

# Analisis Dan Perancangan Modul Aplikasi Penerimaan Mahasiswa Baru Berbasiskan *Service-Oriented Architecture* (SOA)

Toninetti

**Abstrak**— *Service Oriented Architecture* (SOA) merupakan sebuah representasi model baru untuk membangun aplikasi yang terdistribusi dan memodularisasi sistem informasi menjadi *services*. Tujuan penelitian ini adalah untuk menguji coba sebuah solusi kombinasi tahapan baru untuk mengatasi masalah proses pengembangan sistem informasi yang lambat dengan menggunakan modul aplikasi Penerimaan Mahasiswa Baru (PMB) sebagai model. Penelitian ini berisi masalah yang ada dalam pengembangan sebuah sistem informasi, seperti lambatnya pemenuhan terhadap perubahan yang diminta oleh pihak pengguna terhadap sebuah sistem informasi. Solusi dari permasalahan yang ada adalah menggunakan *Service Oriented Architecture* (SOA) sebagai metodologi baru untuk keseluruhan proses pengembangan sistem informasi mulai dari pengumpulan *User Requirements* sampai dengan *Maintenance*. Pada bagian tahap *development* di SOA, *Agile Software Development* merupakan solusi yang tepat sebagai panduan untuk menghasilkan sebuah aplikasi yang memiliki arsitektur yang dinamis, sehingga pemenuhan terhadap perubahan yang diminta pada aplikasi dapat dengan cepat dilakukan tanpa mengganggu apa yang telah ada. Penelitian menghasilkan sebuah kombinasi metodologi pengembangan sistem informasi dan tiga modul aplikasi penerimaan mahasiswa baru yang dibangun dengan berbasiskan metodologi yang diteliti. Simpulan dari penelitian ini adalah bahwa metodologi pengembangan sistem informasi yang dibangun dengan menggunakan kombinasi *Service Oriented Architecture* dan *Agile Software Development* mempercepat pengembangan sistem informasi dan meningkatkan kemampuan metodologi pengembangan sistem informasi dalam mengadopsi perubahan proses bisnis yang terjadi.

**Indeks - Metodologi Pengembangan Sistem Informasi, *Service Oriented Architecture*, *Agile Software Development*.**

## I. INTRODUCTION

Dalam kemajuan teknologi, informasi merupakan kebutuhan yang esensial untuk mendukung operasional sebuah

Makalah ini dibuat pada tanggal 18 maret 2009. Makalah ini bagian dari skripsi Jurusan Database, BINUS University Jakarta.

Toninetti adalah mahasiswa Jurusan Database, BINUS University Jakarta. Toninetti mengucapkan terimakasih kepada Gintoro S.kom., M.M. yang telah membimbing selama pembuatan skripsi.

Toninetti mengucapkan terimakasih kepada Renan Prasta Jenie S.T.P.,M.T. yang telah membantu dalam pembuatan jurnal ini.

Toninetti mengucapkan terimakasih kepada karyawan BINUS UNIVERSITY jurusan IT yang telah membantu untuk pembuatan skripsi.

organisasi. Sebuah sistem informasi yang handal dan terintegrasi dibutuhkan dalam menyajikan informasi yang bermanfaat guna mendukung pengambilan keputusan secara tepat dan bermanfaat bagi semua pihak. Sistem informasi juga diharapkan dapat mengadopsi perubahan proses bisnis yang terjadi pada perusahaan dengan cepat.

*Service Oriented Architecture* (SOA) dan *Agile Software Development* adalah metodologi pengembangan sistem yang dapat bergerak dinamis saat pengembangan sebuah sistem informasi. *Agile Software Development* memberikan sebuah panduan pengembangan sistem informasi dengan arsitektur yang dinamis sehingga kelak apabila terjadi perubahan yang harus dapat diadopsi dengan cepat oleh sebuah sistem informasi, hal itu tidak akan menjadi sebuah masalah. SOA dan *Agile Software Development* dapat dijadikan pilihan yang digunakan dalam IT Directorate BINUS Group sehingga pengembangan sistem informasi akan berlangsung dinamis dan cepat.

Sebagai implementasi atas metodologi pengembangan sistem informasi solusi baru maka digunakan tiga modul pada aplikasi Penerimaan Mahasiswa Baru (PMB) sebagai model uji coba integrasi solusi baru, yaitu modul penjualan formulir, registrasi dan seleksi penerima beasiswa.

## II. REFERENSI LITERATUR

### A. Sistem Informasi

Informasi [8] adalah data yang telah diatur sehingga memiliki makna dan nilai bagi penerimanya. Sistem informasi [9] adalah pengaturan orang, data, proses, dan teknologi informasi yang berinteraksi untuk mengumpulkan, memproses, menyimpan, dan menyediakan keluaran informasi yang diperlukan untuk mendukung sebuah organisasi.

### B. Basis Data

Basis Data [5] adalah kumpulan data yang terhubung secara logis yang digunakan bersama-sama dan deskripsi dari data tersebut yang dirancang untuk memenuhi kebutuhan informasi sebuah organisasi. Terdapat beberapa istilah penting dalam basis data, antara lain:

1. Relasi: sebuah tabel dengan kolom dan baris.
2. Atribut: kolom yang diberi nama pada sebuah relasi.
3. *Relationship*: asosiasi antar tabel.
4. *Primary key*: *candidate key* yang terpilih untuk mengidentifikasi *tuple* secara unik dalam sebuah relasi.

5. *Multiplicity*: jumlah kejadian yang mungkin dari sebuah tipe entitas yang berhubungan dengan kejadian tunggal dari tipe entitas lain yang berhubungan melalui *relationship* tertentu, seperti pada Tabel I.

**TABEL I**  
**Batasan Multiplicity**

Batasan Multiplicity	Arti
0..1	Kejadian entitas nol atau satu.
1..1	Kejadian entitas tepat satu.
0..*	Kejadian entitas nol atau lebih.
1..*	Kejadian entitas satu atau lebih.
5..10	Kejadian entitas minimum 5 sampai maksimum 10
0, 3, 6-8	Kejadian entitas nol atau tiga atau enam atau tujuh atau delapan.

*Database Management System* (DBMS) adalah sistem perangkat lunak yang memungkinkan pengguna untuk mendefinisikan, membuat, memelihara, dan mengontrol akses ke basis data.

C. Proses Pengembangan Sistem

Proses pengembangan sistem [9] adalah satu set aktivitas, metode, praktik terbaik, baranng siap dikirim, dan peralatan terotomasi yang digunakan pada stakeholder untuk mengembangkan dan secara berkesinambungan memperbaiki sistem informasi dan perangkat lunak.

D. *Service-Oriented Architecture* (SOA)

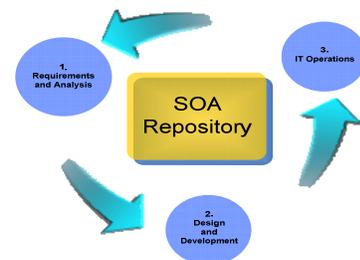
*Service Oriented Architecture* (SOA) [7] merupakan sebuah representasi model baru untuk membangun aplikasi yang terdistribusi. SOA [4] adalah sebuah gaya arsitektural yang memodularisasi sistem informasi menjadi *services*. SOA [3] adalah sebuah framework yang mengintegrasikan proses bisnis dan mendukung infrastruktur IT yang aman, berkomponen terstandarisasi (*services*) yang dapat digunakan kembali dan disertakan dalam prioritas bisnis yang berubah.

Terdapat beberapa aspek kunci pada prinsip SOA [6], yaitu:

1. *Loose coupling*, yaitu bahwa *services* tersebut mempertahankan sebuah hubungan yang meminimalisasi ketergantungan dan mereka hanya perlu menjaga kesadaran antar satu sama lain.
2. *Service contract*, *services* melekat dan taat pada sebuah kesepakatan komunikasi, yang didefinisikan secara kolektif oleh satu atau lebih deskripsi service dan dokumen yang berhubungan.
3. *Autonomy*, bahwa *services* mempunyai kendali berdasarkan logika yang dikapsulasi.
4. *Abstraction*, di luar apa yang dideskripsikan pada *service contract*, *services* menyembunyikan logika dari dunia luar.

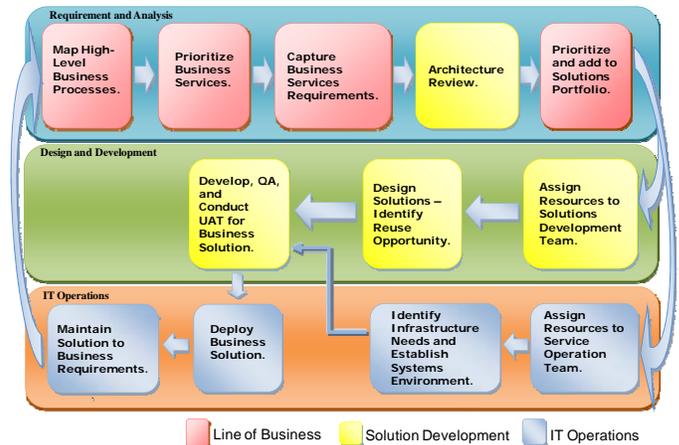
5. *Reusability*, logika terbagi menjadi *services* dengan tujuan untuk digunakan kembali.
6. *Composability*, kumpulan dari *services* dapat dikoordinasikan dan dihimpun untuk membentuk *services* yang berbeda.
7. *Statelessness*, *services* meminimalisasi sifat berpegang teguh pada informasi tertentu untuk sebuah aktivitas.
8. *Discoverability*, *services* dirancang dengan sifat yang deskriptif sehingga mereka dapat ditemukan dan ditentukan dengan menggunakan mekanisme-mekanisme penemuan yang ada.

Siklus hidup *service* memungkinkan penempatan kemampuan *service* melalui tiga tahap [10], yaitu: *requirements and analysis*, *design and development*, dan *IT operations*, seperti pada Gambar I.



**Gambar I Service Lifecycle**

Gambar untuk keseluruhan siklus besar pada SOA dapat dilihat pada Gambar II.



**Gambar II Keseluruhan Siklus Besar Tahapan Metode Pengembangan Sistem Informasi dengan SOA**

1. *Requirement and Analysis*
  - a. *Map High-Level Business Processes*  
Ini merupakan tahap awal dari tahap *requirements and analysis*. Pada tahap ini dilakukan pemetaan terhadap proses-proses bisnis yang akan tercakup dalam sistem. Proses-proses bisnis yang akan tercakup dalam sistem dianalisis dan dipetakan. Pada akhir tahap ini akan menghasilkan proses-proses bisnis yang akan tercakup dalam sistem yang telah terpetakan.
  - b. *Prioritize Business Services*

Pemetaan proses-proses bisnis dalam sistem yang telah dikerjakan pada tahap *Map High-Level Business Processes*, kemudian dilanjutkan dengan melakukan penentuan urutan prioritas proses-proses bisnis sehingga dihasilkan proses-proses bisnis yang akan tercakup dalam sistem yang telah diberikan urutan prioritas.

c. *Capture business services requirements*

Untuk masing-masing proses bisnis yang telah memiliki urutan prioritas, kemudian dilakukan pengumpulan kebutuhan-kebutuhan yang akan digunakan dalam *business services*, sehingga diperoleh daftar kebutuhan-kebutuhan yang akan digunakan dalam *business services*

d. *Architecture Review*

Pada tahap ini dilakukan peninjauan kembali arsitektur yang akan digunakan untuk mengembangkan sistem baru dengan menggunakan daftar proses-proses bisnis yang akan tercakup dalam sistem dan telah terpetakan beserta dengan daftar kebutuhan pada masing-masing proses bisnis.

Architecture Views (Gambar III) adalah gambaran dari keseluruhan arsitektur yang penuh arti yang berguna bagi satu atau lebih stakeholders dalam sistem. Sebuah arsitektur menggambarkan satu atau lebih model arsitektur yang bergabung membentuk sebuah deskripsi yang koheren atas arsitektur sistem yang akan dibangun.

To address the concerns of the following stakeholders...			
Users, Planners, Business Management	Database Designers and Administrators, System Engineers	System and Software Engineers	Acquirers, Operators, Administrators, & Managers
... the following views may be developed			
Business Architecture Views	Data Architecture Views	Applications Architecture Views	Technology Architecture Views
Business Function View	Data Entity View	Software Engineering View	Networked Computing Hardware View
Business Services View			Communications Engineering View
Business Process View			Processing View
Business Information View			
Business Locations View			
Business Logistics View	Data Flow View (Organization Data Use)	Applications Interoperability View	Cost View
People View (Organization Chart)	Logical Data View	Software Distribution View	Standards View
Workflow View			
Usability View			
Business Strategy and Goals View			
Business Objectives View			
Business Rules View	System Engineering View		
Business Events View	Enterprise Security View		
Business Performance View	Enterprise Manageability View		
	Enterprise Quality of Service View		
	Enterprise Mobility View		

**Gambar III Architecture Views**

Sudut pandang arsitektur yang ditinjau kembali meliputi 4 aspek, yaitu :

i. *Business Architecture Views:*

*Business Architecture Views* menunjuk pada hal-hal yang harus diperhatikan oleh *User*, *Planner* dan *Business Manager*, dan fokus pada aspek fungsionalitas sistem dari segi perspektif *User*, hal-hal yang termasuk di dalamnya adalah performa, fungsionalitas sistem, dan kegunaan sistem. Sebuah implementasi SOA bisa dikatakan berhasil bila SOA diarahkan pada *business architecture*. Kemampuan untuk digunakan kembali dari sebuah proses bisnis akan memberikan nilai *Return On Investments* (ROI) yang lebih tinggi dibandingkan kemampuan penggunaan kembali dari sebuah infrastruktur atau komponen data yang potensial.

Aspek-aspek dalam *Business Architecture Views*, antara lain :

- *People View:*  
Fokus pada aspek sumber daya manusia yang terlibat dalam sistem.
- *Business Process View:*  
Berhubungan dengan proses-proses bisnis yang tercakup dalam sistem.
- *Business Function View:*  
Berhubungan dengan fungsi yang diperlukan untuk mendukung masing-masing proses bisnis dalam sistem.
- *Business Information View:*  
Berhubungan dengan informasi yang dibutuhkan untuk mendukung dalam alur proses bisnis.
- *Usability View:*  
Memperhatikan aspek kegunaan dari sistem dan lingkungan sistem yang akan terbentuk.
- *Business Performance View:*  
Memperhatikan aspek performa sistem dan performa lingkungan sistem.

ii. *Data Architecture Views:*

Arsitektur data berhubungan dengan model logikal dan fisik data yang akan digunakan. Termasuk dalam hal ini adalah tipe dan panjang data yang akan digunakan untuk merepresentasikan data yang akan disimpan. Dengan arsitektur data yang baik, maka data yang disimpan juga akan memiliki kualitas yang baik.

iii. *Application Architecture Views:*

Berhubungan dengan bagaimana cara memodelkan setiap bagian yang akan terdapat dalam aplikasi. *Application Architecture Views* mengarahkan terbentuknya komunikasi yang baik antar komponen dalam aplikasi namun tidak memiliki tingkat ketergantungan yang tinggi.

iv. *Technology Architecture Views:*

Menunjuk pada kepedulian *acquirers*, *operators*, *communications engineers*, *administrators*, dan *manager*. Beberapa aspek dalam *Technology Architecture Views*, antara lain:

- *Communications Engineering View:*  
Menunjuk pada hal-hal yang diperhatikan oleh seorang *communications engineer*. Aspek ini akan menguji beberapa cara yang bervariasi untuk menyusun fasilitas komunikasi guna menyederhanakan perencanaan dan perancangan jaringan bisnis.

- *Acquirer's View:*

Menyediakan sebuah panduan yang cocok untuk pendayagunaan komponen. *Acquirer's View* berhubungan dengan biaya dan standard-standard yang harus diikuti untuk mengarah pada efektifitas biaya.

- e. *Prioritize and add to Solutions Portfolio*

Tahap ini merupakan tahap akhir dari *requirements and analysis*. Pada tahap ini dimasukkan data-data yang telah dianalisa dan rancangan arsitektur untuk sistem baru. Semuanya akan digabungkan menjadi *solutions portfolio*.

## 2. *Design and Development:*

- a. *Assign Resources to Solutions Development Team:*

Awal dari tahap *design and development* dimulai dengan proses *Assign Resources to Solutions Development Team*. Tim pengembang merupakan salah satu aspek penting untuk pengembangan sistem baru. Setelah memperoleh semua data mengenai kebutuhan untuk pengembangan sistem baru, tahap berikutnya adalah melakukan penentuan anggota tim pengembangan sistem yang akan terlibat sehingga diperoleh daftar anggota tim pengembang yang akan terlibat.

- b. *Design Solutions – Identify Reuse Opportunity:*

Tahap ini merupakan tahap perancangan solusi yang akan dibangun. Hal ini dilakukan dengan mengacu pada *solutions portfolio* dan *services* apa saja yang telah ada sekarang yang masih dapat digunakan.

- c. *Develop, QA, and Conduct UAT for Business Solution:*

Setelah memperoleh rancangan solusi baru yang akan dikembangkan, maka berikutnya adalah tahap melakukan pengembangan sistem baru, menjaga kualitas sistem yang akan dibangun, dan memberikan kesempatan kepada *user* untuk melakukan *testing*. Pada tahap pengembangan sistem, digunakan pula pendekatan *Agile Software Development* untuk mendukung perancangan arsitektur aplikasi. *Develop, QA, and Conduct UAT for Business Solution* merupakan tahap akhir dari tahap *design and development*. Pada akhir tahap ini dihasilkan solusi baru yang telah jadi dan siap diimplementasikan.

## 3. *IT Operations:*

- a. *Assign Resources to Service Operation Team:*

Tahap ini merupakan awal dari tahap *IT operations*. Tahap ini adalah tahap melakukan penentuan anggota tim *service operation* sistem yang akan terlibat. Pada akhir tahap ini diperoleh daftar anggota tim *service operation* yang akan terlibat.

- b. *Identify Infrastructure Needs and Establish Systems Environment:*

Pada tahap ini dilakukan identifikasi kebutuhan lingkungan sistem dan mempersiapkan segala hal yang diperlukan sehingga diperoleh lingkungan implementasi sistem yang siap digunakan.

- c. *Deploy Business Solution:*

Setelah tahap pengembangan sistem baru selesai dan persiapan untuk implementasi sistem telah terpenuhi, maka langkah berikutnya adalah melakukan implementasi sistem baru. Setelah tahap ini diselesaikan, diperoleh sistem baru yang telah diimplementasikan dan siap digunakan.

- d. *Maintain Solution to Business Requirements:*

Tahap ini merupakan akhir dari *Maintain Solution to Business Requirements* satu siklus besar SOA, yaitu akhir dari tahap *IT operations*. Namun apabila setelah tahap ini berakhir dan terjadi perubahan proses bisnis pada sistem, maka dimungkinkan untuk dilakukan pengulangan dalam langkah-langkah tahap pengembangan sistem baru. Tahap ini merupakan tahap perawatan solusi baru yang telah dijalankan dan memungkinkan untuk dilakukan persiapan untuk pengembangan sistem berikutnya.

## E. *Agile Software Development*

*Agile Software Development* [11] merupakan sebuah model pengembangan sistem yang memungkinkan tim pengembang sistem untuk mengembangkan sistem dan responsif terhadap perubahan dengan cepat, di dalamnya terdapat beberapa nilai utama yang dikandung, yaitu:

1. *Individuals and interactions over processes and tools.*
2. *Working software over comprehensive documentation.*
3. *Customer collaboration over contract negotiation.*
4. *Responding to change over following a plan.*

Terdapat beberapa prinsip utama dalam *agile software development*, yaitu:

1. Prioritas utama kami adalah untuk memuaskan pelanggan melalui penyelesaian perangkat lunak yang berkualitas yang dilakukan dengan cepat dan diberikan secara berkelanjutan.
2. Menyambut dengan terbuka terjadinya perubahan kebutuhan sistem, meskipun pada akhir pengembangan. Proses *agile* memanfaatkan perubahan sebagai keuntungan yang bersaing bagi pelanggan.
3. Menyerahkan piranti lunak yang telah beroperasi kepada pelanggan secara berjangka, dari beberapa minggu sampai dengan beberapa bulan, dengan sebuah kecenderungan pada jangka waktu yang lebih pendek.
4. Pelaku bisnis dan pengembang sistem harus bekerja secara bersama sehari-hari di dalam proyek.
5. Melaksanakan pengerjaan proyek di sekitar individu yang bermotivasi. Menyediakan bagi mereka sebuah lingkungan dan memenuhi kebutuhan mereka, serta percaya bahwa mereka dapat menyelesaikan pekerjaan tersebut.
6. Metode yang paling efisien dan efektif dalam menyampaikan informasi kepada dan dengan sebuah tim adalah percakapan tatap muka.
7. Piranti lunak yang dapat beroperasi adalah ukuran utama dari perkembangan yang telah dilaksanakan.
8. Proses-proses *agile* mengajukan sebuah pengembangan yang dapat bertahan. Para sponsor, pengembang sistem (*developers*), dan para pengguna harus mampu mempertahankan sebuah langkah yang terus-menerus dan tidak terbatas.
9. Perhatian yang berkelanjutan pada keunggulan teknikal dan rancangan yang bagus menambah tingkat kelincuhan sistem.
10. Kesederhanaan keahlian untuk memaksimalkan jumlah kerja bukanlah hal yang penting.
11. Arsitektur, persyaratan sistem dan rancangan yang terbaik, muncul dari tim yang dapat mengurus dirinya sendiri .

12. Pada sebuah selang waktu yang teratur, tim akan berpikir bagaimana untuk menjadi lebih efektif, kemudian menyesuaikan irama perilakunya sesuai dengan hal tersebut.

1. *Single-Responsibility Principle (SRP)*

*Single-Responsibility Principle* [11] memiliki definisi: "A class should have only one reason to change."

Jika sebuah class memiliki lebih dari satu fungsi (*responsibility*), maka fungsi-fungsi tersebut akan menjadi bergantung. Perubahan pada sebuah fungsi akan melemahkan atau menghentikan kemampuan *class* tersebut untuk memenuhi fungsi yang lain. Hal ini mengarah pada desain yang rapuh yang akan hancur pada cara yang tidak diharapkan ketika perubahan dilakukan.

2. *Open/Closed Principle (OCP)*

*Open/Closed Principle* [11] memiliki definisi:

"Software entities (classes, modules, functions, etc.) should be open for extension but closed for modification."

Jika OCP dilakukan dengan baik, perubahan tambahan dilakukan dengan penambahan kode, tanpa merubah kode yang sebelumnya telah ada yang telah berjalan.

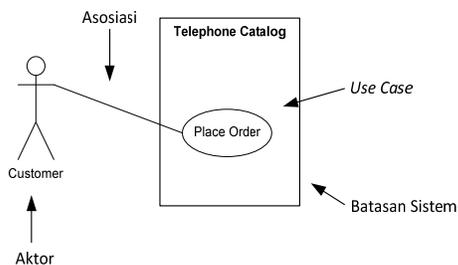
F. *Unified Modelling Language (UML)*

*Unified Modeling Language (UML)* [12] merupakan bahasa pemodelan visual yang bersifat *general-purpose* yang digunakan untuk menspesifikasikan, memvisualisasikan, membangun, dan mendokumentasikan artifak-artifak dari sebuah sistem perangkat lunak. UML menangkap keputusan-keputusan serta pemahaman mengenai sistem yang akan dibangun. UML ditujukan untuk penggunaan pada semua metode pengembangan, tahapan daur hidup, *domain* aplikasi, dan media.

1. *Use Case Diagram*

*Use Case Diagram* [9] adalah diagram yang secara grafis menggambarkan siapa yang akan menggunakan sistem dan dengan cara apa pengguna mengharapkan untuk berinteraksi dengan sistem.

*Use Case Diagram* [12] menangkap perilaku dari sistem ataupun subsistem yang dilihat dari sudut pandang pengguna. Sebuah *use case* menggambarkan interaksi antara aktor dengan sistem. Simbol-simbol yang digunakan dalam *Use Case Diagram* yang dapat dilihat pada Gambar IV.



Gambar IV Notasi *Use Case Diagram*

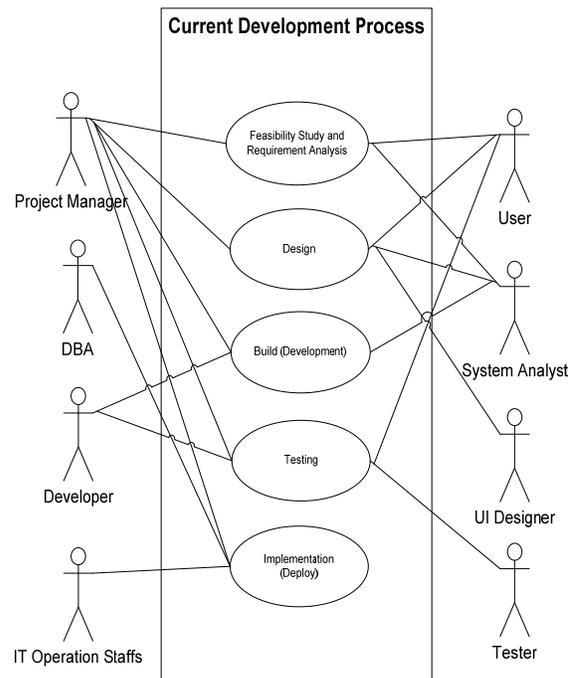
Hal-hal yang perlu diperhatikan dalam mendokumentasikan *use case* adalah sebagai berikut:

1. *Aktor*, adalah segala sesuatu yang berinteraksi dengan sistem. Setiap aktor memiliki peran tertentu.
2. *Precondition*, mengindikasikan apa yang terjadi sebelum *use case* dan menyatakan kondisi sistem pada saat *use case* dimulai.
3. *Post condition*, mengindikasikan apa yang terjadi sesudah *use case* dan menyatakan kondisi sistem pada akhir *use case*. *Post condition* harus selalu bernilai benar dengan alternatif *use case* apapun.
4. *Flow of events*, merupakan serangkaian pernyataan deklaratif yang menyatakan langkah-langkah *use case* dari sudut pandang aktor.
5. *Basic path*, digunakan untuk kondisi dimana segala hal berjalan dengan benar. Harus ada satu *basic path* untuk setiap scenario.
6. *Alternative path* merupakan sesuatu yang memungkinkan urutan kejadian yang berbeda dengan yang terjadi pada *basic path*.

III. ANALISIS

A. *Analisa Observasi*

Proses pengembangan sebuah sistem baru yang saat ini diimplementasikan oleh IT *Directorate* Bina Nusantara dapat diperinci menjadi beberapa proses yang lebih rinci. Proses pengembangan sistem baru dimulai oleh proses *Feasibility Study and Requirement Analysis* yaitu mempelajari dan mempersiapkan kebutuhan untuk sistem baru, dan diakhiri oleh proses *Implementation (Deploy)*. Proses-proses tersebut beserta aktor yang terlibat dapat dilihat pada Gambar V di bawah ini.



Gambar V Proses Pengembangan Sistem yang Berjalan Beserta Aktor yang Terlibat

Setiap proses pada Gambar V akan dijelaskan lebih lanjut dengan menggunakan Deskripsi *Use Case* seperti yang dapat dilihat dalam Tabel II hingga Tabel VI di bawah ini.

**TABEL II**  
**Deskripsi Use Case untuk Proses *Feasibility Study and Requirement Analysis***

<b>Actor</b> :	<i>User, Project Manager, System Analyst</i>
<b>Precondition</b> :	<i>User</i> menginginkan adanya sebuah sistem informasi untuk bagian <i>Business Unit</i> tertentu.
<b>Flow of Events</b> :	<p><b>Basic Path</b> :</p> <ol style="list-style-type: none"> <li><i>User</i> menentukan <i>requirements</i> dan <i>business process</i> yang hendak dipenuhi oleh sistem informasi yang baru.</li> <li><i>Project manager</i> dan <i>system analyst</i> melakukan pertemuan (wawancara) dengan <i>user</i> untuk mendapatkan deskripsi kebutuhan dan <i>business process</i> untuk sistem informasi baru yang akan dibangun.</li> <li><i>Project manager</i> dan <i>system analyst</i> merampungkan uraian kebutuhan dan <i>business process</i> yang akan tercakup untuk sistem baru, dan melakukan <i>feasibility study</i>.</li> </ol> <p><b>Alternative Path</b> :</p> <p><b>Alternative 1</b> : Hasil dari <i>feasibility study, requirements</i> dan <i>business process</i> yang diperlukan untuk sistem informasi yang baru masih belum cukup.</p> <ol style="list-style-type: none"> <li><i>Project manager</i> dan <i>system analyst</i> melakukan pertemuan (wawancara) lagi dengan <i>user</i> untuk mendapatkan deskripsi lebih lanjut mengenai kebutuhan dan <i>business process</i> untuk sistem informasi baru yang akan dibangun.</li> </ol> <p><b>Alternative 2</b> : Ada <i>requirements</i> dan <i>business process</i> untuk sistem informasi baru yang lupa <i>user</i> sampaikan dalam wawancara</p> <ol style="list-style-type: none"> <li><i>User</i> mengirimkan lagi <i>requirements</i> dan <i>business process</i> untuk sistem informasi baru yang lupa disampaikan pada wawancara melalui <i>email</i> atau menyusun kembali jadwal untuk wawancara.</li> </ol>
<b>Post Condition</b> :	Rangkuman uraian kebutuhan dan proses bisnis untuk sistem informasi baru.

**TABEL III**  
**Deskripsi Use Case untuk Proses *Design***

<b>Actor</b> :	<i>Project Manager, System Analyst, UI Designer</i>
<b>Precondition</b> :	<i>Project manager</i> merampungkan rangkuman uraian kebutuhan dan <i>business process</i> hasil dari wawancara dan <i>requirements gathering</i> untuk sistem informasi baru.
<b>Flow of Events</b> :	<p><b>Basic Path</b> :</p> <ol style="list-style-type: none"> <li><i>Project manager</i> dan <i>system analyst</i> membuat rancangan <i>database</i> untuk aplikasi baru.</li> <li><i>Project manager</i> membuat rancangan menu untuk aplikasi baru.</li> </ol>

3. <i>UI (User Interface) designer</i> membuat rancangan tampilan antarmuka untuk aplikasi baru.
<b>Alternative Path</b> :
<b>Alternative 1</b> : <i>User</i> tidak menyukai rancangan tampilan <i>UI</i> atau <i>prototype</i> yang disampaikan.
1. <i>UI designer</i> akan merancang kembali tampilan <i>UI</i> dan <i>prototype</i> sesuai dengan kebutuhan <i>user</i> .
<b>Post Condition</b> :
Rancangan menu, <i>database</i> , tampilan antarmuka dan <i>prototype</i> untuk sistem informasi baru.

**TABEL IV**  
**Deskripsi Use Case untuk Proses *Build (Development)***

<b>Actor</b> :	<i>Project Manager, System Analyst, Developer</i>
<b>Precondition</b> :	<i>Project Manager, System Analyst</i> , dan <i>UI Designer</i> membuat rancangan menu, <i>database</i> dan tampilan antarmuka untuk sistem informasi baru.
<b>Flow of Events</b> :	<p><b>Basic Path</b> :</p> <ol style="list-style-type: none"> <li><i>Project manager</i> memantau langkah pelaksanaan proyek yang telah berlangsung.</li> <li><i>Project manager</i> bersama dengan <i>System Analyst</i> dan <i>Developer</i> mulai menulis <i>code</i> untuk membangun aplikasi sistem informasi yang baru.</li> </ol>
<b>Post Condition</b> :	Aplikasi sistem informasi baru yang telah selesai dikembangkan.

**TABEL V**  
**Deskripsi Use Case untuk Proses *Testing***

<b>Actor</b> :	<i>Project Manager, Tester, Developer</i>
<b>Precondition</b> :	<i>Project manager</i> bersama dengan <i>System Analyst</i> dan <i>Developer</i> selesai mengembangkan aplikasi sistem informasi yang baru.
<b>Flow of Events</b> :	<p><b>Basic Path</b> :</p> <ol style="list-style-type: none"> <li><i>Project manager</i> memantau langkah pelaksanaan proyek yang telah berlangsung.</li> <li><i>Tester</i> melakukan pengecekan/uji coba (<i>testing</i>) atas aplikasi yang telah selesai dikembangkan. Hal ini berguna untuk mendapatkan sebuah aplikasi sistem informasi yang berguna dan memenuhi kebutuhan <i>user</i> serta bersih dari <i>bugs</i>. <i>User</i> akan melakukan pengecekan terhadap kesesuaian aplikasi dalam hubungannya dengan pemenuhan kebutuhan proses bisnis dan tampilan antarmuka</li> <li><i>Developer</i> memperbaiki <i>bugs/kesalahan code</i> yang masih terdapat pada aplikasi.</li> </ol> <p><b>Alternative Path</b> :</p> <p><b>Alternative 1</b> : <i>User</i> sedang tidak ada di tempat atau tidak memiliki waktu untuk melakukan <i>testing</i>.</p> <ol style="list-style-type: none"> <li>Pihak IT akan menunggu sampai ada hasil konfirmasi dari <i>user</i> atas <i>testing</i> yang telah dilakukan.</li> </ol>

<i>Alternative 2</i>	: Ada modul pada sistem informasi baru yang tidak diterima oleh <i>user</i> , namun pihak IT sedang tidak bisa memperbaiki karena sedang banyak pekerjaan.
1.	<i>User</i> akan menunggu hasil perbaikan dari IT sampai perbaikan tersebut selesai bila perbaikan yang diperlukan sedang tidak mendesak. Bila perbaikan yang diperlukan merupakan hal mendesak, maka <i>user</i> akan memberikan batas waktu kepada pihak IT untuk memperbaiki.
<b>Post Condition</b>	: Aplikasi sistem informasi baru yang telah selesai dikembangkan, bebas dari <i>bugs</i> /kesalahan, dan siap dipakai.

**TABEL VI**  
**Deskripsi Use Case untuk Proses Implementation (Deploy)**

<b>Actor</b>	: <i>Project Manager, DBA (Database Administrator), Developer, IT Operation Staffs.</i>
<b>Precondition</b>	: <i>Tester</i> melakukan pengecekan terhadap aplikasi yang telah selesai dikembangkan, dan <i>developer</i> melakukan perbaikan terhadap kesalahan <i>code</i> yang masih terdapat pada aplikasi sehingga diperoleh aplikasi sistem informasi yang siap dipakai.
<b>Flow of Events</b>	: <i>Basic Path</i> :
1.	<i>Project manager</i> bersama dengan <i>DBA</i> melakukan implementasi <i>database</i> pada <i>database server</i> yang akan digunakan.
2.	<i>Project manager</i> bersama dengan <i>developer</i> dan <i>IT Operation Staffs</i> melakukan implementasi aplikasi pada <i>application server</i> yang akan digunakan.
3.	<i>IT Operation Staffs</i> menjaga ( <i>maintain</i> ) aplikasi sistem informasi yang telah diimplementasikan ke <i>application server</i> sehingga <i>user</i> dapat menggunakannya tanpa ada masalah teknis.
<b>Alternative Path</b>	: <i>Alternative 1</i> : Server atau kebutuhan teknis untuk implementasi mengalami gangguan.
1.	Pihak IT akan segera memperbaiki gangguan yang terjadi dan implementasi akan dilaksanakan setelahnya.
<i>Alternative 2</i>	: Aplikasi yang telah dikembangkan oleh <i>developer</i> tidak sesuai ( <i>compatible</i> ) dengan <i>server</i> untuk implementasi.
1.	<i>Developer</i> akan segera mencari jalan keluar agar aplikasi dapat <i>compatible</i> dan diimplementasikan pada <i>server</i> .
<b>Post Condition</b>	: Aplikasi sistem informasi baru yang sudah bisa digunakan oleh <i>user</i> .

Dari hasil observasi terhadap proses pengembangan sebuah sistem baru yang saat ini berjalan yang telah disebutkan di atas, terdapat beberapa masalah dapat diidentifikasi berdasar pada sumber-sumber literatur yang ada seperti yang dapat dilihat pada Tabel VII.

**TABEL VII**  
**Permasalahan yang Teridentifikasi Berdasarkan Hasil Observasi pada Proses Pengembangan Sistem Informasi yang Sedang Berjalan**

No.	Nama Proses	Masalah
1	<i>Build (Development)</i>	Pemenuhan terhadap permintaan perubahan yang diajukan oleh <i>User</i> lambat untuk dilakukan.
2	<i>Design</i>	Frekuensi keterlibatan <i>User</i> dalam perancangan sistem sangat sedikit.
3	<i>Feasibility Study and Requirement Analysis</i>	Pengumpulan <i>User Requirement</i> untuk sistem tidak dapat terselesaikan pada satu periode <i>Feasibility Study and Requirement Analysis</i> .

#### B. Analisa Wawancara

Melalui wawancara yang dilakukan pada beberapa aktor yang terlibat dalam pengembangan sebuah sistem informasi di *IT Directorate* Bina Nusantara. Data yang didapat seperti pada Tabel VIII hingga Tabel X di bawah ini.

**TABEL VIII**  
**Draft Wawancara I**

Nama:	San Karya
Jabatan:	<i>Project Manager</i>
Tanggal wawancara:	4 November 2008
Waktu wawancara:	10.00 – 10.45 WIB
Tempat wawancara:	Ruang Lab Skripsi ATL
1.	Apakah metodologi <i>Software Development LifeCycle (SDLC)</i> yang digunakan sekarang?
2.	Bagaimana urutan proses pengembangan sistem yang digunakan saat ini? Berapa lama waktu masing-masing proses?
3.	Berapa kali proses pengumpulan <i>User Requirement</i> dilakukan?
4.	Berapa lama waktu <i>development</i> yang diperlukan untuk merampungkan sebuah aplikasi besar?
5.	Apa saja kendala teknis yang sering dihadapi?
6.	Apa saja kendala non teknis yang sering dihadapi?
7.	Apakah dalam proses <i>development</i> ada dilakukan <i>n-unit testing</i> ?
8.	Setelah <i>development</i> selesai, faktor apa saja pada aplikasi yang dilakukan pengecekan/ <i>testing</i> ?
9.	Seberapa cepat <i>development methodology</i> sekarang dapat memenuhi perubahan yang dibutuhkan?
10.	Apa kekurangan/masalah sekarang yang paling perlu untuk dibenahi?
11.	Apakah yang diharapkan dapat diberikan oleh <i>development methodology</i> yang baru? Apakah saat ini ada menggunakan <i>framework</i> dari <i>vendor</i> ?

**TABEL IX**  
**Draft Wawancara II**

Nama:	Martinus
Jabatan:	<i>System Analyst</i>
Tanggal wawancara:	4 November 2008
Waktu wawancara:	11.30 – 12.30
Tempat wawancara:	Ruang Lab Skripsi ATL
<ol style="list-style-type: none"> <li>1. Apakah metodologi <i>Software Development LifeCycle</i> (SDLC) yang digunakan sekarang?</li> <li>2. Bagaimana urutan proses pengembangan sistem yang digunakan saat ini? Berapa lama waktu masing-masing proses ?</li> <li>3. Berapa kali proses pengumpulan <i>User Requirement</i> dilakukan?</li> <li>4. Berapa lama waktu <i>development</i> yang diperlukan untuk merampungkan sebuah aplikasi besar?</li> <li>5. Apa saja kendala teknis yang sering dihadapi?</li> <li>6. Apakah dalam proses <i>development</i> ada dilakukan <i>n-unit testing</i>?</li> <li>7. Setelah <i>development</i> selesai, faktor apa saja pada aplikasi yang dilakukan pengecekan/<i>testing</i>?</li> <li>8. Seberapa cepat <i>development methodology</i> sekarang dapat memenuhi perubahan yang dibutuhkan?</li> <li>9. Untuk memenuhi requirement baru tersebut, berapa lama waktu yang diperlukan untuk versi yang berikutnya?</li> <li>10. Apa kekurangan/masalah sekarang yang paling perlu untuk dibenahi?</li> <li>11. Apakah yang diharapkan dapat diberikan oleh <i>development methodology</i> yang baru? Apakah saat ini ada menggunakan <i>framework</i> dari <i>vendor</i>?</li> </ol>	

**TABEL X**  
**Draft Wawancara III**

Nama:	Hunvrey
Jabatan:	<i>System Developer</i>
Tanggal wawancara:	4 November 2008
Waktu wawancara:	14.30 – 15.00
Tempat wawancara:	Ruang Lab Skripsi ATL
<ol style="list-style-type: none"> <li>1. Bagaimana urutan proses pengembangan sistem yang digunakan saat ini?</li> <li>2. Berapa lama waktu <i>development</i> yang diperlukan untuk merampungkan sebuah aplikasi besar?</li> <li>3. Apa saja kendala teknis yang sering dihadapi?</li> <li>4. Apa saja kendala non teknis yang sering dihadapi?</li> <li>5. Apakah dalam proses <i>development</i> ada dilakukan <i>n-unit testing</i>?</li> <li>6. Setelah <i>development</i> selesai, faktor apa saja pada aplikasi yang dilakukan pengecekan/<i>testing</i>?</li> <li>7. Seberapa cepat <i>development methodology</i> sekarang dapat memenuhi perubahan yang dibutuhkan?</li> <li>8. Apa kekurangan/masalah sekarang yang paling perlu untuk dibenahi?</li> <li>9. Apakah yang diharapkan dapat diberikan oleh <i>development methodology</i> yang baru?</li> <li>10. Apakah saat ini ada menggunakan <i>framework</i> dari <i>vendor</i>?</li> </ol>	

Evaluasi permasalahan yang diperoleh hasil wawancara yang dilakukan dapat dilihat pada Tabel XI di bawah ini.

**TABEL XI**  
**Evaluasi Hasil Wawancara Terhadap Permasalahan**

No.	Permasalahan yang Muncul	Target Pengguna yang Mengalami Permasalahan	Evaluasi dari:
1	Kemampuan <i>software development methodology</i> yang digunakan saat ini lambat dalam memenuhi perubahan yang diminta.	<i>Project Manager, System Analyst dan System Developer.</i>	Pertanyaan nomor 8 dan nomor 11 pada wawancara kedua. Jawaban dari pertanyaan ini berperan sebagai masukan untuk mengetahui kekurangan apa saja yang ada pada saat ini, sehingga implementasi metodologi baru diharapkan dapat memberikan hasil sesuai dengan diinginkan.
2	<i>User</i> sangat sulit menyediakan waktu untuk keterlibatan dalam perancangan proses bisnis.	<i>Project Manager dan System Analyst.</i>	Pertanyaan nomor 10 pada wawancara pertama. Pertanyaan nomor 3 pada wawancara kedua. Jawaban dari pertanyaan ini berperan sebagai masukan bagi metodologi baru untuk mengetahui masalah non teknis apa saja yang dihadapi saat ini.
3	Fleksibilitas <i>software development methodology</i> yang digunakan	<i>Project Manager, System Analyst dan System Developer.</i>	Pertanyaan nomor 11 pada wawancara pertama. Pertanyaan

No.	Permasalahan yang Muncul	Target Pengguna yang Mengalami Permasalahan	Evaluasi dari:
	saat ini rendah untuk menangani perubahan proses.		nomor 10 dan nomor 11 pada wawancara kedua. Dari pertanyaan-pertanyaan ini diperoleh informasi mengenai masalah utama yang ada pada metodologi pengembangan sistem saat ini sehingga implementasi metodologi baru diharapkan dapat memberikan solusi untuk permasalahan yang dihadapi.
4	Sulit dalam melakukan <i>maintenance</i> terhadap modul-modul aplikasi yang telah ada, karena kekhawatiran perbaikan kecil yang dilakukan dalam modul akan berpengaruh besar terhadap aplikasi.	<i>Project Manager, System Analyst dan System Developer.</i>	Pertanyaan no.11 pada wawancara pertama. Hal ini akan menjadi panduan untuk mengetahui perbaikan-perbaikan utama apa saja yang perlu dilakukan pada metodologi yang baru.
5	Pengumpulan <i>user requirement</i> tidak dapat terselesaikan semuanya pada satu periode pengumpulan <i>user requirement</i> .	<i>Project Manager dan System Analyst.</i>	Pertanyaan nomor 3 dan nomor 10 pada wawancara pertama. Jawaban dari pertanyaan ini berperan sebagai masukan untuk mengetahui kekurangan apa saja yang ada pada saat ini,

No.	Permasalahan yang Muncul	Target Pengguna yang Mengalami Permasalahan	Evaluasi dari:
			sehingga implementasi metodologi baru diharapkan dapat memberikan hasil sesuai dengan diinginkan.
6	Modul dengan fungsi yang sama yang dikembangkan pada sistem sebelumnya tidak dapat digunakan kembali untuk sistem baru yang akan dibangun karena tingkat <i>coupling</i> antar modul tinggi.	<i>Project Manager dan System Developer.</i>	Pertanyaan 9 pada wawancara pertama. Hasil dari jawaban untuk kedua pertanyaan ini menjadi salah satu tolak ukur kemampuan metodologi yang baru menjawab permasalahan yang ada saat ini.

### C. Usulan Solusi

Solusi yang dapat digunakan untuk memecahkan masalah yang telah diidentifikasi sebelumnya adalah dengan menggunakan *Service Oriented Architecture (SOA)* sebagai metodologi baru untuk keseluruhan proses pengembangan sistem mulai dari pengumpulan *User Requirements* sampai dengan *Maintenance*. Pada bagian tahap *development* di *SOA*, *Agile Software Development* merupakan solusi yang tepat sebagai panduan untuk menghasilkan sebuah aplikasi yang memiliki arsitektur yang dinamis, sehingga pemenuhan terhadap perubahan yang diminta pada aplikasi dapat dengan cepat dilakukan tanpa mengganggu apa yang telah ada. Rangkuman solusi terhadap permasalahan yang telah diidentifikasi dapat dilihat pada Tabel XII di bawah ini.

**TABEL XII**  
**Rangkuman Solusi untuk Permasalahan Studi Kasus**

No.	Permasalahan yang Diidentifikasi	Solusi untuk Permasalahan	Verifikasi dengan Landasan Teori
1	Kemampuan <i>software development methodology</i> yang digunakan saat	Dalam pelaksanaan pengembangan sistem, dilakukan dengan	“ <i>Welcome changing requirements, even late in development. Agile processes</i> ”

No.	Permasalahan yang Diidentifikasi	Solusi untuk Permasalahan	Verifikasi dengan Landasan Teori
	ini lambat dalam memenuhi perubahan yang diminta.	pendekatan SOA ( <i>Service Oriented Architecture</i> ) dan <i>Agile Software Development</i> . Metodologi SOA dan <i>Agile Software Development</i> memungkinkan para pelaku yang terlibat dalam pengembangan sistem informasi untuk tetap melakukan pengembangan sistem ketika ada permintaan untuk melakukan perubahan, baik perubahan dalam sebuah proses bisnis, maupun perubahan alur proses bisnis. <i>Agile Software Development</i> memberikan sebuah rancangan arsitektur modul yang dapat bergerak dinamis terhadap perubahan.	<i>harness change for the customer's competitive advantage. This is a statement of attitude. The participants in an agile process are not afraid of change. They view changes to the requirements as good things, because those changes mean that the team has learned more about what it will take to satisfy the customer.</i> " (Micah, dan Robert, 2006)  "As the requirements of the application change, we can extend the module with new behaviors that satisfy those changes. In other words, we are able to change what the module does." (Micah, dan Robert, 2006)  "...Focusing on the business processes is important for enterprises to mature to a more flexible goal-oriented model. However, business and IT operations teams frequently differ in their approaches. For example, some

No.	Permasalahan yang Diidentifikasi	Solusi untuk Permasalahan	Verifikasi dengan Landasan Teori
			<i>business operations teams prefer to demonstrate "quick wins" to validate an approach, while IT operations prefer to build out the infrastructure. Fortunately, SOA offers both.</i> " (Durvasula, 2006)
2	User sangat sulit menyediakan waktu untuk keterlibatan dalam perancangan proses bisnis.	Menggunakan <i>Agile Software Development</i> untuk pengembangan dalam aplikasi. Metodologi <i>Agile Software Development</i> san mengutamakan keterlibatan user dalam pengembangan aplikasi mulai dari tahap awal.	" Successful projects involve customer feedback on a regular and frequent basis. Rather than depending on a contract, or a statement of work, the customer of the software works closely with the development team, providing frequent feedback on its efforts." (Micah, dan Robert, 2006)
3	Fleksibilitas software development methodology yang digunakan saat ini rendah untuk menangani perubahan proses.	Dalam pelaksanaan pengembangan sistem, dilakukan dengan pendekatan SOA ( <i>Service Oriented Architecture</i> ). Sistem yang dibangun dengan metodologi SOA memiliki arsitektur yang berorientasi	" Services are loosely coupled Services must be designed to interact without the need for tight, cross-service dependencies." (Erl, 2005)

No.	Permasalahan yang Diidentifikasi	Solusi untuk Permasalahan	Verifikasi dengan Landasan Teori
		terhadap fungsi/ <i>service</i> yang akan diberikan. Sistem yang dikembangkan akan menjadi sebuah sistem yang berpotensi karena kebutuhan user dapat terpenuhi dan sistem akan responsif terhadap perubahan.	
4	Sulit dalam melakukan <i>maintenance</i> terhadap modul-modul aplikasi yang telah ada, karena kekhawatiran perbaikan kecil yang dilakukan dalam modul akan berpengaruh besar terhadap aplikasi	Menggunakan <i>Agile Software Development</i> dan SOA untuk pengembangan dalam aplikasi. Konsep <i>abstraction</i> pada <i>Agile Software Development</i> menghasilkan sebuah arsitektur aplikasi yang mudah untuk dipelihara karena modul-modul yang ada dalam aplikasi dapat dengan mudah diperbaiki tanpa mempengaruhi modul lain apabila diperlukan adanya perbaikan. SOA memperkuat arsitektur sistem yang dibangun karena modul-modul yang ada telah	<p>“<i>Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely. An agile project is not run like a 50-yard dash; it is run like a marathon. The team does not take off at full speed and try to maintain that speed for the duration. Rather, it runs at a fast but sustainable pace.</i>” (Micah, dan Robert, 2006, p)</p> <p>“<i>Service loose coupling establishes an inherent independence that frees a service from immediate ties to others. This makes it a great deal easier to</i></p>

No.	Permasalahan yang Diidentifikasi	Solusi untuk Permasalahan	Verifikasi dengan Landasan Teori
		terklasifikasi berdasarkan <i>services</i> yang akan diberikan.	<i>realize reuse.</i> ” (Erl, 2005)
5	Pengumpulan user requirement tidak dapat terselesaikan semuanya pada satu periode pengumpulan <i>user requirement</i> .	Kebutuhan untuk sistem yang tercakup dalam pengumpulan user requirement sulit dimungkinkan terpenuhi apabila dalam satu kali tahap pelaksanaan sudah dapat diperoleh semua data. Oleh karena itu diperlukan sebuah metodologi yang memungkinkan terjadinya penambahan user requirement di tengah proses pengembangan yang sedang berlangsung. Hal ini dapat diatasi dengan menggunakan metodologi SOA dan agile.	<p>“<i>Businesspeople and developers must work together daily throughout the project.</i>” (Micah, dan Robert, 2006)</p>
6	Modul dengan fungsi yang sama yang dikembangkan pada sistem sebelumnya tidak dapat digunakan kembali untuk sistem baru yang akan dibangun karena tingkat <i>coupling</i> antar modul tinggi.	Menggunakan <i>Agile Software Development</i> dan SOA untuk pengembangan dalam aplikasi. <i>Microservice Architecture</i> dan SOA akan menghasilkan sebuah aplikasi yang memiliki arsitektur dengan tingkat	<p>“<i>Service loose coupling establishes an inherent independence that frees a service from immediate ties to others. This makes it a great deal easier to realize reuse.</i>” (Erl, 2005)</p> <p>“<i>When a single</i></p>

No.	Permasalahan yang Diidentifikasi	Solusi untuk Permasalahan	Verifikasi dengan Landasan Teori
		<i>coupling</i> yang rendah. Semakin rendah tingkat <i>coupling</i> sebuah aplikasi, dan semakin tinggi tingkat kohesi di dalamnya, akan menambah kualitas modul-modul yang terdapat dalam aplikasi tersebut dan memiliki kemampuan yang tinggi untuk dikonsumsi kembali oleh aplikasi lainnya.	<i>change to a program results in a cascade of changes to dependent modules, the design smells of rigidity. OCP advises us to refactor the system so that further changes of that kind will not cause more modifications. If OCP is applied well, further changes of that kind are achieved by adding new code, not by changing old code that already works.”</i> (Micah, dan Robert, 2006)

#### IV. PERANCANGAN

##### A. Perancangan Proses Pengembangan Sistem Diusulkan Pada Solusi.

###### 1. *Requirement and Analysis*.

Proses *Requirement and Analysis* dilakukan oleh *Project Manager, System Analyst, Solution Architect* dan *User*. Proses ini merupakan proses mengidentifikasi dan memberikan urutan prioritas pada kebutuhan bisnis. Proses *Requirement and Analysis* terbagi lagi menjadi beberapa tahap subproses yang lebih kecil.

###### a. Subproses *Mapping High Level Business Processes*

Pada tahap ini dilakukan pemetaan terhadap proses-proses bisnis yang akan tercakup dalam sistem. Proses-proses bisnis yang akan tercakup dalam sistem dianalisis dan dipetakan. Pada akhir tahap ini akan menghasilkan proses-proses bisnis yang akan tercakup dalam sistem yang telah terpetakan.

Untuk mendapatkan proses-proses bisnis yang akan tercakup dalam sistem, dilakukan wawancara terhadap seorang staf bagian admisi yang melayani penerimaan mahasiswa baru, seperti pada Tabel XIII.

**TABEL XIII**  
**Data Staff Penerimaan Mahasiswa Baru**

Nama:	Hendy
Jabatan:	<i>Admission Section Head</i>
Tanggal wawancara:	16 Januari 2009
Waktu wawancara:	13.00 – 14.00 WIB
Tempat wawancara:	Ruang Bagian Admisi <i>BINUS University</i>

Hasil dari wawancara merupakan masukan untuk uraian proses-proses bisnis yang ada dalam penerimaan mahasiswa baru. Berikut ini merupakan uraian proses-proses yang terdapat dalam siklus penerimaan mahasiswa baru pada aplikasi Penerimaan Mahasiswa Baru (PMB) pada *BINUS University* yang diperoleh dari hasil wawancara:

Proses penerimaan mahasiswa baru pada *BINUS University* diawali dengan penentuan jumlah gelombang pendaftaran yang akan dibuka untuk satu tahun akademik. Pada tahap ini dilakukan pula penentuan rentang tanggal masing-masing gelombang pendaftaran, tanggal pengembalian formulir per gelombang pendaftaran, tanggal pelaksanaan ujian per gelombang pendaftaran, tanggal pelaksanaan orientasi masing-masing gelombang pendaftaran, beasiswa atau keringanan biaya yang tersedia untuk masing-masing gelombang pendaftaran, *shift* perkuliahan yang dibuka, jurusan perkuliahan yang dibuka, biaya pendaftaran masing-masing jurusan per gelombang pendaftaran, jumlah cicilan dan nominal masing-masing cicilan, dan tanggal pendaftaran ulang per gelombang pendaftaran.

Kemudian melangkah ke proses berikutnya adalah alokasi ruangan. Alokasi ruangan dilakukan untuk dua keperluan, yakni alokasi ruang ujian seleksi pendaftaran mahasiswa baru dan alokasi ruangan untuk orientasi.

Nomor formulir pendaftaran pada *BINUS University* tidak memiliki format yang baku. Format nomor formulir dapat berbeda antara tahun yang satu dan tahun sebelumnya. Yang dilakukan adalah hanya menyediakan tempat pada *database* dengan ukuran delapan karakter untuk *field* nomor formulir. Format nomor formulir ditentukan oleh staf *marketing*. Petugas administrasi memasukkan format formulir untuk yang nomor pertama dan kemudian memasukkan angka jumlah formulir yang hendak dicetak, maka sistem akan menghasilkan nomor-nomor formulir yang berurutan sesuai dengan format pertama yang telah ditentukan sebelumnya.

Formulir pendaftaran didistribusikan melalui beberapa cara, yaitu melalui *agent (education counselor)*, melalui penjualan pada kantor admisi *BINUS University*, dan melalui sekolah-sekolah yang bekerja sama dengan *BINUS University*.

Pendaftar dapat juga melakukan pendaftaran melalui internet. Pendaftar yang melakukan pendaftaran di internet, akan mengisi formulir *online*, dan mendapatkan kode booking. Setelah dilakukan pembayaran, barulah kemudian akan mendapatkan nomor formulir. Pada saat seseorang akan melakukan pembelian formulir secara *online*, maka akan

ditanyakan kepada siapa ia akan merekomendasikan BINUS *University* untuk berikutnya.

Pendaftar yang telah melakukan pembelian formulir pendaftaran kemudian mengisinya di rumah dan dikumpulkan kembali ke kantor admisi BINUS *University* beserta dengan dokumen-dokumen yang dibutuhkan, inilah yang disebut dengan proses registrasi. Setelah memperoleh pengumpulan formulir pendaftaran yang telah diisi beserta dengan dokumen yang diperlukan, maka berikutnya adalah pertama-tama petugas bagian admisi akan melakukan penyaringan nama-nama pendaftar yang berhak menerima beasiswa. Bagi para pendaftar yang ditolak permohonan beasiswanya, maka akan langsung dialokasikan untuk mengikuti ujian penyaringan pendaftaran mahasiswa baru.

Setiap peserta yang mengikuti ujian seleksi masuk akan mendapatkan kartu peserta ujian yang harus dibawa saat ujian. Kartu peserta ujian ini diberikan kepada pendaftar setelah mengembalikan formulir pendaftaran yang telah dilengkapi data-data di dalamnya. Untuk pendaftar yang melakukan pendaftaran secara *online* karena berada di luar kota dan tidak dapat datang ke admisi BINUS *University*, maka pencetakan kartu peserta ujian dapat dilakukan sendiri dengan mengunduh kartu peserta ujian yang bersangkutan dari situs pendaftaran mahasiswa baru BINUS *University*.

Pelaksanaan ujian penerimaan mahasiswa baru terbagi menjadi 2 jenis, yaitu *computer-based test* dan *paper-based test*. *Paper-based test* dilakukan terhadap ujian yang sifatnya tidak mungkin untuk dilaksanakan secara komputerisasi, misalnya ujian melukis.

Hal-hal yang perlu disiapkan untuk pelaksanaan ujian penerimaan mahasiswa baru adalah daftar pertanyaan untuk *computer-based test*, daftar absensi peserta ujian, daftar nama petugas penjaga ujian dan ruangan ujian yang diawasi, pencetakan data peserta ujian pada lembar jawaban ujian, dan daftar *survey question* yang akan diberikan kepada peserta ujian untuk keperluan analisa pasar. Pertanyaan-pertanyaan yang dimasukkan dalam ujian penerimaan mahasiswa baru disimpan dalam *database* beserta dengan opsi jawaban dan jawaban yang benar. Oleh karena itu, sebelum pelaksanaan ujian yang berbasis komputer, petugas yang bersangkutan sebelumnya harus melakukan input data pertanyaan, opsi jawaban dan jawaban yang benar. Hal ini dilakukan dengan menggunakan sistem.

Setelah pelaksanaan ujian, kemudian dilakukan pengecekan jawaban peserta dengan menggunakan komputer dan dilanjutkan dengan proses *test-data importing*, yaitu proses memasukkan data nilai hasil ujian dari *access* atau *excel* ke *database*. Untuk *paper-based test*, penilaian dilakukan secara manual dan kemudian nilai tersebut dimasukkan ke dalam *database*.

Untuk proses pengumuman nilai hasil ujian penerimaan mahasiswa baru, sebelumnya dilakukan beberapa langkah. Pertama-tama hasil ujian dicetak dan dilaporkan kepada manajemen yang berwenang untuk disetujui. Setelah mendapatkan persetujuan, baru kemudian dilakukan pencetakan hasil ujian penerimaan mahasiswa baru yang akan ditempelkan pada papan pengumuman kantor admisi BINUS *University*. Pengumuman hasil ujian dapat

diketahui oleh para pendaftar melalui situs BINUS *University* dan melalui telepon (*BINUS Phone Service*).

Bagi para pendaftar yang lulus dalam ujian penerimaan mahasiswa baru, proses berikutnya yang dilakukan adalah pendaftaran ulang. Pada tahap ini, pendaftar harus melengkapi data-data pada formulir pendaftaran ulang, melampirkan dokumen-dokumen yang diperlukan dan telah melakukan pembayaran administrasi sampai pada tahap yang telah ditentukan. Untuk beberapa dokumen yang masih belum dapat dilampirkan oleh pendaftar karena belum dimiliki, seperti surat tanda kelulusan SLTA atau dokumen lainnya yang sejenis, maka dapat diberikan keringanan dengan mengizinkan pendaftar untuk mengumpulkan dokumen tersebut menyusul kemudian sesuai dengan batas tanggal waktu yang ditentukan. Apabila semua kebutuhan administrasi dokumen dan keuangan telah dipenuhi oleh pendaftar, maka staf admisi akan melakukan proses penentuan ruang orientasi bagi pendaftar dan pencetakan kartu *briefing*.

b. Subproses *Prioritize Business Services*

Setelah melakukan pemetaan atas proses bisnis apa saja yang akan tercakup dalam sistem penerimaan mahasiswa baru BINUS *University*, maka proses berikutnya adalah melakukan penentuan urutan prioritas setiap proses bisnis yang akan tercakup dalam sistem.

Daftar prioritas proses bisnis penerimaan mahasiswa baru yang merupakan rangkuman dari keseluruhan proses penerimaan mahasiswa baru diberikan pada Tabel XIV.

**TABEL XIV**  
**Daftar Proses Bisnis Penerimaan Mahasiswa Baru**  
**Berdasarkan Urutan Prioritasnya**

Prioritas	Proses Bisnis	Subproses
1.	<i>Setting and Scheduling</i>	<i>Enrollment Setting</i>
		<i>Enrollment Phases Scheduling</i>
		<i>Enrollment Phases Setting</i>
		<i>Enrollment Test Setting</i>
		<i>Orientation Scheduling</i>
		<i>Grade Setting</i>
		<i>Payment Setting</i>
2.	<i>Room Allocation</i>	<i>Scholarship Setting</i>
		<i>Test Room Allocation</i>
3.	<i>Form Number Generation and Distribution</i>	<i>Orientation Room Allocation</i>
		<i>Form Number Generation</i>
4.	<i>Application Form Purchase Offline</i>	<i>Form Number Distribution</i>
		<i>Buyer Information Inputting</i>
		<i>Online Form Number Ordering</i>
	<i>Application Form Purchase Online</i>	<i>Application Form</i>

Prioritas	Proses Bisnis	Subproses
		<i>Payment Transferring</i>
		<i>Transfer Receipt Confirmation</i>
5.	<i>Registration</i>	<i>Primary Data Inputting</i>
		<i>Secondary Data Inputting</i>
		<i>Application Form Information Inputting (untuk applicant)</i>
		<i>Application Form Submitting (untuk applicant)</i>
		<i>Test Room Placement</i>
6.	<i>Scholarship Filtering</i>	<i>Scholarship Decision</i>
		<i>Scholarship Approval</i>
		<i>Scholarship Announcement</i>
7.	<i>Test Preparation</i>	<i>Computer-based Test Question Ordering</i>
		<i>Paper-based Test Question Ordering</i>
		<i>Attendance List Printing</i>
		<i>Test Card Printing</i>
		<i>Applicant's Data Exporting</i>
8.	<i>Test Data Inputting</i>	<i>Test Score Importing</i>
		<i>Manual Test Score Inputting</i>
9.	<i>Test Result Announcement</i>	<i>Approval Grading Test Result</i>
		<i>Test Result Publishing</i>
		<i>Test Result Printing</i>
10.	<i>Re-registration</i>	<i>Re-registration Document Checking</i>
		<i>Primary Re-registration Information Inputting</i>
		<i>Secondary Re-registration Information Inputting</i>
		<i>Orientation Room Placement</i>
		<i>Briefing Card Printing</i>
		<i>Fee Payment Inputting</i>

c. Subproses *Capture Business Services Requirements*

Untuk masing-masing proses bisnis yang telah memiliki urutan prioritas, kemudian dilakukan pengumpulan kebutuhan-kebutuhan yang terdapat pada masing-masing proses bisnis. Hasil dari pengumpulan kebutuhan-kebutuhan proses bisnis akan diperoleh daftar kebutuhan-kebutuhan yang akan digunakan dalam *business services*. Daftar kebutuhan untuk masing-masing proses bisnis diberikan pada Tabel XV.

**TABEL XV**  
**Daftar Kebutuhan untuk Masing-Masing Proses**  
**Bisnis Penerimaan Mahasiswa Baru**

No.	Nama Proses	Aktor	Dokumen
1.	<i>Setting and Scheduling</i>	<i>Back Admission Staff</i>	<ul style="list-style-type: none"> <li>- <i>Enrollment Setup</i></li> <li>- <i>Enrollment Phase</i></li> <li>- <i>Required Registration Document</i></li> <li>- <i>Required Re-registration Document</i></li> <li>- <i>Opened Department</i></li> <li>- <i>Opened Lecture Shift</i></li> <li>- <i>Opened Enrollment Track</i></li> <li>- <i>Orientation Schedule</i></li> <li>- <i>Grade List</i></li> <li>- <i>Payment Rule</i></li> </ul>
2.	<i>Room Allocation</i>	<i>Back Admission Staff</i>	<ul style="list-style-type: none"> <li>- <i>Room</i></li> <li>- <i>Enrollment Phase</i></li> <li>- <i>Test Schedule</i></li> <li>- <i>Test Room Allocation</i></li> <li>- <i>Orientation Room Allocation</i></li> </ul>
3.	<i>Form Number Generation and Distribution</i>	<i>Back Admission Staff</i>	<ul style="list-style-type: none"> <li>- <i>Form Number</i></li> <li>- <i>Enrollment Phase</i></li> <li>- <i>Distribution Location</i></li> <li>- <i>Distributed Form Number</i></li> </ul>
4.	<i>Application Form Purchase Offline</i>	<i>Front Admission Staff</i>	<ul style="list-style-type: none"> <li>- <i>Distributed Form Number Buyer Information</i></li> </ul>
	<i>Application Form Purchase Online</i>	<i>Front Admission Staff</i>	<ul style="list-style-type: none"> <li>- <i>Distributed Form Number Buyer Information</i></li> </ul>
		<i>Applicant</i>	<ul style="list-style-type: none"> <li>- <i>Purchase Information Application Form Payment Transfer Receipt</i></li> </ul>
5.	<i>Registration</i>	<i>Back Admission</i>	<ul style="list-style-type: none"> <li>- <i>Primary Application</i></li> </ul>

No.	Nama Proses	Aktor	Dokumen
		<i>Staff</i>	<ul style="list-style-type: none"> <li><i>Form Information</i></li> <li>– <i>Secondary Application Form Information</i></li> <li>– <i>Application Form Information</i></li> </ul>
		<i>Front Admission Staff</i>	<ul style="list-style-type: none"> <li>– <i>Buyer Information</i></li> <li>– <i>Primary Application Form Information</i></li> <li>– <i>Application Form Information</i></li> <li>– <i>Required Registration Document</i></li> <li>– <i>Submitted Registration Document</i></li> <li>– <i>Test Room Allocation</i></li> <li>– <i>Applicant's Test Room</i></li> </ul>
		<i>Applicant</i>	<ul style="list-style-type: none"> <li>– <i>Buyer Information</i></li> <li>– <i>Application Form Information</i></li> <li>– <i>Application Form</i></li> </ul>
6.	<i>Scholarship Filtering</i>	<i>Back Admission Staff</i>	<ul style="list-style-type: none"> <li>– <i>Application Form Information</i></li> <li>– <i>Submitted Registration Document</i></li> <li>– <i>Test Room Allocation</i></li> <li>– <i>Applicant's Test Room</i></li> <li>– <i>Scholarship Receiver Data</i></li> </ul>
7.	<i>Test Preparation</i>	<i>Back Admission Staff</i>	<ul style="list-style-type: none"> <li>– <i>Application Form Information</i></li> <li>– <i>Applicant's Test Room</i></li> <li>– <i>Test Room Allocation</i></li> <li>– <i>Printed</i></li> </ul>

No.	Nama Proses	Aktor	Dokumen
			<ul style="list-style-type: none"> <li><i>Attendance List</i></li> <li>– <i>Applicant's Data</i></li> </ul>
		<i>Front Admission Staff</i>	<ul style="list-style-type: none"> <li>– <i>Application Form Information</i></li> <li>– <i>Applicant's Test Room</i></li> <li>– <i>Test Schedule</i></li> <li>– <i>Test Card</i></li> </ul>
8.	<i>Test Data Inputting</i>	<i>Back Admission Staff</i>	<ul style="list-style-type: none"> <li>– <i>Test Score</i></li> <li>– <i>Test Score Data</i></li> </ul>
9.	<i>Test Result Announcement</i>	<i>Back Admission Staff</i>	<ul style="list-style-type: none"> <li>– <i>Grade List</i></li> <li>– <i>Grade Score Range</i></li> <li>– <i>Test Score Data</i></li> <li>– <i>Test Result</i></li> <li>– <i>Fee Payment Invoice</i></li> <li>– <i>Application Form Information</i></li> <li>– <i>Test Result</i></li> <li>– <i>Scholarship Receiver Data</i></li> <li>– <i>Fee Payment Invoice</i></li> <li>– <i>Printed Test Result</i></li> </ul>
		<i>Applicant</i>	<ul style="list-style-type: none"> <li>– <i>Application Form Information</i></li> <li>– <i>Test Result</i></li> <li>– <i>Scholarship Receiver Data</i></li> <li>– <i>Fee Payment Invoice</i></li> </ul>
10.	<i>Re-registration</i>	<i>Front Admission Staff</i>	<ul style="list-style-type: none"> <li>– <i>Fee Payment Invoice</i></li> <li>– <i>Transfer Receipt</i></li> <li>– <i>Fee Payment Data</i></li> <li>– <i>Payment Receipt</i></li> <li>– <i>Primary Re-registration Information</i></li> <li>– <i>Re-registration Information</i></li> <li>– <i>Required Re-registration Document</i></li> </ul>

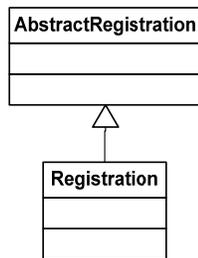
No.	Nama Proses	Aktor	Dokumen
			<ul style="list-style-type: none"> <li>- Submitted Re-registration Document</li> <li>- Orientation Room Allocation</li> <li>- Applicant's Orientation Class</li> <li>- Briefing Card</li> </ul>
		Back Admission Staff	<ul style="list-style-type: none"> <li>- Primary Re-registration Information</li> <li>- Secondary Re-registration Information</li> <li>- Re-registration Information</li> </ul>

d. Subproses *Architecture Review*

Pengembangan sistem Penerimaan Mahasiswa Baru dengan SOA memetakan dengan jelas setiap proses bisnis yang akan tercakup dalam sistem Penerimaan Mahasiswa Baru.

Untuk rancangan arsitektur data bagi sistem Penerimaan Mahasiswa Baru akan menggunakan pendekatan *relational database*. *Relational database* memiliki kemampuan untuk menghubungkan relasi antar tabel sehingga hubungan yang terjadi di antaranya dapat terlihat jelas dan dapat mempercepat proses pengambilan data pada tabel-tabel yang berhubungan.

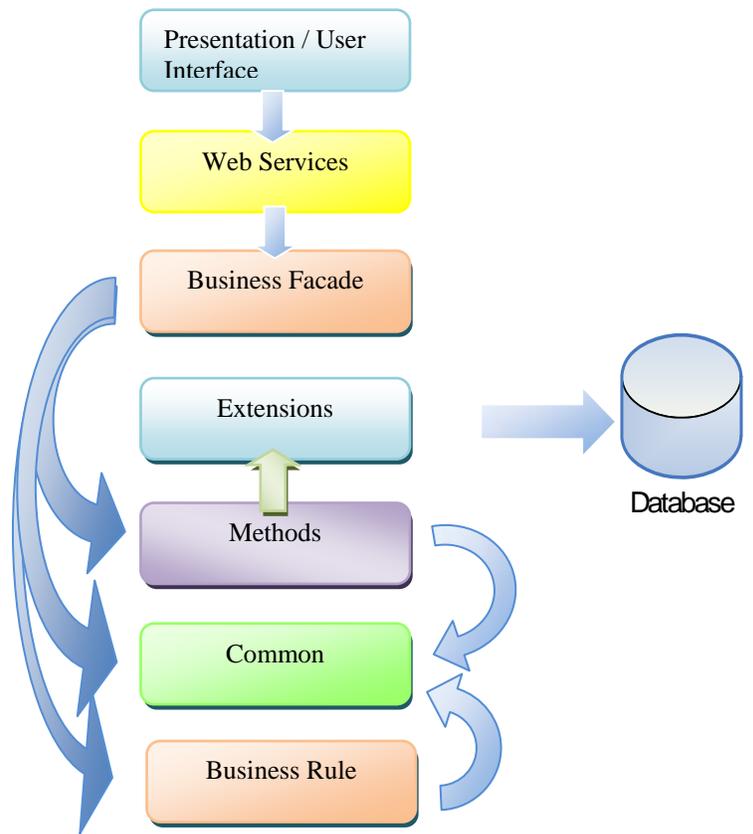
Perancangan arsitektur aplikasi pada sistem Penerimaan Mahasiswa Baru, dilakukan dengan menggunakan pendekatan *Agile Software Development* dan SOA. Berdasarkan pendekatan SOA, arsitektur aplikasi dibangun dengan menggunakan dua langkah, yaitu: Membuat *interface* umum dan Menggunakan *extended schema*. Hal ini dapat dilihat pada Gambar VI.



**Gambar VI Implementasi Pendekatan SOA pada Arsitektur Aplikasi PMB**

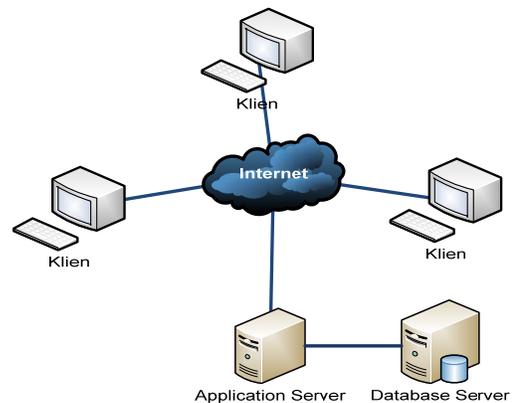
Pendekatan *Agile Software Development* berperan dalam mendukung pendekatan SOA untuk menghasilkan arsitektur aplikasi yang dinamis bagi aplikasi PMB. Arsitektur aplikasi PMB dengan menggunakan *Agile*

*Software Development*, akan berpegang pada prinsip *Open/Closed Principle (OCP)* dan *Single-Responsibility Principle (SRP)*. Model arsitektur untuk aplikasi PMB secara keseluruhan hasil dari pendekatan SOA dan *Agile Software Development* dapat dilihat pada Gambar VII.



**Gambar VII Model Arsitektur untuk Aplikasi PMB**

Untuk kebutuhan arsitektur teknologi untuk sistem Penerimaan Mahasiswa Baru, digunakan arsitektur untuk jaringan *klien-server* dengan memanfaatkan teknologi internet, seperti pada Gambar VIII.



**Gambar VIII Arsitektur Teknologi untuk Sistem Penerimaan Mahasiswa Baru**

e. Subproses *Prioritize and add to Solutions Portfolio*

Setelah semua data untuk kebutuhan analisis terkumpul, maka langkah berikutnya adalah menggabungkan semua data hasil analisis sampai dengan tahap *Architecture Review* menjadi sebuah *Solution Portfolio*. *Solution Portfolio* merupakan rangkuman dari keseluruhan kebutuhan sistem yang telah teranalisa dan akan menjadi panduan bagi langkah pengembangan berikutnya.

2. *Design and Development.*

Pada tahap *design*, *system analyst* akan bekerja bersama *architect* untuk menangani *requirements* sistem informasi baru. *Architect* bertanggung jawab atas hasil rancangan dan menangani anggota tim pengembang sistem informasi. Tim pengembang sistem informasi bertanggung jawab untuk mengembangkan, membuat kode pemrograman, mengecek aplikasi yang dihasilkan agar kualitas sistem informasi terjaga dan menangani penyerahan aplikasi untuk sistem informasi baru yang telah jadi kepada *IT operations*.

a. Subproses *Assign Resources to Solutions Development Team*

Tim pengembang merupakan salah satu aspek penting dalam pengembangan sebuah sistem baru. Setelah memperoleh semua data mengenai kebutuhan untuk pengembangan sistem Penerimaan Mahasiswa Baru, tahap berikutnya adalah melakukan penentuan anggota tim pengembangan sistem yang akan terlibat sesuai dengan kemampuan yang dimiliki masing-masing sehingga diperoleh daftar anggota tim pengembang yang akan terlibat seperti yang dapat dilihat pada Tabel XVI.

**TABEL XVI**  
**Daftar Anggota Tim Pengembangan Sistem Penerimaan Mahasiswa Baru**

No.	Posisi	Nama
1.	<i>Project Manager</i>	Eli Suryani
2.	<i>System Analyst</i>	Robby Wijaya
3.	<i>System Developer</i>	Lisa Agnes Pratiwi
		Rudy Yulianto
4.	<i>User Interface Designer</i>	Herman
5.	<i>Architect</i>	Martinus

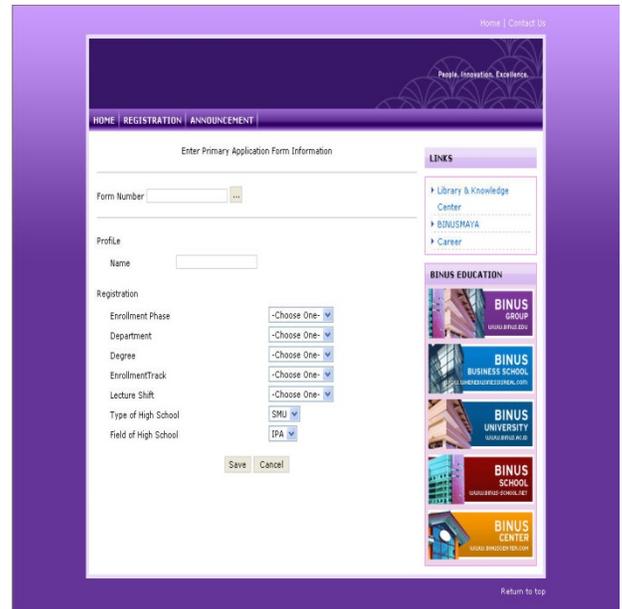
b. Subproses *Design Solutions – Identify Reuse Opportunity*

Rancangan solusi untuk Sistem Informasi Penerimaan Mahasiswa Baru dimulai dengan melakukan rancangan layar untuk masing-masing proses bisnis yang akan tercakup dalam aplikasi untuk Sistem Informasi Penerimaan Mahasiswa Baru, kemudian dilanjutkan dengan melakukan perancangan basis data.

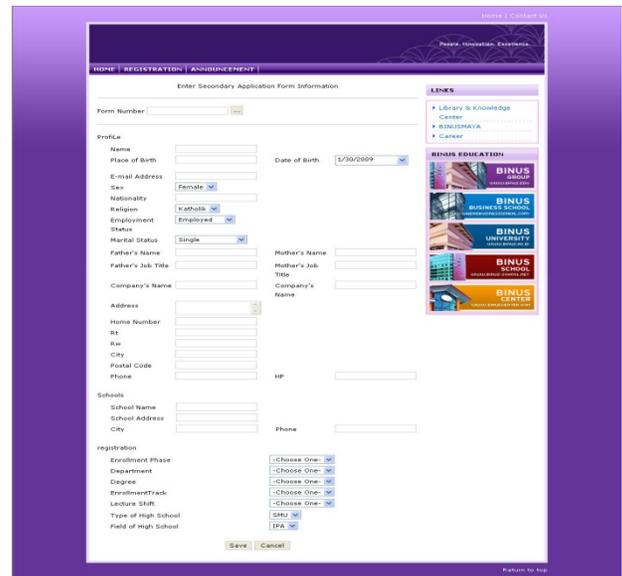
c. Subproses *Develop, QA, and Conduct UAT for Business Solution*

Tahap berikutnya setelah melakukan perancangan adalah menerjemahkan dalam bentuk kode pemrograman semua hasil analisis dan desain untuk pengembangan Sistem Penerimaan Mahasiswa Baru. Contoh tampilan layar untuk

Sistem Informasi Penerimaan Mahasiswa Baru dapat dilihat pada gambar IX dan Gambar X.



**Gambar IX** Tampilan untuk *Enter Primary Application Form Information*



**Gambar X** Tampilan untuk *Enter Secondary Application Form Information*

3. *IT Operations.*

a. Subproses *Assign Resources to Service Operation Team*

Tim implementasi atas solusi *service* yang telah dihasilkan merupakan salah satu aspek penting bagi sebuah sistem baru. Setelah memperoleh solusi *service* baru untuk Sistem Penerimaan Mahasiswa Baru, tahap berikutnya adalah melakukan penentuan anggota tim implementasi atas solusi *service* sistem yang akan terlibat sesuai dengan kemampuan yang dimiliki masing-masing sehingga diperoleh daftar anggota tim pengembang yang akan terlibat seperti yang dapat dilihat pada Tabel XVII.

**TABEL XVII**

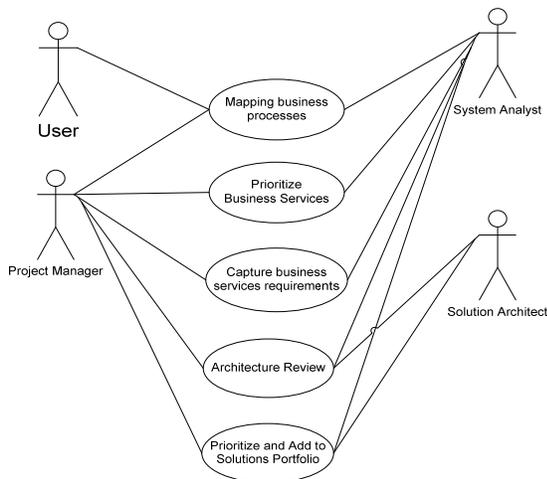
**Daftar Anggota Tim Implementasi Solusi Service Sistem Penerimaan Mahasiswa Baru**

No.	Posisi	Nama
1.	<i>Project Manager</i>	Eli Suryani
2.	<i>System Analyst</i>	Robby Wijaya
3.	<i>Data Services Officer</i>	Choirul Huda
4.	<i>System and Application Services Office</i>	Heribertus Wirono Prakosa

- b. Subproses *Identify Infrastructure Needs and Establish Systems Environment*  
Setelah melakukan pengembangan untuk solusi baru Sistem Penerimaan Mahasiswa Baru dan menentukan daftar nama anggota tim implementasi solusi service baru yang akan terlibat, langkah berikutnya adalah mengidentifikasi kebutuhan lingkungan yang digunakan untuk implementasi sistem.
- c. Subproses *Deploy Business Solution*  
Tahap ini merupakan tahap implementasi sistem baru.

**V. IMPLEMENTASI**

Setelah melakukan perancangan solusi baru metodologi pengembangan sistem informasi dengan menggunakan SOA, maka berikutnya adalah melakukan implementasi atas solusi tersebut. Berikut ini adalah deskripsi tahapan-tahapan pelaksanaan SOA dengan menggunakan diagram *use case* dan deskripsi *use case*, seperti pada Gambar XI.



**Gambar XI Diagram Use Case untuk Requirement and Analysis**

Implementasi untuk *use case Map High-Level Business Processes* dapat dilihat pada Tabel XVIII.

**TABEL XVIII**

**Deskripsi Use Case Map High-Level Business Processes**

<b>Actor</b> :	<i>Project Manager, System Analyst</i>
<b>Precondition</b> :	<i>User</i> menginginkan adanya sebuah sistem informasi yang berbasis service.
<b>Flow of Events</b> :	<p><b>Basic Path</b> :</p> <ol style="list-style-type: none"> <li>1. <i>User</i> menentukan uraian proses bisnis yang hendak dipenuhi oleh sistem informasi yang baru.</li> <li>2. <i>Project manager</i> dan <i>system analyst</i> melakukan pertemuan (wawancara) dengan <i>user</i> untuk mendapatkan deskripsi proses bisnis untuk sistem informasi baru yang akan dibangun.</li> <li>3. <i>Project manager</i> dan <i>system analyst</i> merampungkan uraian proses bisnis yang akan tercakup untuk sistem baru, dan melakukan pemetaan proses bisnis mana yang memiliki kepentingan / <i>level</i> tinggi.</li> </ol> <p><b>Alternative Path</b> :</p> <p><b>Alternative 1</b> : Hasil dari pemetaan terhadap proses bisnis, masih terdapat proses bisnis yang diperlukan untuk sistem informasi yang baru masih belum cukup.</p> <ol style="list-style-type: none"> <li>1. <i>Project manager</i> dan <i>system analyst</i> melakukan pertemuan (wawancara) lagi dengan <i>user</i> untuk mendapatkan deskripsi lebih lanjut mengenai proses bisnis untuk sistem informasi baru yang akan dibangun.</li> <li>2. <i>Project manager</i> dan <i>system analyst</i> melakukan pemetaan terhadap penambahan proses bisnis baru.</li> </ol> <p><b>Alternative 2</b> : Terdapat proses bisnis untuk sistem informasi baru yang lupa <i>user</i> sampaikan dalam wawancara.</p> <ol style="list-style-type: none"> <li>1. <i>User</i> dapat menyampaikan proses bisnis tambahan untuk sistem informasi baru yang lupa disampaikan pada wawancara dengan beberapa cara, yaitu : datang langsung menemui pihak <i>project manager</i> dan <i>system analyst</i> di IT, mengirimkan melalui <i>email</i> atau menyusun kembali jadwal untuk wawancara.</li> <li>2. <i>Project manager</i> dan <i>system analyst</i> melakukan pemetaan terhadap penambahan proses bisnis baru.</li> </ol>
<b>Post Condition</b> :	Rangkuman proses-proses bisnis yang telah terpetakan untuk sistem informasi baru.

Implementasi untuk *use case Prioritize Business Services* dapat dilihat pada Tabel XIX.

**TABEL XIX**

**Deskripsi Use Case Prioritize Business Services**

<b>Actor</b> :	<i>Project Manager dan System Analyst</i>
<b>Precondition</b> :	<i>Project Manager dan System Analyst</i> melakukan pemetaan

<p>proses bisnis mana yang memiliki kepentingan / level tinggi.</p> <p><b>Flow of Events</b> :</p> <p><b>Basic Path</b> :</p> <ol style="list-style-type: none"> <li>1. <i>Project manager</i> dan <i>system analyst</i> mempelajari kembali rangkuman proses-proses bisnis untuk sistem informasi baru yang telah terpetakan.</li> <li>2. <i>Project manager</i> dan <i>system analyst</i> melakukan penentuan urutan prioritas proses-proses bisnis yang akan tercakup dalam sistem informasi baru yang akan dibangun.</li> </ol> <p><b>Alternative Path</b> :</p> <p><b>Alternative 1</b> : Terdapat proses bisnis yang belum lengkap sehingga penentuan urutan prioritas proses bisnis untuk sistem informasi baru tidak dapat dilakukan.</p> <ol style="list-style-type: none"> <li>1. <i>Project manager</i> dan <i>system analyst</i> melakukan pertemuan kembali dengan <i>user</i> untuk mendapatkan deskripsi lebih lanjut mengenai proses bisnis untuk sistem informasi baru yang akan dibangun.</li> </ol> <p><b>Alternative 2</b> : Terdapat proses bisnis untuk sistem informasi baru yang lupa <i>user</i> sampaikan dalam wawancara atau terdapat penambahan proses bisnis baru untuk sistem informasi yang akan dibangun.</p> <ol style="list-style-type: none"> <li>1. <i>User</i> dapat menyampaikan penambahan proses bisnis atau melengkapi uraian proses bisnis dalam sistem informasi baru yang lupa disampaikan pada wawancara dengan beberapa cara, yaitu : datang langsung menemui pihak <i>project manager</i> dan <i>system analyst</i> di IT, mengirimkan melalui <i>email</i> atau menyusun kembali jadwal untuk wawancara.</li> <li>2. <i>Project manager</i> dan <i>system analyst</i> melakukan pemetaan terhadap penambahan proses bisnis baru.</li> <li>3. <i>Project manager</i> dan <i>system analyst</i> mempelajari kembali rangkuman urutan proses bisnis yang telah dihasilkan.</li> <li>4. <i>Project manager</i> dan <i>system analyst</i> menentukan urutan prioritas yang akan diberikan terhadap penambahan proses bisnis baru.</li> <li>5. <i>Project manager</i> dan <i>system analyst</i> merangkum kembali urutan prioritas untuk semua proses bisnis yang akan tercakup dalam sistem informasi baru yang akan dibangun.</li> </ol> <p><b>Post Condition</b> :</p> <p>Rangkuman proses-proses bisnis yang akan ada dalam sistem yang telah diberikan urutan prioritas.</p>
--

Implementasi untuk *use case Capture business services requirements* dapat dilihat pada Tabel XX.

**TABEL XX**  
**Deskripsi Use Case Capture business services requirements**

<b>Actor</b> :	<i>Project Manager</i> dan <i>System Analyst</i>
<b>Precondition</b> :	

<p><i>Project manager</i> dan <i>system analyst</i> melakukan penentuan urutan prioritas proses-proses bisnis yang akan tercakup dalam sistem informasi baru yang akan dibangun.</p> <p><b>Flow of Events</b> :</p> <p><b>Basic Path</b> :</p> <ol style="list-style-type: none"> <li>1. <i>Project manager</i> dan <i>system analyst</i> melakukan pertemuan dengan <i>user</i> untuk mendapatkan deskripsi kebutuhan pada masing-masing proses bisnis untuk sistem informasi baru yang akan dibangun.</li> <li>2. <i>Project manager</i> dan <i>system analyst</i> merampungkan uraian kebutuhan pada masing-masing proses bisnis yang akan tercakup untuk sistem baru.</li> </ol> <p><b>Alternative Path</b> :</p> <p><b>Alternative 1</b> : Terdapat kebutuhan pada proses bisnis tertentu untuk sistem informasi baru yang lupa <i>user</i> sampaikan dalam wawancara.</p> <ol style="list-style-type: none"> <li>1. <i>Project manager</i> dan <i>system analyst</i> melakukan pertemuan (wawancara) lagi dengan <i>user</i> untuk mendapatkan deskripsi lebih lanjut mengenai kebutuhan proses bisnis untuk sistem informasi baru yang akan dibangun.</li> </ol> <p><b>Alternative 2</b> : Terdapat proses bisnis untuk sistem informasi baru yang lupa <i>user</i> sampaikan dalam wawancara atau terdapat penambahan proses bisnis baru untuk sistem informasi yang akan dibangun.</p> <ol style="list-style-type: none"> <li>1. <i>User</i> dapat menyampaikan penambahan proses bisnis atau melengkapi uraian proses bisnis dalam sistem informasi baru yang lupa disampaikan pada wawancara dengan beberapa cara, yaitu : datang langsung menemui pihak <i>project manager</i> dan <i>system analyst</i> di IT, mengirimkan melalui <i>email</i> atau menyusun kembali jadwal untuk wawancara.</li> <li>2. <i>Project manager</i> dan <i>system analyst</i> melakukan pemetaan terhadap penambahan proses bisnis baru.</li> <li>3. <i>Project manager</i> dan <i>system analyst</i> mempelajari kembali rangkuman urutan proses bisnis yang telah dihasilkan.</li> <li>4. <i>Project manager</i> dan <i>system analyst</i> menentukan urutan prioritas yang akan diberikan terhadap penambahan proses bisnis baru.</li> <li>5. <i>Project manager</i> dan <i>system analyst</i> merangkum kembali urutan prioritas untuk semua proses bisnis yang akan tercakup dalam sistem informasi baru yang akan dibangun.</li> <li>6. <i>Project manager</i> dan <i>system analyst</i> melakukan pertemuan dengan <i>user</i> untuk mendapatkan deskripsi kebutuhan pada proses bisnis tambahan untuk sistem informasi baru yang akan dibangun.</li> </ol> <p><b>Post Condition</b> :</p> <p>Rangkuman uraian kebutuhan setiap proses bisnis yang akan dimasukkan dalam <i>business services</i> pada sistem informasi baru.</p>
--

Implementasi untuk *use case Architecture Review* dapat dilihat pada Tabel XXI.

**TABEL XXI**  
**Deskripsi Use Case Architecture Review**

<b>Actor</b> :	<i>Project Manager, System Analyst dan Solution Architect.</i>
<b>Precondition</b> :	<i>Project manager dan system analyst merampungkan uraian kebutuhan pada masing-masing proses bisnis yang akan tercakup untuk sistem baru.</i>
<b>Flow of Events</b> :	
<b>Basic Path</b> :	
	<ol style="list-style-type: none"> <li><i>Project manager, system analyst dan solution architect mempelajari kembali rangkuman uraian kebutuhan setiap proses bisnis yang akan dimasukkan dalam business services pada sistem informasi baru.</i></li> <li><i>Project manager, system analyst dan solution architect melakukan perancangan arsitektur yang akan digunakan untuk pengembangan sistem informasi baru, termasuk di dalamnya business architecture, data architecture, application architecture dan technology architecture.</i></li> </ol>
<b>Alternative Path</b> :	
<b>Alternative 1</b> :	Terdapat proses bisnis untuk sistem informasi baru yang lupa user sampaikan dalam wawancara atau terdapat penambahan proses bisnis baru untuk sistem informasi yang akan dibangun.
	<ol style="list-style-type: none"> <li><i>User dapat menyampaikan penambahan proses bisnis atau melengkapi uraian proses bisnis dalam sistem informasi baru yang lupa disampaikan pada wawancara dengan beberapa cara, yaitu : datang langsung menemui pihak project manager dan system analyst di IT, mengirimkan melalui email atau menyusun kembali jadwal untuk wawancara.</i></li> <li><i>Project manager dan system analyst melakukan pemetaan terhadap penambahan proses bisnis baru.</i></li> <li><i>Project manager dan system analyst mempelajari kembali rangkuman urutan proses bisnis yang telah dihasilkan.</i></li> <li><i>Project manager dan system analyst menentukan urutan prioritas yang akan diberikan terhadap penambahan proses bisnis baru.</i></li> <li><i>Project manager dan system analyst merangkum kembali urutan prioritas untuk semua proses bisnis yang akan tercakup dalam sistem informasi baru yang akan dibangun.</i></li> <li><i>Project manager dan system analyst melakukan pertemuan dengan user untuk mendapatkan deskripsi kebutuhan pada proses bisnis tambahan untuk sistem informasi baru yang akan dibangun.</i></li> <li><i>Project manager, system analyst dan solution architect mempelajari kembali rangkuman uraian kebutuhan setiap proses bisnis yang akan dimasukkan dalam business services pada sistem informasi baru.</i></li> <li><i>Project manager, system analyst dan solution</i></li> </ol>

*architect melakukan perancangan arsitektur yang akan digunakan untuk pengembangan sistem informasi baru, termasuk di dalamnya business architecture, data architecture, application architecture dan technology architecture.*

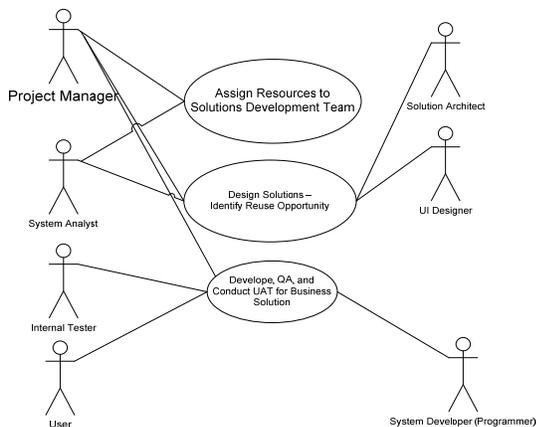
**Post Condition** :  
Rancangan arsitektur yang akan digunakan untuk mengembangkan sistem informasi baru.

Implementasi untuk *use case Prioritize and Add to Solutions Portfolio* dapat dilihat pada Tabel XXII dan Gambar XII.

**TABEL XXII**  
**Deskripsi Use Case Prioritize and Add to Solutions Portfolio**

<b>Actor</b> :	<i>Project Manager, System Analyst dan Solution Architect.</i>
<b>Precondition</b> :	<i>Project manager, system analyst dan solution architect melakukan perancangan arsitektur yang akan digunakan untuk pengembangan sistem informasi baru, termasuk di dalamnya business architecture, data architecture, application architecture dan technology architecture.</i>
<b>Flow of Events</b> :	
<b>Basic Path</b> :	
	<ol style="list-style-type: none"> <li><i>Project manager, system analyst dan solution architect mempelajari kembali rangkuman prioritas dan uraian kebutuhan setiap proses bisnis yang akan dimasukkan dalam business services serta rancangan arsitektur yang akan digunakan pada sistem informasi baru.</i></li> <li><i>Project manager, system analyst dan solution architect memasukkan semua data dan rancangan arsitektur yang diperoleh untuk digabungkan menjadi sebuah solution portfolio.</i></li> <li><i>Project manager menyerahkan solution portfolio yang telah dihasilkan kepada user untuk dilakukan peninjauan kembali dan disetujui.</i></li> </ol>
<b>Alternative Path</b> :	
<b>Alternative 1</b> :	<i>User tidak setuju dengan salah satu bagian yang terdapat dalam solution portfolio.</i>
	<ol style="list-style-type: none"> <li><i>Project manager akan berdiskusi dengan user mengenai kekurangan atau kesalahan yang terdapat dalam solution portfolio.</i></li> <li><i>Project manager dan system analyst akan melakukan ulang tahapan yang hasilnya masih kurang atau salah dalam solution portfolio. Apabila diperlukan, solution architect akan diikutsertakan.</i></li> </ol>
<b>Alternative 2</b> :	Terdapat proses bisnis untuk sistem informasi baru yang lupa user sampaikan dalam wawancara atau terdapat penambahan proses bisnis baru untuk sistem informasi yang akan dibangun.
	<ol style="list-style-type: none"> <li><i>User dapat menyampaikan penambahan proses bisnis atau melengkapi uraian proses bisnis dalam sistem informasi baru yang lupa disampaikan pada wawancara dengan beberapa cara, yaitu : datang</i></li> </ol>

<p>langsung menemui pihak <i>project manager</i> dan <i>system analyst</i> di IT, mengirimkan melalui <i>email</i> atau menyusun kembali jadwal untuk wawancara.</p> <ol style="list-style-type: none"> <li><i>Project manager</i> dan <i>system analyst</i> melakukan pemetaan terhadap penambahan proses bisnis baru.</li> <li><i>Project manager</i> dan <i>system analyst</i> mempelajari kembali rangkuman urutan proses bisnis yang telah dihasilkan.</li> <li><i>Project manager</i> dan <i>system analyst</i> menentukan urutan prioritas yang akan diberikan terhadap penambahan proses bisnis baru.</li> <li><i>Project manager</i> dan <i>system analyst</i> merangkum kembali urutan prioritas untuk semua proses bisnis yang akan tercakup dalam sistem informasi baru yang akan dibangun.</li> <li><i>Project manager</i> dan <i>system analyst</i> melakukan pertemuan dengan <i>user</i> untuk mendapatkan deskripsi kebutuhan pada proses bisnis tambahan untuk sistem informasi baru yang akan dibangun.</li> <li><i>Project manager</i>, <i>system analyst</i> dan <i>solution architect</i> merancang arsitektur untuk sistem informasi baru.</li> <li><i>Project manager</i>, <i>system analyst</i> dan <i>solution architect</i> merancang arsitektur untuk sistem informasi baru.</li> </ol> <p><b>Post Condition</b> :  <i>Solution portfolio</i> yang akan digunakan sebagai panduan dalam mengembangkan sistem informasi baru.</p>
--



**Gambar XII Diagram Use Case untuk Design and Development**

Implementasi untuk use case *Assign Resources to Solutions Development Team* dapat dilihat pada Tabel XXII.

**TABEL XXII**  
**Deskripsi Use Case Assign Resources to Solutions Development Team**

<p><b>Actor</b> :  <i>Project Manager</i> dan <i>System Analyst</i>.</p>
<p><b>Precondition</b> :  <i>Project manager</i>, <i>system analyst</i> dan <i>solution architect</i> memasukkan semua data dan rancangan arsitektur yang diperoleh untuk digabungkan menjadi sebuah <i>solution</i></p>

<p><i>portfolio</i>.</p> <p><b>Flow of Events</b> :</p> <p><b>Basic Path</b> :</p> <ol style="list-style-type: none"> <li><i>Project manager</i> mempelajari kembali rangkuman prioritas dan uraian kebutuhan untuk sistem informasi baru yang terdapat dalam <i>solution portfolio</i>.</li> <li>Berdasarkan <i>solution portfolio</i> mengenai sistem informasi baru yang akan dibangun, <i>project manager</i> dan <i>system analyst</i> menentukan anggota tim pengembangan sistem yang akan terlibat.</li> </ol> <p><b>Alternative Path</b> :</p> <p><b>Alternative 1</b> : Terdapat proses bisnis untuk sistem informasi baru yang lupa <i>user</i> sampaikan dalam wawancara atau terdapat penambahan proses bisnis baru untuk sistem informasi yang akan dibangun.</p> <ol style="list-style-type: none"> <li><i>User</i> dapat menyampaikan penambahan proses bisnis atau melengkapi uraian proses bisnis dalam sistem informasi baru yang lupa disampaikan pada wawancara dengan beberapa cara, yaitu : datang langsung menemui pihak <i>project manager</i> dan <i>system analyst</i> di IT, mengirimkan melalui <i>email</i> atau menyusun kembali jadwal untuk wawancara.</li> <li><i>Project manager</i> dan <i>system analyst</i> melakukan pemetaan terhadap penambahan proses bisnis baru.</li> <li><i>Project manager</i> dan <i>system analyst</i> mempelajari kembali rangkuman urutan proses bisnis yang telah dihasilkan.</li> <li><i>Project manager</i> dan <i>system analyst</i> menentukan urutan prioritas yang akan diberikan terhadap penambahan proses bisnis baru.</li> <li><i>Project manager</i> dan <i>system analyst</i> merangkum kembali urutan prioritas untuk semua proses bisnis yang akan tercakup dalam sistem informasi baru yang akan dibangun.</li> <li><i>Project manager</i> dan <i>system analyst</i> melakukan pertemuan dengan <i>user</i> untuk mendapatkan deskripsi kebutuhan pada proses bisnis tambahan untuk sistem informasi baru yang akan dibangun.</li> <li><i>Project manager</i>, <i>system analyst</i> dan <i>solution architect</i> merancang arsitektur untuk sistem informasi baru.</li> <li><i>Project manager</i>, <i>system analyst</i> dan <i>solution architect</i> mempelajari kembali rangkuman prioritas dan uraian kebutuhan setiap proses bisnis yang akan dimasukkan dalam <i>business services</i> serta rancangan arsitektur yang akan digunakan pada sistem informasi baru.</li> <li><i>Project manager</i>, <i>system analyst</i> dan <i>solution architect</i> memasukkan semua data dan rancangan arsitektur yang diperoleh untuk digabungkan menjadi sebuah <i>solution portfolio</i>.</li> <li>Berdasarkan <i>solution portfolio</i> mengenai sistem informasi baru yang akan dibangun, <i>project manager</i> dan <i>system analyst</i> menentukan anggota tim pengembangan sistem yang akan terlibat.</li> </ol> <p><b>Post Condition</b> :</p>
--

Daftar anggota tim pengembang sistem informasi yang akan terlibat.

Implementasi untuk *use case Design Solutions – Identify Reuse Opportunity* dapat dilihat pada Tabel XXIV.

**TABEL XXIV**

**Deskripsi Use Case Design Solutions – Identify Reuse Opportunity**

<b>Actor</b> :	<i>Project Manager, System Analyst, Solution Architect dan User Interface (UI) Designer.</i>
<b>Precondition</b> :	Berdasarkan <i>solution portfolio</i> mengenai sistem informasi baru yang akan dibangun, <i>project manager</i> dan <i>system analyst</i> menentukan anggota tim pengembangan sistem yang akan terlibat.
<b>Flow of Events</b> :	<p><b>Basic Path</b> :</p> <ol style="list-style-type: none"> <li><i>Project manager, system analyst</i> dan <i>solution architect</i> melakukan perancangan solusi yang akan digunakan pada sistem informasi baru, yaitu meliputi perancangan basis data, perancangan layar antarmuka pengguna.</li> <li><i>Project manager</i> menyerahkan rancangan layar antarmuka pengguna yang telah dihasilkan kepada <i>user</i> untuk dilakukan peninjauan kembali dan disetujui.</li> <li><i>Project manager</i> menyerahkan rancangan layar antarmuka pengguna yang telah dibuat untuk dikerjakan oleh <i>UI Designer</i>.</li> </ol> <p><b>Alternative Path</b> :</p> <p><b>Alternative 1</b> : <i>User</i> tidak setuju dengan rancangan layar antarmuka pengguna yang diajukan.</p> <ol style="list-style-type: none"> <li><i>Project manager</i> akan berdiskusi dengan <i>user</i> mengenai kekurangan atau kesalahan yang terdapat dalam rancangan layar antarmuka pengguna.</li> <li><i>Project manager</i> dan <i>system analyst</i> akan melakukan penambahan dalam rancangan solusi yang diajukan sesuai dengan masukan dari <i>user</i>.</li> </ol> <p><b>Alternative 2</b> : Terdapat proses bisnis untuk sistem informasi baru yang lupa <i>user</i> sampaikan dalam wawancara atau terdapat penambahan proses bisnis baru untuk sistem informasi yang akan dibangun.</p> <ol style="list-style-type: none"> <li><i>User</i> dapat menyampaikan penambahan proses bisnis atau melengkapi uraian proses bisnis dalam sistem informasi baru yang lupa disampaikan pada wawancara dengan beberapa cara, yaitu : datang langsung menemui pihak <i>project manager</i> dan <i>system analyst</i> di IT, mengirimkan melalui <i>email</i> atau menyusun kembali jadwal untuk wawancara.</li> <li><i>Project manager</i> dan <i>system analyst</i> melakukan pemetaan terhadap penambahan proses bisnis baru.</li> <li><i>Project manager</i> dan <i>system analyst</i> mempelajari kembali rangkuman urutan proses bisnis yang telah dihasilkan.</li> </ol>

- Project manager* dan *system analyst* menentukan urutan prioritas yang akan diberikan terhadap penambahan proses bisnis baru.
- Project manager* dan *system analyst* merangkum kembali urutan prioritas untuk semua proses bisnis yang akan tercakup dalam sistem informasi baru yang akan dibangun.
- Project manager* dan *system analyst* melakukan pertemuan dengan *user* untuk mendapatkan deskripsi kebutuhan pada proses bisnis tambahan untuk sistem informasi baru yang akan dibangun.
- Project manager, system analyst* dan *solution architect* merancang arsitektur untuk sistem informasi baru.
- Project manager, system analyst* dan *solution architect* mempelajari kembali rangkuman prioritas dan uraian kebutuhan setiap proses bisnis yang akan dimasukkan dalam *business services* serta rancangan arsitektur yang akan digunakan pada sistem informasi baru.
- Project manager, system analyst* dan *solution architect* memasukkan semua data dan rancangan arsitektur yang diperoleh untuk digabungkan menjadi sebuah *solution portfolio*.
- Berdasarkan *solution portfolio* mengenai sistem informasi baru yang akan dibangun, *project manager* dan *system analyst* menentukan anggota tim pengembangan sistem yang akan terlibat.
- Project manager, system analyst* dan *solution architect* melakukan perancangan solusi yang akan digunakan pada sistem informasi baru, yaitu meliputi perancangan basis data, perancangan layar antarmuka pengguna.
- Project manager* menyerahkan rancangan layar antarmuka pengguna yang telah dihasilkan kepada *user* untuk dilakukan peninjauan kembali dan disetujui.
- Project manager* menyerahkan rancangan layar antarmuka pengguna yang telah dibuat untuk dikerjakan oleh *UI Designer*.

**Post Condition** :

Rancangan solusi yang akan digunakan pada sistem informasi baru, yaitu meliputi perancangan basis data dan perancangan layar antarmuka pengguna.

Implementasi untuk *use case Develop, QA, and Conduct UAT for Business Solution* dapat dilihat pada Tabel XXV dan Gambar XIII.

**TABEL XXV**

**Deskripsi Use Case Develop, QA, and Conduct UAT for Business Solution**

<b>Actor</b> :	<i>Project Manager, System Developer (Programmer), Internal Tester dan User.</i>
<b>Precondition</b> :	<i>Project manager, system analyst</i> dan <i>solution architect</i> melakukan perancangan solusi yang akan digunakan pada sistem informasi baru, yaitu meliputi perancangan basis data, perancangan layar antarmuka pengguna..
<b>Flow of Events</b> :	<p><b>Basic Path</b> :</p> <ol style="list-style-type: none"> <li><i>Project manager</i> menyerahkan rancangan solusi</li> </ol>

sistem informasi baru yang telah dihasilkan dan disetujui oleh *user* kepada *system developer (programmer)* untuk dikembangkan lebih lanjut dan diterjemahkan ke dalam bahasa pemrograman sehingga menghasilkan sebuah sistem informasi baru.

2. *System developer (programmer)* melakukan *n-unit testing* saat dalam tahap pengembangan terhadap setiap modul yang telah diselesaikan sehingga sistem informasi yang dihasilkan memiliki ketepatan proses terhadap *input* yang dimasukkan dan *output* yang dihasilkan.
3. *System developer (programmer)* menyerahkan sistem informasi yang telah selesai dikembangkan kepada *project manager* dan *project manager* memberikan kepada *internal tester* untuk dilakukan pengecekan terhadap sistem informasi tersebut apakah telah sesuai dengan permintaan pemenuhan kebutuhan *user*.
4. *Project manager* menyerahkan sistem informasi yang telah selesai dilakukan pengecekan oleh *internal tester* kepada *user* untuk dilakukan pengecekan akhir pada sistem informasi.

*Alternative Path* :

*Alternative 1* : Terdapat kesalahan yang ditemukan oleh *internal tester* pada satu atau beberapa modul dalam sistem informasi saat dilakukan pengecekan.

1. *Internal tester* akan memberitahukan kepada *project manager* dan *system developer (programmer)* tentang bagian modul sistem informasi yang masih perlu untuk diperbaiki.
2. *System developer (programmer)* akan memperbaiki kesalahan yang terdapat pada modul yang bersangkutan dan memberikan kembali *project manager*.
3. *Project manager* memberikan kepada *internal tester* untuk dilakukan pengecekan terhadap perbaikan yang telah dilakukan *system developer (programmer)* apakah telah sesuai dengan permintaan pemenuhan kebutuhan *user*.
4. *Project manager* menyerahkan sistem informasi yang telah selesai dilakukan pengecekan oleh *internal tester* kepada *user* untuk dilakukan pengecekan akhir pada sistem informasi.

*Alternative 2* : *User* tidak setuju dengan salah satu bagian atau salah satu modul yang terdapat dalam sistem informasi baru yang telah selesai dikembangkan.

1. *Project manager* akan berdiskusi dengan *user* mengenai kekurangan atau kesalahan yang terdapat dalam sistem informasi.
2. *Project manager* akan memberitahukan kepada *system developer (programmer)* mengenai kekurangan atau kesalahan yang dikeluhkan oleh *user*.
3. *System developer (programmer)* akan memperbaiki kesalahan yang terdapat pada modul yang bersangkutan dan memberikan kembali *project manager*.
4. *Project manager* memberikan kepada *internal tester* untuk dilakukan pengecekan terhadap perbaikan yang telah dilakukan *system developer (programmer)*.

5. *Project manager* menyerahkan sistem informasi yang telah selesai dilakukan pengecekan oleh *internal tester* kepada *user*.

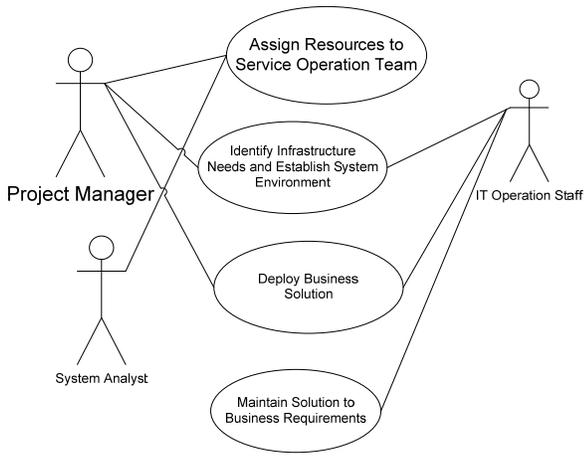
*Alternative 3* : Terdapat proses bisnis untuk sistem informasi baru yang lupa *user* sampaikan dalam wawancara atau terdapat penambahan proses bisnis baru untuk sistem informasi yang akan dibangun.

1. *User* dapat menyampaikan penambahan proses bisnis atau melengkapi uraian proses bisnis dalam sistem informasi baru yang lupa disampaikan pada wawancara dengan beberapa cara, yaitu : datang langsung menemui pihak *project manager* dan *system analyst* di IT, mengirimkan melalui *email* atau menyusun kembali jadwal untuk wawancara.
2. *Project manager* dan *system analyst* melakukan pemetaan terhadap penambahan proses bisnis baru.
3. *Project manager* dan *system analyst* mempelajari kembali rangkuman urutan proses bisnis yang telah dihasilkan.
4. *Project manager* dan *system analyst* menentukan urutan prioritas yang akan diberikan terhadap penambahan proses bisnis baru.
5. *Project manager* dan *system analyst* merangkum kembali urutan prioritas untuk semua proses bisnis yang akan tercakup dalam sistem informasi baru yang akan dibangun.
6. *Project manager* dan *system analyst* melakukan pertemuan dengan *user* untuk mendapatkan deskripsi kebutuhan pada proses bisnis tambahan untuk sistem informasi baru yang akan dibangun.
7. *Project manager, system analyst* dan *solution architect* merancang arsitektur untuk sistem informasi baru.
8. *Project manager, system analyst* dan *solution architect* mempelajari kembali rangkuman prioritas dan uraian kebutuhan setiap proses bisnis yang akan dimasukkan dalam *business services* serta rancangan arsitektur yang akan digunakan pada sistem informasi baru.
9. *Project manager, system analyst* dan *solution architect* memasukkan semua data dan rancangan arsitektur yang diperoleh untuk digabungkan menjadi sebuah *solution portfolio*.
10. Berdasarkan *solution portfolio* mengenai sistem informasi baru yang akan dibangun, *project manager* dan *system analyst* menentukan anggota tim pengembangan sistem yang akan terlibat.
11. *Project manager, system analyst* dan *solution architect* melakukan perancangan solusi yang akan digunakan pada sistem informasi baru, yaitu meliputi perancangan basis data, perancangan layar antarmuka pengguna.
12. *Project manager* menyerahkan rancangan layar antarmuka pengguna yang telah dihasilkan kepada *user* untuk dilakukan peninjauan kembali dan disetujui.
13. *Project manager* menyerahkan rancangan layar antarmuka pengguna yang telah dibuat untuk dikerjakan oleh *UI Designer*.
14. *Project manager* menyerahkan rancangan solusi sistem informasi baru yang telah dihasilkan dan disetujui oleh *user* kepada *system developer (programmer)* untuk

dikembangkan lebih lanjut dan diterjemahkan ke dalam bahasa pemrograman sehingga menghasilkan sebuah sistem informasi baru.

15. *System developer (programmer)* melakukan *n-unit testing* saat dalam tahap pengembangan terhadap setiap modul yang telah diselesaikan sehingga sistem informasi yang dihasilkan memiliki ketepatan proses terhadap *input* yang dimasukkan dan *output* yang dihasilkan.
16. *System developer (programmer)* menyerahkan sistem informasi yang telah selesai dikembangkan kepada *project manager* dan *project manager* memberikan kepada *internal tester* untuk dilakukan pengecekan terhadap sistem informasi tersebut apakah telah sesuai dengan permintaan pemenuhan kebutuhan *user*.
17. *Project manager* menyerahkan sistem informasi yang telah selesai dilakukan pengecekan oleh *internal tester* kepada *user* untuk dilakukan pengecekan akhir pada sistem informasi.

**Post Condition :**  
Sebuah sistem informasi sebagai solusi bisnis baru yang telah jadi dan siap diimplementasikan.



**Gambar XIII Diagram Use Case untuk IT Operations**

Implementasi untuk *use case Assign Resources to Service Operation Team* dapat dilihat pada Tabel XXVI.

**TABEL XXVI**

**Deskripsi Use Case Assign Resources to Service Operation Team**

<b>Actor</b> :	<i>Project Manager</i> dan <i>System Analyst</i> .
<b>Precondition</b> :	<i>Project manager</i> , <i>system analyst</i> dan <i>solution architect</i> memasukkan semua data dan rancangan arsitektur yang diperoleh untuk digabungkan menjadi sebuah <i>solution portfolio</i> .
<b>Flow of Events</b> :	
<b>Basic Path</b> :	
	<ol style="list-style-type: none"> <li>1. <i>Project manager</i> mempelajari kembali rangkuman prioritas dan uraian kebutuhan untuk sistem informasi baru yang terdapat dalam <i>solution portfolio</i>.</li> <li>2. Berdasarkan <i>solution portfolio</i> mengenai sistem</li> </ol>

informasi baru yang akan dibangun, *project manager* dan *system analyst* menentukan anggota tim IT *Operations* untuk melakukan implementasi dan *maintenance* sistem informasi baru yang akan terlibat.

**Alternative Path :**  
**Alternative 1 :** Terdapat proses bisnis untuk sistem informasi baru yang lupa *user* sampaikan dalam wawancara atau terdapat penambahan proses bisnis baru untuk sistem informasi yang akan dibangun.

1. *User* dapat menyampaikan penambahan proses bisnis atau melengkapi uraian proses bisnis dalam sistem informasi baru yang lupa disampaikan pada wawancara dengan beberapa cara, yaitu : datang langsung menemui pihak *project manager* dan *system analyst* di IT, mengirimkan melalui *email* atau menyusun kembali jadwal untuk wawancara.
2. *Project manager* dan *system analyst* melakukan pemetaan terhadap penambahan proses bisnis baru.
3. *Project manager* dan *system analyst* mempelajari kembali rangkuman urutan proses bisnis yang telah dihasilkan.
4. *Project manager* dan *system analyst* menentukan urutan prioritas yang akan diberikan terhadap penambahan proses bisnis baru.
5. *Project manager* dan *system analyst* merangkum kembali urutan prioritas untuk semua proses bisnis yang akan tercakup dalam sistem informasi baru yang akan dibangun.
6. *Project manager* dan *system analyst* melakukan pertemuan dengan *user* untuk mendapatkan deskripsi kebutuhan pada proses bisnis tambahan untuk sistem informasi baru yang akan dibangun.
7. *Project manager*, *system analyst* dan *solution architect* merancang arsitektur untuk sistem informasi baru.
8. *Project manager*, *system analyst* dan *solution architect* mempelajari kembali rangkuman prioritas dan uraian kebutuhan setiap proses bisnis yang akan dimasukkan dalam *business services* serta rancangan arsitektur yang akan digunakan pada sistem informasi baru.
9. *Project manager*, *system analyst* dan *solution architect* memasukkan semua data dan rancangan arsitektur yang diperoleh untuk digabungkan menjadi sebuah *solution portfolio*.
10. Berdasarkan *solution portfolio* mengenai sistem informasi baru yang akan dibangun, *project manager* dan *system analyst* menentukan anggota tim IT *Operations* yang akan terlibat.

**Post Condition :**  
Daftar anggota tim IT *Operations* untuk melakukan implementasi dan *maintenance* sistem informasi baru yang akan terlibat.

Implementasi untuk *use case Identify Infrastructure Needs and Establish Systems Environment* dapat dilihat pada Tabel XXVII.

**TABEL XXVII**  
**Deskripsi Use case Identify Infrastructure Needs and Establish Systems Environment**

<b>Actor</b> :
<i>Project Manager</i> dan <i>IT Operations Staff</i> .
<b>Precondition</b> :
<i>Project manager, system analyst</i> dan <i>solution architect</i> memasukkan semua data dan rancangan arsitektur yang diperoleh untuk digabungkan menjadi sebuah <i>solution portfolio</i> .
<b>Flow of Events</b> :
<b>Basic Path</b> :
<ol style="list-style-type: none"> <li><i>Project manager</i> dan <i>IT Operations Staff</i> mengidentifikasi kebutuhan infrastruktur untuk lingkungan implementasi sistem informasi baru sesuai dengan kebutuhan yang terdapat untuk sistem dalam <i>solution portfolio</i>.</li> <li><i>Project manager</i> dan <i>IT Operations Staff</i> mempersiapkan dan membangun lingkungan operasional untuk implementasi sistem informasi baru.</li> </ol>
<b>Alternative Path</b> :
<b>Alternative 1</b> : Terdapat proses bisnis untuk sistem informasi baru yang lupa <i>user</i> sampaikan dalam wawancara atau terdapat penambahan proses bisnis baru untuk sistem informasi yang akan dibangun.
<ol style="list-style-type: none"> <li><i>User</i> dapat menyampaikan penambahan proses bisnis atau melengkapi uraian proses bisnis dalam sistem informasi baru yang lupa disampaikan pada wawancara dengan beberapa cara, yaitu : datang langsung menemui pihak <i>project manager</i> dan <i>system analyst</i> di IT, mengirimkan melalui <i>email</i> atau menyusun kembali jadwal untuk wawancara.</li> <li><i>Project manager</i> dan <i>system analyst</i> melakukan pemetaan terhadap penambahan proses bisnis baru.</li> <li><i>Project manager</i> dan <i>system analyst</i> mempelajari kembali rangkuman urutan proses bisnis yang telah dihasilkan.</li> <li><i>Project manager</i> dan <i>system analyst</i> menentukan urutan prioritas yang akan diberikan terhadap penambahan proses bisnis baru.</li> <li><i>Project manager</i> dan <i>system analyst</i> merangkum kembali urutan prioritas untuk semua proses bisnis yang akan tercakup dalam sistem informasi baru yang akan dibangun.</li> <li><i>Project manager</i> dan <i>system analyst</i> melakukan pertemuan dengan <i>user</i> untuk mendapatkan deskripsi kebutuhan pada proses bisnis tambahan untuk sistem informasi baru yang akan dibangun.</li> <li><i>Project manager, system analyst</i> dan <i>solution architect</i> merancang arsitektur untuk sistem informasi baru.</li> <li><i>Project manager, system analyst</i> dan <i>solution architect</i> mempelajari kembali rangkuman prioritas dan uraian kebutuhan setiap proses bisnis yang akan dimasukkan dalam <i>business services</i> serta rancangan arsitektur yang</li> </ol>

<p>akan digunakan pada sistem informasi baru.</p> <ol style="list-style-type: none"> <li><i>Project manager, system analyst</i> dan <i>solution architect</i> memasukkan semua data dan rancangan arsitektur yang diperoleh untuk digabungkan menjadi sebuah <i>solution portfolio</i>.</li> <li>Berdasarkan <i>solution portfolio</i> mengenai sistem informasi baru yang akan dibangun, <i>project manager</i> dan <i>system analyst</i> menentukan anggota tim <i>IT Operations</i> yang akan terlibat.</li> <li><i>Project manager</i> dan <i>IT Operations Staff</i> mengidentifikasi kebutuhan infrastruktur untuk lingkungan implementasi sistem informasi baru sesuai dengan kebutuhan yang terdapat untuk sistem dalam <i>solution portfolio</i>.</li> <li><i>Project manager</i> dan <i>IT Operations Staff</i> mempersiapkan dan membangun lingkungan operasional untuk implementasi sistem informasi baru.</li> </ol>
<b>Post Condition</b> :
Infrastruktur dan lingkungan untuk implementasi sistem informasi baru yang siap digunakan.

Implementasi untuk *use case Deploy Business Solution* dapat dilihat pada Tabel XXVIII.

**TABEL XXVIII**  
**Deskripsi Use case Deploy Business Solution**

<b>Actor</b> :
<i>Project Manager</i> dan <i>IT Operations Staff</i> .
<b>Precondition</b> :
Berdasarkan <i>solution portfolio</i> mengenai sistem informasi baru yang akan dibangun, <i>project manager</i> dan <i>system analyst</i> menentukan anggota tim <i>IT Operations</i> untuk melakukan implementasi dan <i>maintenance</i> sistem informasi baru yang akan terlibat. <i>Project manager</i> dan <i>IT Operations Staff</i> mempersiapkan dan membangun lingkungan operasional untuk implementasi sistem informasi baru. <i>System developer (programmer)</i> selesai mengembangkan sistem informasi untuk solusi baru dan dilakukan pengecekan oleh <i>internal tester</i> , dan <i>user</i> sehingga sistem informasi tersebut benar-benar telah sesuai dengan keinginan <i>user</i> .
<b>Flow of Events</b> :
<b>Basic Path</b> :
<ol style="list-style-type: none"> <li><i>Project manager</i> dan <i>IT Operations Staff</i> melakukan implementasi terhadap sistem informasi baru pada lingkungan operasional yang telah disediakan.</li> </ol>
<b>Post Condition</b> :
Sebuah sistem informasi baru yang telah selesai diimplementasikan pada lingkungan operasional yang telah disediakan dan siap untuk digunakan.

Implementasi untuk *use case Maintain Solution to Business Requirements* dapat dilihat pada Tabel XXIX.

**TABEL XXIX**  
**Deskripsi Use Case Maintain Solution to Business Requirements**

<b>Actor</b> :
<i>IT Operations Staff</i> .

<p><b>Precondition</b> :</p> <p><i>Project manager</i> dan <i>IT Operations Staff</i> melakukan implementasi terhadap sistem informasi baru pada lingkungan operasional yang telah disediakan.</p>
<p><b>Flow of Events</b> :</p> <p><b>Basic Path</b> :</p> <p>1. <i>IT Operations Staff</i> melakukan perawatan (maintenance) terhadap sistem informasi yang telah diimplementasikan sehingga keberadaan sistem informasi dapat selalu terjaga untuk memenuhi kebutuhan penggunaan <i>user</i>.</p> <p><b>Alternative Path</b> :</p> <p><b>Alternative 1</b> : Terdapat permintaan dari <i>user</i> atas sistem informasi baru atau penambahan proses bisnis baru pada sistem informasi yang sudah berjalan.</p> <p>1. <i>User</i> menentukan uraian proses bisnis baru untuk sistem informasi yang sedang berjalan atau untuk mengembangkan sistem informasi baru yang hendak dipenuhi.</p> <p>2. <i>Project manager</i> dan <i>system analyst</i> melakukan pertemuan (wawancara) dengan <i>user</i> untuk mendapatkan deskripsi proses bisnis.</p> <p>3. <i>Project manager</i> dan <i>system analyst</i> merampungkan uraian proses bisnis yang akan tercakup untuk sistem baru, dan melakukan pemetaan proses bisnis mana yang memiliki kepentingan / level tinggi.</p>
<p><b>Post Condition</b> :</p> <p>Sistem informasi yang kondisinya selalu terjaga sehingga pemenuhan kebutuhan <i>user</i> dapat berjalan dengan baik dan lancar.</p>

pada BINUS *Online Learning*, yaitu untuk komponen informasi yang tidak diperlukan, maka dapat dihilangkan. Untuk perancangan arsitektur kode pengembangan aplikasi, pada tahapan ini juga dilakukan adopsi langsung terhadap arsitektur kode yang telah ada sebelumnya untuk Sistem Informasi Penerimaan Mahasiswa Baru BINUS *University S1* regular.

Kemudahan arsitektur sistem informasi berbasis SOA dalam mengadopsi perubahan proses bisnis yang terjadi akan sangat mempermudah kegiatan pengembangan sebuah sistem informasi salah satunya dikarenakan oleh tingkat *reuseability* yang tinggi pada modul-modul yang ada. Rangkuman perbandingan tahapan pengembangan sistem informasi antara metodologi yang digunakan sekarang dengan SOA, dapat dilihat pada Tabel XXX.

**TABEL XXX**  
**Tabel Perbandingan Pengembangan Sistem Informasi dengan Menggunakan Metodologi SOA dan Metodologi yang Sekarang**

No.	Nama Proses	Lama Waktu yang Dibutuhkan	
		Metodologi Saat Ini	SOA
1.	Requirement and Analysis	10-15 hari	5 hari
2.	Development	60-100 hari	20 hari - Modul <i>Asses Scholarship</i> = 4 hari - Modul <i>Application Form Purchase</i> = 6 - Modul <i>Registration</i> = 10 hari

## VI. EVALUASI

Setelah dilakukan implementasi metodologi pengembangan sistem informasi Penerimaan Mahasiswa Baru untuk Bina Nusantara *University S1* Regular dengan menggunakan kombinasi SOA dan *Agile Software Development*, maka dilakukan uji coba untuk menerapkan kombinasi metodologi yang sama untuk pengembangan sistem informasi Penerimaan Mahasiswa Baru untuk Bina Nusantara *University Online Learning*. Hal ini dimaksudkan untuk mendapatkan hasil evaluasi kemampuan metodologi baru dalam menghadapi perbedaan proses bisnis sistem informasi.

Untuk pengembangan Sistem Penerimaan Mahasiswa Baru BINUS *Online Learning*, segala tahapan SOA yang berhubungan dengan proses bisnis dilakukan dari awal, mulai dari *Mapping High-Level Business Processes* sampai dengan *Assign Resources to Solutions Development Team*. Namun, untuk tahapan perancangan desain solusi, tahapan ini menjadi sangat mudah karena terdapat tahapan yang dapat dilewati, yaitu tahapan untuk perancangan *database*. Sedangkan untuk perancangan layar, tahapan ini dapat menggunakan rancangan layar yang telah ada sebelumnya yaitu pada Sistem Informasi Penerimaan Mahasiswa Baru BINUS *University S1* regular, hanya perlu dilakukan sedikit penyesuaian dengan kebutuhan

Untuk proses perancangan antarmuka aplikasi Penerimaan Mahasiswa Baru BINUS *Online Learning*, waktu yang digunakan tidak sampai dengan satu hari, karena dapat menggunakan rancangan yang ada dari aplikasi Penerimaan Mahasiswa Baru BINUS Regular. Sedangkan untuk perancangan *database* aplikasi Penerimaan Mahasiswa Baru BINUS *Online Learning* menggunakan perancangan *database* yang telah dihasilkan untuk aplikasi Penerimaan Mahasiswa Baru BINUS Regular karena *database* tersebut telah dirancang berbasis *service* yang akan terjadi dalam proses penerimaan mahasiswa baru.

## VII. SIMPULAN DAN SARAN

### A. Simpulan

Berdasarkan hasil penelitian dan evaluasi yang dilakukan terhadap kemampuan metodologi pengembangan sistem informasi dalam mengadopsi perubahan proses bisnis untuk BINUS *Online Learning*, maka dapat disimpulkan bahwa:

1. Metodologi baru pengembangan sistem informasi sebagai solusi yang ditawarkan mempercepat proses pengembangan sistem informasi.
  2. Modul-modul yang dikembangkan sebelumnya pada sistem informasi lain, dapat digunakan kembali untuk sistem informasi yang baru. Dengan berbasiskan pada service yang akan dibangun untuk sebuah sistem informasi, modul-modul pada aplikasi yang dikembangkan dengan SOA memiliki tingkat *reusability* yang tinggi.
  3. *Database* yang dirancang dengan berbasiskan SOA memiliki kemampuan untuk menyimpan data dengan akurat dan dapat digunakan kembali untuk sistem informasi lain yang memiliki *service* sistem yang sama sehingga waktu perancangan database akan menjadi sangat singkat.
- [7] Hasan, Jeffrey. *Expert Service-Oriented Architecture in C# Using the Web Services Enhancements 2.0*. Appress, New York.
  - [8] Turban E., Rainer, R. K. Jr., Potter, R. E. (2006). *Pengantar Teknologi Informasi*. Edisi 3. Penerbit Salemba Infotek, Jakarta.
  - [9] Whitten, J. L., Bentley, L. D., Dittman, K. C. (2004). *Systems Analysis and Design Methods*. 6<sup>th</sup> Edition. McGraw-Hill, New York.
  - [10] Durvasula, S. (2006). *SOA Practitioner's Guide Part 3, Introduction to Services Lifecycle*.
  - [11] Robert, M.C, Micah, M. (2006). *Agile Principles, Patterns, and Practices in C#*. Prentice Hall PTR, Indiana.
  - [12] Rumbaugh, J., Jacobson, I., Booch, G. (1999). *The Unified Modeling Language Reference Manual*. Addison-Wesley, Massachusetts

#### B. Saran

Adapun saran yang dapat diberikan dari hasil penelitian yang telah dilakukan untuk penelitian hal sama di masa yang akan datang adalah:

1. Perlu adanya memperhatikan aspek keamanan dalam sebuah sistem informasi, maka sistem informasi yang dikembangkan dengan berbasiskan SOA harus sangat memperhatikan aspek keamanan pada *services* yang dibuat. Hal ini dikarenakan *services* yang dibuat dapat dikonsumsi oleh beberapa pihak, maka untuk menghindari terjadinya penggunaan sebuah *service* oleh akses yang tidak diperbolehkan, maka pada *service* tersebut perlu diberikan segi keamanan yang menjaga agar *service* tersebut hanya diakses oleh pihak yang memiliki otorisasi.
2. Perlunya sumber daya manusia yang memiliki pengetahuan dan pengalaman yang dalam mengenai SOA dan *Agile Software Development* apabila hendak merancang pengembangan sistem informasi dengan menggunakan metodologi ini, karena pemahaman yang salah akan menyebabkan jumlah *service* yang terbentuk berlebihan dan memiliki tingkat *reusability* yang rendah.

#### REFERENCES

- [1] Service-oriented architecture (SOA) definition, [http://www.service-architecture.com/web\\_services/articles/service-oriented\\_architecture\\_soa\\_definition.html](http://www.service-architecture.com/web_services/articles/service-oriented_architecture_soa_definition.html)
- [2] Service, [http://www.service-architecture.com/web\\_services/articles/service.html](http://www.service-architecture.com/web_services/articles/service.html)
- [3] Bieberstein, et al. (2008). *Executing SOA: A Practical Guide for The Service Oriented Architecture*. IBM Press, Indiana.
- [4] Brown, Paul C. (2008). *Implementating SOA : Total Architecture in Practice*. Addison Wesley Professional, Massachusetts.
- [5] Connolly, T. dan C. Begg. (2002). *Basis Data Systems*. Third Edition. Addison Wesley, United States of America.
- [6] Erl, Thomas. (2005). *Service Oriented Architecture : Concepts, Technology, and Design*. Prentice Hall PTR, Indiana.