

Rancang Bangun *Web Service* Perpustakaan Universitas Sam Ratulangi

Bramwell A. Kasaedja, Rizal Sengkey, ST., MT., Oktavian A. Lantang, ST., MT.,
Jurusan Teknik Elektro-FT, UNSRAT, Manado-95115, Email: bramwell.kasaedja@gmail.com

Abstract - Library now has information that is stored in physical or digital form. Advances in technology are increasingly supporting the development of a modern library. UNSRAT have a Special Service Unit Library in each Faculty. With a lot of information that needs a library data management such as book catalog management, book-lending management and membership management, supported with interesting appearance and easy to understand.

Thesis with the title "Design of Web Service Sam Ratulangi University Library" aims to design and develop an integrated system for managing catalogs of books, loan management, and management members. This integrated system will be built using web service. Web services built using the Java programming language with SOAP architecture. End user applications built using the C # programming language in the form of stand-alone applications and web pages.

In designing and developing web services UNSRAT library, the author will use software development methodologies Diciplined Agile Delivery (DAD).

Keywords: *Web services, SOAP, Application Library, Diciplined Agile Delivery*

Abstrak - Perpustakaan saat ini sudah memiliki informasi yang tersimpan dalam bentuk fisik maupun digital. Kemajuan teknologi semakin mendukung berkembangnya perpustakaan modern. UNSRAT memiliki Unit Pelayanan Khusus Perpustakaan pada masing-masing Fakultas. Dengan informasi yang banyak perlu adanya pengelolaan data perpustakaan berupa pengelolaan katalog buku, pengelolaan peminjaman, dan pengelolaan anggota yang didukung dengan tampilan *interface* menarik dan mudah dipahami.

Skripsi dengan judul “Rancang Bangun *Web Service* Perpustakaan Universitas Sam Ratulangi” bertujuan merancang dan mengembangkan sebuah sistem terintegrasi untuk pengelolaan katalog buku, pengelolaan peminjaman, dan pengelolaan anggota. Sistem terintegrasi ini akan dibangun menggunakan *web service*. *Web service* dibangun menggunakan bahasa pemrograman Java dengan arsitektur SOAP. Aplikasi *end user* dibangun menggunakan bahasa pemrograman C# dalam bentuk aplikasi *stand alone* dan halaman web.

Dalam merancang dan mengembangkan *web service* perpustakaan unsrat, penulis akan menggunakan metodologi pengembangan perangkat lunak *Diciplined Agile Delivery* (DAD).

Kata Kunci: *Web service, SOAP, Aplikasi Perpustakaan, Diciplined Agile Delivery*

I. PENDAHULUAN

Perpustakaan pada umumnya memiliki informasi yang tersimpan dalam bentuk fisik, tetapi dengan perkembangan teknologi maka data-data tersebut dapat dibuat dalam bentuk digital. Dengan jumlah informasi yang banyak perlu adanya pengelolaan data perpustakaan. Saat ini sudah banyak dikembangkan aplikasi yang membantu pengelolaan data perpustakaan, baik dalam bentuk aplikasi *stand alone*, maupun dalam bentuk halaman *web*. Selain mempermudah pengelolaan, dengan adanya aplikasi ini informasi-informasi tersebut juga mudah diakses oleh pengguna.

Unsrat memiliki perpustakaan yang dapat diakses oleh semua civitas akademik. Perpustakaan Unsrat terdiri dari perpustakaan pusat dan perpustakaan di setiap fakultas. Informasi perpustakaan pada beberapa fakultas belum dikelola dengan baik, dan masih tertutup. Akibatnya masyarakat kota Manado, dan para civitas akademika Unsrat sulit untuk mengakses informasi-informasi yang berhubungan dengan perpustakaan pada Fakultas-Fakultas tersebut. Saat ini Unsrat sudah memiliki aplikasi yang secara khusus mengelola informasi tersebut tetapi belum dilengkapi dengan *interface* yang baik. Aplikasi tersebut juga belum dilengkapi dengan fitur – fitur yang dapat mengelola data peminjaman dan keanggotaan perpustakaan. Unsrat juga sudah memiliki *website* khusus bagi *user*, tetapi belum menampilkan informasi buku dari semua perpustakaan di setiap fakultas. Oleh sebab itu dibutuhkan sebuah layanan yang mampu mengintegrasikan kedua aplikasi tersebut sehingga mempermudah dalam mengelola informasi digital yang berhubungan dengan perpustakaan, dan dapat diakses dengan mudah oleh Masyarakat Manado dan para civitas akademik.

Web service memiliki peluang besar untuk menyelesaikan masalah pertukaran data, integrasi data dan pengelolaan data. *Web Service* adalah sebuah aplikasi dengan beberapa *service* dalam *web*. *Service* yang dibangun dapat dikembangkan atau digunakan oleh beberapa aplikasi dengan *platform* yang berbeda. *Web service* yang telah dikembangkan akan didistribusikan menggunakan *hosting web* dan domain. Setiap *client* (Fakultas) akan memiliki aplikasi perpustakaan masing-masing dengan *interface* yang lebih baik dan akan terintegrasi langsung untuk mengembangkan *service* yang akan dibuat.

Dengan latar belakang diatas maka penulis akan mengembangkan sebuah sistem yang berhubungan dengan perpustakaan Unsrat dengan judul “RANCANG BANGUN *WEB SERVICE* PERPUSTAKAAN UNIVERSITAS SAM RATULANGI”

II. LANDASAN TEORI

A. Rancang – Bangun

Menurut R. Pressman dalam bukunya “*Software Engineering A Practitioner’s Approach 7th*” pengertian rancang adalah proses menganalisa kebutuhan dan mendeskripsikan dengan detail komponen-komponen yang akan di implementasi. Dalam buku tersebut juga di jelaskan pengertian dari bangun yaitu menciptakan sistem baru atau memperbaiki sistem yang sudah ada.

George M. Scott mengemukakan bahwa proses rancang sistem adalah bagaimana suatu sistem akan menyelesaikan apa yang harus diselesaikan. Kegiatan rancang sistem juga termasuk megkonfigurasi komponen perangkat lunak dan perangkat keras dari suatu sistem.

Dengan demikian dapat disimpulkan bahwa rancang – bangun adalah kegiatan menganalisa kebutuhan, dan mendeskripsikan semua komponen – komponen yang terlibat dengan sistem kemudian menciptakan sistem berdasarkan kebutuhan dan komponen-komponen yang sudah dideskripsikan sebelumnya. Tujuan akhir dari kegiatan rancang – bangun adalah menjawab sebuah permasalahan dengan memberikan solusi yang sesuai dengan kebutuhan.

B. Web Service

Dalam sebuah buku yang di terbitkan oleh O’Reilly, penulis David A Chappel dan Tyler Jewell mengemukakan bahwa *web service* adalah kumpulan logika bisnis dalam internet yang dapat di akses melalui protokol internet. Dalam buku tersebut juga diuraikan Terdapat tiga teknologi dalam *Web service* yaitu, Simple Object Acces Protocol (SOAP), *Web Service Description Language* (WSDL), dan *Universal Description, Discoveri, Integration* (UDDI)

Ketiga teknologi diatas saling berkomunikasi, ketika aplikasi client meminta *service*, UDDI akan memberikan informasi tentang letak dari dokumen WSDL. WSDL berisi sebuah pesan dengan skema XML. Dengan skema tersebut pesan yang diminta dari client akan di proses.

Pada tahun 1997 sebuah perusahaan *software* yaitu “Sun” memperkenalkan sebuah platform yang disebut dengan Jini Platform. Platform ini berkomunikasi menggunakan *web services*. Kemudian perusahaan – perusahaan besar mulai turut mengembangkan *web service* milik mereka. Dalam perkembangannya munculah badan-badan consortium seperti W3C, Oasis, JCP, dan OMG yang memiliki standar berbeda-beda dalam mengembangkan *web service*.

Implementasi sistem terdistribusi sudah ada sebelumnya seperti CORBA. Sistem terdistribusi ini mengimplementasikan layanannya dengan format komunikasi yang berbeda – beda sehingga menimbulkan masalah interoperabilitas. *Web service* menggunakan XML sebagai standar komunikasi, sehingga memungkinkan terjadinya komunikasi antara *platform* yang berbeda. Sebagai contoh dengan adanya *web service* maka aplikasi berbasis Java dengan aplikasi berbasis .Net dapat saling berkomunikasi. Selain masalah interoperabilitas, sistem terdistribusi sebelumnya tidak menggunakan *port* standar sehingga tidak semua komunikasi dapat berjalan dengan dinamis. *Web*

service menggunakan protokol standar HTTP sehingga komunikasi dinamis antara aplikasi dapat terus berjalan.

C. Perpustakaan

Menurut UU No. 43 tahun 2007, perpustakaan adalah “Institusi pengelola karya tulis, karya cetak, dan karya rekam secara professional dengan sistem yang baku guna memenuhi kebutuhan pendidikan, penelitian, pelestarian, informasi, dan rekreasi pemustaka”. Secara tradisional perpustakaan adalah sebuah koleksi buku atau majalah. Perpustakaan sudah ada sebelum jaman modern, sebelum masehi manusia mencatat kegiatan mereka dan menyimpannya dalam bentuk pahatan kayu, batu, atau lempengan. Perkembangan zaman mempengaruhi media tulis manusia hingga sekitar tahun 700 Masehi mulai digunakan kulit binatang dan terus berkembang sampai ditemukannya kertas sebagai media tulis. Setelah ditemukannya mesin pencetak koleksi – koleksi tulisan tersebut sudah berbentuk cetak.

Kehadiran pepustakaan dalam masyarakat tetap bertahan, karena perpustakaan mempunyai fungsi yang berkaitan dengan kepentingan masyarakat. Fungsi perpustakaan bagi masyarakat adalah : sebagai media penyimpanan karya tulis, sebagai sumber informasi, sebagai sumber rekreasi, mendukung sarana pendidikan informal, dan fungsi Kultural

Dengan berkembangnya teknologi informasi maka saat ini perpustakaan tradisional telah berkembang menjadi perpustakaan modern, sehingga segala informasi tidak hanya berbentuk kumpulan buku cetak, tetapi juga dalam bentuk digital.

D. Web Service Perpustakaan

Berdasarkan pengertian dari *web service* dan perpustakaan, maka *web service* perpustakaan dapat diartikan sebagai layanan *web* dengan *service* yang akan dibangun berupa logika bisnis pertukaran data-data perpustakaan. *Web service* perpustakaan dapat mendukung fungsi-fungsi perpustakaan bagi masyarakat. Layanan ini memberikan informasi bibliografis dari buku-buku yang tersedia di perpustakaan, juga informasi peminjaman buku dan keanggotaan perpustakaan.

Dalam pengembangannya akan dibangun aplikasi *end user* berbasis aplikasi *windows* dan *website* yang akan diimplementasikan pada setiap fakultas. Aplikasi *end-user* tersebutlah yang akan mengembangkan *service* yang sudah dibuat. Data-data yang akan saling berkomunikasi melalui *service* ini adalah data katalog buku, peminjaman buku, dan keanggotaan perpustakaan. Semua *database* katalog buku akan disentralisasikan menggunakan *service* yang akan dibuat.

Berbeda dengan aplikasi *client – server* pada umumnya, jika implementasi diperluas maka semua perpustakaan dalam lingkungan unsrat akan berperan sebagai *client*. Sedangkan *service* akan di-*deploy* pada media internet. Sehingga dalam lingkungan pengembangan tidak diperlukan perangkat keras tambahan yang berperan sebagai *server*.

E. SOAP Web Service Menggunakan JAX – WS

SOAP adalah sebuah protokol pertukaran informasi dengan konsep sentralisasi dan terdistribusi. SOAP menyediakan mekanisme sederhana dan ringan dalam pertukaran informasi menggunakan XML (*World Wide*

Consortium). Dalam teknologi *web service* terdapat WSDL, yaitu sebuah bahasa yang digunakan untuk mendeskripsikan *web service* tersebut. WSDL akan mendeskripsikan tipe data yang akan saling berkomunikasi, operasi-operasi yang akan dikembangkan, protokol yang akan digunakan dan dimana *service* akan ditempatkan.

Jax-WS adalah sebuah API yang dikembangkan oleh java untuk membangun sebuah *web service*. Dalam membangun *web service* menggunakan Jax-WS, pesan-pesan SOAP yang kompleks dapat disederhanakan. Setiap operasi-operasi didefinisikan menggunakan *interface* bahasa pemrograman java.

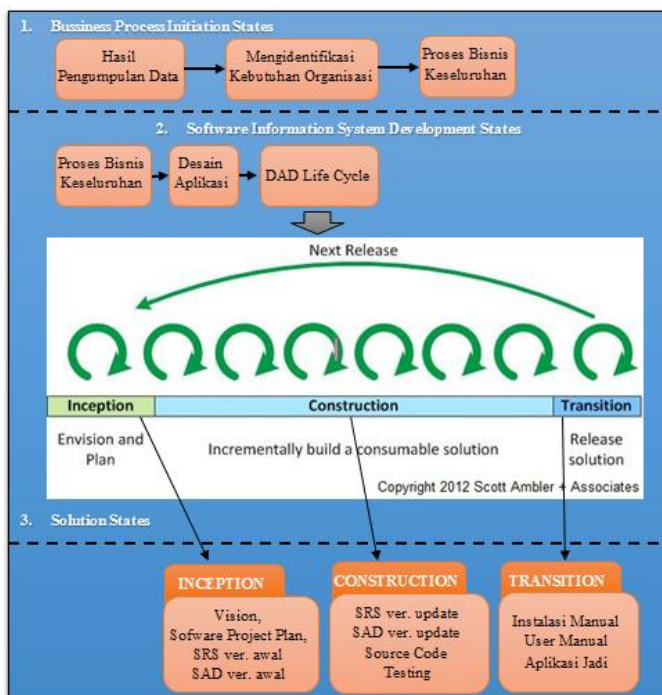
F. Metodologi Pengembangan *Disciplined Agile Delivery* (DAD)

DAD merupakan salah satu metodologi pengembangan perangkat lunak. Berdasarkan sumber yang didapatkan maka definisi dari DAD adalah “*The Disciplined Agile Delivery (DAD) decision process framework is a people-first, learning-oriented hybrid agile approach to IT solution delivery. It has a risk-value delivery lifecycle, is goal-driven, is enterprise aware, and is scalable.*” (IBM’s White Paper “Disciplined Agile Delivery : An Introduction”).

DAD merupakan metodologi dengan kerangka kerja *hybrid* karena mengadopsi beberapa metodologi seperti Scrum, *Agile Modeling*, *Extreme Programming*, *Unified Process*, Kanban, dan *Agile Data*.

G. Kerangka Kerja

Kerangka kerja (gambar 1) dibuat untuk menggambarkan prosedur kegiatan rancang bangun (*input*, *output* dan *process*) yang akan dilakukan oleh penulis. Dalam kerangka ini akan terlihat keterkaitan antara setiap tahap rancang bangun dengan pengembangan perangkat lunak.



Gambar 1. Kerangka Kerja Rancang Bangun

Bussines Process Initiation States

Bagian ini adalah proses awal dengan melakukan identifikasi bisnis serta besar sistem yang akan di rancang. Pada bagian ini dilakukan kegiatan pengumpulan data yang dibutuhkan untuk pengembangan sistem. Hasil dari kegiatan pengumpulan data adalah data input. Kemudian dilakukan identifikasi kebutuhan organisasi dan menghasilkan output berupa proses bisnis keseluruhan yang berisi proses bisnis, lingkungan bisnis dan proyek yang dikerjakan.

Software Information System Development States

Bagian ini adalah proses rancang bangun *Web Service* Perpustakaan Unsrat dengan menggunakan *framework* pengembangan perangkat lunak DAD. Terdapat dua jenis input dalam bagian ini yaitu *output* dari *Bussines Process Initiation States* yang berupa proses bisnis keseluruhan dan desain *Web Service* Perpustakaan Unsrat. *Input* tersebut kemudian diproses dalam 3 fase pengembangan perangkat lunak menggunakan *framework* DAD (*Inception*, *Construction*, dan *Transition*).

Solution States

Bagian ini adalah bagian akhir, yang merupakan *output* dari *Software Information System Development States*. Pada bagian ini juga akan dihasilkan dokumen – dokumen hasil dari perancangan aplikasi.

III. ANALISA DAN PERANCANGAN

A. Gambaran Umum Lingkungan Pengembangan

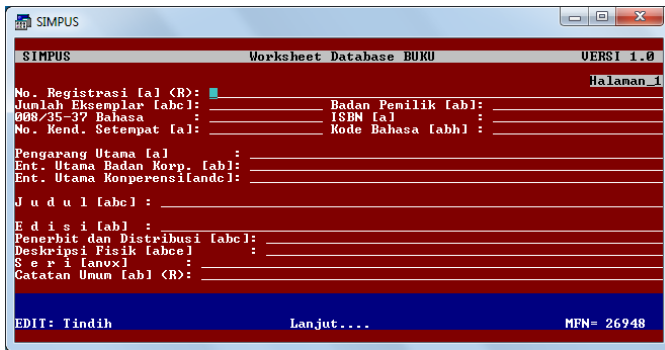
Unsrat mempunyai unsur penunjang yang berbentuk unit pelaksana teknis (UPT), UPT terdiri dari pimpinan, kelompok tenaga ahli, dan unit pelaksana pelayanan administrasi, dan bertugas memberikan pelayanan untuk menunjang penyelenggaraan universitas. Perpustakaan Unsrat adalah salah satu unit Pelaksana Teknis di bidang perpustakaan yang berada di bawah dan bertanggung jawab langsung kepada Rektor dan pembinaannya dilakukan oleh Pembantu Rektor I. Perpustakaan dipimpin oleh Kepala yang ditunjuk di antara pustakawan senior di lingkungan Perpustakaan. (sumber : www.unsrat.ac.id)

Sistem Informasi Manajemen Perpustakaan

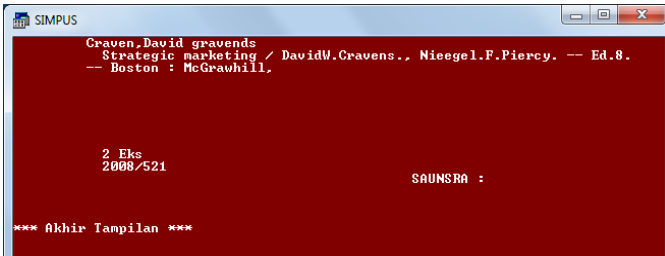
Dalam mengelola informasi, perpustakaan – perpustakaan unsrat menggunakan sebuah aplikasi yang di sebut SIMPUS. Aplikasi yang saat ini digunakan belum dilengkapi dengan GUI dan masih berbasis *text*. Navigasi menu dilakukan dengan mengetik Huruf tertentu sesuai dengan keterangan yang ada. Tampilan Aplikasi SIMPUS dapat dilihat pada gambar 2 dan gambar 3.



Gambar 2. Tampilan Awal Aplikasi SIMPUS



Gambar 3. Tampilan Input Data Baru



Gambar 4. Tampilan Pencarian Buku

Aplikasi ini dapat menyimpan data, menampilkan kembali data yang tersimpan, dan menampilkan data hasil penelusuran (Gambar 4).

B. Inception Phase

Fase ini merupakan awal fase awal, pada tahap ini akan dilakukan pengumpulan semua kebutuhan – kebutuhan yang akan digunakan dalam mengembangkan sistem. Melalui fase ini akan dihasilkan dokumen *Story Card* (Lampiran 1) dan *Vision* (Lampiran 2)

Identifying Business Value

Web Service perpustakaan Unsrat ini dikembangkan dengan tujuan untuk memudahkan pengelolaan data-data perpustakaan, data-data peminjaman dan data keanggotaan dalam bentuk digital yang dapat diakses dengan mudah oleh masyarakat manado termasuk civitas akademika Unsrat. Identifikasi kebutuhan dapat dilihat pada tabel I.

Stakeholder Request

Pada bagian ini akan dideskripsikan permasalahan yang dihadapi *stakeholder* serta mendeskripsikan kebutuhan-kebutuhan *stakeholder* atas aplikasi. Kebutuhan *stakeholder* ini dapat dilihat pada tabel II.

C. Construction Phase

Pada fase ini akan dilakukan pemodelan sistem dengan menggunakan UML. Fase ini akan menghasilkan dokumen *Software Architecture Document*, *Software Requirement Specification* dan *Source Code*. Diagram-diagram yang dihasilkan adalah *Use Case*, *Class Diagram*, *Sequence Diagram*, *Deployment View* dan *Navigation Diagram*.

TABEL I. *PROBLEM STATEMENT*

<i>The problem of</i>	Aplikasi yang sedang berjalan belum memiliki <i>interface</i> yang baik website perpustakaan belum menampilkan informasi buku dari setiap fakultas. Pengelolaan informasi peminjaman dan katalog buku masih manual.
<i>Affects</i>	Aplikasi yang sedang berjalan masih sulit dipahami dan sulit dioperasikan. Proses pertukaran informasi menjadi kurang cepat dan tepat. Pengelolaan data memakan waktu cukup lama.
<i>The impact of which is</i>	Perkembangan informasi yang berhubungan dengan perpustakaan terhambat dan data tidak terorganisir dengan baik. Pegawai perpustakaan belum puas dengan aplikasi yang sedang berjalan.
<i>A successful solution would be</i>	Membuat aplikasi lebih menarik, mudah dipahami dan mudah dioperasikan sehingga dapat memudahkan proses pengelolaan informasi perpustakaan dan informasi tersebut lebih mudah diakses oleh civitas akademik maupun masyarakat manado.

Tahap Perancangan dan Pengembangan Sistem

Perancangan dan Pengembangan *Web Service* Perpustakaan Unsrat ini terdiri dari 4 tahapan (Gambar 5) yang didalamnya dilakukan kegiatan pengumpulan data perpustakaan, pengolahan *database* perpustakaan, pembuatan *service* dan pembuatan aplikasi untuk mengembangkan *service* yang sudah dibuat. Data yang dibutuhkan adalah informasi tentang sistem yang sedang berjalan sekarang, proses pengelolaan katalog, proses peminjaman dan proses pengelolaan anggota perpustakaan

Functional Model

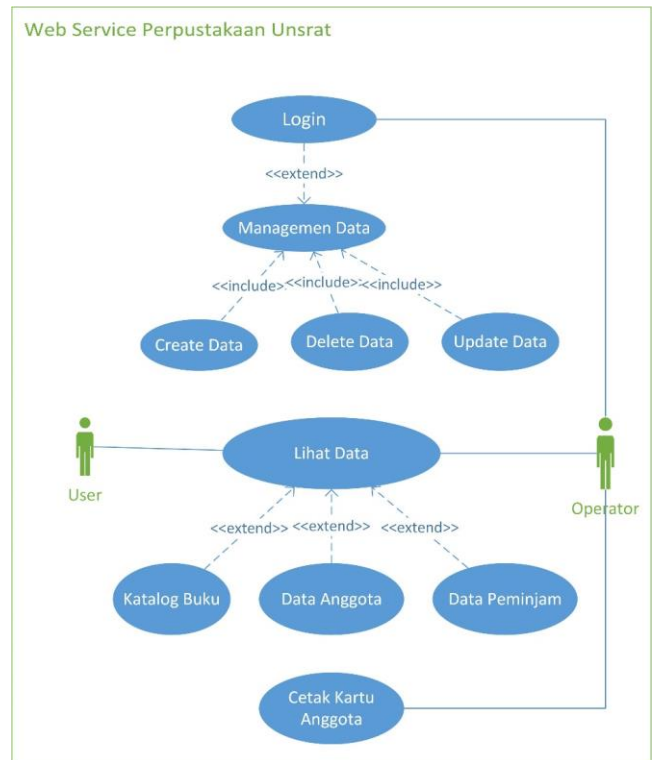
Berdasarkan *stakeholder request*, maka seluruh kebutuhan untuk pengembangan sistem akan di gambarkan dalam bentuk pemodelan fungsional. Dalam pemodelan ini akan digunakan diagram UML Diagram *Use Case*. Diagram ini dibuat untuk menggambarkan setiap aktor yang akan berinteraksi dengan sistem. Diagram *use case* dapat dilihat pada gambar 6.

Use Case Description

Tabel *use case description* akan menjelaskan secara rinci dari kegiatan aktor berdasarkan diagram, *use case* dan reaksi sistem yang akan dibangun. *Use case description* dapat dilihat pada tabel III, tabel IV, tabel V, dan tabel VI.

TABEL II. STAKEHOLDER REQUEST

<i>Problems</i>	<p>Aplikasi yang sedang berjalan tidak dilengkapi dengan <i>interface</i> yang baik.</p> <p>Kesulitan dalam mengelola data peminjam dan katalog buku perpustakaan</p> <p>Kesulitan dalam mengakses informasi perpustakaan</p>
<i>Affects</i>	<p>Aplikasi masih sulit dipahami dan dioperasikan</p> <p>Proses mendapatkan informasi menjadi kurang cepat dan tepat.</p> <p>Proses mengelola data memakan waktu yang lama</p>
<i>The impact of which is</i>	<p>Perkembangan informasi yang berhubungan dengan perpustakaan terhambat</p>
<i>Solution to implemented would be</i>	<p>Membuat aplikasi dengan <i>interface</i> yang lebih baik dan dapat memudahkan proses pengelolaan informasi-informasi yang berhubungan dengan perpustakaan.</p>
<i>Stackholder needed</i>	<p>Aplikasi memiliki <i>interface</i> yang lebih baik.</p> <p>Aplikasi dapat menampilkan informasi katalog buku, data peminjam, dan anggota perpustakaan.</p> <p>Aplikasi dapat mendukung proses pengelolaan informasi dengan baik.</p>



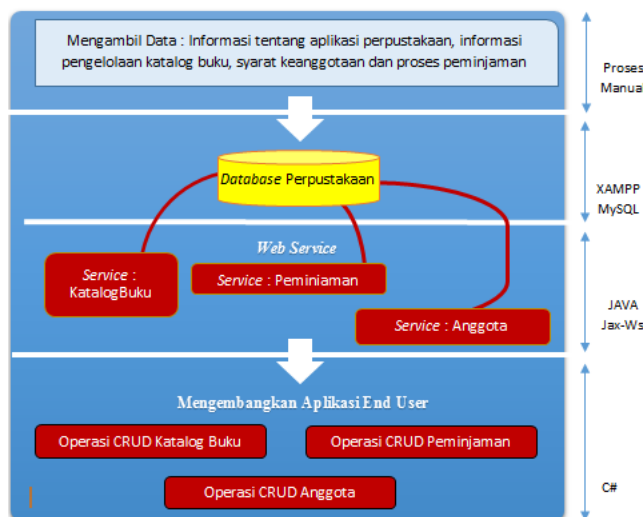
Gambar 6. Use Case Diagram

TABLE III. USE CASE DESCRIPTION : LOGIN

Nama Use Case	Login	
Aktor	Operator	
Deskripsi	Menjelaskan tentang proses autentikasi aktor sebelum mengakses aplikasi	
Pre - condition		
Normal Course	Aktor	Sistem
	2. Mengisi <i>username</i> dan <i>password</i> kemudian <i>login</i>	1. Menampilkan <i>form login</i>
		3. Menampilkan <i>form utama</i>
Alternate Course	Jika input tidak valid sistem akan menampilkan pesan gagal <i>login</i>	
Post - condition		

TABLE IV. USE CASE DESCRIPTION : MANAGEMEN DATA

Nama Use Case	Managemen Data
Aktor	Operator
Deskripsi	Menjelaskan tentang proses aktor melakukan tambah data, <i>update</i> data, dan hapus data
Pre - condition	



Gambar 5. Tahap Pengembangan Web Service

	Aktor	Sistem
	2. Memilih menu pada menu utama	1. Menampilkan form utama 3. Menampilkan form sesuai pilihan menu 5. Menyimpan hasil proses ke database
Normal Course	4. Melakukan proses Creat, Update, dan Delete	
Alternate Course		
Post - condition		

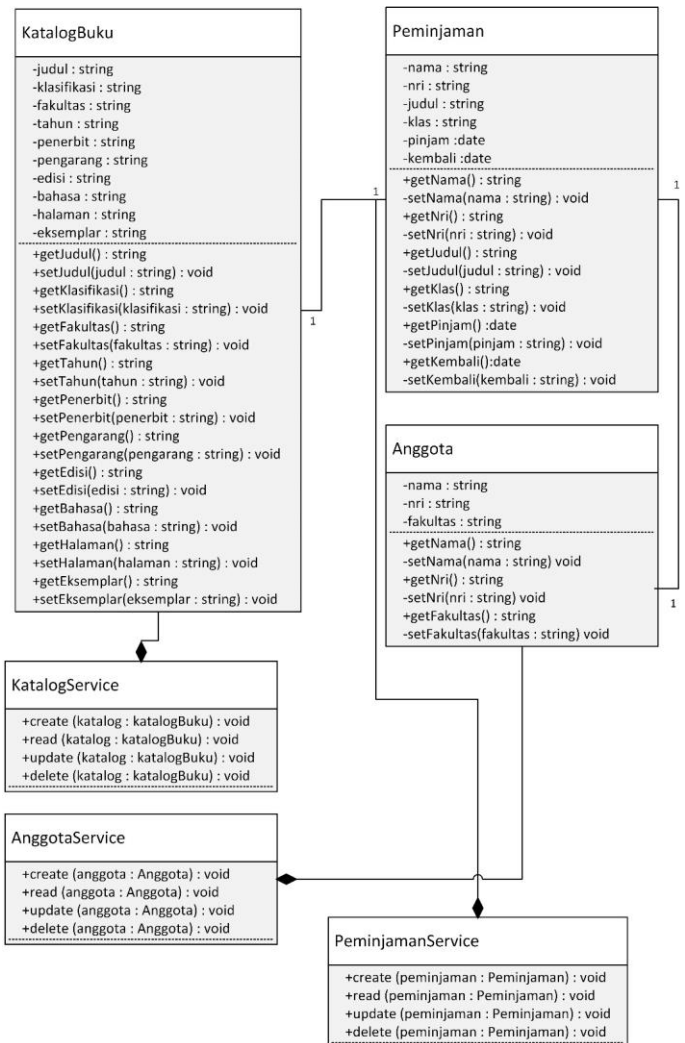
	yang akan dilihat dengan memilih menu pada menu utama	Utama
		3. Menampilkan data dalam bentuk table
Alternate Course		
Post - condition		

TABLE V. USE CASE DESCRIPTION : CETAK ANGGOTA

Nama Use Case	Cetak Anggota	
Aktor	Operator	
Deskripsi	Menjelaskan tentang proses aktor mencetak anggota	
Pre - condition		
Normal Course	Aktor	Sistem
	2. Memilih menu Anggota Perpustakaan	1. Menampilkan Form Utama 3. Menampilkan form anggota 6. Menampilkan form cetak
	4. Memilih data anggota yang akan di cetak	
	5. Mengakses tombol cetak	
	7. Mencetak kartu anggota	
Alternate Course		
Post - condition		

TABLE VI. USE CASE DESCRIPTION : LIHAT DATA

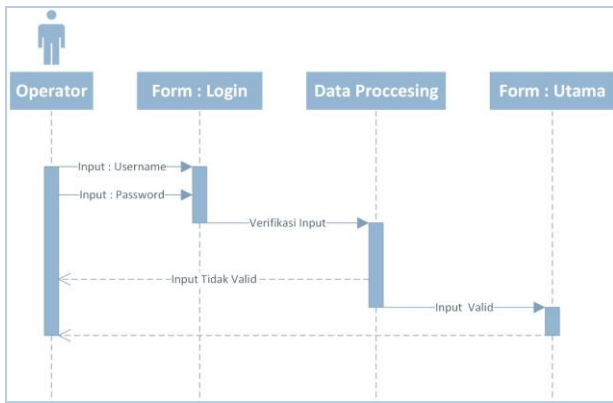
Nama Use Case	Lihat Data	
Aktor	Operator dan User	
Deskripsi	Menjelaskan tentang proses aktor mengakses database untuk melihat data yang telah tersimpan	
Pre - condition		
Normal Course	Aktor	Sistem
	2. Menentukan data	1. Menampilkan Form



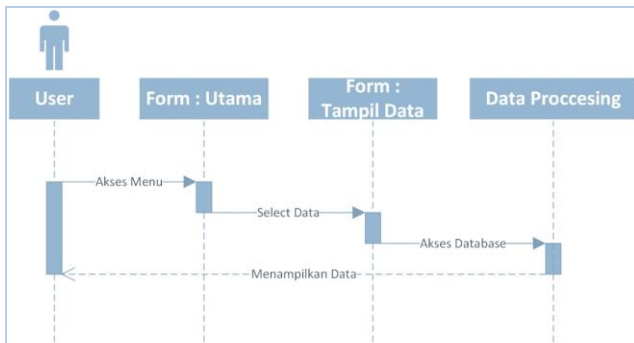
Gambar 7. Class Diagram

Structural Model

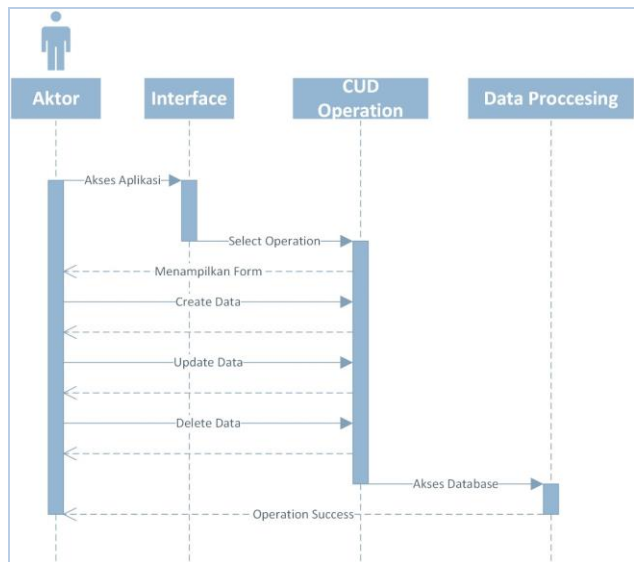
Structural model dibuat menggunakan UML Class Diagram (gambar 7). Dengan demikian hubungan setiap class beserta method class masing-masing dapat terlihat.



Gambar 8. Sequence Diagram : Login



Gambar 9. Sequence Diagram Menampilkan Data



Gambar 10. Sequence Diagram Manajemen Data

Behavioral Model

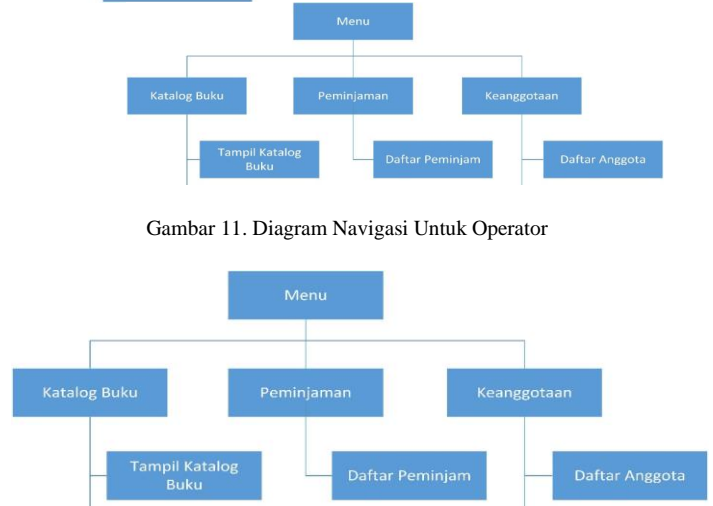
Perancangan sistem yang akan dikembangkan dimodelkan berdasarkan gambar perilaku objek. Dalam pemodelan ini akan digunakan UML Sequence Diagram.

Human Computer Interaction Layer Design

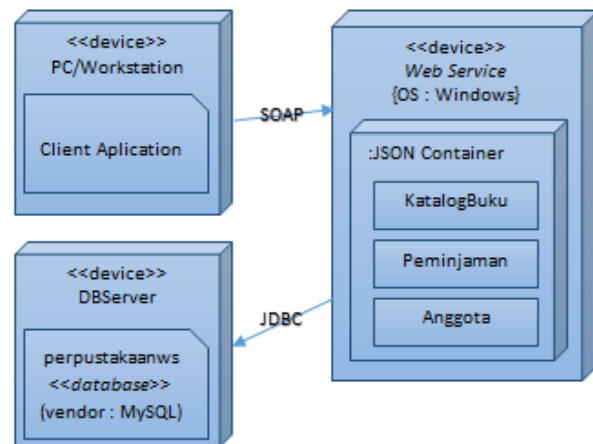
Pemodelan ini adalah untuk menggambarkan bagaimana interaksi sistem dengan entitas lain.



Gambar 11. Diagram Navigasi Untuk Operator



Gambar 12. Diagram Navigasi Untuk User



Gambar 13. Deployment Diagram

Dalam pemodelan ini akan digunakan Diagram Navigasi untuk menggambarkan fitur-fitur yang ada pada aplikasi. Diagram Navigasi untuk operator dapat dilihat pada gambar 10, sedangkan untuk user dapat dilihat pada gambar 11.

Design Physical Layer Architecture

Desain arsitektur dari lapisan fisik akan dimodelkan menggunakan *deployment diagram* (gambar 13). Diagram ini akan menggambarkan hubungan setiap komponen perangkat lunak maupun perangkat keras yang ada dalam sistem, juga dimana sistem akan didistribusikan.

User Interface Design

Dalam kegiatan rancang - bangun ini selain dibuat *service* yang akan mendukung beberapa proses bisnis perpustakaan, juga dibuat aplikasi *end-user* dalam bentuk *desktop application* untuk masing-masing perpustakaan. Dalam penerapannya, aplikasi *end-user* ini akan mengembangkan *service* yang sudah dibuat. Pembuatan aplikasi ini menggunakan IDE Visual Studio 2010 dengan bahasa pemrograman C#. Pembuatan *interface* menggunakan *tools windows form* yang sudah disediakan oleh IDE. *Service* dibangun menggunakan IDE Java Netbeans. *screen capture project* PerpustakaanWS dapat dilihat pada gambar 14.

Dalam *project* PerpustakaanWS terdapat beberapa *package* yang berisi beberapa *class*, yaitu :

Package wellz.ws

berisi *class* DataPerpustakaan.Java. *Class* ini adalah *namespace* untuk *webservice* yang dibuat. Pada *class* inilah operasi-operasi *service* didefinisikan.

Package wellz.ws.model

terdapat *class* Anggota.java, KatalogBuku.java, dan Peminjaman.java. Tiga *class* dalam *package* ini berisi atribut-atribut yang akan menjadi struktur xml dari *webservice*.

Class Katalog.java

berisikan atribut-atribut yang akan digunakan untuk *service* pengelolaan katalog buku.

Class Anggota.java

berisikan atribut-atribut yang akan digunakan untuk *service* pengelolaan data anggota perpustakaan.

Class Peminjaman.Java

berisikan atribut-atribut yang akan digunakan untuk *service* pengelolaan data peminjaman buku perpustakaan

Package wellz.ws.util

Terdapat *class* Database.java. *Class* ini dibuat untuk membangun koneksi dengan *database* MySQL. Untuk membangun koneksi perlu didefinisikan *username*, *password*, *hostname*, dan nama *database* yang dibuat. Referensi driver untuk *database* yang digunakan adalah *com.mysql.jdbc.Driver*.

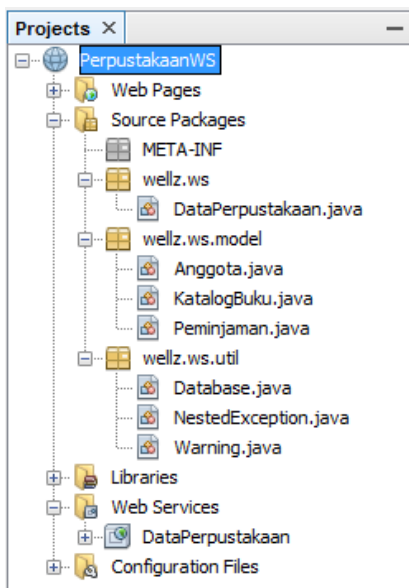
Berikut ini rancangan *user interface* dari Aplikasi Perpustakaan.

Login Form

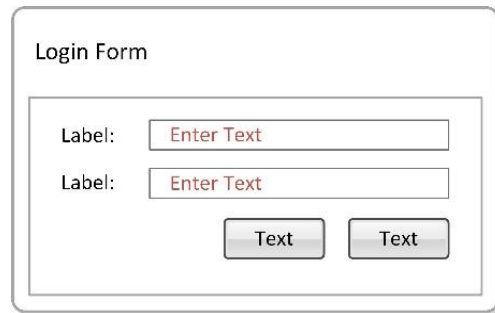
Sebelum mengakses *Main Form*, user terlebih dahulu harus mengisi *username* dan *password* pada *field* yang sudah disediakan. Terdapat dua tombol akses untuk membatalkan *login* dan untuk *login*. Rancangan tampilan Login dapat dilihat pada gambar 15.

Main Form

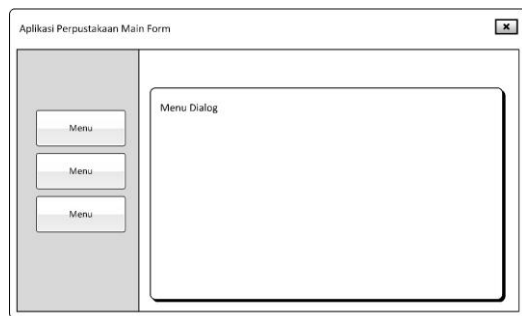
Main Form (gambar 16) adalah jendela utama ketika user berhasil login untuk mengakses aplikasi. *Main Form* berisikan tombol menu Katalog Buku, Peminjaman dan Keanggotaan yang terletak pada panel kiri . Setiap menu tersebut memiliki dialog masing masing sesuai dengan kebutuhan.



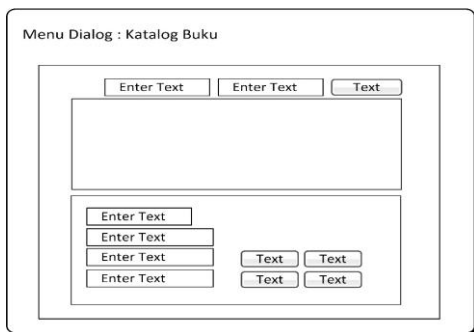
Gambar 14. Project PerpustakaanWS menggunakan Java NetBeans



Gambar 15. Rancangan Interface Login



Gambar 16. Rancangan Interface Main Form

Gambar 17. Rancangan *Interface* Menu Dialog Katalog BukuGambar 18. *Login Form*

Menu Dialog : Katalog Buku

Pada menu dialog katalog buku akan ditampilkan tabel daftar buku yang sudah tersimpan dalam *database*. Pada bagian atas table terdapat *field* dan tombol untuk fungsi *filter* atau pencarian buku. Untuk mempermudah navigasi maka tepat dibawah table sudah disediakan *field-field* yang akan menampilkan informasi buku sesuai kolom yang dipilih. Juga disediakan tombol operasi *database* sehingga secara langsung operator juga dapat melakukan penambahan, hapus, *update* atau *edit* buku.

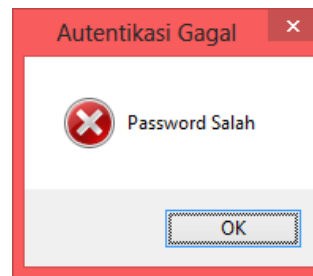
Interface yang sama juga akan digunakan pada menu dialog peminjaman dan menu dialog keanggotaan, dengan penyesuaian fungsi masing-masing menu. Rancangan tampilan menu katalog buku dapat dilihat pada gambar 17.

BAB IV HASIL DAN PEMBAHASAN

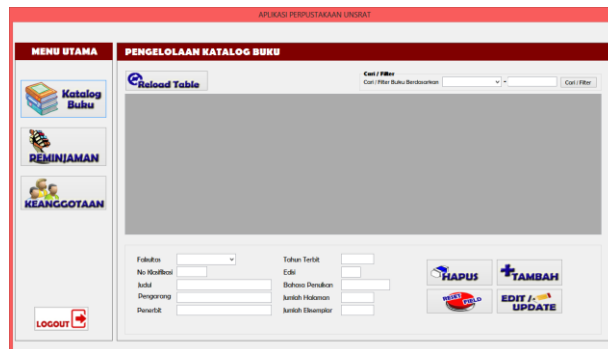
A. *Web Service*

Hasil dari *project* PerpustakaanWS adalah sebuah *webservice* yang sudah terhubung dengan *database* MySQL. Ketika dilakukan testing maka browser akan menampilkan *method* yang sudah dibuat, dan kita dapat melakukan pengujian terhadap *method* tersebut. Sebagai contoh ketika melakukan pengujian terhadap *method* *getKatalogBuku* maka browser akan menampilkan SOAP *Request* dan SOAP *Respons*. *Method* yang dipanggil akan merespon dengan menampilkan data sesuai dengan yang sudah didefinisikan dalam *class* *DataPerpustakaan.java*.

Selain melakukan testing, kita juga dapat melihat WSDL dari *webservice*. WSDL akan mendeskripsikan tipe data yang akan saling berkomunikasi, operasi-operasi yang akan dikembangkan, protokol yang akan digunakan dan dimana *service* akan ditempatkan.



Gambar 19. Pemberitahuan Autentikasi Gagal



Gambar 20. Jendela Utama

B. *Aplikasi End User Desktop*

Dalam pengembangan *web service* ini dikembangkan aplikasi utama bagi *end user* berupa aplikasi windows. Aplikasi inilah yang akan digunakan oleh setiap fakultas dalam mengelola informasi perpustakaan. *Source code* lengkap aplikasi *end-user* dilampirkan pada lampiran 5. Berdasarkan perancangan sebelumnya, berikut ini adalah tampilan *interface* yang telah dibuat.

Autentikasi Operator

Sebelum mengakses jendela utama operator harus memasukkan *password* pada *form login* (gambar 18).

Jika *Password* salah, maka akan diberikan pemberitahuan (gambar 19) dan *Login* dibatalkan. Jika *password* benar, maka jendela utama akan ditampilkan.

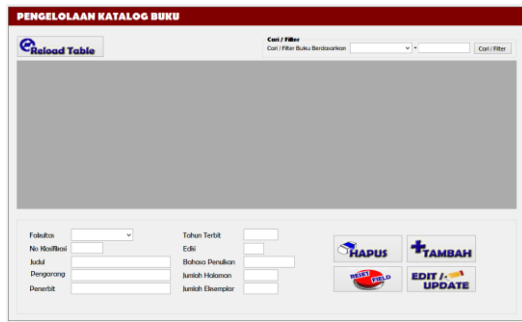
Disediakan dua tombol, yaitu tombol LOGIN untuk mengakses aplikasi, dan tombol CANCEL untuk membatalkan autentikasi.

Jendela Utama

Setelah berhasil *login*, maka jendela utama (gambar 20) akan ditampilkan. Pada tampilan ini terdapat panel Menu Utama dan panel Pengelolaan.

Panel Menu Utama berfungsi untuk mengakses panel pengelolaan. Tombol Katalog Buku untuk membuka panel Pengelolaan Katalog Buku, tombol PEMINJAMAN untuk membuka panel Pengelolaan Peminjaman Buku, dan tombol KEANGGOTAAN untuk membuka panel Pengelolaan Anggota Perpustakaan.

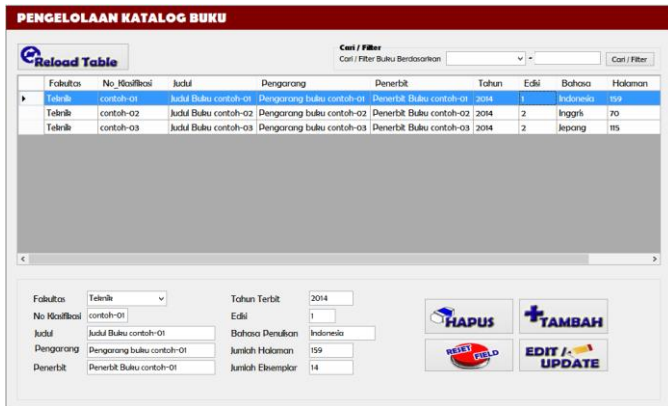
Pada jendela utama juga terdapat tombol LOGOUT untuk menutup aplikasi. *Login Form* akan kembali ditampilkan setelah operator *logout*.



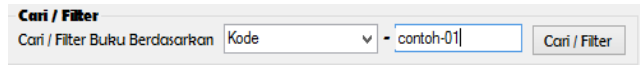
Gambar 21. Panel Pengelolaan Katalog Buku



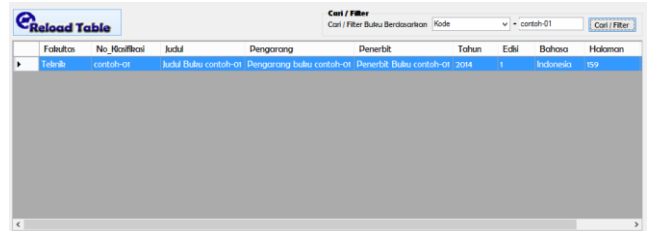
Gambar 22. Form Input Data Katalog Buku



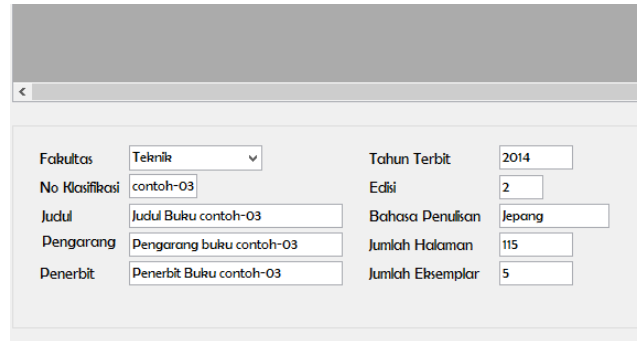
Gambar 23. Menampilkan database dalam bentuk tabel



Gambar 24. Form Pencarian / Filter Katalog Buku



Gambar 25. Menampilkan hasil pencarian Katalog Buku



Gambar 26. Fitur Edit/Update Data Katalog Buku



Gambar 27. Tombol akses untuk manajemen data Katalog Buku

Panel Pengelolaan Katalog Buku

Secara otomatis panel ini akan ditampilkan ketika jendela utama terbuka. Pada panel ini akan ditampilkan data katalog buku dalam bentuk tabel (gambar 23). Terdapat beberapa tombol dan *field* yang berfungsi untuk manajemen data perpustakaan. Tampilan panel katalog buku dapat dilihat pada gambar 21.

Input Data Katalog Buku

Untuk menambahkan data katalog buku maka telah disediakan *form input* tepat dibawah table Setelah mengisi *field* tersebut, tekan tombol TAMBAH untuk menyimpan data ke dalam database. Kemudian data yang berhasil di simpan akan ditampilkan dalam bentuk tabel. Tampilan *field* untuk *input* data katalog buku dapat dilihat pada gambar 22.

Fitur pencarian / filter Katalog Buku

Bagian atas tabel terdapat *form* untuk melakukan pencarian atau *filter* (gambar 24). Terdapat pilihan pencarian berdasarkan kode buku atau judul buku, sebagai contoh jika pencarian berdasarkan kode maka user mengetik kode klasifikasi buku pada *field* yang disediakan.

Setelah menekan tombol *Cari/Filter*, tabel akan menampilkan data yang dibutuhkan.

Fitur Edit / Update dan Hapus Katalog Buku

Untuk merubah data atau menghapus data, operator memilih data yang akan diubah atau dihapus, kemudian secara otomatis keseluruhan data tersebut akan terisi pada *field input* data. Tampilan Edit/Update dan Hapus Data dapat dilihat pada gambar 26.

Untuk merubah data, operator mengedit data pada *field* tersebut sesuai dengan kebutuhan. Kemudian menekan tombol EDIT/UPDATE DATA. Untuk menghapus data, setelah memilih data yang akan dihapus kemudian menekan tombol Hapus. Terdapat tombol optional “RESET FIELD” untuk mengosongkan *field input* data. Tombol akses Edit/Update dan Hapus dapat dilihat pada gambar 27.

Panel Pengelolaan Peminjaman

Panel pengelolaan peminjaman akan terbuka jika menekan tombol PEMINJAMAN pada Menu Utama. Pada panel ini akan ditampilkan data peminjaman buku yang telah tersimpan dalam bentuk tabel. Tampilan pengelolaan peminjaman dapat dilihat pada gambar 28.

Input Data Peminjam

Input data baru dengan mengisikan data pada *form input* yang disediakan (Gambar 31). Agar dapat meminjam buku harus terdaftar sebagai anggota perpustakaan. Apabila calon peminjam sudah menjadi anggota, maka NRI calon peminjam akan ditampilkan pada dalam *list combo box*. Nama calon peminjam akan secara otomatis terisi sesuai dengan NRI.

Demikian juga dengan buku yang akan dipinjam akan terisi otomatis setelah memilih no. klasifikasi yang sudah ditampilkan dalam *list combo box*. Tanggal peminjaman akan sesuai dengan tanggal setempat waktu peminjaman. Tanggal pengembalian akan terisi *default* dengan *min date C#*.

Edit / Update dan Hapus Data Peminjam

Operator memilih data yang akan diubah pada tabel, kemudian secara otomatis *field input* data akan terisi. Operator dapat merubah status pengembalian dengan menekan tombol SUDAH, maka secara otomatis tanggal pengembalian akan berubah menjadi tanggal waktu pengembalian. Tampilan *edit/update* data peminjam dapat dilihat pada gambar 29.

Untuk menghapus data peminjam, operator memilih data yang akan dihapus kemudian menekan tombol HAPUS PEMINJAM.

Untuk menghapus data anggota, pada tabel operator memilih data yang akan dihapus kemudian menekan tombol HAPUS.v Tombol akses *edit/update* dan hapus dapat dilihat pada gambar 30.

Panel Pengelolaan Anggota

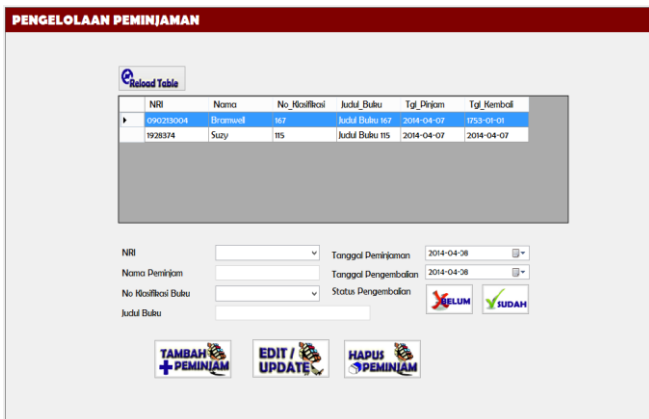
Panel pengelolaan anggota (gambar 32) akan terbuka jika operator menekan tombol KEANGGOTAAN pada Menu Utama Pada panel ini akan ditampilkan anggota yang sudah terdaftar dalam bentuk tabel.

Input Data Anggota Baru

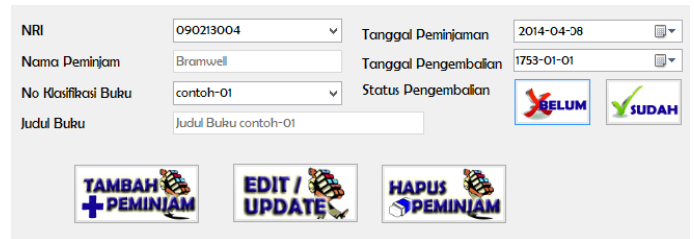
Mengambahkan anggota baru dengan cara *Input* data anggota baru pada *field* yang sudah disediakan. Kemudian menekan tombol TAMBAH. Tampilan input data anggota baru dapat dilihat pada gambar 33.

Fitur Cetak Kartu Anggota

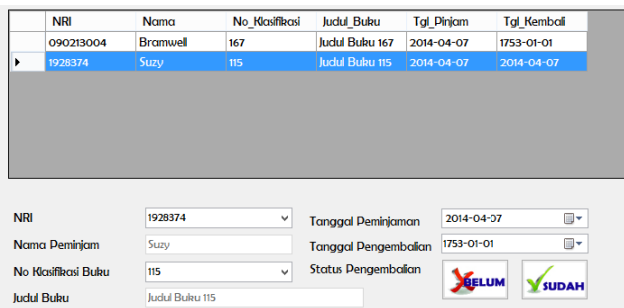
Untuk mencetak kartu anggota operator memilih pada tabel data yang akan dicetak. Kemudian menekan tombol CETAK KARTU ANGGOTA untuk membuka *form* Cetak. Tampilan mencetak kartu anggota dapat dilihat pada gambar 34.



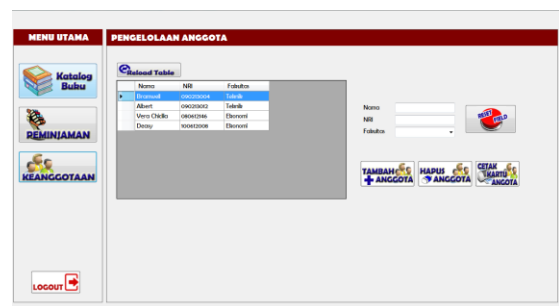
Gambar 28. Panel Pengelolaan Peminjaman



Gambar 31. Form Input Data Peminjam



Gambar 29. Edit / Update Data Peminjam



Gambar 32. Panel Pengelolaan Anggota

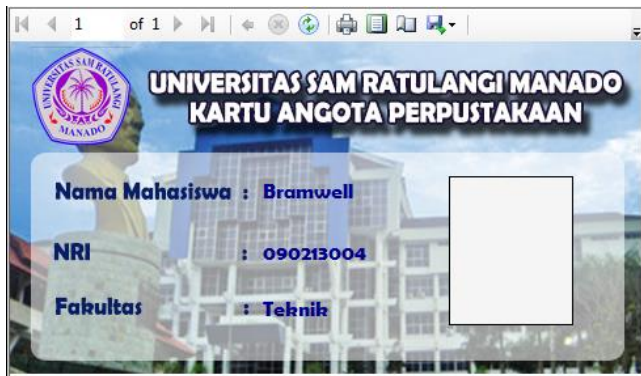


Gambar 30. Tombol untuk manajemen data peminjaman buku

Fitur Hapus



Gambar 33. Input data anggota baru



Gambar 34. Form Cetak Kartu Anggota




Gambar 36. Halaman Database Anggota



Gambar 35. Halaman Web Katalog Buku



Gambar 37. Halaman Peminjaman Buku

Operator menekan tombol  untuk membuka jendela printer dan mulai mencetak. Selain itu kartu anggota juga dapat di *export* kedalam format PDF dengan menekan tombol *export* pada barisan menu.

C. Aplikasi End-User Web

Selain aplikasi utama untuk pengelolaan informasi perpustakaan, juga dikembangkan aplikasi pendukung bagi mahasiswa untuk melihat *database* katalog buku, anggota, dan peminjaman buku perpustakaan. Aplikasi ini dikembangkan dalam bentuk halaman web. Web dibuat dengan 3 halaman utama.

Halaman Katalog Buku dan Database Anggota

Halaman ini (gambar 35, 36) menampilkan *database* katalog buku pada setiap perpustakaan, halaman ini adalah halaman pertama yang akan ditampilkan ketika mengakses *website* pengelolaan data perpustakaan. Disediakan tombol pada sebelah kiri untuk mengakses *link database* katalog buku.

Halaman Peminjaman Buku

Halaman ini (gambar 37) akan menampilkan *database* peminjaman buku, disediakan tombol “PEMINJAMAN” pada halaman utama untuk mengakses halaman pengelolaan data peminjaman buku.

KESIMPULAN

Berdasarkan perancangan yang sudah dilakukan serta pengembangan *web service* perpustakaan Unsrat maka diperoleh hasil yaitu *Service* dengan fungsi-fungsi yang mendukung proses pengelolaan katalog, peminjaman, dan keanggotaan perpustakaan, dikembangkan dua buah aplikasi *end-user* dalam bentuk *desktop application* dan *web* yang sudah terintegrasi dengan *service* yang dibuat, sehingga fungsi-fungsi *service* yang sudah dibuat dapat dikembangkan melalui aplikasi *end-user*. Dengan adanya perancangan dan pengembangan ini maka informasi-informasi perpustakaan dapat diakses dengan mudah oleh siapa saja dan dimana saja melalui *web client*. Proses pengelolaan katalog, data peminjam, dan keanggotaan perpustakaan menjadi lebih mudah. Aplikasi didesain sederhana mungkin, dengan tampilan *interface* yang menarik, sehingga mudah dipahami dan mudah dioperasikan.

DAFTAR PUSTAKA

- [1] A. Wijanarko, Integrasi Aplikasi Perpustakaan Untuk Berbagi Informasi Penelitian Perguruan Tinggi Berbasis Web Service. *S2 thesis, UAJY*. 2013.
- [2] IBM, *Web Service : The Basic*, tersedia di : www.ibm.com/developerworks/webservice, diakses 27 Juli 2013.

- [3] IBM, *Unified Modeling Language*, tersedia di : <http://www-01.ibm.com/software/rational/uml/>, diakses 09 Maret 2014.
- [4] L. Williams, *An Introduction to the Unified Modeling Language*, 2004.
- [5] Oracle, *The Java EE 6 Tutorial*. United States : Oracle, 2013.
- [6] R. Pressman, *Software Engineering A Practitioner's Approach 7th Edition*. United States : Mc-Graw Hill, 2010.
- [7] S. W. Ambler, M.Lines, *Disciplined Agile Delivery A Practitioner's Guide to Agile Software Delivery in the enterprise*. United States : 2012.