industri atau perusahaan dapat berjalan secara optimal dan dapat mengurangi biaya-biaya operasional yang tidak efisien, seperti biaya *inventory* maupun biaya kerugian akibat kesalahan teknis (Yasin, 2013). *Enterprise Resource Planning (ERP)* merupakan salah satu solusi sistem informasi terintegrasi dan terpadu yang digunakan oleh sebuah perusahaan dalam menjalankan bisnisnya. ERP mencakup beberapa domain fungsi yang saling terintegrasi sehingga menjadi sebuah kesatuan sistem yang terpadu. Integrasi dalam ERP mencakup integrasi dalam hal proses bisnis serta integrasi dalam hal manajemen pengguna (Sarno, Putra, & Sunaryono, 2011).

Enterprise Resource Planning adalah sistem informasi yang digunakan untuk mengintegrasikan dan mengotomasikan proses bisnis perusahaan, dari proses pengadaan barang, produksi, hingga proses penjualan. Dewasa ini, banyak perusahaan-perusahaan yang tertarik dan berpindah menggunakan sistem ERP ini (Herlambang, Sarno, & Sunaryono, 2013). Sistem ERP sekarang ini banyak

menganut sistem arsitektur 3 tingkat atau lebih. Dalam sistem arsitektur 3 tingkat, *user interface* berjalan di *client*.

1. *Presentation Layer: Graphical User Interface (GUI)* atau *browser* untuk memasukkan data atau mengakses fungsi system.
2. *Application Layer: aturan bisnis, logika fungsi, dan program yang menerima/mengirim dari/ke server database.*
3. *Database Layer: Manajemen transaksi data termasuk pula metadatanya.*

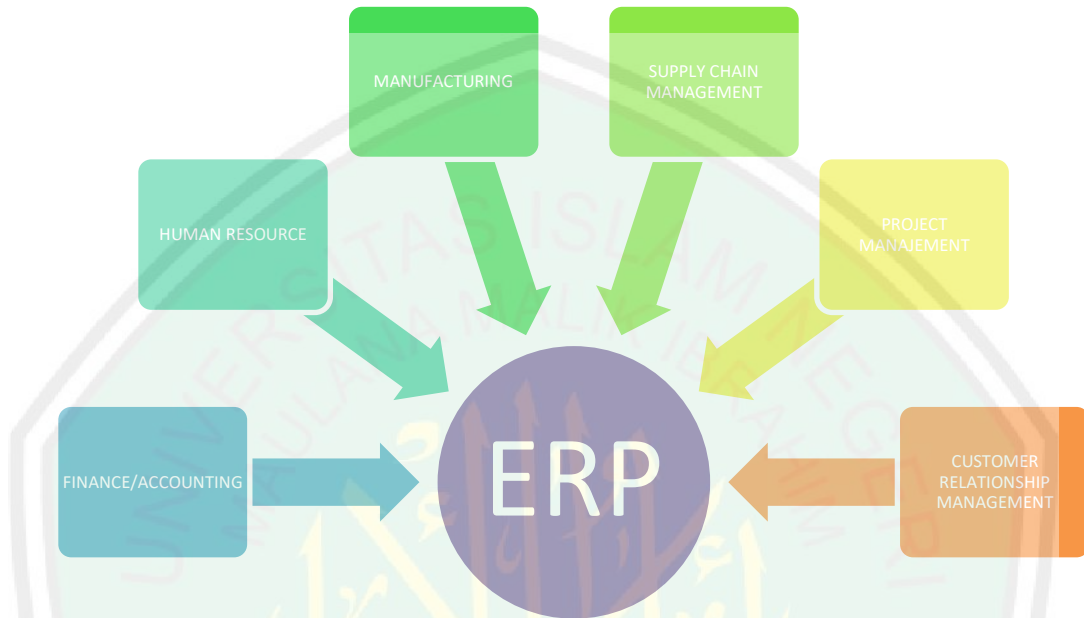
Arsitektur ERP dapat diilustrasikan pada Gambar 2.1 sebagai berikut.



Gambar 2.1 Arsitektur ERP (Wibisono, 2005)

Teknologi *Enterprise Resources Planning (ERP)* dapat mengintegrasikan fungsi marketing, fungsi produksi, fungsi logistik, fungsi *finance*, fungsi sumber daya, fungsi produksi, dan fungsi lainnya. *Enterprise Resources Planning* telah berkembang sebagai alat integrasi yang memiliki tujuan untuk mengintegrasikan

semua aplikasi perusahaan ke pusat penyimpanan data sehingga dengan mudah diakses oleh semua bagian yang membutuhkan (Yasin, 2013). Sistem Global ERP dapat dilihat pada Gambar 2.2 berikut.



Gambar 2.2 Sistem Global ERP (Sunaryono, Sarno, Hariadi, & Kurniawan, 2013)

Dalam SAP-ERP dijelaskan bahwa ada enam bagian utama dalam ERP, diantaranya yaitu. (Wibisono, 2005)

1. *Sales and Distribution.*

Sales and distribution ini adalah salah satu modul SAP yang membantu meningkatkan efisiensi kegiatan operasional berkaitan dengan proses pengelolaan *customer order* (*sales support, sales, shipping, dan machineing*).

Sales and distribution berfungsi untuk menangani proses penjualan (*sales*) dan pengelolaan pengiriman barang ke *customer*. *Sales and distribution* merupakan dasar dari semua transaksi yang dijalankan yang terdiri dari seluruh master data, *system configuration* dan transaksi untuk menyelesaikan proses pemesanan.

Modul *Sales and Distribution* ditekankan pada penggunaan strategi penjualan yang mampu mengantisipasi perubahan pasar. Prioritas utama dari penggunaan modul ini adalah untuk membuat struktur data yang mampu merekam, menganalisis, dan mengontrol aktifitas untuk memberikan kepuasan kepada pelanggan dan menghasilkan keuntungan yang layak dalam periode akuntansi yang akan datang.

2. *Production Planning.*

Production planning merupakan proses untuk menentukan yang akan diproduksi oleh perusahaan serta menentukan mesin, bahan, dan kebutuhan lainnya dalam suatu siklus produksi. Modul *production planning* berfungsi dalam merencanakan dan mengendalikan jalannya *material* sampai kepada proses pengiriman produk.

3. *Material Management.*

Proses bisnis pada *material management* merupakan proses bisnis yang menerangkan jalanya bisnis dari permintaan sampai proses pembayaran *procure-to-pay process*. Lebih tepatnya *material management* membantu menjalankan proses pembelian (*procurement*) dan mengelola *inventory*. Fungsi utama dari modul *Material Management* adalah untuk membantu manajemen dalam aktifitas sehari-hari dalam tipe bisnis apapun yang memerlukan konsumsi *material*, termasuk energi dan pelayanan.

4. *Manufacturing Execution.*

Modul ini meliputi beberapa langkah, yaitu penciptaan dan pelepasan tatanan produksi, masalah komponen barang, konfirmasi aktivitas produksi, dan penerimaan barang dari barang jadi.

5. *Financial Accounting.*

Pada bagian ini dilakukan proses pencatatan transaksi keuangan dengan mengikuti kaidah-kaidah dalam akuntansi keuangan. Semua transaksi dicatat dalam jurnal transaksi dan secara berkala diposting ke dalam buku besar. Dalam pencatatan transaksi disertakan juga kode-kode akunnya. Kemudian dari data yang ada dalam buku besar tersebut dapat dibuat berbagai laporan keuangan.

6. *Control.*

Pada bagian ini dilakukan pengendalian terhadap kegiatan-kegiatan yang terkait dengan transaksi keuangan, sehingga bisa diketahui tingkat keuntungan yang didapatkan oleh *enterprise*.

Dalam pencapaian harapan yang tinggi meliputi penghematan biaya dan peningkatan layanan dalam penerapan ERP sangat bergantung pada seberapa jauh kesesuaian sistem ERP dengan fungsional perusahaan dan seberapa optimal modifikasi dan konfigurasi ulang yang dilakukan atas proses-proses yang ada pada sistem agar sesuai dengan kultur bisnis, strategi, dan struktur perusahaan.

Siklus Hidup Pengembangan Sistem ERP terdiri dari beberapa fase, yaitu (Yasin, 2013).

1. Fase 1: Perencanaan.

- a. Mengidentifikasi tujuan utama dan ruang lingkup proyek ERP, menentukan manajer proyek dan anggota tim lainnya.
- b. Mendefinisikan masalah yang akan diselesaikan oleh sistem ERP dan menentukan ruang lingkup proyek secara lebih spesifik.

- c. Mengevaluasi alternatif pendekatan pada ERP dan biasanya berupa solusi kostumisasi, integrasi, dan kombinasi paket yang akan digunakan.
- d. Membuat jadwal dan anggaran proyek, dengan memperhatikan kelayakan dan melaporkan temuan kepada komite pengarah baik secara tertulis maupun lisan.

2. Fase 2: Analisis.

- a. Tim proyek membentuk kelompok kerja pada berbagai fungsi di organisasi untuk mengumpulkan informasi dan mendefinisikan kebutuhan.
- b. Tim proyek mengevaluasi *vendor* yang diperkirakan dapat memenuhi kebutuhan dan membuat rekomendasi kepada tim pengarah.
- c. Beberapa pendekatan khusus dapat dilakukan untuk mengevaluasi beberapa alat bantu pengembangan *software*.
- d. Tim proyek memilih *vendor* dan melakukan evaluasi lebih terinci atas *vendor* yang terpilih.
- e. Tanggung jawab utama tim proyek pada fase ini adalah mengidentifikasi inisiatif rekayasa ulang proses bisnis yang mungkin diperlukan, berdasarkan paket *software* yang dipilih dengan bekerja sama dengan fungsi terkait.
- f. Jumlah rekayasa proses bisnis yang harus dilakukan bergantung pada pendekatan ERP dan paket yang dipilih.
- g. Memilih beberapa paket dan kombinasi beberapa alternatif paket yang tersedia dan disesuaikan dengan kebutuhan.

- h. Idealnya pada fase ini akan dihasilkan sebuah *Prototype* sistem ERP di berbagai area untuk menyimulasikan dan menunjukkan integrasi antarmodul kepada *user*.
 - i. Dilakukan evaluasi ulang atas alternatif yang pernah diajukan sebelumnya, sehingga didapatkan persetujuan dan verifikasi kelanjutan proyek.
 - j. Pada fase ini biasanya lebih singkat waktunya jika menggunakan pendekatan satu kesatuan paket dan lebih memakan waktu jika perusahaan memilih menggunakan pendekatan kustomisasi.
3. Fase 3: Desain.
- a. Fase disain dimulai setelah perusahaan memutuskan *vendor* mana yang dipilih.
 - b. Tingkat disain tergantung pada pendekatan ERP yang dipilih.
 - c. Pada pendekatan kustomisasi, perancangan aplikasi, *prototype* dan *database* dilakukan sangat insentif.
 - d. Pendekatan *prototype* sangat bermanfaat dalam melengkapi identifikasi kebutuhan baik pendekatan kustomisasi, kesatuan paket ataupun kombinasi beberapa paket.
 - e. Melakukan desain *hardware* dan teknologi jaringan yg akan digunakan, termasuk didalamnya memilih arsitektur *client-server*, serta mempertimbangkan *platform* yang digunakan pada saat ini.
 - f. Pengguna akhir (*end-user*) harus mendapatkan pelatihan secara intensif atas paket-paket ERP.
 - g. Pada fase ini dimungkinkan untuk merekayasa ulang proses bisnis dalam tingkatan yang terinci dan perlunya dokumentasi yang baik.

4. Fase 4: Implementasi.
 - a. Implementasi atau konstruksi terhadap sistem ERP disesuaikan dengan jenis proses bisnis yang pada masing-masing fungsi bisnis.
 - b. Pada pendekatan kombinasi paket, program dari beberapa *vendor* yang berbeda harus terintegrasi menjadi satu kesatuan sistem dengan menggunakan *middleware*.
 - c. Jadi pada intinya pada fase ini lebih fokus bagaimana cara mengintegrasikan paket- paket yang ada pada sistem ERP.
 - d. Melakukan evaluasi dari implementasi yang sudah dilakukan serta melakukan verifikasi dan pengujian terhadap keseluruhan system.
 - e. Membuat rencana *roll out* sistem yang meliputi jadwal instalasi sistem diseluruh organisasi dengan pendekatan yang bisa digunakan, misalnya: *pilot, parallel, dan cut-over*.
 - f. Selama fase ini, semua rencana rekayasa ulang proses bisnis diterapkan, sehingga yang perlu dikaji selanjutnya adalah orang dan prosedur.
 - g. Biasanya dengan adanya perubahan sistem dalam organisasi maka akan ada prosedur kerja baru yang harus disusun dan diterapkan.
5. Fase 5: Dukungan Teknis.
 - a. Tujuan dari fase ini adalah untuk menjamin keberhasilan sistem jangka pendek dan jangka panjang.
 - b. Dukungan teknis sangat diperlukan dalam transisi sistem yang berlangsung dalam organisasi.
 - c. Termasuk juga dalam fase ini adalah pemeliharaan sistem ERP, pemeliharaan bisa saja meliputi koreksi kesalahan yang ditentukan oleh

user, sehingga dalam tahap analisis dan disain sebaiknya dapat meminimalkan kesalahan.

- d. Jika terjadi kesalahan diharapkan adanya respon yang cepat dari konsultan yang berpengalaman untuk tetap menjaga kepercayaan *user* terhadap sistem.
- e. Pemeliharaan bisa saja dilakukan secara adaptif dengan adanya *upgrade* versi paket atau modul atau kostumisasi akibat adanya penambahan kebutuhan.
- f. Perlu adanya audit sistem yang dilakukan secara berperiodik, sehingga dapat menjaga kinerja sistem secara optimal.

Keuntungan implementasi ERP bagi perusahaan-perusahaan tersebut adalah (Yasin, 2013).

1. ERP membantu memperlancar proses bisnis dan membuatnya jadi lebih mudah, murah, cepat, dan efisien.
2. Mengurangi biaya-biaya berupa penghematan biaya operasional perusahaan. Hal ini disebabkan karena sistem ERP sudah didesain sedemikian rupa sehingga dapat mengurangi dan menghilangkan duplikasi data.
3. Pengambilan keputusan, yaitu sistem ERP yang merupakan sistem yang mengintegrasikan seluruh data dan informasi sangat membantu perusahaan dalam pengambilan keputusan, terutama apabila akan muncul masalah dalam perusahaan maka dengan cepat mereka dapat mengetahuinya dan segera mencari dan mengambil keputusan guna memecahkan masalah tersebut.
4. Meningkatkan etos kerja karyawan, karena proses kerja tersusun sesuai dengan standar operasi perusahaan yang sudah dibakukan.

5. Meningkatkan jumlah penjualan, karena sistem ERP ini membantu dalam keluar masuknya arus barang.
6. Menambah daya saing perusahaan, karena ERP membantu dalam distribusi produk yang dihasilkan perusahaan dengan memberikan informasi yang cepat dan akurat bagi konsumen.
7. Mengurangi kecurangan dan biaya dengan menghapuskan aktifitas yang tidak memiliki nilai tambah.

Jadi, ERP memiliki beberapa bagian yang memiliki fungsi yang berbeda-beda pula. Dalam siklus hidupnya, ERP memiliki beberapa fase yang harus dilewati

2.2 Akuntansi

Di Indonesia istilah akuntansi relatif belum lama dikenal bila dibandingkan dengan istilah tata buku. Sampai dengan akhir tahun 1950-an masyarakat hanya mengenal istilah tata buku yang merupakan warisan dari Belanda. Istilah akuntansi mulai dikenal pada awal tahun 1960-an, ketika ilmu akuntansi Amerika Serikat mulai masuk ke Indonesia. Tata buku merupakan bagian dari proses akuntansi. Badan yang berwenang dan beberapa ahli memberi pengertian yang bervariasi tergantung pada sudut pandang dan penekanan yang dianut. Pada prinsipnya, apa yang diungkapkan para ahli tersebut menuju ke satu pengertian akuntansi, yaitu informasi keuangan (Bastian, 2006).

Definisi akuntansi dapat dirumuskan dari dua sudut pandang, yaitu dari sudut pandang pemakai jasa akuntansi dan dari proses kegiatannya. Ditinjau dari sudut pandang pemakainya, akuntansi dapat didefinisikan sebagai suatu disiplin ilmu yang menyediakan informasi yang diperlukan untuk melaksanakan kegiatan

secara efisien dan mengevaluasi kegiatan-kegiatan suatu organisasi. Dari definisi tersebut dapat disimpulkan bahwa (Bastian, 2006).

1. Akuntansi diselenggarakan dalam suatu organisasi (contohnya berupa organisasi institusi pendidikan). Informasi akuntansi yang dihasilkan adalah informasi tentang organisasi.
2. Informasi akuntansi sangat penting dalam menyelenggarakan kegiatan institusi pendidikan. Informasi tersebut dipergunakan dalam pengambilan keputusan internal dan eksternal organisasi.

Ditinjau dari sudut pandang proses kegiatannya, akuntansi dapat didefinisikan sebagai proses pencatatan, penggolongan, peringkasan, pelaporan, dan penganalisisan data keuangan suatu organisasi. Dari pengertian di atas dapat disimpulkan bahwa akuntansi pada dasarnya (Bastian, 2006).

1. Mengidentifikasi data yang berkaitan atau relevan dengan keputusan yang akan diambil.
2. Memproses atau menganalisis data yang relevan.
3. Mengubah data menjadi informasi yang dapat digunakan untuk pengambilan keputusan.

2.2.1 Akuntansi Pendidikan

Siklus akuntansi di Sektor Pendidikan pada dasarnya sama dengan siklus akuntansi pada umumnya, namun aplikasinya di lapangan mengalami sejumlah perubahan, agar sesuai dengan tujuan pencatatan akuntansi dan status lembaga pendidikan yang bersangkutan. Lembaga Pendidikan adalah bersifat nirlaba, sehingga memiliki karakteristik yang berbeda dengan karakteristik akuntansi bisnis. Sebagai lembaga yang bersifat nirlaba, pelaporan keuangan disusun

berdasarkan Pernyataan Standar Akuntansi Keuangan No.45. Pelaporan keuangan di sektor pendidikan dimaksudkan untuk menyajikan dan mengungkapkan secara penuh aktivitas lembaga pendidikan termasuk unit-unit di dalamnya dan sumber daya ekonomi yang dipercayakan oleh para penyumbang, anggota organisasi lembaga pendidikan tersebut, kreditur dan pihak lain serta untuk mempertanggung jawabkannya sesuai dengan peraturan perundang-undangan yang berlaku dengan memperhatikan prinsip-prinsip akuntabilitas dan transparansi. Ciri dasar atau karakteristik Akuntansi Sektor Pendidikan adalah.

1. Tidak bertujuan untuk mengukur laba.

Tujuan lembaga pendidikan tidak untuk memperoleh laba tetapi memberikan pelayanan kepada masyarakat dalam bidang pendidikan dan pengajaran, penelitian serta pengabdian pada masyarakat sehingga harus memberikan informasi keuangan mengenai sumber daya ekonomi/keuangan yang digunakan untuk pelayanan dan asal sumber-sumber tersebut. Sesuai dengan PSAK No: 45 sumber daya tersebut diklasifikasikan sebagai berikut.

- a. Terikat yaitu sumber daya yang penggunaannya dibatasi untuk tujuan tertentu oleh penyedia dana. Pembatasan tersebut dapat bersifat temporer/semantara atau permanen. Pembatasan temporer/semantara adalah pembatasan penggunaan sumber daya oleh penyumbang yang menetapkan agar sumber daya tersebut dipertahankan sampai dengan terpenuhinya keadaan tertentu. Pembatasan permanen adalah pembatasan penggunaan sumber daya yang ditetapkan oleh penyumbang agar sumber daya tersebut dipertahankan secara permanen, tetapi organisasi diizinkan

untuk menggunakan sebagian atau semua penghasilan atau manfaat ekonomi lainnya yang berasal dari sumber daya tersebut.

- b. Tidak Terikat adalah sumber daya yang penggunaannya tidak dibatasi untuk tujuan tertentu oleh penyedia dana.

2. Akuntansi Dana.

Lembaga Pendidikan sebagai organisasi nirlaba menggunakan sistem akuntansi dana yang mencakup akuntansi yang memisahkan dana menjadi dana lancar dan dana tidak lancar sesuai dengan masing-masing sumber dana yang dipercayakan. Dana diidentifikasi sebagai suatu entitas akuntansi dan pelaporannya disajikan terpisah untuk setiap dana. Aktiva bersih (seperti ekuitas dalam akuntansi komersial) dikelompokkan dalam berbagai rekening seperti: aktiva bersih terikat permanen, aktiva bersih terikat temporer, dan aktiva bersih tidak terikat.

Adapun tahapan-tahapan akuntansi pendidikan adalah sebagai berikut (Bastian, 2006).

1. Tahap Pencatatan.
 - a. Kegiatan pengidentifikasian dan pengukuran bukti transaksi serta bukti pencatatan.
 - b. Kegiatan pencatatan bukti transaksi ke dalam buku harian atau jurnal.
 - c. Memindah bukukan (*posting*) dari jurnal berdasarkan kelompok atau jenisnya ke dalam akun buku besar.
2. Tahap Pengikhtisaran.
 - a. Penyusunan neraca saldo (*trial balance*) berdasarkan akun-akun buku besar.

- b. Penyusunan ayat jurnal penyesuaian (*adjusting entries*).
 - c. Penyusunan kertas kerja (*work sheet*) atau neraca lajur.
 - d. Pembuatan ayat jurnal penutup (*closing entries*).
 - e. Pembuatan neraca saldo setelah penutupan (*post-closing trial balance*).
 - f. Pembuatan ayat jurnal pembalik (*reversing entries*).
3. Tahap Pelaporan.
- a. Laporan Surplus Defisit.
 - b. Laporan Arus Kas.
 - c. Neraca.
 - d. Catatan atas Laporan Keuangan.

Tujuan pembahasan sistem akuntansi biaya di sektor pendidikan sekolah adalah sebagai berikut.

1. Mengefektifkan dan mengefisienkan penggunaan dana sekolah.
2. Mengetahui penyebab utama biaya yang terjadi di sekolah.
3. Memberikan informasi berupa laporan biaya yang akurat.
4. Memberikan jaminan akuntabilitas dan transparansi penggunaan dana dan pelaporannya.
5. Menghasilkan laporan biaya yang terkini sebagai bahan pertimbangan dalam pengambilan keputusan pengelola sekolah, terutama pada aspek keuangan.

2.2.2 Sistem Informasi Akuntansi

Sistem Informasi Akuntansi (SIA) adalah sebuah sistem yang mengumpulkan, mencatat, menyimpan, dan memproses data sehingga menghasilkan informasi yang berguna dalam membuat keputusan. SIA bisa berupa kertas dan alat tulis (manual) maupun terkomputerisasi penuh (serba

otomatis) ayau kondisi di antara keduangan (gabungan manual dan komputerisasi) (Sarosa, 2009). Sistem informasi akuntansi oleh perusahaan dalam menangani kegiatan operasional sehari-hari untuk menghasilkan informasi-informasi akuntansi yang diperlukan oleh manajemen dan pihak-pihak yang terkait lainnya yang berhubungan dengan pengambilan keputusan dan kebijakan-kebijakan lainnya. Sistem Informasi Akuntansi yang terkomputerisasi akan membantu pengguna laporan keuangan karena secara otomatis komputer akan melakukan pengelolaan terhadap data yang dimasukkan kedalam sistem

Sistem informasi akuntansi adalah suatu subsistem dari SIM yang menyediakan informasi akuntansi dan keuangan, juga informasi lain yang diperoleh dari pengolahan rutin atas transaksi akuntansi. SIA menelusuri sejumlah besar informasi mengenai pesanan penjualan, penjualan dalam satuan unit dan mata uang, penagihan kas, pesanan pembelian, penerimaan barang, pembayaran, gaji, dan jam kerja (Rama & Jones, 2008).

Menurut Krismiaji (Putri & Widyawati, 2013), komponen sistem informasi akuntansi terdiri dari.

1. Tujuan.

Setiap sistem informasi dirancang untuk mencapai satu atau lebih tujuan yang memberikan arah bagi sistem tersebut secara keseluruhan.

2. Input.

Data harus dikumpulkan dan dimasukkan sebagai input kedalam sistem. Sebagian besar input berupa data transaksi. Namun, dalam perkembangannya sebuah sistem informasi akuntansi tidak hanya mengolah data dan menghasilkan informasi keuangan saja tetapi juga mengolah data dan

menghasilkan informasi non keuangan. Oleh karena itu sebagian input adalah berupa data non keuangan.

3. Output.

Informasi yang dihasilkan oleh sebuah sistem. Output dari sebuah sistem informasi akuntansi biasanya berupa laporan keuangan dan laporan internal seperti daftar umur piutang, anggaran, dan proyeksi arus kas. 4. Penyimpanan Data sering disimpan untuk dipakai lagi dimasa yang akan datang.

4. Pemroses.

Data harus diproses untuk menghasilkan informasi dengan menggunakan komponen pemroses.

5. Instruksi dan prosedur.

Sistem informasi tidak dapat memproses data untuk menghasilkan informasi tanpa instruksi dan prosedur rinci.

Lingkup yang dikerjakan oleh Sistem Informasi Akuntansi dan lima macam penggunaan informasi akuntansi adalah sebagai berikut. (Rama & Jones, 2008)

1. Membuat Laporan Eksternal.

Perusahaan menggunakan sistem informasi akuntansi untuk menghasilkan laporan-laporan khusus untuk memenuhi kebutuhan informasi dari para investor, kreditor, dinas pajak, badan-badan pemerintah, dan yang lain.

Laporan-laporan ini mencakup laporan keuangan, SPT pajak, dan laporan yang diperlukan oleh badan-badan pemerintah yang mengatur perusahaan dalam industri perbankan dan utilitas.

2. Mendukung Aktivitas Rutin.

Para manager memerlukan satu sistem informasi akuntansi untuk menangani aktivitas operasi rutin sepanjang siklus operasi perusahaan itu. Contohnya antara lain menerima pesanan pelanggan, dan menagih ke pelanggan.

3. Mendukung Pengambilan Keputusan.

Informasi juga diperlukan untuk mendukung pengambilan keputusan yang tidak rutin pada semua tingkat dari suatu organisasi. Contohnya antara lain mengetahui produk-produk yang penjualannya bagus dan pelanggan mana yang paling banyak melakukan pembelian.

4. Perencanaan dan Pengendalian.

Suatu sistem informasi juga diperlukan untuk aktivitas perencanaan dan pengendalian. Informasi mengenai anggaran dan biaya standar disimpan oleh sistem informasi, dan laporan dirancang untuk membandingkan angka anggaran dengan jumlah aktual.

5. Menerapkan Pengendalian Internal.

Pengendalian internal mencakup kebijakan-kebijakan, prosedur-prosedur, dan sistem informasi yang digunakan untuk melindungi aset-aset perusahaan dari kerugian atau korupsi, dan untuk memelihara keakuratan data keuangan. Dimungkinkan untuk membangun pengendalian ke dalam suatu sistem informasi akuntansi yang terkomputerisasi untuk membantu mencapai tujuan ini.

2.3 Business Process Modelling (BPM)

Proses bisnis merupakan serangkaian aktifitas yang saling terkait untuk mencapai tujuan bisnis tertentu yang diselesaikan secara berurutan ataupun paralel, oleh manusia atau sistem, baik di dalam maupun di luar organisasi.

Kompleksitas proses bisnis yang terjadi membuat perusahaan mencari cara untuk menggambarkan proses bisnis. Pemodelan proses bisnis digunakan untuk mengevaluasi dan melakukan perbaikan proses bisnis di masa mendatang. Analisa proses bisnis umumnya melibatkan pemetaan proses dan subproses di dalamnya hingga tingkatan aktivitas atau kegiatan. Analisa tersebut dapat dilakukan melalui pemodelan proses bisnis yang menggambarkan cara orang-orang atau pihak-pihak saling berinteraksi di dalam sistem, dan dijelaskan dengan cara atau standar tertentu (Ramdhani, 2015).

Business Process Modelling (BPM) atau Pemodelan Proses Bisnis (PPB) merupakan diagram yang umum mewakili urutan kegiatan secara implisit berfokus pada sebuah proses, tindakan dan kegiatan (*job*). Sumber Daya (*resource*) yang digambarkan dalam PPB menunjukkan bagaimana mereka akan diproses. Pemodelan proses bisnis adalah lintas fungsional, biasanya penggabungan pekerjaan dan dokumentasi lebih dari satu departemen dalam sebuah institusi, organisasi, atau perusahaan. Dalam situasi lebih rumit, pemodelan proses juga dimasukkan pada aktivitas proses eksternal pada organisasi dan sistem yang dimasukkan ke dalam sebuah proses *primer* / utama (Ramdhani, 2015).

BPMN adalah singkatan dari *Business Process Modelling Notation*, yaitu suatu metodologi baru yang dikembangkan oleh *Business Process Modeling Initiative* sebagai suatu standar baru pada pemodelan proses bisnis, dan juga sebagai alat desain pada sistem yang kompleks seperti sistem *e-Business* yang berbasis pesan. BPMN tidak hanya terdiri atas model diagram tetapi juga dilengkapi dengan serangkaian alat bantu untuk menurunkan proses tersebut

menjadi bahasa eksekusi bisnis. Kelebihan metoda BPMN dalam memodelkan masalah terletak pada kemampuannya untuk memodelkan bisnis yang bertipe e-bisnis dengan aliran informasi berupa pesan-pesan (message) yang dilewatkan antar entitas bisnis (Rosmalal & Falahah, 2007). Notasi yang digunakan dalam BPMN adalah (Sari & Asniar).

1. Notasi *Event*.

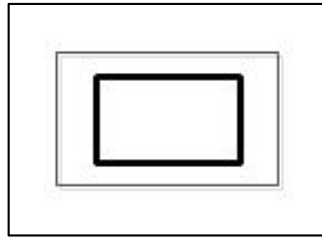
Event digambarkan dengan sebuah lingkaran dan merupakan sesuatu yang “terjadi” selama berlangsungnya proses bisnis. *Event-event* ini mempengaruhi aliran proses dan biasanya memiliki penyebab (*trigger*) atau hasil (*result*). *Event* adalah lingkaran dengan pusat terbuka untuk memungkinkan pembedaan *trigger* dan *result* yang berbeda. Terdapat 3 tipe *event* berdasarkan kapan mereka mempengaruhi aliran, yaitu: *Start*, *Intermediate*, dan *End*. Notasi *Event* dapat dilihat pada Gambar 2.3.



Gambar 2.3 Notasi *Event*

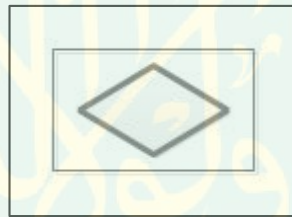
2. *Activity*.

Sebuah *activity* ditunjukkan dengan persegi panjang dengan ujung-ujung bulat dan merupakan bentuk umum untuk pekerjaan yang dilakukan oleh perusahaan. Sebuah aktivitas dapat berdiri sendiri atau gabungan. Jenis-jenis *activity* adalah: *task* dan sub-proses. Subproses dibedakan oleh kecil tanda plus di bagian tengah bawah dari bentuk. Notasi *Activity* dapat dilihat pada Gambar 2.4.

Gambar 2.4 Notasi *Activity*

3. *Gateway*.

Sebuah *gateway* digambarkan dengan bentuk wajik dan digunakan untuk mengontrol percabangan dan penggabungan Sequence Flow. Jadi, *gateway* menentukan keputusan tradisional, penggabungan, dan penggabungan aliran.. *Internal marker* akan menunjukkan jenis kontrol perilaku. Notasi *Gateway* dapat dilihat pada Gambar 2.5.

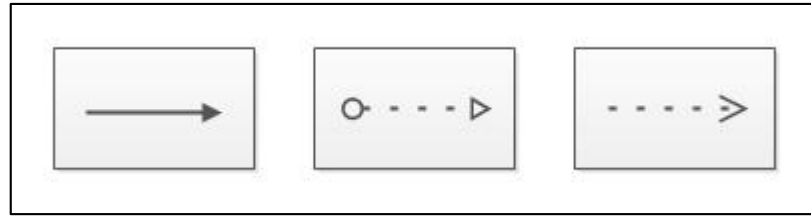
Gambar 2.5 Notasi *Gateway*

4. Penghubung.

Notasi penghubung terdiri dari tiga jenis yaitu.

- a. *Sequence flow*, yang menunjukkan kegiatan yang dituju merupakan kelanjutan kegiatan sebelumnya.
- b. *Message flow*, yang menunjukkan adanya aliran pesan dimana proses yang dituju bukan kelanjutan proses sebelumnya.
- c. *Association*, yang menunjukkan bentuk hubungan antara dua proses atau kegiatan.

Notasi Penghubung dapat dilihat pada Gambar 2.6.



Gambar 2.6 Notasi Penghubung

5. Data.

Notasi data bertujuan untuk mengidentifikasi elemen grafis yang spesifik secara visual. Menjelaskan apa yang dilakukan, mendeskripsikan tanda, jenis yang digunakan. Notasi data juga menggambarkan keterhubungan antar elemen grafis.

6. Artifak.

BPMN dirancang untuk memungkinkan pemodel dan alat-alat pemodelan fleksibel dalam memperluas notasi dasar dan dalam memberikan kemampuan untuk konteks tambahan sesuai dengan situasi pemodelan spesifik. Setiap jumlah artefak dapat ditambahkan ke diagram yang sesuai untuk konteks proses bisnis yang dimodelkan. BPMN mendefinisikan tiga jenis *artifacts*.

7. *Swimlanes*.

Swimlanes digunakan untuk memvisualisasikan kemampuan fungsional atau tanggung jawab yang berbeda.

2.4 SOA

Service Oriented Architecture (SOA) merupakan pendekatan yang menjadikan fungsi-fungsi aplikasi sebagai *service*, yang dipaket sebagai komponen yang dapat digunakan secara berulang serta bersifat independen, hal ini akan memudahkan apabila ada perubahan ataupun penambahan fungsi (Shofa, Aradea, & Kurnia,

2013). Selain itu SOA dapat diterjemahkan sebagai susunan dari aturan-aturan arsitektural yang diatur berdasarkan beberapa elemen dimana setiap elemennya dapat saling berhubungan dengan berbagai rancangan dan gaya arsitektur yang berbeda. SOA membagi fungsi-fungsi menjadi unit-unit yang berbeda (layanan), yang dapat didistribusikan melalui suatu jaringan dan dikombinasikan serta digunakan ulang untuk membentuk aplikasi bisnis. Layanan-layanan ini saling berkomunikasi dengan mempertukarkan data antar mereka atau dengan mengkoordinasikan aktivitas antara dua atau lebih layanan. Konsep SOA sering dianggap didasari atau berkembang dari konsep-konsep yang lebih lama dari komputasi terdistribusi dan pemrograman modular (Sunaryono, Sarno, Hariadi, & Kurniawan, 2013).

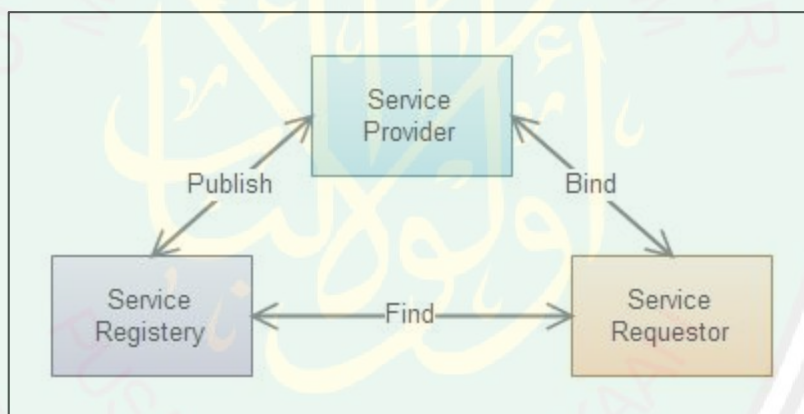
Service Oriented Architecture (SOA) adalah suatu cara perancangan aplikasi dengan menggunakan komponen-komponen atau pelayanan yang sudah ada tanpa harus merubah sistem yang telah ada. Hanya memerlukan penambahan modul-modul yang digunakan untuk mengintegrasikan sistem atau dengan kata lain disusun dalam bentuk modul (modular). Pendekatan SOA dilakukan dengan cara membagi permasalahan yang besar menjadi beberapa *service*. *Service* ini tidak hanya dapat digunakan oleh sistem yang menaunginya namun dapat digunakan juga oleh sistem lain yang berbeda, sehingga integrasi antar sistem dapat dicapai. SOA lebih sesuai untuk mengintegrasikan sistem yang heterogen dan lebih mudah beradaptasi dengan perubahan lingkungan, lebih efisien dan efektif. Salah satu bentuk dari implementasi SOA adalah teknologi *web service* dimana pertukaran data dapat dimungkinkan antar aplikasi yang berbeda *platform*. Teknologi ini banyak diterapkan pada aplikasi-aplikasi yang berbasis *web*. (Slameto, 2015)

Pada dasarnya SOA adalah sebuah arsitektur aplikasi dimana semua fungsi, layanan- layanan didefinisikan menggunakan sebuah bahasa yang terdeskripsi dan dapat mengakses antarmuka yang dipanggil untuk melakukan proses-proses bisnis. Setiap interaksi adalah independen terhadap yang lain. Karena antarmuka bersifat independen terhadap *platform*, setiap klien dari sembarang *device* dapat menggunakan *service* yang disediakan. SOA menghubungkan sistem operasi yang beragam dan mampu melakukan otomatisasi terhadap proses bisnis suatu organisasi secara internal atau *enterprise* (Slameto, 2015). Tujuan utama dari *Service Oriented Architecture* (SOA) adalah untuk menunjang dunia usaha dengan dunia teknologi informasi (TI) dengan cara membuat keduanya lebih efektif. SOA adalah sebuah jembatan yang menciptakan suatu simbiosis dan hubungan sinergis antara keduanya yang lebih kuat. Selain itu, SOA adalah tentang hasil usaha yang dapat dicapai dengan keselarasan yang lebih baik antara bisnis dan TI (Rifai, Sarno, & Sunaryono, 2016).

2.5 Web Service

Web Service merupakan teknologi yang tepat untuk menerapkan konsep SOA, karena *cycle* proses *web service* sama dengan SOA. Dalam *web service* bekerja pada infrastruktur *service*. Penggunaan infrastruktur *requester entity* ini dapat mengikat dan menggunakan *web service* dengan beberapa cara, tetapi *cycle* interaksi dalam *web service* akan selalu sesuai dengan *cycle* SOA. Beberapa alasan mengapa digunakannya *web service* adalah sebagai berikut (Slameto, 2015).

1. *Web service* dapat digunakan untuk mentransformasikan satu atau beberapa bisnis logic atau class dan obyek yang terpisah dalam satu ruang lingkup yang menjadi satu, sehingga tingkat keamanan dapat ditangani dengan baik.
2. *Web service* memiliki kemudahan dalam proses *deployment*nya, karena tidak memerlukan registrasi khusus ke dalam suatu sistem operasi. *Web service* cukup diupload ke *web server* dan siap diakses oleh pihak-pihak yang telah diberikan otorisasi.
3. *Web service* berjalan di port 80 yang merupakan protokol standar HTTP, dengan demikian *web service* tidak memerlukan konfigurasi khusus di sisi firewall.



Gambar 2.7 Arsitektur *Web Service* (Brittenham, 2002)

Pada Gambar 2.7 diatas, ada tiga komponen utama dari *web service* yaitu.

1. *Service provider* adalah penyedia *web service* yang berfungsi menyediakan kumpulan *web services* yang dapat diakses oleh pengguna.
2. *Service requestor* adalah aplikasi yang bertindak sebagai pengguna yang melakukan permintaan layanan (berupa *web services*) ke *service provider*.
3. *Service registry* adalah tempat dimana *service provider* mempublikasikan layanannya. Pada arsitektur *Web service*, *Service registry* bersifat opsional.

Dari gambar 2.7 diatas terdapat beberapa operasi dari *web service*, yaitu.

1. *Publish/Unpublish*: Menerbitkan/menghapus layanan ke dalam atau dari *registry*.
2. *Find*: *Service requestor* mencari dan menemukan layanan yang dibutuhkan.
3. *Bind*: *Service requestor* setelah menemukan layanan yang dicarinya, kemudian melakukan binding ke *service provider* untuk melakukan interaksi dan mengakses layanan/*service* yang disediakan oleh *service provider*.

Dalam perkembangannya, model *web service* memiliki dua metode yang berorientasi pada layanan dan sumberdaya informasi, yaitu: SOAP (*Simple Object Access Protocol*) dan REST (*REpresentational State Transfer*). Impementasi model SOA telah banyak dilakukan dan dikembangkan oleh banyak *vendor*.

SOAP (*Simple Object Application Protocol*) dan beberapa teknologi yang didukung seperti WSDL (*Web service Description Language*) dan UDDI (*Universal Description Discovery, and Integration*) merupakan kombinasi dari XML (*eXtensible Markup Language*) yang dikirimkan melalui HTTP (*HyperText Transport Protocol*). Adanya kombinasi dari XML dan HTTP membuat *web service* yang berbasis XML sangat mungkin untuk diimplementasikan sehingga menjanjikan banyak kemudahan dan perbaikan dalam mendukung integrasi berbagai *platform* sistem dan aplikasi, baik melalui infrastruktur Intranet maupun Internet/Ekstranet. Proses layanan dengan arsitektur SOAP memiliki tiga komponen utama, yaitu: 1) *service provider*, 2) *service requester*, dan 3) *service broker*, serta komponen pendukung yaitu: 1) XML, 2) SOAP-XML (terdiri atas *header* dan *body*), 3) WSDL, serta 4) UDDI.



Gambar 2.8 Blok Bangunan *Web Service* (Deviana, 2011)

Gambar 2.8 merupakan blok bangunan *web service* yang menyediakan fasilitas komunikasi jarak jauh antara dua aplikasi yang merupakan layer arsitektur *web service*.

1. *Layer 1* adalah Protokol internet standar yang digunakan sebagai sarana transportasi adalah HTTP dan TCP/IP.
2. *Layer 2* adalah *Simple Object Access Protocol (SOAP)* berbasis XML dan digunakan untuk pertukaran informasi antar sekelompok layanan.
3. *Layer 3* adalah *Web Service Definition Language (WSDL)* yang digunakan untuk mendeskripsikan atribut layanan.
4. *Layer 4* adalah *Universal Description, Discovery and Integration* yang mana merupakan direktori pusat untuk deskripsi layanan.

BAB III

ANALISIS DAN DESAIN SISTEM

3.1 Analisis Sistem

3.1.1 Gambaran Umum

Gambaran umum sistem informasi akuntansi melingkupi tiga hal sebagai berikut.

1. Data.

Data pada sistem informasi akuntansi terdiri dari data internal dan eksternal.

Data eksternal pada sistem ini berupa data akun dan data transaksi yang berasal dari sistem informasi keuangan. Sedangkan data internal berupa data akun.

2. Proses.

Dari data yang sudah ada baik data eksternal maupun data internal, kemudian data tersebut diolah untuk menghasilkan proses akuntansi berupa jurnal akuntansi, buku besar, dan neraca.

3. Komunikasi.

Sistem informasi akuntansi yang akan dibangun membutuhkan beberapa data yang diperoleh dari sistem informasi keuangan. Begitu juga data akun pada sistem informasi akuntansi akan dikirimkan ke sistem informasi keuangan.

Untuk melakukan pengiriman data dari sistem lainnya maka dibuatlah sebuah layanan yang disebut dengan *service*. *Service* tersebut kemudian diolah pada ESB WSO2 untuk mengkomunikasikan data yang diperlukan.

3.1.2 Sumber Data

Sumber data yang dipakai pada penelitian ini terdapat dua macam yaitu data primer dan data sekunder.

1. Data primer pada penelitian ini diperoleh dari literatur-literatur yang berkaitan dengan akuntansi pendidikan diantaranya buku akuntansi pendidikan, buku akuntansi umum, dokumentasi program-program akuntansi seperti Wave Accounting dan Zipbooks Accounting.
2. Data sekunder diperoleh dari jurnal maupun sumber lain yang berkaitan dengan akuntansi.

3.1.3 Analisis Proses Bisnis

3.1.3.1 Identifikasi dan Analisis Proses Bisnis

1. Identifikasi Proses Bisnis.

Terdapat empat proses bisnis pada sistem informasi akuntansi. Proses bisnis tersebut adalah sebagai berikut.

- a. Penentuan Akun.

Saat melakukan transaksi, user menentukan akun yang akan digunakan untuk transaksi tersebut. Akun yang diinputkan berupa akun debit, kredit, dan akun hutang piutang apabila transaksi tersebut merupakan transaksi hutang/piutang. Dengan begitu akun untuk suatu transaksi sudah tersedia dan dapat saat pembuatan jurnal, buku besar, dan neraca.

- b. Membuat Jurnal Akuntansi.

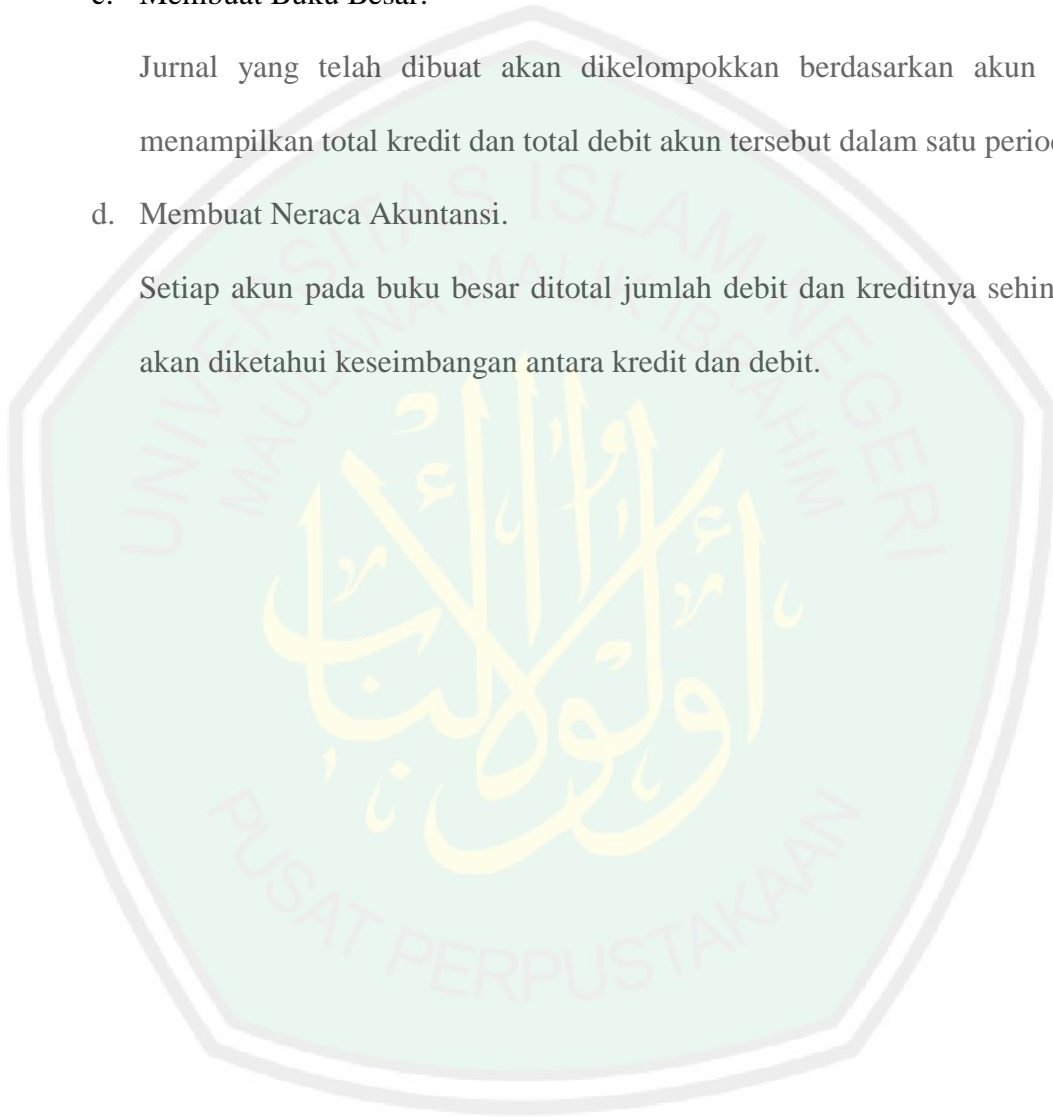
Transaksi-transaksi yang ada akan dicocokkan dengan akun yang telah dibuat. Jurnal dibuat dengan menampilkan kode akun dan posisi kredit dan debit akun tersebut termasuk utang piutang.

c. **Membuat Buku Besar.**

Jurnal yang telah dibuat akan dikelompokkan berdasarkan akun dan menampilkan total kredit dan total debit akun tersebut dalam satu periode.

d. **Membuat Neraca Akuntansi.**

Setiap akun pada buku besar ditotal jumlah debit dan kreditnya sehingga akan diketahui keseimbangan antara kredit dan debit.



2. Analisis Proses Bisnis.

Tabel 3.1 berikut adalah analisis proses bisnis yang telah disebutkan diidentifikasi proses bisnis di atas.

Tabel 3.1 Analisis Proses Bisnis

| No | Proses Bisnis | Siapa yang terlibat | Dimana proses bisnis terjadi | Kapan proses bisnis terjadi | Bagaimana proses bisnis terjadi | Dokumen Terkait |
|----|--------------------|----------------------------------|---|-----------------------------|---|---|
| 1. | Penentuan Akun | Bag. Keuangan, Bag. Akuntansi | Pondok Pesantren, masing-masing departemen | Saat terjadi transaksi | Bagian keuangan harus menentukan akun debit dan akun kredit untuk transaksi yang dilakukannya. | <ul style="list-style-type: none"> • Data Akun • Data Transaksi |
| 2. | Membuat Jurnal | Bag. Akuntansi | Pondok Pesantren | Setelah terjadi transaksi | Setelah terjadi transaksi dan akun yang terlibat ditentukan, maka sistem akan memasukkan transaksi tersebut kedalam jurnal. | <ul style="list-style-type: none"> • Data Akun • Data Transaksi |
| 3. | Membuat Buku Besar | Bag. Akuntansi | Pondok Pesantren | Setelah terjadi transaksi | Data transaksi akan dikelompokkan menurut akun oleh sistem, sehingga masing-masing akun akan memiliki jumlah total debit dan kreditnya masing-masing. | <ul style="list-style-type: none"> • Data Akun • Data Transaksi |
| 4. | Membuat Neraca | Bag. Akuntansi | Pondok Pesantren | Setelah terjadi transaksi | Data debit dan kredit masing-masing akun pada buku besar akan dijumlah untuk mengetahui apakah sudah seimbang atau belum. | <ul style="list-style-type: none"> • Data Akun • Data Transaksi |

3. Data Terkait.
 - a. Data Akun.
 - b. Data Transaksi.

3.1.3.2 Identifikasi dan Analisis Kebutuhan

1. Identifikasi dan Analisis Kebutuhan Fungsional.
 - a. Identifikasi Kebutuhan Fungsional.

Identifikasi kebutuhan fungsional penentuan akun dapat dilihat pada Tabel 3.2 berikut.

Tabel 3.2 Penentuan Akun

| No. | Keuangan | Akuntansi | Kebutuhan Fungsional SI |
|-----|------------------------------------|-------------------------------|-------------------------|
| 1. | Menentukan akun yang akan terlibat | Menentukan akun yang terlibat | Form tambah akun |

Identifikasi kebutuhan fungsional pembuatan jurnal dapat dilihat pada Tabel 3.3 berikut.

Tabel 3.3 Pembuatan Jurnal

| No. | Keuangan | Akuntansi | Kebutuhan Fungsional SI |
|-----|-------------------|----------------|-------------------------|
| 1. | Mendata transaksi | Membuat jurnal | Tabel Jurnal |

Identifikasi kebutuhan fungsional penentuan buku besar dapat dilihat pada Tabel 3.4 berikut.

Tabel 3.4 Penentuan Buku Besar

| No. | Keuangan | Akuntansi | Kebutuhan Fungsional SI |
|-----|-------------------|--------------------|-------------------------|
| 1. | Mendata transaksi | Membuat buku besar | Tabel Buku Besar |

Identifikasi kebutuhan fungsional pembuatan neraca dapat dilihat pada Tabel 3.5 berikut.

Tabel 3.5 Pembuatan Neraca

| No. | Keuangan | Akuntansi | Kebutuhan Fungsional SI |
|-----|-------------------|----------------|-------------------------|
| 1. | Mendata transaksi | Membuat neraca | Tabel Neraca |

b. Analisis Kebutuhan Fungsional.

Analisis kebutuhan fungsional sistem yang akan dikerjakan terlampir pada Tabel 3.6 berikut.

Tabel 3.6 Analisis Kebutuhan Fungsional

| No. | Kebutuhan Fungsional Sistem | Siapa yang terlibat | Dimana kegiatan dilakukan | Kapan Sistem melakukan | Bagaimana sistem bekerja | Dokumen Terkait |
|-----|-----------------------------|---------------------|---------------------------|------------------------|--|-----------------|
| 1. | Form tambah akun | Bag. Akuntansi | Pondok Pesantren | Sesuai kebutuhan | SI Akuntansi akan menambahkan akun pada table akun pada <i>database</i> . | Daftar akun |
| 2. | Tabel jurnal | Bag. Akuntansi | Pondok Pesantren | Sesuai kebutuhan | SI Akuntansi mengambil data transaksi dari SI Keuangan dan mengolahnya menjadi tabel Jurnal. | Data Transaksi |
| 3. | Tabel buku besar | Bag. Akuntansi | Pondok Pesantren | Sesuai kebutuhan | SI Akuntansi mengambil data transaksi dari SI Keuangan dan mengolahnya menjadi tabel buku besar. | Data Transaksi |
| 4. | Tabel neraca | Bag. Akuntansi | Pondok Pesantren | Sesuai kebutuhan | SI Akuntansi mengambil data transaksi dari SI Keuangan dan mengolahnya menjadi tabel neraca. | Data Transaksi |

2. Identifikasi dan Analisis Kebutuhan Non Fungsional.

Berikut adalah identifikasi dan analisis non-fungsional sistem yang terlampir pada Tabel 3.7.

Tabel 3.7 Identifikasi dan Analisis Kebutuhan Non-Fungsional

| Perangkat | Komponen Sistem Informasi | Spesifikasi | Siapa yang mengadakan | Kapan harus diadakan | Dimana harus diadakan | Bagaimana pengadaannya |
|-----------------|--|---|-----------------------|---|----------------------------|--|
| <i>Hardware</i> | <i>Server</i> | Intel Core2Duo : - RAM 2gb - Hardisk 80gb - LAN card | Pondok Pesantren | Pada awal pembuatan sistem informasi Pondok Pesantren | Di ruang <i>server</i> | Pengadaan <i>server</i> dengan membeli yang sesuai dengan keperluan sistem. Pemasangan baik itu dari segi Instalasi maupun <i>security</i> dilakukan dengan disesuaikan dengan keperluan sistem. |
| <i>Software</i> | Sistem Operasi Untuk <i>Server</i> | Windows7 | Pondok Pesantren | Pada awal pembuatan sistem informasi Pondok Pesantren | Di Kantor Pondok Pesantren | Instalasi manual dalam <i>hardisk</i> |
| | Sistem Operasi Untuk <i>Developing Website</i> | Windows7 dan Windows10 | Pondok Pesantren | Pada awal pembuatan sistem informasi Pondok Pesantren | Di Kantor Pondok Pesantren | Instalasi manual pada PC masing-masing |
| | <i>Editor Development Web</i> | - Notepad++ - Sublime | Pondok Pesantren | Pada awal pembuatan sistem informasi Pondok Pesantren | Di Kantor Pondok Pesantren | Instalasi manual di PC masing-masing |
| | ERD Designer | - CAERwin - XAMPP | Pondok Pesantren | Pada awal perancangan sistem | Di Kantor Pondok Pesantren | Download di internet |

Tabel 3.7 Identifikasi dan Analisis Kebutuhan Non-Fungsional (Sambungan)

| Perangkat | Komponen Sistem Informasi | Spesifikasi | Siapa yang mengadakan | Kapan harus diadakan | Dimana harus diadakan | Bagaimana pengadaannya |
|-----------------|--|--|-----------------------|---|----------------------------|--|
| <i>Software</i> | DFD Desiger | Power Designer ProcessAnalyst 6 | Pondok Pesantren | Pada awal perancangan sistem | Di Kantor Pondok Pesantren | Download di internet |
| | Arsitektur Website Designer | - EdrawMax 6.5 - Visio 2007 | Pondok Pesantren | Pada awal perancangan sistem | Di Kantor Pondok Pesantren | Membeli di toko <i>software</i> . |
| | Aplikasi untuk mendemokan website/produk | Balsmaiq Mockup dan Adobe Air | Pondok Pesantren | Pada awal perancangan sistem informasi Pondok Pesantren | Di Kantor Pondok Pesantren | Download di internet |
| | <i>Browser</i> | Google Chrome 22.0.1229.94, Mozilla Firefox 13 | Pondok Pesantren | Pada awal pembuatan sistem informasi Pondok Pesantren | Di Kantor Pondok Pesantren | Download di internet |
| | Bahasa Pemrograman Website | PHP 5.3.8, Ajax, Java Script, CSS 2&3, HTML5, JQuery, XML, Java. | Pondok Pesantren | Pada awal pembuatan sistem informasi Pondok Pesantren | Di Kantor Pondok Pesantren | tutorial <i>script</i> dari materi di internet, <i>e-book</i> , dan buku |
| | Aplikasi Transfer File | Filezilla 3.5.2 | Pondok Pesantren | Pada awal pembuatan sistem informasi Pondok Pesantren | Di Kantor Pondok Pesantren | Download secara gratis di Internet: http://filezilla-project.org/download.php |
| | <i>Web Server</i> | Apache 2.2.21 | Pondok Pesantren | Pada awal pembuatan sistem informasi Pondok Pesantren | Di Kantor Pondok Pesantren | Download di internet: http://http.apache.org/download.cgi |

Tabel 3.7 Identifikasi dan Analisis Kebutuhan Non-Fungsional (Sambungan)

| Perangkat | Komponen Sistem Informasi | Spesifikasi | Siapa yang mengadakan | Kapan harus diadakan | Dimana harus diadakan | Bagaimana pengadaannya |
|--|--|---|-----------------------|---|----------------------------|---|
| <i>Software</i> | DBMS | -Oracle 11g | Pondok Pesantren | Pada awal pembuatan sistem informasi Pondok Pesantren | Di Kantor Pondok Pesantren | Mendownload secara gratis di Internet |
| | <i>Network</i> | WLAN | Pondok Pesantren | Pada awal pembuatan sistem informasi Pondok Pesantren | Di Kantor Pondok Pesantren | Membeli peralatan WLAN di toko komputer |
| Orang yang terlibat dalam Operasional Sistem | Analisis Sistem | | Pondok Pesantren | Pada awal pembuatan sistem informasi Pondok Pesantren | Di Kantor Pondok Pesantren | Disediakan oleh pengembang |
| | Tampilan (<i>Interface</i>) <i>Web Designer</i> | Mempunyai dasar IMK yang bagus. | Pondok Pesantren | Pada awal desain sistem informasi Pondok Pesantren | Di Kantor Pondok Pesantren | Disediakan oleh pengembang |
| | <i>Programmer</i> | Menguasai pemrograman PHP, HTML, dan XML. | Pondok Pesantren | Pada awal perancangan sistem informasi Pondok Pesantren | Di Kantor Pondok Pesantren | Disediakan oleh pengembang |
| | <i>Administrator</i> | Mampu memajemen sistem dengan baik. | Pondok Pesantren | Pada awal perancangan sistem informasi Pondok Pesantren | Di Kantor Pondok Pesantren | Koordinator Pondok Pesantren |
| | Teknisi Jaringan | Mempunyai kemampuan di bidang jaringan. | Pondok Pesantren | Pada awal perancangan sistem informasi Pondok Pesantren | Di Kantor Pondok Pesantren | Disediakan oleh pengembang |

Tabel 3.7 Identifikasi dan Analisis Kebutuhan Non-Fungsional (Sambungan)

| Perangkat | Komponen Sistem Informasi | Spesifikasi | Siapa yang mengadakan | Kapan harus diadakan | Dimana harus diadakan | Bagaimana pengadaannya |
|--|---------------------------|--|-----------------------|---|----------------------------------|----------------------------|
| Orang yang terlibat dalam pengoperasian sistem | <i>Database designer</i> | Mampu mendesain secara efektif. | Pondok Pesantren | Pada awal perancangan sistem informasi Pondok Pesantren | Di Kantor Pondok Pesantren | Disediakan oleh pengembang |
| | <i>Moderator</i> | Orang yang mempunyai keahlian hampir sama dengan Administrator | Pondok Pesantren | Pada awal perancangan sistem informasi Pondok Pesantren | Di Kantor Dinas Pondok Pesantren | Semua Pegawai |



3.1.3.3 Identifikasi Output

Berikut ini adalah Tabel 3.8 yang menjabarkan identifikasi *output* sistem informasi akuntansi pondok pesantren tipe D.

Tabel 3.8 Identifikasi *Output*

| No. | Nama Laporan | Alat untuk menampilkan | Bentuk Laporan | Pembuat Laporan | Penerima Laporan | Periode Laporan | Deskripsi Laporan | Data/informasi yang ditampilkan |
|-----|-----------------|------------------------|----------------|-----------------|------------------|-----------------|--|--|
| 1. | Daftar Akun | Monitor | Tabel | Bag. Keuangan | Bag. Keuangan | Setiap Hari | Berisikan data akun yang telah terdaftar. | Kode akun, nama akun |
| 2. | Jurnal | Monitor | Tabel | Bag. Keuangan | Bag. Keuangan | Setiap Hari | Berisikan data transaksi yang telah diolah menjadi jurnal akuntansi. | Tanggal transaksi, kode akun, nama akun, debit, kredit |
| 3. | Buku Besar | Monitor | Tabel | Bag. Keuangan | Bag. Keuangan | Setiap Hari | Berisikan data transaksi yang telah diolah menjadi buku besar. | Tanggal transaksi, kode akun, nama akun, keterangan transaksi, debit, kredit |
| 4. | Neraca Keuangan | Monitor | Tabel | Bag. Keuangan | Bag. Keuangan | Setiap Hari | Berisikan data transaksi yang telah diolah menjadi neraca keuangan. | Kode akun, nama akun, debit, kredit |

3.1.3.4 Identifikasi Input

Identifikasi *input* sistem yang akan dikerjakan terlampir pada Tabel 3.9 berikut.

Tabel 3.9 Identifikasi Input

| No | Nama Proses Input | Alat Input | Bentuk Input | Yang Menyediakan Data | Periode Input | Deskripsi Input | Data/Informasi |
|----|-------------------|--------------------|--------------|-----------------------|------------------|-----------------|-----------------------|
| 1. | Input Akun Baru | Keyboard dan mouse | Teks | Admin | Sesuai Kebutuhan | Input data Akun | Nama akun, nomor akun |

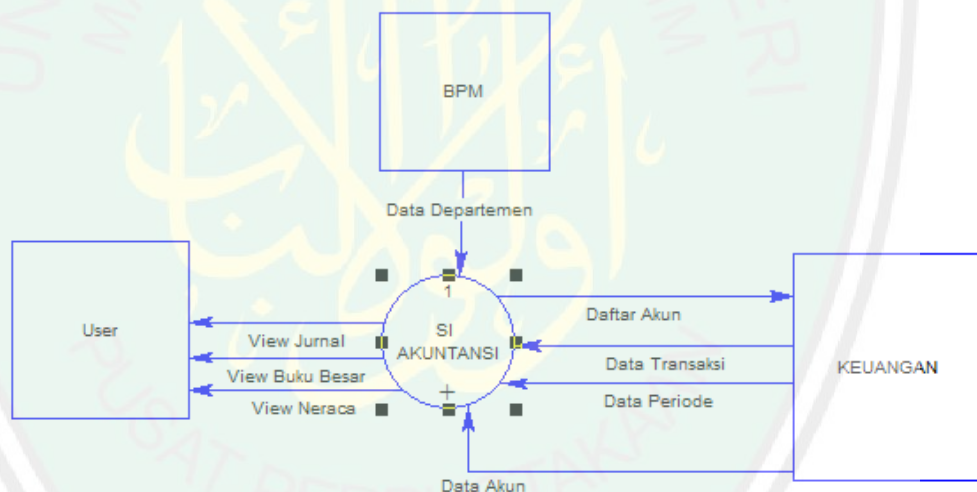


3.2 Desain Sistem

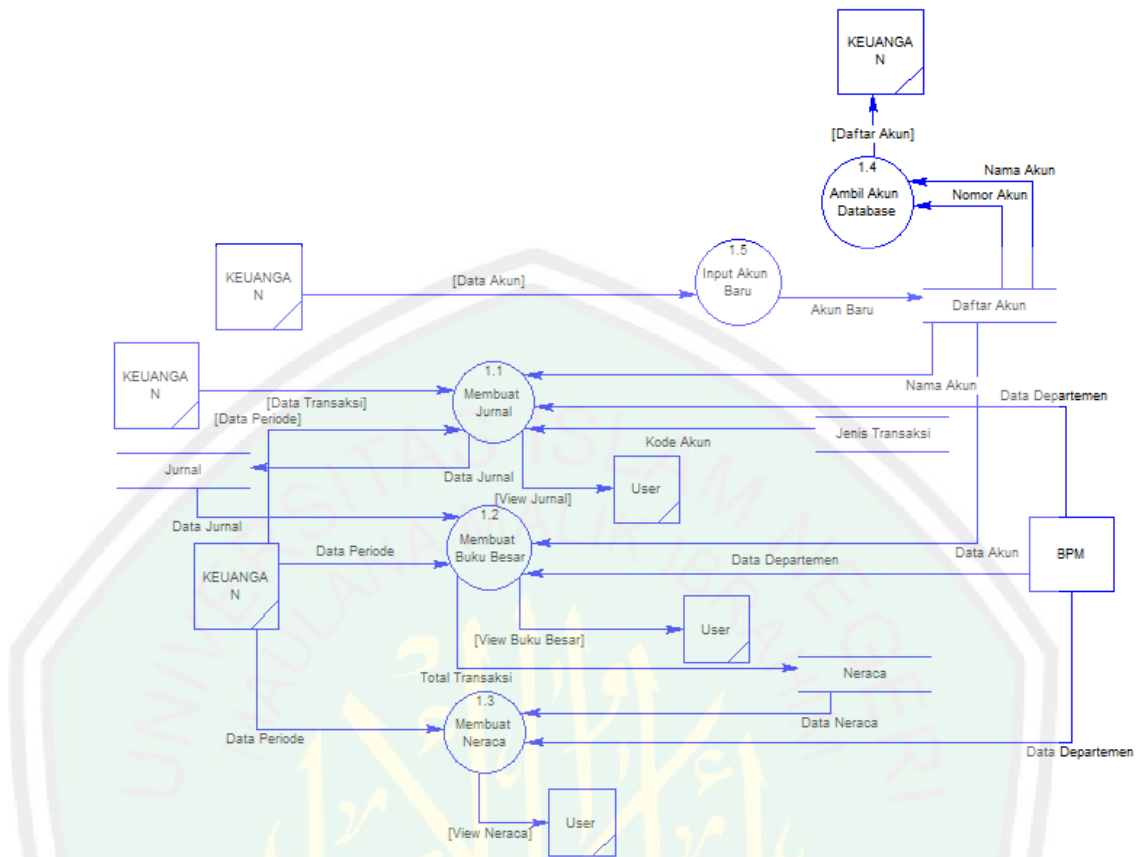
3.2.1 Pemodelan Sistem Informasi Akuntansi

1. DFD (*Data Flow Diagram*).

Context Diagram dari SI Akuntansi dapat dilihat pada Gambar 3.1. Gambar 3.1 menunjukkan alur data pada sistem informasi akuntansi yang akan dibangun secara global. Sedangkan DFD Level 1 Sistem Informasi Akuntansi dapat dilihat pada Gambar 3.2. Pada Gambar 3.2 terdapat lima proses yang menunjukkan proses-proses yang ada pada sistem informasi akuntansi.



Gambar 3. 1 *Context Diagram*



Gambar 3.2 DFD Level 1

2. ERD (Entity Relationship Diagram).

Database Sistem Informasi Akuntansi terdiri dari beberapa tabel berikut.

a. Tabel Akun.

Kolom pada tabel akun dapat dilihat pada Tabel 3.10.

Tabel 3.10 Tabel Akun

| Nama kolom | Tipe Data | Lebar | Key |
|------------|-----------|-------|-------------|
| Kode_akun | INT | 11 | PRIMARY KEY |
| Nama_akun | VARCHAR | 50 | NOT NULL |

b. Tabel Jenis Transaksi.

Kolom pada tabel jenis transaksi dapat dilihat pada Tabel 3.11.

Tabel 3.11 Tabel Jenis Transaksi

| Nama kolom | Tipe Data | Lebar | Key |
|----------------------|------------------|--------------|--------------------|
| id_jenis_transaksi | INT | 11 | <i>PRIMARY KEY</i> |
| nama_transaksi | VARCHAR | 100 | <i>NOT NULL</i> |
| Kode_rekening_debet | INT | 11 | <i>NOT NULL</i> |
| Kode_rekening_kredit | INT | 11 | <i>NOT NULL</i> |

c. Tabel Jurnal

Kolom pada tabel jurnal dapat dilihat pada Tabel 3.12.

Tabel 3.12 Tabel Jurnal

| Nama kolom | Tipe Data | Lebar | Key |
|-------------------|------------------|--------------|--------------------|
| id_jurnal | INT | 11 | <i>PRIMARY KEY</i> |
| kode_akun | INT | 11 | <i>NOT NULL</i> |
| id_transaksi | INT | 11 | <i>NOT NULL</i> |
| nama_transaksi | VARCHAR | 50 | <i>NOT NULL</i> |
| jumlah | INT | 11 | <i>NOT NULL</i> |

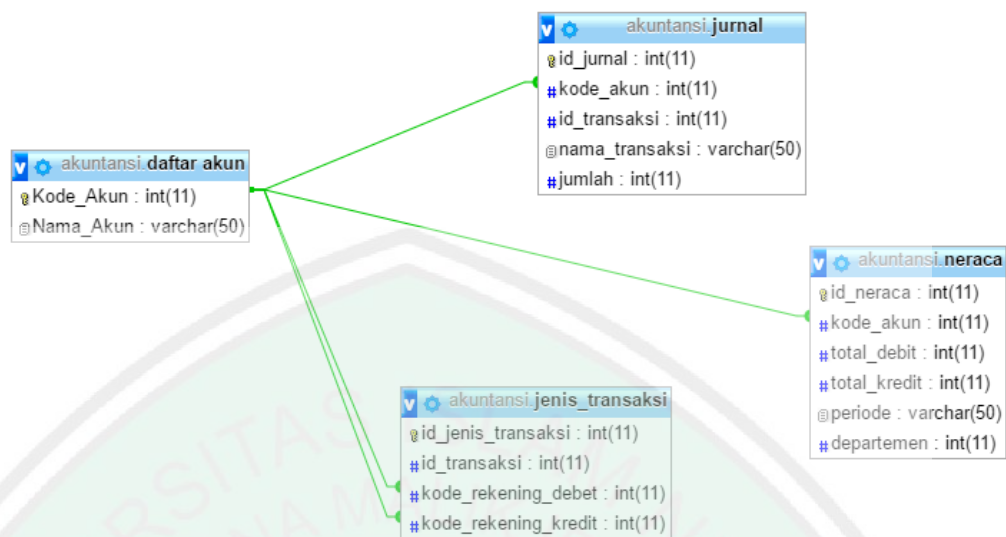
d. Tabel Neraca

Kolom pada tabel neraca dapat dilihat pada Tabel 3.13.

Tabel 3.13 Tabel Neraca

| Nama kolom | Tipe Data | Lebar | Key |
|-------------------|------------------|--------------|--------------------|
| id_neraca | INT | 11 | <i>PRIMARY KEY</i> |
| kode_akun | INT | 11 | <i>NOT NULL</i> |
| Total debit | INT | 11 | <i>NOT NULL</i> |
| Total kredit | INT | 11 | <i>NOT NULL</i> |
| periode | VARCHAR | 50 | <i>NOT NULL</i> |
| departemen | INT | 11 | <i>NOT NULL</i> |

Untuk ERD Sistem Informasi Akuntansi dapat dilihat pada Gambar 3.3 berikut.



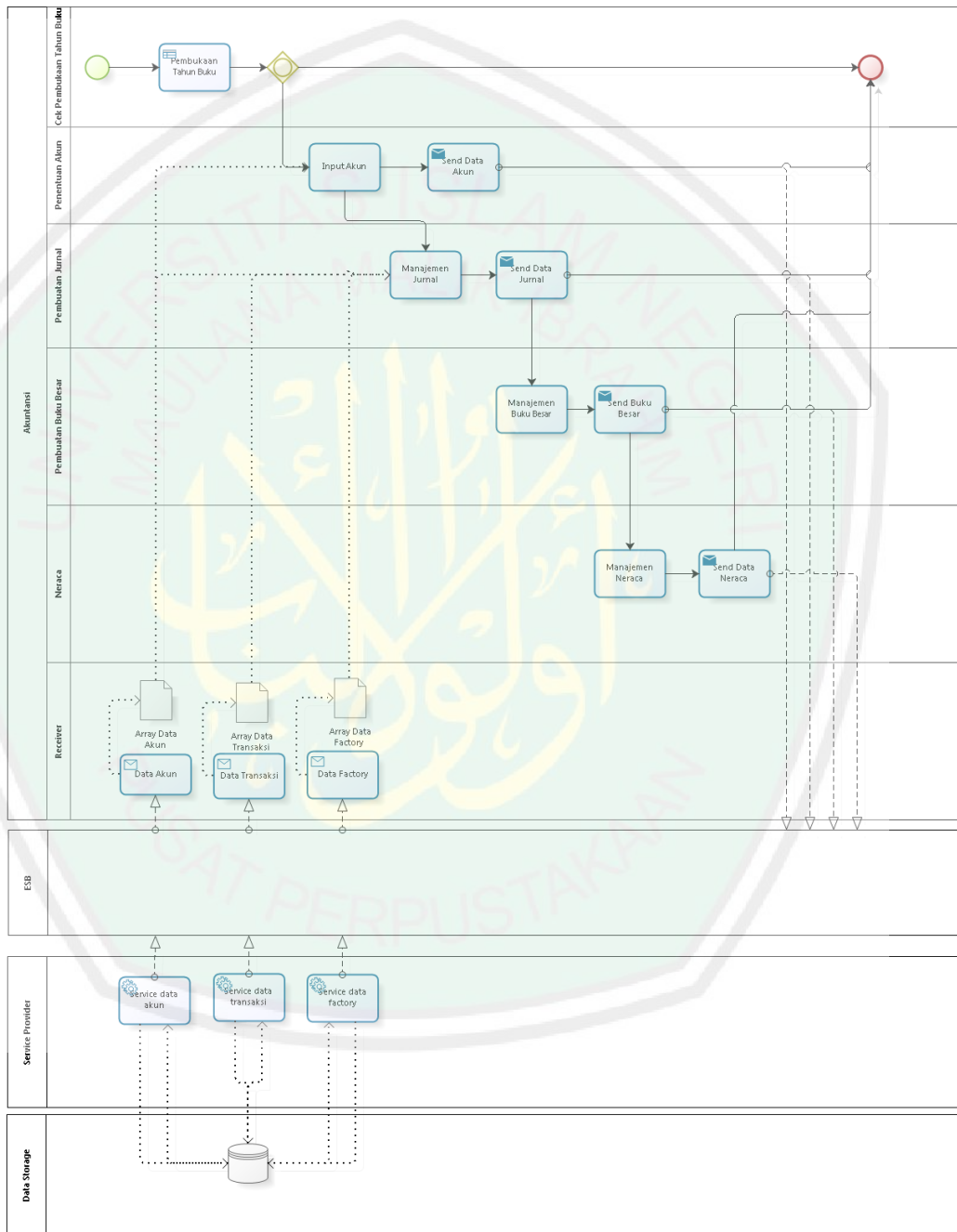
Gambar 3.3 Entity Relationship Diagram

3. BPMN.

Proses bisnis sistem informasi akuntansi pada pondok pesantren yang akan dibangun dimodelkan menggunakan BPMN. Pemodelan tersebut dapat dilihat pada Gambar 3.4. Terdapat empat proses penting pada akuntansi yang akan dibangun yaitu pendataan akun, pembuatan jurnal, pembuatan buku besar, dan neraca dimana proses-proses tersebut saling berhubungan dan masing-masing proses diletakkan pada *swimline* yang berbeda.

Proses pendataan akun berhubungan dengan sistem informasi keuangan dimana data yang akan disimpan dari sistem informasi keuangan dan sistem informasi akuntansi berada dalam *database* akuntansi. Untuk pembuatan jurnal dibutuhkan data akun dan data transaksi yang diperoleh dari bagian keuangan. Setelah dibuat jurnal keuangan, maka dapat dibuat buku besar untuk selanjutnya masuk kepada proses pembuatan neraca. Masing-masing proses digambarkan dengan simbol *Task*. Data yang diperlukan diperoleh dari ESB dan diterima oleh *task receiver* yang berisi *object array*. Untuk

mengambil, *input* data, maupun mengubah data yang terdapat dalam *database* maka harus melewati ESB terlebih dahulu sehingga fungsi tersebut harus berupa *service* yang digambarkan dengan simbol *Task Send*.



Gambar 3.4 Pemodelan SI Akuntansi dengan Sistem Lainnya

4. Desain *Output*.

Desain *output* dari Sistem Akuntansi sebagai berikut.

a. Daftar Akun.

Desain *output* dari halaman daftar akun dapat dilihat pada Gambar 3.5.

The screenshot shows a web browser window titled 'SI Akuntansi' with the URL 'http://SIA.com'. The page layout includes a left sidebar with a home icon and 'SIA' text, a 'Welcome, Admin' message with a user icon, and a navigation menu with 'Beranda', 'Daftar Akun', and 'Laporan'. The main content area features a 'Tambah Akun' section with two input fields for 'No. Akun' and 'Nama Akun', and a 'Tambah' button. Below this is a table titled 'Tabel Akun' with three columns: 'No.', 'No. Akun', and 'Nama Akun'. The table is currently empty.

Gambar 3.5 Desain *Output* Daftar Akun

b. Jurnal.

Desain *output* dari halaman jurnal dapat dilihat pada Gambar 3.6.

The screenshot shows a web browser window titled 'SI Akuntansi' with the URL 'http://SIA.com'. The page layout is similar to the previous one, with a left sidebar and a main content area. The main content area features a 'Cari Jurnal' section with two dropdown menus for 'Periode' and 'Departemen', and a 'Cari' button. Below this is a table titled 'Tabel Akun' with five columns: 'Tanggal', 'No. Akun', 'Nama Akun', 'Debet', and 'Kredit'. The table is currently empty.

Gambar 3.6 Desain *Output* Jurnal

c. Buku Besar.

Desain *output* dari halaman buku besar dapat dilihat pada Gambar 3.7.

The screenshot shows a web browser window titled "SI Akuntansi" with the URL "http://SIA.com". The page layout includes a left sidebar with navigation options: "SIA", "Welcome, Admin", "Beranda", "Daftar Akun", and "Laporan". The main content area is titled "Cari Buku Besar" and contains three dropdown menus labeled "Akun", "Periode", and "Departemen", each with a "Pilih" button. Below these is a "Cari" button. The "Buku Besar" section displays a table with the following structure:

| Tanggal | Keterangan | Debet | Kredit |
|---------|------------|-------|--------|
| | | | |
| | | | |
| | | | |

Gambar 3.7 Desain *Output* Buku Besar

d. Neraca.

Desain *output* dari halaman neraca dapat dilihat pada Gambar 3.8 berikut.

The screenshot shows a web browser window titled "SI Akuntansi" with the URL "http://SIA.com". The page layout is similar to the previous one, with a left sidebar and a main content area. The "Cari Neraca" section contains two dropdown menus labeled "Periode" and "Departemen", each with a "Pilih" button, and a "Cari" button. The "Neraca" section displays a table with the following structure:

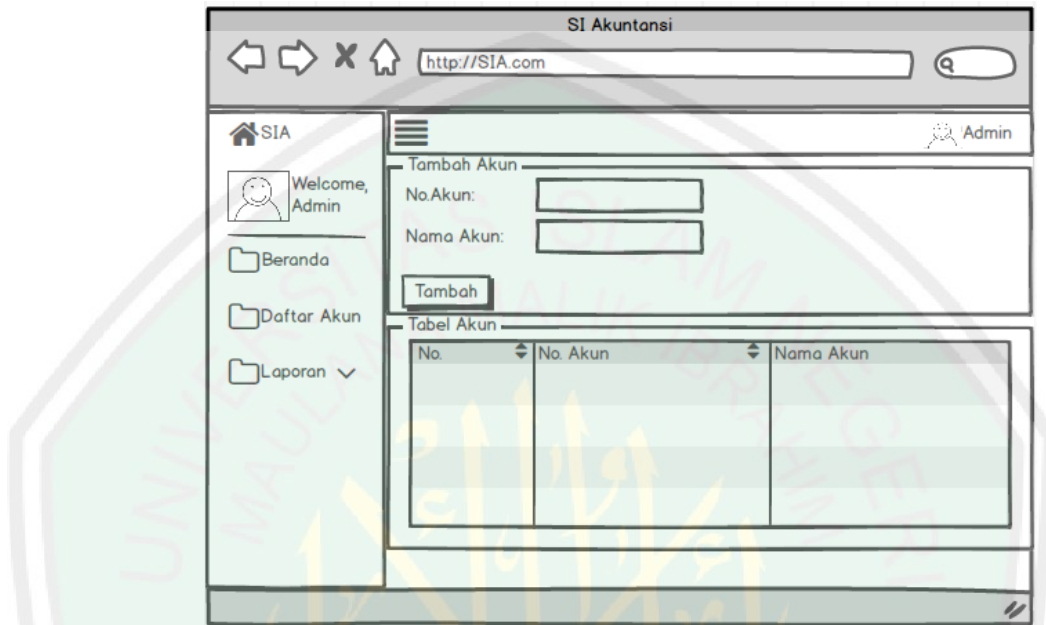
| No.Akun | Nama Akun | Debet | Kredit |
|---------|-----------|-------|--------|
| | | | |
| | | | |
| | | | |

Gambar 3.8 Desain *Output* Neraca

5. Desain *Input*.

a. *Input* Akun Baru.

Desain *input* halaman *input* akun baru dapat dilihat pada Gambar 3.9.



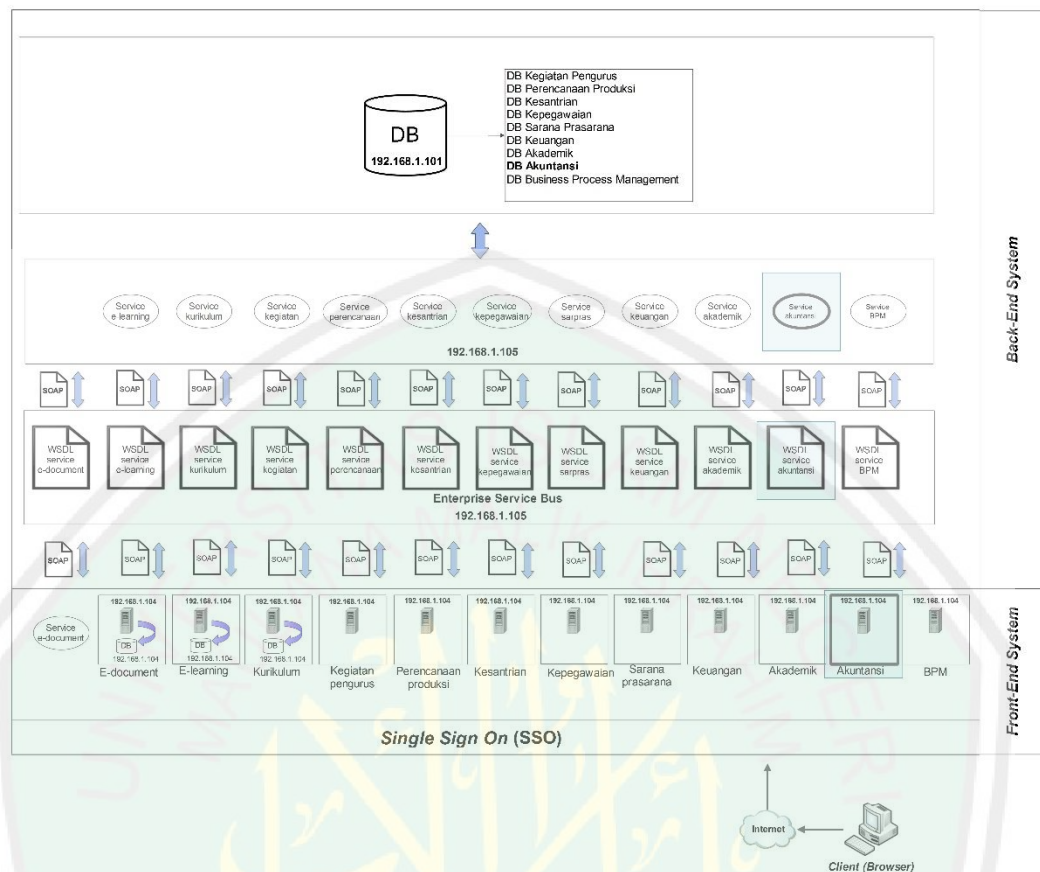
Gambar 3.9 Desain *Input* Akun Baru

3.2.2 Penerapan SOA

Sistem informasi akuntansi merupakan salah satu bagian dari Sistem besar ERP pesantren. Dalam pengembangannya sistem informasi akuntansi akan terintegrasi dengan beberapa sistem lainnya yaitu: Sistem Informasi Kegiatan Pengurus, Sistem Informasi Perencanaan Produksi, Sistem Informasi Kesantrian, Sistem Informasi Kepegawaian, Sistem Informasi Sarana Prasarana, Sistem Informasi Keuangan, Sistem Informasi Akademik, dan Sistem Informasi *Business Process Management*. Integrasi tersebut akan menggunakan arsitektur berorientasi *service*. Setiap fungsi yang ada dibentuk dalam *service-service* yang akan didaftarkan pada ESB WSO2. *Service-service* terdaftar tersebut akan

menghasilkan file WSDL yang bisa diakses oleh *requester*. Arsitektur SOA pada Sistem besar ERP Pondok Pesantren dapat dilihat pada Gambar 3.10.

Pada Gambar 3.10 digambarkan basis data dari sistem besar pondok pesantren diletakkan pada IP Local: 192.168.1.101 yang akan diakses oleh sistem informasi yang tergabung dalam ERP Pondok Pesantren. Terkecuali beberapa sistem informasi seperti sistem informasi e-learning, sistem informasi *e-document*, dan sistem informasi kurikulum *database* melekat pada aplikasinya dan diletakkan bersamaan di IP Local: 192.168.1.104 karena sistem informasi tersebut merupakan *legacy* dari penelitian sebelumnya. Tiap fungsi dari masing-masing sistem informasi dijadikan sebuah *service* yang akan didaftarkan pad ESB WSO2 dan diletakkan pada IP Local: 192.168.1.105. *Service-service* tersebut akan diakses oleh masing-masing aplikasi yang terdapat pada IP Local:192.168.1.104 termasuk di dalamnya aplikasi akuntansi pondok pesantren.



Gambar 3.10 Arsitektur SOA

Dengan arsitektur seperti yang digambarkan pada Gambar 3.10 maka dapat memudahkan pengembangan dan pemeliharaan sistem. Karena pada umumnya ketika diadakan perbaikan sistem yang dirubah adalah proses bisnisnya, akan tetapi fungsi-fungsi yang dijalankan tetap sama.

3.2.3 Pemodelan Service

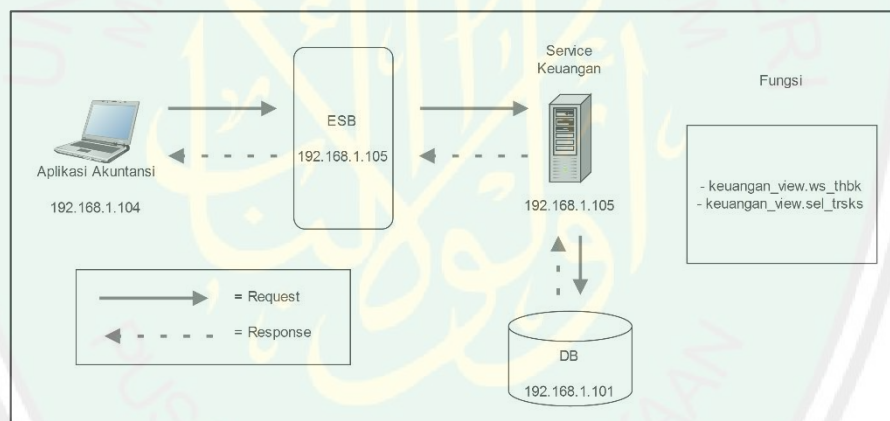
Sistem Informasi Akuntansi hanya terhubung dengan Sistem Informasi Keuangan dan BPM. Data yang akan diminta oleh bagian akuntansi dapat dilihat pada Tabel 3.14. Untuk data akun dapat diinputkan melalui dua sistem yaitu melalui sistem informasi keuangan dan sistem informasi akuntansi untuk kemudian disimpan

dalam *database* akuntansi. Data tersebut kemudian diolah untuk proses-proses yang ada pada akuntansi.

Tabel 3.14 Data Service Akuntansi Keuangan

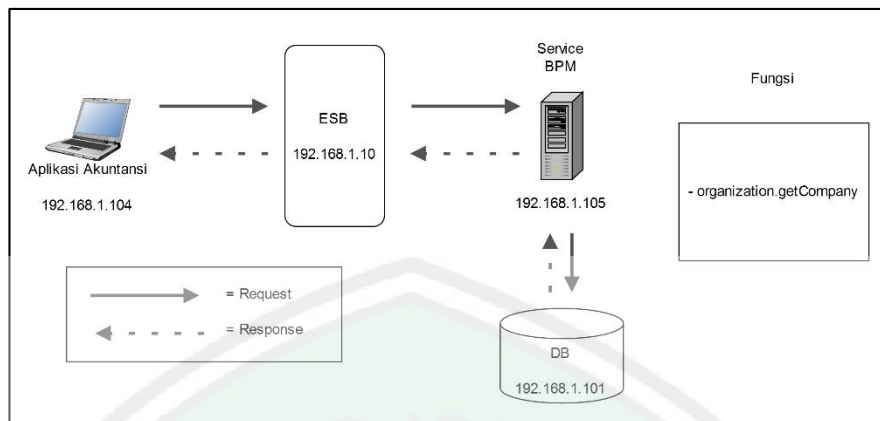
| No | Nama | Penyedia | Pengguna | Deskripsi |
|----|-------------------------|----------------------------|----------------------------|---|
| 1. | keuangan_view.ws_thbk | Sistem Informasi Keuangan | Sistem Informasi Akuntansi | Layanan untuk mendapatkan data akun. |
| 2. | keuangan_view.sel_trks | | | Layanan untuk mendapatkan data transaksi. |
| 3. | organization.getFactory | BPM | Sistem Informasi Akuntansi | Layanan untuk mendapatkan data <i>factory</i> . |
| 4. | akuntansi.getAkun | Sistem Informasi Akuntansi | Sistem Informasi Keuangan | Layanan untuk mendapatkan data akun. |

Pemodelan hubungan antara sistem akuntansi dengan sistem keuangan dapat dilihat pada Gambar 3.11 berikut.



Gambar 3.11 Model Integrasi Sistem Informasi Akuntansi ke *Service* Keuangan

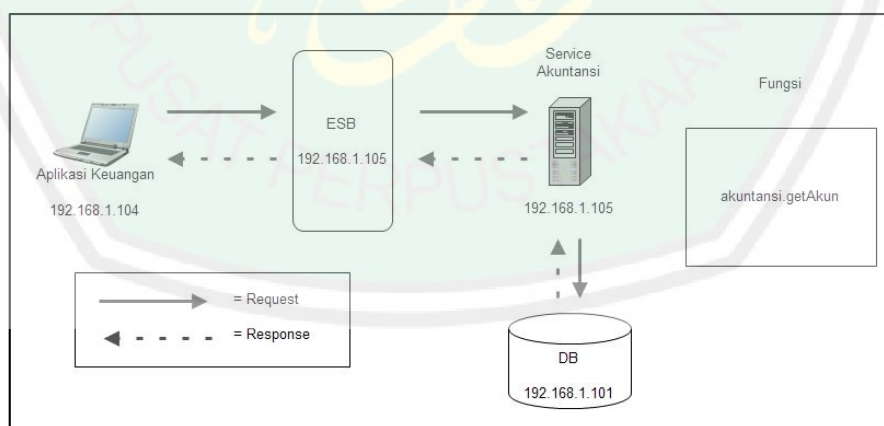
Pemodelan hubungan antara sistem akuntansi dengan sistem BPM dapat dilihat pada Gambar 3.12 berikut.



Gambar 3.12 Model Integrasi Sistem Informasi Akuntansi ke *Service* BPM

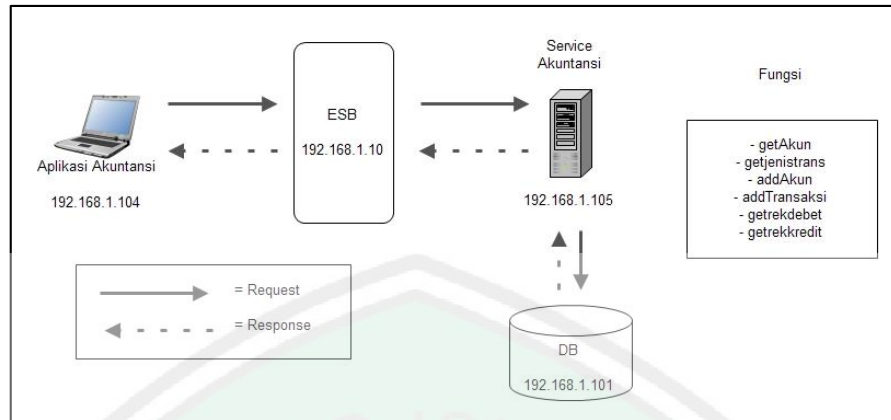
Pada Gambar 3.12 dan Gambar 3.12 bagian keuangan menjadi *provider* dan bagian akuntansi menjadi *requester*. Bagian akuntansi melakukan *request* ke bagian keuangan yang akan ditangani oleh ESB yang berisi file WSDL. Dari ESB tersebut akan dihasilkan URL protokol yang akan diakses oleh *requester* untuk mengambil data dari *provider*.

Selain itu sistem informasi keuangan juga dapat mengambil data akun yang ada pada akuntansi. Proses tersebut dapat dilihat pada Gambar 3.13 berikut.



Gambar 3.13 Model Integrasi Sistem Informasi Keuangan ke *Service* Akuntansi

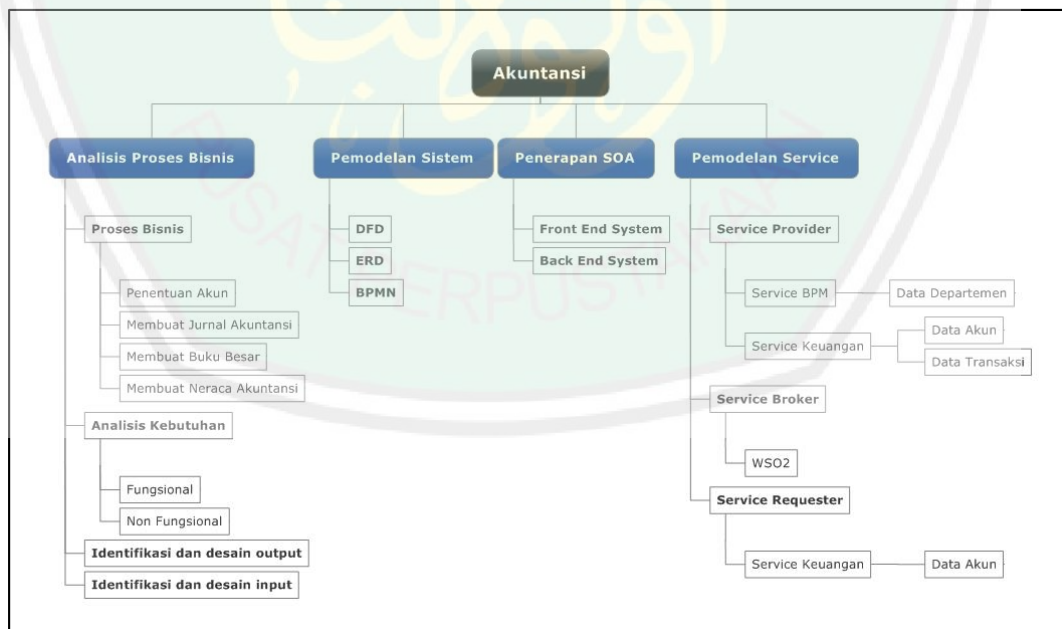
Untuk mengambil data milik akuntansi sendiri, sistem juga membuat layanan yang akan dialirkan melalui ESB. Proses pengambilan data tersebut dapat dilihat pada Gambar 3.14 berikut.



Gambar 3.14 Pengambilan Data SI Akuntansi Melalui ESB

3.3 Prosedur Penelitian

Prosedur penelitian digambarkan dalam bentuk *mindmap* seperti pada Gambar 3.15. yang menggambarkan proses yang dikerjakan pada penelitian ini. Berikut merupakan rincian prosedur penelitian dari sistem informasi akuntansi yang akan dibangun.



Gambar 3. 15 Mindmap Penelitian

Berikut merupakan prosedur penelitian dari sistem informasi akuntansi terintegrasi dengan ERP yang akan dibangun.

1. Analisis Proses Bisnis.

Dalam analisis proses bisnis akan dipaparkan identifikasi proses bisnis yang ada beserta analisisnya. Selain proses bisnis dipaparkan juga identifikasi dan analisis kebutuhan, identifikasi dan desain *output*, serta identifikasi dan desain *input*.

2. Pemodelan Sistem.

Pemodelan sistem berisi pemodelan sistem informasi akuntansi serta pemodelan sistem informasi akuntansi dengan sistem lainnya. Pemodelan sistem ini meliputi desain arsitektur *website*, *Context Diagram* (CD), *Data Flow Diagram* (DFD), dan *Entity Relational Diagram* (ERD) yang berkenaan dengan desain *database*.

3. Pemodelan SOA.

Pemodelan SOA berisi pemodelan *Service Oriented Architecture* dalam yang akan diterapkan pada sistem besar pondok pesantren menggunakan konsep ERP. Termasuk di dalamnya Sistem Informasi BPM, Sistem Informasi Kegiatan Pengurus, Sistem Informasi Akademik, Sistem Informasi Kurikulum, Sistem Informasi Kesantrian, Sistem Informasi Kepegawaian, Sistem Informasi Sarana Prasarana, Sistem Informasi Perencanaan Produksi, Sistem Informasi Keuangan, Sistem Informasi Akuntansi, Sistem Informasi *E-learning*, Sistem Informasi *E-document*, dan Sistem Informasi Pengadaan.

4. Pemodelan *Service*.

Pemodelan *service* berisi pemodelan keterkaitan antara sistem informasi akuntansi dengan sistem lainnya yaitu sistem informasi keuangan.

BAB IV

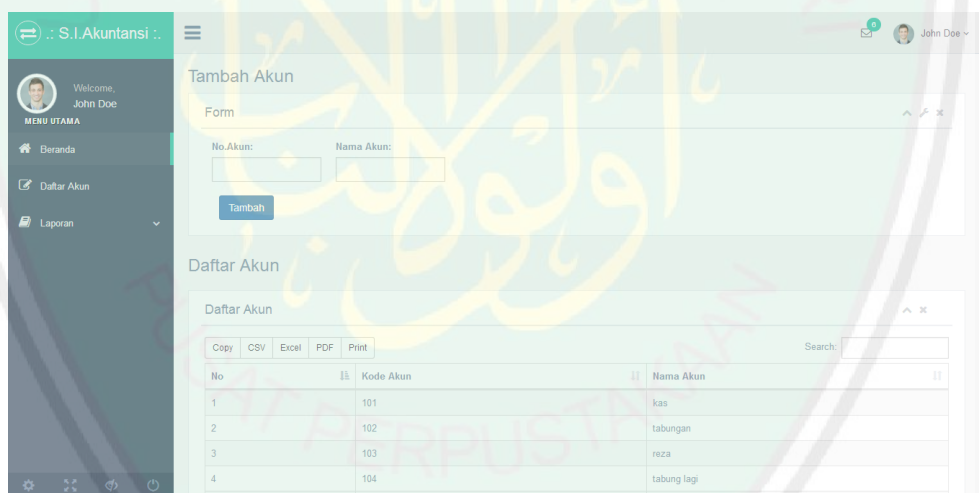
HASIL DAN PEMBAHASAN

4.1 Sistem Informasi Akuntansi

Berikut ini akan dijelaskan implementasi antarmuka sistem informasi akuntansi yang terintegrasi dengan ERP.

1. Daftar Akun.

Halaman daftar akun berisi tabel yang menampilkan data akun yang terdaftar pada database. Pada halaman ini juga user dapat menginputkan akun baru pada field yang telah disediakan. Data yang harus diisi adalah nomor akun dan nama akun. Halaman daftar akun dapat dilihat pada Gambar 4.1.

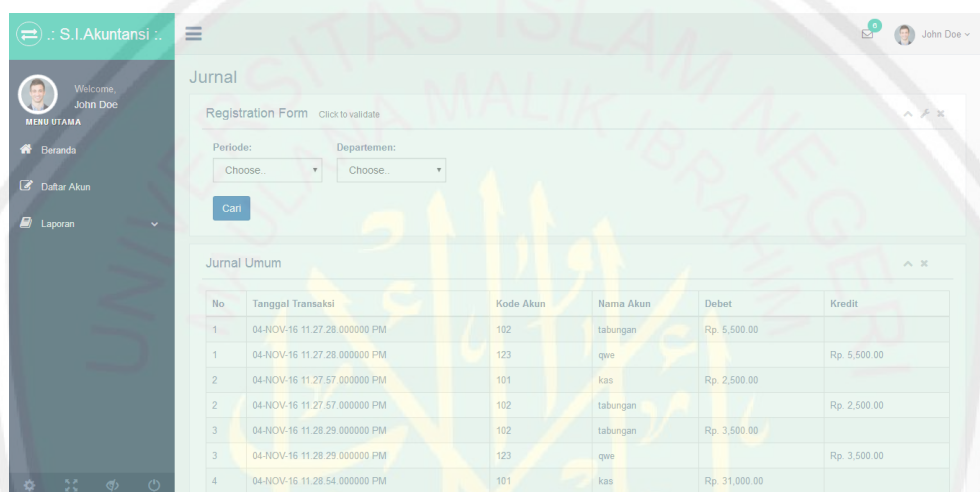


Gambar 4.1 Halaman Daftar Akun

Sistem informasi akuntansi yang dibangun pada penelitian ini memiliki beberapa tahapan sesuai dengan tahapan siklus akuntansi pendidikan pada umumnya. Tahap pertama pada akuntansi yang dibangun adalah penentuan akun. Saat terjadi transaksi, sistem informasi keuangan akan menyertakan daftar akun pada Gambar 4.1 tersebut sehingga user dapat memilih akun yang akan digunakan.

2. Jurnal.

Halaman jurnal menampilkan data jurnal umum yang berisi beberapa informasi terkait transaksi yang telah dilakukan, antara lain informasi tanggal transaksi, kode akun, nama akun, debet, kredit. Halaman jurnal dapat dilihat pada Gambar 4.2. Pada halaman jurnal user dapat melihat data berdasarkan periode dan departemen yang diinginkan oleh user.



Gambar 4.2 Halaman Jurnal

4.2 Konfigurasi Service

4.2.1 Konfigurasi Service dengan nuSoap

Front-end dan *back-end* pada sistem informasi akuntansi pondok pesantren pada penelitian ini dibangun secara terpisah. *Back-end* berisi sekumpulan *service* yang menggunakan *library* nuSOAP. NuSOAP memiliki kemampuan untuk membangun WSDL pada saat *service* digunakan. Dokumen WSDL menyediakan metadata untuk suatu *service*. NuSOAP mengizinkan programmer menentukan WSDL yang dibangun untuk suatu layanan dengan menggunakan beberapa

method dalam *class soap_server*. Untuk membangun WSDL diperlukan untuk mengeksekusi *method* “configureWSDL”.

Aplikasi menyediakan *service* dengan menggunakan WSDL dengan cara mengeksekusi *method* configureWSDL. WSDL yang dikonfigurasi menggunakan dua *parameter input*. Parameter pertama menggambarkan nama *web service*, dan parameter kedua yang bersifat optional yang menggambarkan namespace WSDL, yaitu:

“\$this->namespace”.

Contoh pembuatan *service* dengan menggunakan WSDL dapat dilihat pada Gambar 4.3 berikut.



```

public function __construct($function=false) {
    if ($function) {
        $this->function = $function;
    }
    $this->server = new soap_server();
    $this->model=new model();
    $this->server->soap_defencoding = $this->encoding;
    $this->server->configureWSDL("Web Service ",$this->namespace);
    public function registerFunction($function,$sttus) {
        $this->function = $function;
        $this->setParameters();
        if($sttus=="array"){
            $this->server->wsdl->addComplexType(
                $this->functionarray,
                'complexType',
                'struct',
                'all',
                '',
                $this->outputParams
            );
            $this->server->wsdl->addComplexType(
                $this->functionarray.'Array',
                'complexType',
                'array', '',
                'SOAP-ENC:Array', array(),
                array(
                    array('ref' => 'SOAP-ENC:arrayType', 'wsdl:arrayType' => 'tns:'.$this->functionarray.'[]')
                ),
                'tns:'.$this->functionarray
            );
            $this->server->register(
                'sipp.'.$this->function,
                $this->inputParams,
                array('return' => 'tns:'.$this->functionarray.'Array'),
                $this->namespace,
                $this->namespace."#".$this->function, // soapaction
                'rpc', // style
                'encoded', // use
                'Fetch array '
            );
        }else if($sttus=="string"){
            $this->server->register(
                'sipp.'.$this->function,
                $this->inputParams,
                $this->outputParams,
                $this->namespace
            );
        }
        return $this;
    }
}

```

Gambar 4.3 Registrasi *Service*

Keterangan dari detail registrasi *service* tersebut dapat dilihat pada Tabel 4.1 berikut.

Tabel 4.1 Detail Registrasi *Service*

| Fungsi | Keterangan |
|---|--|
| Name | Nama method <i>service</i> yang disediakan |
| <i>InputParam</i> | Nilai <i>input</i> berupa array asosiatif (param name => param type) |
| <i>outputParams</i> | Nilai output berupa array asosiatif (param name => type) |
| <i>\$this->namespace</i> | Informasi namespace pada <i>service</i> yang disediakan |
| <i>\$this->namespace."#".\$this->function</i> | Informasi soap action pada <i>service</i> yang disediakan |
| <i>rpc</i> | Optional style atau bernilai false |
| Use | Optional use (decoded literal) atau bernilai false |
| encoded | Optional deskripsi dokumentasi WSDL |

| | |
|-------------|-------------------------|
| Fetch array | Optional style encoding |
|-------------|-------------------------|

Detail semua *service* yang disediakan oleh aplikasi server di atas, dilakukan dengan cara mengetikkan pada *browser web* URL yang mengacu pada file aplikasi *server* tersebut. Hasil konfigurasi *service* diatas adalah tampilan method-method pada WSDL. Dengan mengklik pada tulisan WSDL maka akan keluar *format* WSDL dari konfigurasi yang telah dibuat. Contoh tampilan WSDL yang telah dibuat dapat dilihat pada Gambar 4.4 berikut ini.

```

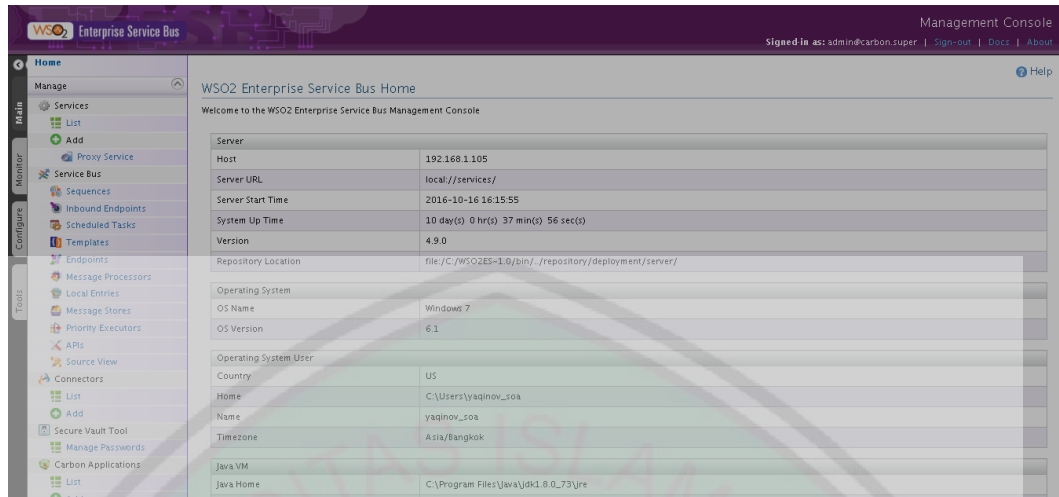
▼ <message name="akuntansi.addAkunRequest">
  <part name="dbname" type="xsd:string"/>
  <part name="kode" type="xsd:int"/>
  <part name="nama" type="xsd:string"/>
</message>
▼ <message name="akuntansi.addAkunResponse">
  <part name="return" type="xsd:string"/>
</message>

```

Gambar 4.4 WSDL *Service* Tambah Akun

4.2.2 Konfigurasi *Service* pada ESB

ESB merupakan *broker* yang menjembatani komunikasi dari *service* satu ke *service* lainnya. Pada dasarnya ESB berfungsi mengganti protokol yang telah kita buat seperti “http://localhost:8181/progres_ws/ws/_akun_aziz//akun.php?wsdl” diganti dengan “<http://yaqinov-pc:8280/services/akun>”. Berikut ini adalah beberapa gambar yang menjelaskan mengkonfigurasi *web service* yang dibuat dengan ESB-WSO2.



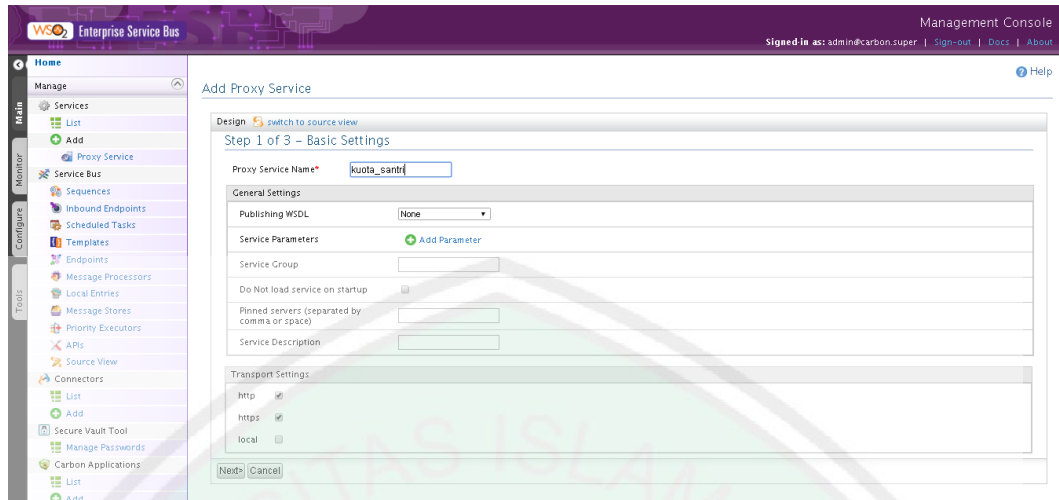
Gambar 4.5 Konfigurasi ESB Langkah ke-1

Pada menu yang ditunjukkan pada Gambar 4.5 diatas, pilih menu *Add* -> *Proxy Service*. Setelah itu akan muncul halaman min seperti pada Gambar 4.6 berikut.



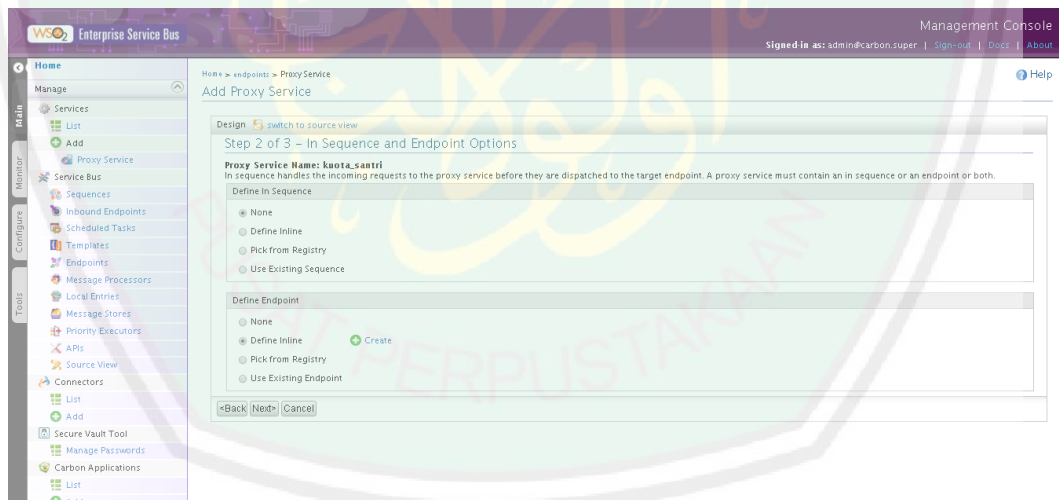
Gambar 4.6 Konfigurasi ESB Langkah ke-2

Pada menu yang ditunjukkan pada Gambar 4.6, pilih menu *Custom Proxy*. Setelah itu akan muncul halaman seperti yang tertera pada Gambar 4.7.



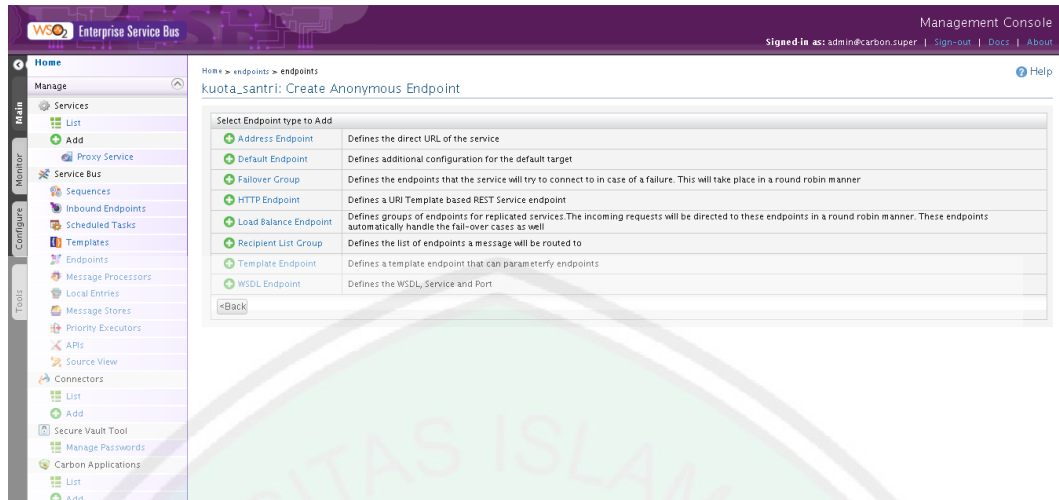
Gambar 4.7 Konfigurasi ESB Langkah ke-3

Pada halaman seperti yang ditunjukkan pada Gambar 4.7 terdapat *field* untuk mengisi *Proxy Service Name* yang harus diisi nama *service* yang akan dibuat. Setelah mengisi nama *service* maka klik tombol *Next*.



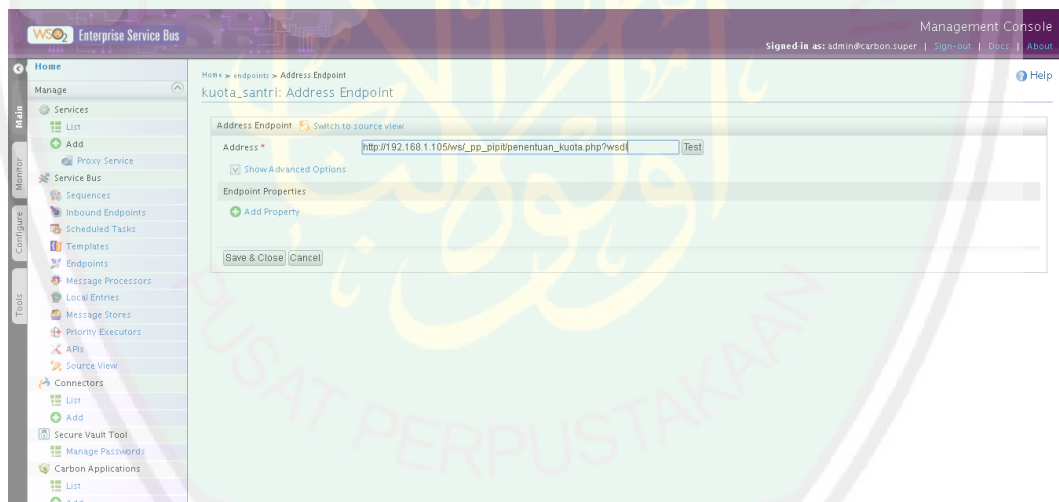
Gambar 4.8 Konfigurasi ESB Langkah ke-4

Setelah klik *Next* maka akan muncul halaman seperti pada Gambar 4.8. Pilih menu *Define Inline* pada *Define Endpoint*, kemudian klik tombol *Create* sehingga muncul halaman seperti pada Gambar 4.9.



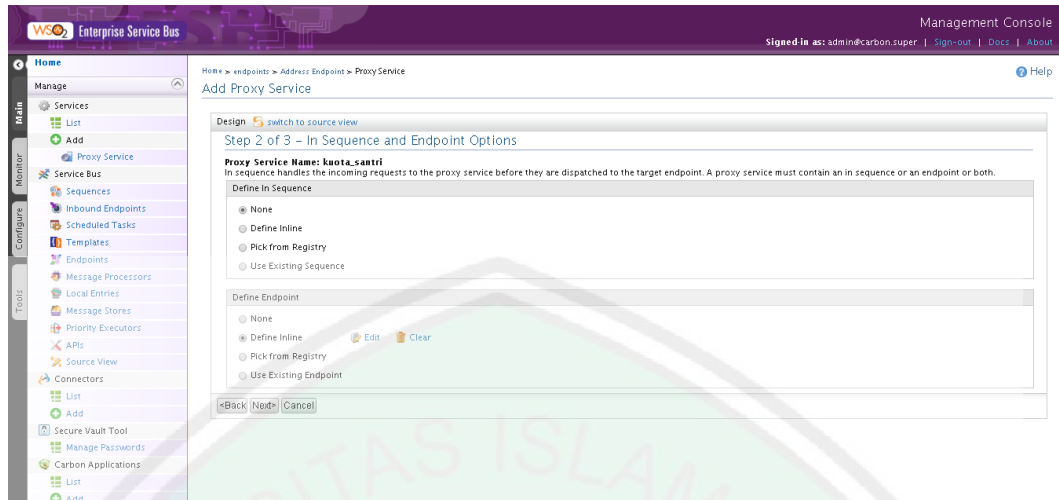
Gambar 4.9 Konfigurasi ESB Langkah ke-5

Pada halaman tersebut pilih menu *Address Endpoint* sehingga muncul halaman seperti pada Gambar 4.10.



Gambar 4.10 Konfigurasi ESB Langkah ke-6

Pada halaman yang ditunjukkan pada Gambar 4.10 terdapat *field* untuk mengisi *Address WSDL* yang telah dibuat sebelumnya. Setelah diisi kemudian klik tombol *Test* untuk pengecekan. Jika tidak terdapat pesan *error* maka dapat dilanjutkan untuk proses selanjutnya.



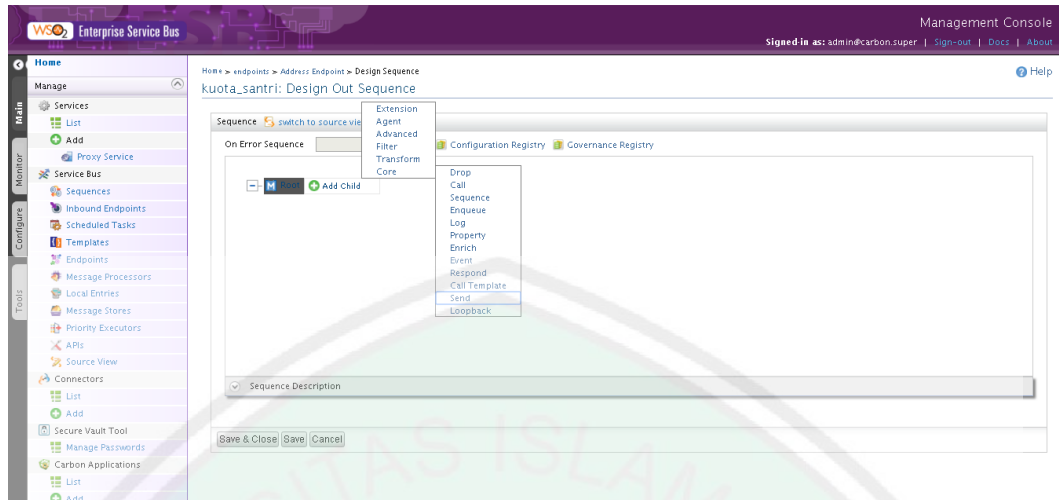
Gambar 4.11 Konfigurasi ESB Langkah ke-7

Setelah langkah yang sebelumnya disimpan maka akan tampilan halaman seperti pada Gambar 4.11. Pilih menu *Define Inline* sehingga muncul halaman seperti pada Gambar 4.12.



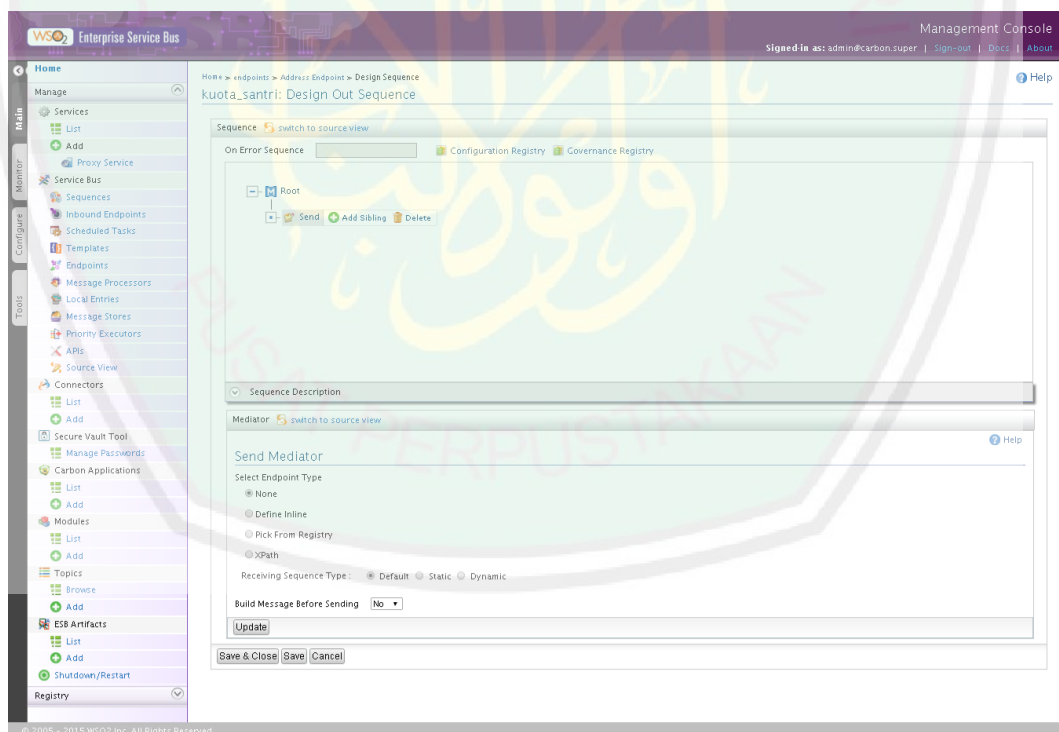
Gambar 4.12 Konfigurasi ESB Langkah ke-8

Pada halaman seperti yang telah ditunjukkan oleh Gambar 4.12 klik tombol klik kemudian pilih *Core* dan kemudian pilih *Send* seperti yang ditunjukkan pada Gambar 4.13.



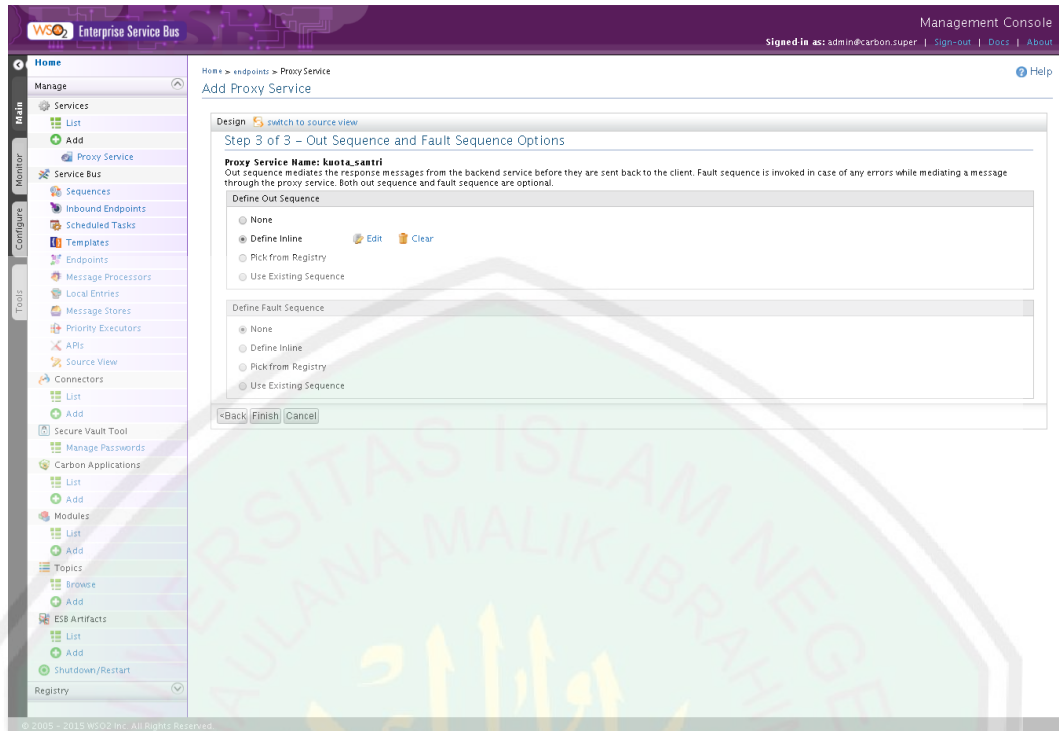
Gambar 4.13 Konfigurasi ESB Langkah ke-9

Setelah proses seperti yang ditunjukkan pada Gambar 4.13 selesai maka akan muncul halaman seperti yang ditunjukkan pada Gambar 4.14.



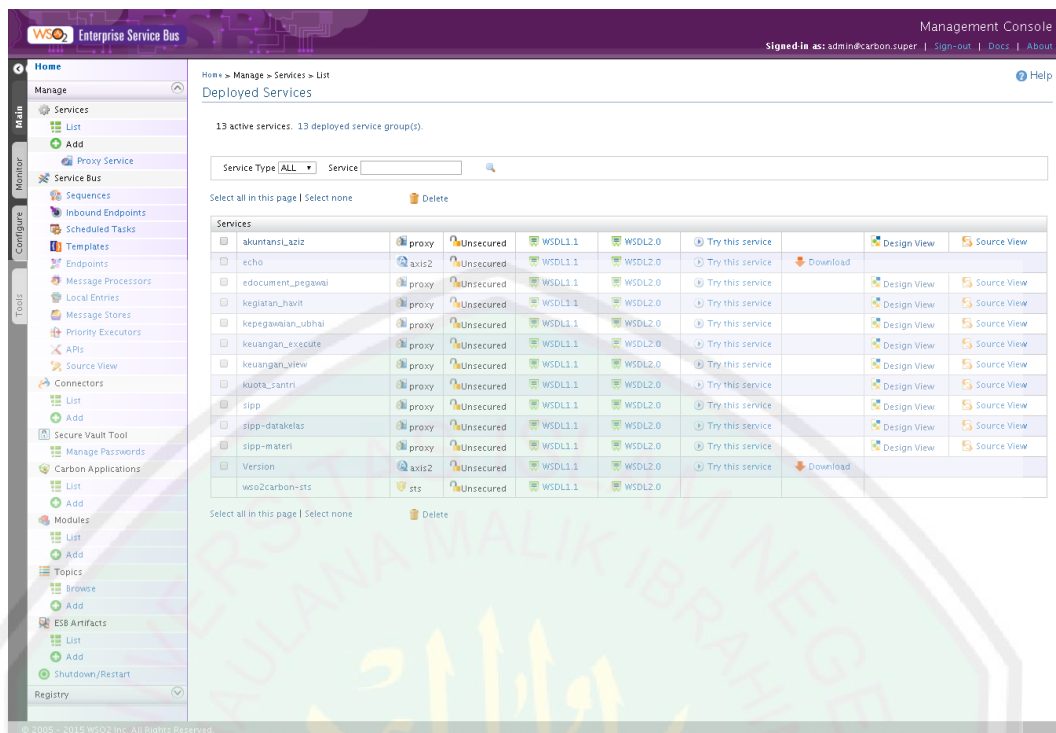
Gambar 4.14 Konfigurasi ESB Langkah ke-10

Setelah itu klik tombol *Save and Close* maka akan muncul halaman seperti pada Gambar 4.15.



Gambar 4.15 Konfigurasi ESB Langkah ke-11

Halaman seperti pada Gambar 4.15 merupakan proses terakhir untuk registrasi ESB. Klik tombol *Finish* kemudian akan muncul halaman seperti pada Gambar 4.16.

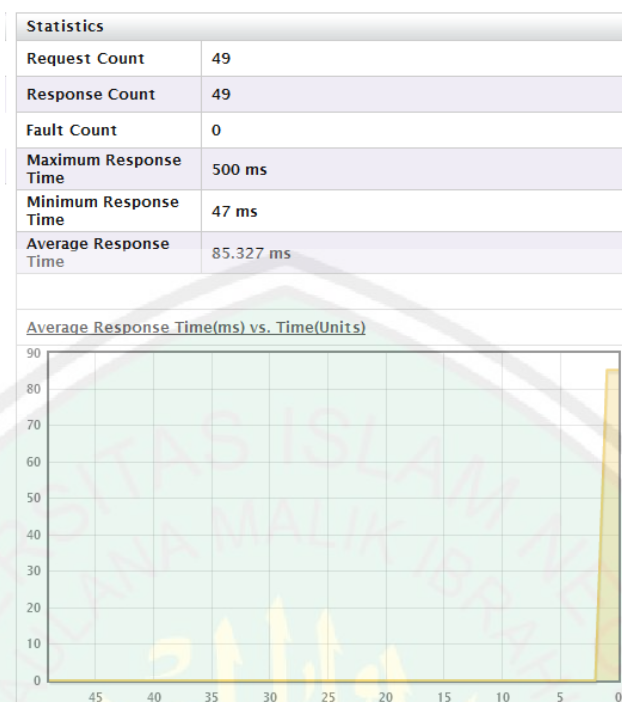


Gambar 4.16 Konfigurasi ESB Langkah ke-12

Dapat dilihat pada Gambar 4.16 bahwa *service* yang telah dibuat telah terdaftar pada ESB WSO2 sehingga dapat diakses oleh *requester*.

4.2.3 Pengujian ESB

ESB WSO2 menyediakan fasilitas pengujian proses komunikasi antar data dari *provider* ke *requester*. Pengujian pada ESB WSO2 dapat diketahui dari statistik *request client* pada *server*. Statistik *request client* dapat dilihat pada Gambar 4.17 berikut.



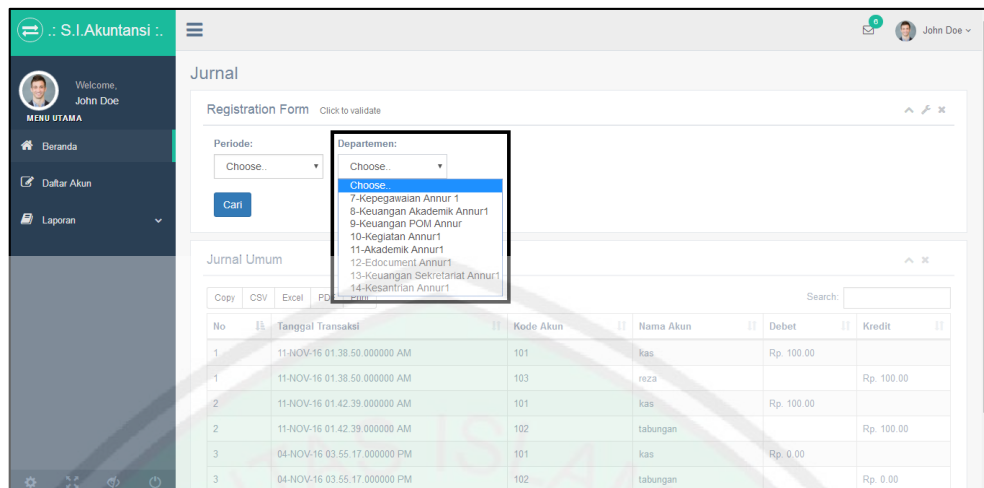
Gambar 4.17 Pengujian ESB

Pada *dashboard* WSO2 ESB, terdapat statistik yang merupakan salah satu bukti aksi dari setiap ada *request* dari *client*. Di dalam data statistik tersebut terdapat beberapa komponen seperti *request count*, *response count*, *fault count*, *maximum response time*, *minimum response time*, dan *average response time*.

4.3 Komunikasi Antar Service

1. Akuntansi dengan BPM.

Dari sistem informasi BPM, bagian akuntansi mengambil departemen departemen yang terdaftar pada masing-masing enterprise. Hasil *request* dari pada BPM akan ditampilkan di halaman jurnal pada combobox departemen seperti yang terlihat pada Gambar 4.18.



Gambar 4.18 Tampilan *Request* Departemen

Hasil pada kotak hitam yang terdapat pada Gambar 4.18 didapatkan dari proses *request* pada *service* BPM dengan sintaksis *call* menggunakan *library* nuSOAP yang kemudian dilewatkan ke ESB. Oleh ESB diarahkan ke alamat *service provider* sesuai permintaan *client*. Setelah direspons oleh *service provider*, dikembalikan lagi ke ESB untuk diarahkan pada *client*. Pesan yang diterima oleh *client* akan diolah sedemikian rupa untuk memperoleh hasil yang baik untuk ditampilkan. Perjalanan sintaksisnya akan digambarkan pada Gambar 4.19.

```
require_once('../vendors/nusoap-master/lib/nusoap.php');
$client = new nusoap_client('http://yaqinov_soa-PC:8280/services/superadmin_company', true);
$status=$client->setCredentials("markamah", "markamahsuperadmin","basic");
$resultc = $client->call('organization.getCompany',array('id_enterprise'=>"101",'dbname'=>"organiz"));
```

Gambar 4.19 *Request* Departemen

Pada Gambar 4.19 $\$client$ menunjukkan alamat *service* tujuan pada ESB sedangkan perintah *setCredential* menunjukkan *credential* dari *service* tujuan untuk mengaksesnya. Perintah *call* berisi *class* dan *method* yang akan diakses pada *service* tujuan. Parameter yang diminta oleh *method* untuk diberikan nilai kembalian ditunjukkan oleh isi dari array.

```

<xsd:complexType name="company">
  <xsd:all>
    <xsd:element name="id_company" type="xsd:int"/>
    <xsd:element name="nm_company" type="xsd:string"/>
    <xsd:element name="almt_company" type="xsd:string"/>
    <xsd:element name="kelurahan_company" type="xsd:string"/>
    <xsd:element name="kecamatan_company" type="xsd:string"/>
    <xsd:element name="kota_company" type="xsd:string"/>
    <xsd:element name="provinsi_company" type="xsd:string"/>
    <xsd:element name="email_company" type="xsd:string"/>
    <xsd:element name="tlp_company" type="xsd:string"/>
    <xsd:element name="tgl_berdiricompany" type="xsd:date"/>
    <xsd:element name="izin_company" type="xsd:string"/>
    <xsd:element name="visi_company" type="xsd:string"/>
    <xsd:element name="misi_company" type="xsd:string"/>
    <xsd:element name="logo" type="xsd:string"/>
    <xsd:element name="typelogo" type="xsd:string"/>
  </xsd:all>
</xsd:complexType>
<xsd:complexType name="companyArray">
  <xsd:complexContent>
    <xsd:restriction base="SOAP-ENC:Array">
      <xsd:attribute ref="SOAP-ENC:arrayType" wsdl:arrayType="tns:company[]"/>
    </xsd:restriction>
  </xsd:complexContent>
</xsd:complexType>

```

Gambar 4.20 WSDL Departemen

Gambar 4.20 merupakan WSDL dari *service* yang disediakan oleh provider dan akan diakses oleh bagian akuntansi. Data yang diperlukan oleh bagian akuntansi dari *service* tersebut antara lain data `id_company` dan `nm_company` yang merupakan nama dari *company*.

2. Akuntansi dengan Keuangan.

Pada aplikasi akuntansi, user akan diminta untuk melakukan pencarian data transaksi yang diinginkan. Gambar 4.21 menunjukkan *form* yang akan digunakan user untuk melakukan pencarian.

The screenshot shows a web application window titled "Jurnal". Inside the window, there is a "Registration Form" with a "Click to validate" link and a close button. Below the form title, there are two dropdown menus: "Periode:" and "Departemen:". Both dropdown menus have "Choose..." as the selected option. Below the dropdown menus is a blue button labeled "Can".

Gambar 4.21 Tampilan *Form* pencarian

Setelah menentukan variabel untuk pencarian, sistem informasi akuntansi mengambil data dari bagian keuangan sesuai dengan yang dicari oleh user. Gambar 4.22 menunjukkan data transaksi yang akan di-request oleh aplikasi akuntansi.

Laporan Transaksi Keuangan Tahun Anggaran 2016/2017

Data Laporan Transaksi Keuangan Tahun Anggaran 2016/2017

Copy CSV Excel Search:

| No. | ID Transaksi/ No. Kwitansi | Waktu Transaksi | Pengirim | Penerima | Jenis Kas | Nama Transaksi | Keterangan | Debit | Kredit |
|-----|-------------------------------|---------------------------------|---------------|----------|-------------|-------------------|------------|------------|---------|
| 1 | 126/ 126 | 22-DEC-16 11.39.36.000000 AM | lima | 12650035 | OPERASIONAL | HUTANG | 45 | 6000 | |
| 2 | 127/ 127 | 22-DEC-16 11.39.35.000000 AM | aaa | 12650035 | OPERASIONAL | PIUTANG | 42 | 6000 | |
| 3 | 128/ | 23-DEC-16 12.26.03.000000 AM | 12650035 | eko | OPERASIONAL | empat | empat | | 16 |
| 4 | 129/ | 23-DEC-16 02.13.21.000000 PM | 12650035 | asad | OPERASIONAL | Nuzulul qur'an | 17 | | 6000000 |
| 5 | 130/ | 23-DEC-16 02.18.47.000000 PM | 12650035 | dfgh | OPERASIONAL | Nuzulul qur'an | 16 | | 6000000 |
| 6 | 131/ | 23-DEC-16 02.36.01.000000 PM | 12650035 | df | OPERASIONAL | Pengajian akhbar2 | 2 | | 6000000 |
| 7 | 132/ | 23-DEC-16 02.44.27.000000 PM | 12650035 | df | OPERASIONAL | dua | 2 | | 45 |
| 8 | 133/ | 23-DEC-16 03.38.22.000000 PM | 12650035 | ASD | OPERASIONAL | dua | 2 | | 45 |
| 9 | 124/ | 21-DEC-16 08.19.01.000000 AM | TAHUN BUKU | 12650035 | OPERASIONAL | KAS AWAL | | 1000000000 | |
| 10 | 125/ 125 | 22-DEC-16 11.28.14.000000 AM | ubai | 12650035 | KEUANGAN | empat | empatt | 16000 | |

Showing 1 to 10 of 11 entries Previous 2 Next

Gambar 4.22 Tampilan Data Transaksi Keuangan

Data transaksi tersebut akan di-request oleh bagian akuntansi melalui WSDL yang telah disediakan oleh sistem informasi keuangan. Gambar 4.23 akan menunjukkan sintaks *request* data transaksi pada *service* keuangan. Sedangkan WSDL keuangan dapat dilihat pada Gambar 4.24.

```
require_once('../vendors/nusoap-master/lib/nusoap.php');
$client = new nusoap_client('http://yaginov_soa-PC:8280/services/keuangan_view', true);
$status=$client->setCredentials("keuangan_om", "financial_om","basic");
$result = $client->call('keuangan_view.sel_trsks_per',array('dbname'=>"annur01101",'tahun_anggaran'=>"$periode", 'id_departemen'=>"$departemen"));
```

Gambar 4.23 Request Transaksi

```

▼<xsd:complexType name="sel_trsk_per">
  ▼<xsd:all>
    <xsd:element name="id_transaksi" type="xsd:int"/>
    <xsd:element name="tanggal_transaksi" type="xsd:string"/>
    <xsd:element name="nama_transaksi" type="xsd:string"/>
    <xsd:element name="jenis_kas" type="xsd:string"/>
    <xsd:element name="harga_satuan" type="xsd:number"/>
    <xsd:element name="jumlah" type="xsd:number"/>
    <xsd:element name="debit" type="xsd:number"/>
    <xsd:element name="kredit" type="xsd:number"/>
    <xsd:element name="no_nota" type="xsd:int"/>
    <xsd:element name="dari_ke" type="xsd:string"/>
    <xsd:element name="keterangan" type="xsd:string"/>
    <xsd:element name="id_anggaran" type="xsd:string"/>
    <xsd:element name="id_buku" type="xsd:string"/>
    <xsd:element name="pengisi_buku" type="xsd:string"/>
    <xsd:element name="id_departemen" type="xsd:string"/>
    <xsd:element name="tahun_anggaran" type="xsd:string"/>
    <xsd:element name="waktu_pembuatan" type="xsd:timestamp"/>
    <xsd:element name="sisahp" type="xsd:number"/>
  </xsd:all>
</xsd:complexType>
▼<xsd:complexType name="sel_trsk_perArray">
  ▼<xsd:complexContent>
    ▼<xsd:restriction base="SOAP-ENC:Array">
      <xsd:attribute ref="SOAP-ENC:arrayType" wsdl:arrayType="tns:sel_trsk_per[]"/>
    </xsd:restriction>
  </xsd:complexContent>
</xsd:complexType>

```

Gambar 4.24 WSDL Transaksi

Dari WSDL yang telah di-*request* tersebut, data yang diambil adalah data `id_transaksi`, `tanggal_transaksi`, `jumlah`, `debit`, dan `kredit` kemudian akan diolah oleh sistem informasi akuntansi dengan source code yang ada pada Gambar 4.25 untuk membentuk jurnal akuntansi.


```

(ws/rq-jenistrans.php");
($i=0;$i<count($result);$i++){
For ($j=0;$j<count($result2);$j++){
if($result[$i]['id_transaksi']==$result2[$j]['id_jenis_transaksi']){
    $debet=$result2[$j]['kode_rekening_debet'];
    include ("ws/rq-debet.php");
    $transaksi=$result[$i]['id_transaksi'];
    $nama_transaksi=$result[$i]['nama_transaksi'];
    $tanggal=$result[$i]['tanggal_transaksi'];
    echo '<tr>';
    echo '<td>'.($i+1).</td>';
    echo '<td>'. $result[$i]['tanggal_transaksi'].</td>';
    echo '<td>'. $result2[$j]['kode_rekening_debet'].</td>';
    echo '<td>'. $nama.</td>';
    if($result[$i]['debit'] != null){
        echo '<td>';
        echo "Rp. " . number_format($result[$i]['debit'], 2, '.', ',');
        echo '</td>';
        $jumlah=$result[$i]['debit'];
        $akun=$result2[$j]['kode_rekening_debet'];
        include ("ws/rq-getjurnal.php");
        //echo count($sjurnal);
        if(count($sjurnal)<2){
            include ("ws/rq-addjurnal.php");
        }
    }
    else{
        echo '<td>';
        echo "Rp. " . number_format($result[$i]['kredit'], 2, '.', ',');
        echo '</td>';
        $jumlah=$result[$i]['kredit'];
        $akun=$result2[$j]['kode_rekening_debet'];
        include ("ws/rq-getjurnal.php");
        if(count($sjurnal)<2){
            include ("ws/rq-addjurnal.php");
        }
    }
}
echo '<td></td>';

```

Gambar 4.25 Source Code Jurnal

Data id_transaksi akan digunakan untuk me-request kode akun yang akan digunakan untuk transaksi tersebut. Gambar 4.26 adalah gambar untuk request kode akun.

```

$client2 = new nusoap_client('http://yaqinov_soa-PC:8280/services/akuntansi_aziz', true);
$status=$client2->setCredentials("aziz", "azizakuntansi","basic");
$result2 = $client2->call('akuntansi.getjenistrans', array('dbname'=>"$dbname"));

```

Gambar 4.26 Request Kode Akun

Request tersebut akan diarahkan pada WSDL yang ada pada Gambar 4.27.

```

<xsd:complexType name="jenistrans">
  <xsd:all>
    <xsd:element name="id_jenis_transaksi" type="xsd:int"/>
    <xsd:element name="nama_transaksi" type="xsd:string"/>
    <xsd:element name="kode_rekening_debet" type="xsd:int"/>
    <xsd:element name="kode_rekening_kredit" type="xsd:int"/>
  </xsd:all>
</xsd:complexType>
<xsd:complexType name="jenistransArray">
  <xsd:complexContent>
    <xsd:restriction base="SOAP-ENC:Array">
      <xsd:attribute ref="SOAP-ENC:arrayType" wsdl:arrayType="tns:jenistrans[]"/>
    </xsd:restriction>
  </xsd:complexContent>
</xsd:complexType>

```

Gambar 4.27 WSDL Kode Akun

Dari WSDL tersebut, data kode_rekening_debet dan kode_rekening_kredit akan diambil sehingga kode akun debit dan kredit untuk masing-masing transaksi akan diketahui. Kemudian, dari kode akun debit dan kredit tersebut, *source code* pada gambar 4.25 akan melakukan *request* seperti pada Gambar 4.28.

```

$client4 = new nusoap_client('http://yaqinov_soa-PC:8280/services/akuntansi_aziz', true);
$status=$client4->setCredentials("aziz", "azizakuntansi", "basic");
$result4 = $client4->call('akuntansi.getrekdebet', array('dbname'=>"$dbname", 'debit'=>"$debit"));

```

Gambar 4.28 Request Nama Akun

Request tersebut akan diarahkan pada WSDL pada Gambar 4.29 untuk mendapatkan nama akun dari kode akun yang ada.

```

<xsd:complexType name="rekdebet">
  <xsd:all>
    <xsd:element name="nama_rekening" type="xsd:string"/>
  </xsd:all>
</xsd:complexType>
<xsd:complexType name="rekdebetArray">
  <xsd:complexContent>
    <xsd:restriction base="SOAP-ENC:Array">
      <xsd:attribute ref="SOAP-ENC:arrayType" wsdl:arrayType="tns:rekdebet[]"/>
    </xsd:restriction>
  </xsd:complexContent>
</xsd:complexType>

```

Gambar 4.29 WSDL Nama Akun

Setelah melakukan *request* tersebut, jurnal yang sudah dibuat akan ditampilkan seperti pada Gambar 4.30.

Journal

Registration Form Click to validate

Periode: 2016/2017 Departemen: 13-Keuangan Se

Cari

Jurnal Umum

| No | Tanggal Transaksi | Kode Akun | Nama Akun | Debet | Kredit |
|----|------------------------------|-----------|-------------|------------------|------------------|
| 1 | 22-DEC-16 11.39.36.000000 AM | 234 | hutang | Rp. 5.000.00 | |
| 1 | 22-DEC-16 11.39.36.000000 AM | 103 | bank | | Rp. 5.000.00 |
| 2 | 22-DEC-16 11.39.35.000000 AM | 101 | kas | Rp. 6.000.00 | |
| 2 | 22-DEC-16 11.39.35.000000 AM | 104 | tabung lagi | | Rp. 6.000.00 |
| 3 | 23-DEC-16 12.26.03.000000 AM | 101 | kas | Rp. 16.00 | |
| 3 | 23-DEC-16 12.26.03.000000 AM | 102 | piutang | | Rp. 16.00 |
| 4 | 23-DEC-16 02.13.21.000000 PM | 234 | hutang | Rp. 5.000.000.00 | |
| 4 | 23-DEC-16 02.13.21.000000 PM | 123 | gedung | | Rp. 5.000.000.00 |
| 5 | 23-DEC-16 02.18.47.000000 PM | 234 | hutang | Rp. 5.000.000.00 | |
| 5 | 23-DEC-16 02.18.47.000000 PM | 123 | gedung | | Rp. 5.000.000.00 |

Showing 1 to 10 of 24 entries

Gambar 4.30 Gambar Jurnal Akuntansi

Dari jurnal diatas, data tanggal transaksi, kode akun, dan nominal transaksi akan disimpan ke dalam database dengan me-request service seperti pada Gambar 4.31.

```
$client3 = new nusoap_client('https://yaqinov_soa-PC:8243/services/akuntansi_aziz', true);
$status=$client3->setCredentials("aziz", "azizakuntansi","basic");
$addj = $client3->call('akuntansi.addjurnal',array('dbname'=>"$dbname",'transaksi'=>"$transaksi",
'akun'=>"$akun",'nama'=>"$nama_transaksi",'jumlah'=>"$jumlah",'tanggal'=>"$tanggal"));
```

Gambar 4.31 Request Tambah Jurnal

Request tambah jurnal tersebut akan diarahkan pada WSDL yang terdapat pada Gambar 4.32.

```
<message name="akuntansi.addjurnalRequest">
  <part name="dbname" type="xsd:string"/>
  <part name="transaksi" type="xsd:int"/>
  <part name="akun" type="xsd:int"/>
  <part name="nama" type="xsd:string"/>
  <part name="jumlah" type="xsd:int"/>
  <part name="tanggal" type="xsd:string"/>
</message>
<message name="akuntansi.addjurnalResponse">
  <part name="return" type="xsd:string"/>
</message>
```

Gambar 4.32 WSDL Tambah jurnal

Setelah berhasil melakukan penambahan jurnal, selanjutnya buku besar untuk periode dan departemen yang sama dengan jurnal yang telah disimpan dapat dibuat. User harus melakukan *input* pada form buku besar seperti pada Gambar 4.33.

Gambar 4.33 *Form* Buku Besar

Sedangkan untuk membuat buku besar, digunakan *source code* pada Gambar 4.34.

```
include ("ws/rq-jurnal.php");
For ($i=0;$i<count($result);$i++){
for ($j=0; $j<count($jurnal); $j++){
if($jurnal[$j]['id_transaksi']==$result[$i]['id_transaksi']){
    $kode=$jurnal[$j]['id_transaksi'];
    echo '<tr>';
    echo '<td>'.($j+1).</td>';
    echo '<td>'. $result[$i]['tanggal_transaksi'].'</td>';
    echo '<td>'. $jurnal[$j]['nama_transaksi'].'</td>';
    include ("ws/rq-jenistrans2.php");
    for($k=0; $k<count($ss); $k++){
        if($ss[$k]['kode_rekening_debet']==$akun ){
            echo '<td>';
            echo "Rp. " . number_format($jurnal[$j]['jumlah'], 2, '.', '');
            echo '</td>';
            $totald+=$jurnal[$j]['jumlah'];
            echo '<td></td>';
        }else{
            echo '<td></td>';
            echo '<td>';
            echo "Rp. " . number_format($jurnal[$j]['jumlah'], 2, '.', '');
            echo '</td>';
            $totalk+=$jurnal[$j]['jumlah'];
        }
    }
    echo '</tr>';
}
}
```

Gambar 4.34 *Source Code* Buku Besar

Source code pada gambar 4.34 akan melakukan request pada WSDL untuk mendapatkan data jurnal yang telah ada pada database. *Request* jurnal

ditampilkan pada Gambar 4.35 dan WSDL yang dituju akan ditampilkan pada Gambar 4.36.

```
$client3 = new nusoap_client('https://yaqinov_soa-PC:8243/services/akuntansi_aziz', true);
$status=$client3->setCredentials("aziz", "azizakuntansi", "basic");
$jurnal = $client3->call('akuntansi.getjurnal2', array('dbname'=>"$dbname", 'akun'=>"$akun"));
```

Gambar 4.35 Request Jurnal

```
▼<xsd:complexType name="jurnal2">
  ▼<xsd:all>
    <xsd:element name="id_transaksi" type="xsd:string"/>
    <xsd:element name="nama_transaksi" type="xsd:string"/>
    <xsd:element name="jumlah" type="xsd:int"/>
  </xsd:all>
</xsd:complexType>
▼<xsd:complexType name="jurnal2Array">
  ▼<xsd:complexContent>
    ▼<xsd:restriction base="SOAP-ENC:Array">
      <xsd:attribute ref="SOAP-ENC:arrayType" wsdl:arrayType="tns:jurnal2[]"/>
    </xsd:restriction>
  </xsd:complexContent>
</xsd:complexType>
```

Gambar 4.36 WSDL Jurnal

Id_transaksi dari wsdl tersebut akan di cocokkan dengan id_transaksi yang didapatkan dari WSDL keuangan pada gambar 4.24 dengan request yang digunakan pada saat pembuatan jurnal seperti pada gambar 4.23. Ketika Id_transaksi pada jurnal dan Id_transaksi dari keuangan cocok, maka akan dilakukan pembuatan buku besar. Buku besar yang telah dibuat seperti pada gambar 4.37.

The screenshot displays the 'Buku Besar' (General Ledger) interface. At the top, there is a 'Registration Form' with the following fields:

- Nama Akun:** kas
- Periode:** 2016/2017
- Departemen:** 13-Kuangan Se

Below the form is a table of transactions:

| No | Tanggal Transaksi | Keterangan | Debet | Kredit |
|----|------------------------------|------------|--------------|--------|
| 1 | 04-NOV-16 03:55:17.000000 PM | KAS AWAL | Rp. 0.00 | |
| 7 | 22-DEC-16 11:28:14.000000 AM | PIUTANG | Rp. 100.00 | |
| 8 | 23-DEC-16 03:38:22.000000 PM | dua | Rp. 45.00 | |
| 11 | 22-DEC-16 11:39:35.000000 AM | HUTANG | Rp. 5,000.00 | |
| 12 | 23-DEC-16 12:26:03.000000 AM | HUTANG | Rp. 1,000.00 | |
| 15 | 21-DEC-16 08:19:01.000000 AM | HUTANG | Rp. 100.00 | |

At the bottom, the 'Total' section shows:

| | | | |
|-------------|--------------|--------------|----------|
| TOTAL DEBIT | Rp. 6,245.00 | TOTAL KREDIT | Rp. 0.00 |
|-------------|--------------|--------------|----------|

Gambar 4.37 Tampilan Buku Besar

Selanjutnya, Total debet dari akun yang telah dipilih oleh user akan di tampilkan pada variabel \$totald sedangkan untuk total kreditnya akan ditampung pada variabel \$totalk. Setelah mendapatkan total debet dan kredit untuk akun tersebut, maka kode akun tersebut beserta total debet, total kredit, periode, dan departemen akan disimpan dengan melakukan *request* seperti pada Gambar 4.38.

```
$client3 = new nusoap_client('https://yaqinov_soa-PC:8243/services/akuntansi_aziz', true);
$status=$client3->setCredentials("aziz", "azizakuntansi", "basic");
$addn = $client3->call('akuntansi.addneraca', array('dbname'=>"$dbname", 'akun'=>"$akun", 'periode'=>"$periode",
'departemen'=>"$departemen", 'debet'=>"$totald", 'kredit'=>"$totalk"));
```

Gambar 4.38 Request Tambah Neraca

Request tersebut akan ditujukan pada WSDL pada Gambar 4.39.

```

<message name="akuntansi.addneracaRequest">
  <part name="dbname" type="xsd:string"/>
  <part name="akun" type="xsd:int"/>
  <part name="periode" type="xsd:string"/>
  <part name="departemen" type="xsd:string"/>
  <part name="debit" type="xsd:int"/>
  <part name="kredit" type="xsd:string"/>
</message>
<message name="akuntansi.addneracaResponse">
  <part name="return" type="xsd:string"/>
</message>

```

Gambar 4.39 WSDL Tambah Neraca

Setelah berhasil menambahkan data dari buku besar tersebut, aplikasi akuntansi ini telah membuat neraca untuk periode dan departemen tersebut. Selanjutnya, neraca akan ditampilkan setelah user melakukan *input* pada form neraca seperti pada Gambar 4.40.

Gambar 4.40 Form Neraca

Setelah user memilih periode dan departemen yang diinginkan, maka *source code* pada Gambar 4.41 akan dijalankan.

```

include ("ws/rq-neraca2.php");
//print_r($neraca);
For ($i=0;$i<count($neraca);$i++){
  echo '<tr>';
  echo '<td>'.($i+1).'/</td>';
  $akun=$neraca[$i]['no_akun'];
  include ("ws/rq-akun2.php");
  echo '<td>'. $result3[0]['nama_rekening'].'/</td>';
  echo '<td>';
  echo "Rp. " . number_format($neraca[$i]['total_debet'], 2, '.', ',');
  $totald+=$neraca[$i]['total_debet'];
  echo '</td>';
  echo '<td>';
  echo "Rp. " . number_format($neraca[$i]['total_kredit'], 2, '.', ',');
  $totalk+=$neraca[$i]['total_kredit'];
  echo '</td>';
  echo '</tr>';
}

```

Gambar 4.41 Source Code Neraca

Source code pada gambar 4.41 tersebut akan melakukan *request* pada service yang akan memberikan data neraca dengan variabel *input* yang berisi data pilihan user. *Request* neraca akan ditampilkan pada Gambar 4.42, sedangkan WSDL yang diakses seperti ditampilkan pada Gambar 4.43.

```
$client3 = new nusoap_client('https://yaginov_soa-PC18243/services/akuntansi_aziz', true);
$status=$client3->setCredentials("aziz", "azizakuntansi","basic");
$neraca = $client3->call('akuntansi.getneraca2',array('dbname'=>"$dbname",'periode'=>"$periode",'departemen'=>"$departemen"));
```

Gambar 4.42 *Request* Neraca

```
▼<xsd:complexType name="neraca">
  ▼<xsd:all>
    <xsd:element name="id_neraca" type="xsd:string"/>
    <xsd:element name="no_akun" type="xsd:string"/>
    <xsd:element name="total_debet" type="xsd:int"/>
    <xsd:element name="total_kredit" type="xsd:int"/>
  </xsd:all>
</xsd:complexType>
▼<xsd:complexType name="neracaArray">
  ▼<xsd:complexContent>
    ▼<xsd:restriction base="SOAP-ENC:Array">
      <xsd:attribute ref="SOAP-ENC:arrayType" wsdl:arrayType="tns:neraca[]"/>
    </xsd:restriction>
  </xsd:complexContent>
</xsd:complexType>
```

Gambar 4.43 *WSDL* Neraca

Setelah *source code* tersebut dijalankan, maka neraca keuangan untuk periode dan departemen yang telah dipilih user akan ditampilkan. Tampilan neraca tersebut seperti pada Gambar 4.44.

The screenshot displays the 'Neraca' (Balance Sheet) report in an accounting system. The interface includes a navigation menu on the left, a registration form at the top, and a main table of accounts. The registration form shows the period '2016/2017' and department '13-Keuangan Se'. The table lists 9 accounts with their respective debit and credit amounts in Indonesian Rupiah (Rp.).

| No | Nama Akun | Debet | Kredit |
|--------------|----------------------------|-------------------|-------------------|
| 1 | kas | Rp. 6,245.00 | Rp. 0.00 |
| 2 | pendapatan non operasional | Rp. 0.00 | Rp. 0.00 |
| 3 | tanah | Rp. 5,000,000.00 | Rp. 45.00 |
| 4 | piutang | Rp. 45.00 | Rp. 1,145.00 |
| 5 | gedung | Rp. 0.00 | Rp. 10,200.00 |
| 6 | hutang | Rp. 27,010,200.00 | Rp. 0.00 |
| 7 | pendapatan | Rp. 0.00 | Rp. 0.00 |
| 8 | bank | Rp. 0.00 | Rp. 27,000,100.00 |
| 9 | tabung lagi | Rp. 0.00 | Rp. 5,005,000.00 |
| Total | | | |
| TOTAL DEBIT | | Rp. 32,016,490.00 | TOTAL KREDIT |
| | | | Rp. 32,016,490.00 |

Gambar 4.44 Tampilan Neraca

Sistem ERP Pondok Pesantren dibangun menggunakan arsitektur data *shared database separate schemas*. Arsitektur ini diterapkan menggunakan DBMS *Oracle XE 11g* yang bisa menerapkan satu *database* untuk beberapa skema. Satu pondok di dalam sistem ini adalah satu *enterprise*, di dalamnya lagi terdapat skema-skema khas milik pondok tersebut. Data dari *enterprise* yang berbeda tidak akan sama karena sudah berbeda *user*, karena saat ada pondok baru yang mendaftar, maka skema akan diberikan tanpa data sama sekali.

Arsitektur data pada sistem informasi yang dibangun menggunakan multitenansi *shared database separate schemas*, maka otomatisasi akun tidak bisa diterapkan pada sistem ini dikarenakan sistem ERP Pondok Pesantren ditujukan untuk berbagai macam pondok.

Selain itu, ketika melakukan otomasi, maka aplikasi yang dibuat harus memberikan fasilitas untuk melakukan *assign* akun pada transaksi yang akan dilakukan oleh pondok yang dituju. Karena itu, dibutuhkan penelitian untuk mengetahui transaksi yang pasti terjadi pada pondok secara umum sehingga dapat dibuatkan daftar transaksi beserta akunya.

Proses bisnis yang digunakan tentu juga berbeda dalam setiap pondok dikarenakan perbedaan penanganan transaksi pondok tersebut. Sebagai contoh, sebuah pondok menerapkan akun yang berbeda untuk pembayaran spp santri setiap tahunnya, sedangkan pondok lain memilih menggunakan akun yang sama. Maka, proses bisnis yang dibutuhkan juga tentu berbeda. Untuk pondok yang memilih menggunakan akun yang berbeda setiap tahun, maka aplikasi akuntansi yang dibuat tentu harus dapat memfasilitasi pembuatan akun setiap tahun untuk akun spp. Setelah pembuatan akun spp tersebut, dibuat kategori transaksi baru secara otomatis untuk pembayaran spp serta proses melakukan *assign* akun baru yang telah dibuat pada transaksi yang tersebut.

Namun, masalah proses bisnis yang berbeda tersebut masih dapat dilakukan dengan implementasi SOA pada aplikasi ini, karena dengan SOA service yang tidak dibutuhkan dapat dilepas dan disambungkan dengan service lain yang dibutuhkan.

Belum adanya proses bisnis yang berbeda dan penelitian pada transaksi umum yang terjadi pada sistem ini menyebabkan tidak berjalannya otomasi akun pada sistem informasi akuntansi ini. Namun, tidak menutup kemungkinan dapat dilakukan dikarenakan arsitektur SOA yang digunakan.

BAB V

PENUTUP

5.1 Kesimpulan

Kesimpulan dari penelitian yang telah dilakukan adalah sebagai berikut:

1. Cara mengintegrasikan sistem informasi akuntansi dengan sistem informasi lainnya yang berbasis ERP adalah (a) Menggunakan *web service* berbasis SOA, (b) Sistem informasi yang dibutuhkan datanya oleh sistem informasi akuntansi membuat layanan *provider* yang kemudian didaftarkan ke ESB, (c) Sistem informasi akuntansi membuat layanan *requester* yang mengakses layanan *provider* yang dibutuhkan pada ESB.
2. Otomasi akun hanya bisa diterapkan pada satu *enterprise* (pondok pesantren) yang memiliki pola tetap pada transaksinya. Sistem Informasi Akuntansi Pondok Pesantren ini belum dapat menerapkan otomasi akun karena adanya perbedaan jenis transaksi pada berbagai *enterprise* (pondok pesantren).

5.2 Saran

Berdasarkan penelitian yang telah dilakukan terdapat beberapa kekurangan. Oleh karena itu diharapkan adanya penelitian lebih lanjut yang meliputi.

1. Perlu adanya pemikiran lebih lanjut atau kecerdasan buatan untuk membuat otomasi akun yang dapat mengenali secara langsung jenis transaksi yang berbeda.
2. Perlu adanya penelitian lebih lanjut tentang pengelolaan proses bisnis dengan menggunakan *Business Process Management Suite (BPMS)* yang berguna

untuk mengkostumisasi proses bisnis, di mana di dalamnya terdapat proses *interface mapping*, *data mapping*, dan *process model mapping*.

3. Perlu adanya penelitian untuk pembuatan kecerdasan dalam pengelolaan proses bisnis yang dimuai dari proses *fragmentasi*, *clustering*, penemuan *common fragment*, *query bussiness process and interface*, dan terakhir *composing*.
4. Diperlukan penelitian tentang penerapan *graph database* untuk *repository bussiness process and data*.



DAFTAR PUSTAKA

- Bastian, I. (2006). *Akuntansi Pendidikan*. Jakarta: Penerbit Erlangga.
- Brittenham, P. (2002). *An overview of the Web Services Inspection Language*. Retrieved Februari 1, 2014, from <http://www.ibm.com/developerworks/library/ws-wslover/>
- Deviana, H. (2011). Penerapan XML Web service Pada Sistem Distribusi Barang. *Jurnal Generic*, 6, 61-70.
- Herlambang, R. W., Sarno, R., & Sunaryono, D. (2013). Implementasi Modul-Modul Enterprise Resource Planning Multi Tenant pada Cloud Computing. *JURNAL TEKNIK POMITS*, 2.
- Putri, N. B., & Widyawati, D. (2013). Penerapan Sistem Informasi Akuntansi Penerimaan Pendapatan pada PT Kereta Api Indonesia Surabaya. *Jurnal Ilmu & Riset Akuntansi*, 2.
- Rama, D. V., & Jones, F. L. (2008). *Sistem Informasi Akuntansi Edisi 18 Buku 1*. Jakarta: Penerbit Salemba Empat.
- Ramdhani, M. A. (2015). Pemodelan Proses Bisnis Sistem Akademik Menggunakan Pendekatan Business Process Modelling Notation (BPMN) (Studi Kasus Institusi Perguruan Tinggi XYZ). *Jurnal Informasi*, VII.
- Rifai, A., Sarno, R., & Sunaryono, D. (2016). Rancang Bangun Sistem Persediaan (Inventory) dengan Model Software As A Service. *ResearchGate*.
- Rosmalal, D., & Falahah. (2007). Pemodelan Proses Bisnis B2B dengan BPMN. *Seminar Nasional Ap/iicasi TelO1ologi Informasi 2007 (SNATI2007)*, 1907-5022.
- Sari, S. K., & Asniar. (n.d.). Analisis dan Pemodelan Proses Bisnis Prosedur Pelaksanaan Proyek Akhir Sebagai Alat Bantu Identifikasi Kebutuhan Sistem. *Jurnal Infotel*, 2085-3688.
- Sarno, R., Putra, A. D., & Sunaryono, D. (2011). Rancang Bangun Orkestrasi Web Service serta Implementasi Single Sign On pada Enterprise Resource Planning. *Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember*.
- Sarosa, S. (2009). *Sistem Informasi Akuntansi*. Jakarta: Grasindo.
- Shofa, R. N., Aradea, & Kurnia, B. B. (2013). Penerapan Service Oriented Architecture (SOA). *Jurnal Penelitian SITROTIKA*, 9, 2.
- Slameto, A. A. (2015). Penerapan Service Oriented Architecture (SOA) dalam Proses Intregasi Sistem Informasi Inventaris Laboratorium dan Sistem Informasi Laporan Kerusakan Komputer pada Laboratorium STMIK AMIKOM. *Jurnal Teknologi Informasi*, X, 1907-2430.

Sunaryono, D., Sarno, R., Hariadi, V., & Kurniawan, Y. (2013). Perancangan dan Pembangunan Perangkat Lunak Berorientasi Arsitektur Servis (SOA) dengan Pendekatan Workflow pada Domain Cash Bank dan General Ledger ERP. *Seminar Nasional Sistem Informasi Indonesia*.

Suyono, H., & Wahyuni, S. (2014). *Jurnal Peranan Pondok Pesantren Dalam Mengatasi Kenakalan Remaja (Studi Kasus Di Pondok Pesantren Al Muayyad Surakarta)*. Retrieved Maret 09 , 2015, from <http://download.portalgaruda.org/article.php?article=151703&val=4059>

Wibisono, S. (2005). Enterprise Resource Planning (ERP) Solusi Sistem Informasi Terintegrasi. *ISSN Jurnal Teknologi Informasi DINAMIK*, X, 150-159.

Yasin, V. (2013). Pentingnya Sistem Enterprise Resource Planning (ERP) dalam Rangka untuk Membangun Sumber Daya pada Suatu Perusahaan. *ISSN*, 4(Manajemen Informatika), 2086 – 1052.



LAMPIRAN
ANGGOTA TIM “SKRIPSI SUKSES”

Skripsi ini adalah bagian dari penelitian besar tentang “ERP Pondok Pesantren” yang dikerjakan oleh beberapa peneliti dalam tim Skripsi Sukses. Berikut penulis lampirkan daftar anggota tim Skripsi Sukses beserta kontribusinya.

| No. | Nama | Bagian Yang Dikerjakan | Kontribusi |
|-----|---------------------------|---|--|
| 1. | Nurika Nadhifatul F. | Sistem Informasi <i>E-document</i> | <i>Project Manager</i> , pengelola data <i>e-document</i> serta pengakses data pembayaran uang gedung. |
| 2. | M. Fajarivan Pratama | Sistem Informasi Keuangan | Penyedia data keuangan |
| 3. | Ahmad Havit Hakim | Sistem Informasi Kegiatan Pengurus | Penyedia data kegiatan. |
| 4. | Dewi Rahmawati | Sistem Informasi Pengadaan | Penyedia data keperluan pengadaan. |
| 5. | Badarudin Syah | Sistem Informasi Kesantrian | Penyedia data santri. |
| 6. | M. Ubaidillah | Sistem Informasi Kepegawaian | Penyedia data pegawai. |
| 7. | M. Eko Suprianto | Sistem Informasi Akademik | Pengakses data pembayaran uang gedung. |
| 8. | Fithrotin Maulidiyah A.F. | Sistem Informasi Perencanaan Produksi | Pengelola data perencanaan produksi. (tidak ada komunikasi data dengan penelitian yang dikerjakan) |
| 9. | Vion Age T. | Sistem Informasi <i>E-learning</i> | Pengelola data <i>e-learning</i> . (tidak ada komunikasi data dengan penelitian yang dikerjakan) |
| 10. | Anni'matul Ma'rifah | Sistem Informasi Kurikulum | Pengelola data kurikulum. (tidak ada komunikasi data dengan penelitian yang dikerjakan) |
| 11. | Aqsari Nufikha Putri | Sistem Informasi Sarana Prasarana | Pengelola data sarana prasarana. (tidak ada komunikasi data dengan penelitian yang dikerjakan) |
| 12. | Abd. Charis Fauzan | Sistem Cerdas Untuk Menemukan Kemiripan Perilaku Pada Model Proses Bisnis Penerimaan Peserta Didik Baru | Pengembang sistem cerdas untuk kemiripan perilaku pada proses bisnis penerimaan peserta didik baru. (tidak ada komunikasi data dengan penelitian yang dikerjakan) |