

USULAN TUGAS AKHIR

1. IDENTITAS PENGUSUL

NAMA : YUSUF KURNIAWAN
NRP : 5109100052
DOSEN WALI : WASKITHO WIBISONO, S.Kom., M.Eng., PhD

2. JUDUL TUGAS AKHIR

“Rancang Bangun Perangkat Lunak Berorientasi Arsitektur Service (SOA) Dengan Pendekatan *Workflow* Pada Domain *Cash Bank* dan *General Ledger Enterprise Resource Planning*”

3. LATAR BELAKANG

Kondisi dunia bisnis saat ini telah berkembang menjadi semakin kompleks, semakin kompetitif, bergerak dengan cepat serta semakin sulit untuk diprediksi. Agar dapat bersaing dan sukses, perusahaan perlu memadukan bisnis dan sumber daya IT yang dimiliki agar dapat secara fleksibel mengakomodasi adanya perubahan untuk kemudian dilakukan adaptasi terhadap perubahan tersebut secara cepat dan tepat.

Dari pembahasan diatas dapat disimpulkan bahwa kebutuhan akan aplikasi *Enterprise Resource Planning* (ERP) dalam perusahaan merupakan kebutuhan yang perlu agar perusahaan dapat memadukan bisnis dan sumber daya. Pada ERP terdapat banyak sekali modul-modul yang dibuat dan di integrasikan, antara lain adalah *Finance / Accounting*, *Sales*, *Customer Relationship Manager*, *Inventory*, *Manufacturing*, *Human Resource Management*, dan lainnya. Pada tugas akhir ini, penulis akan membangun fungsional domain *Cash Bank* dan *General Ledger* yang termasuk dalam paket *Finance / Accounting*.

FI-Financial Accounting, Software bisnis SAP ditujukan untuk menyediakan pengukuran secara kontinu terhadap profitabilitas perusahaan. Modul *FI* juga mengukur kinerja keuangan perusahaan, berdasarkan pada data transaksi intenal maupun eksternal. Modul *FI* menyediakan dokumen keuangan yang mampu melacak (mengaudit) setiap angka yang terdapat dalam suatu laporan keuangan hingga ke data transaksi awalnya.

Dengan berkaca pada modul *SAP FI-CO* dan dengan sedikit penyesuaian sesuai kebutuhan, maka penulis akan membangun sebuah domain ERP *Accounting* yang meliputi sub-domain *Account Payable*, *Account Receivable*, *Cash Bank*, *General Ledger* dan juga *Fixed Asset*. Dan Penulis akan membangun sub-domain *Cash Bank* dan *General Ledger* pada tugas akhir kali ini.

Cash Bank adalah domain fungsional yang menangani pencatatan transaksi penerimaan dan pengeluaran uang. *Cash Bank* melakukan pencatatan transaksi berdasarkan invoice yang diterima dari *Account Payable* dan *Account Receivable*, kemudian setiap pencatatan tersebut akan diakumulasi dan diproses menjadi sebuah jurnal.

Dan *General Ledger*, Dalam sebuah aplikasi manajemen keuangan, bagian yang penting dibutuhkan oleh direksi perusahaan untuk mengambil keputusan adalah general ledger / buku besar. Karena pada bagian ini mereka bisa mengetahui seluruh informasi keuangan yang dibutuhkan dalam bentuk jurnal atau report sesuai dengan kebutuhan.

4. RUMUSAN MASALAH

Detil permasalahan yang diangkat dalam tugas akhir ini adalah sebagai berikut:

1. Bagaimana membuat aplikasi *Cash Bank* dan *General Ledger* suatu perusahaan untuk menggantikan proses pengolahan data manual dan sebagai penyedia informasi terkait *Cash Bank* dan *General Ledger* sebagai usaha peningkatan kinerja perusahaan.
2. Bagaimana mengimplementasikan pemodelan arsitektur *Service Oriented Analysis and Design* dalam merancang servis dari sistem *Cash Bank* dan *General Ledger*.
3. Bagaimana mengimplementasikan orkestrasi *web service* pada aplikasi ERP *Accounting* untuk sub domain *Cash Bank* dan *General Ledger*.
4. Bagaimana mengimplementasikan pendekatan *Workflow* pada aplikasi ERP *Accounting* untuk sub domain *Cash Bank* dan *General Ledger*.

5. BATASAN MASALAH

Masalah yang dibahas pada tugas akhir ini dibatasi lingkupnya pada:

1. Hasil dari Tugas Akhir ini adalah menghasilkan sebuah aplikasi ERP untuk modul *Accounting* yang terdiri dari *Cash Bank* dan *General Ledger* yang servisnya dibangun dengan menggunakan pemodelan arsitektur SOAD
2. Domain Fungsional,
 - a. *Cash Bank* terdiri dari :
 - i. *Cash Management*
 - ii. *Bank Management*
 - iii. *Cheque Management*
 - iv. *Giro Management*
 - v. *Currency Management*
 - b. *General Ledger* terdiri dari :
 - i. Pengaturan akun perkiraan (*chart of account*)
 - ii. Pengaturan periode akuntansi (*accounting period*)
 - iii. Pencatatan jurnal
 - iv. Tutup buku (*closing period*)
 - v. Membuat laporan keuangan

3. ERP yang dibangun berorientasi pada aspek *multi-tenant*.
4. Platform yang digunakan menggunakan framework .NET dengan menggunakan teknologi :
 - a. WCF (*Windows Communication Foundation*)
 - b. ASP.NET
 - c. nHibernate
 - d. WWF (*Windows Workflow Foundation*)
5. Menggunakan SQL Server sebagai *Database Management System*

6. TUJUAN PEMBUATAN TUGAS AKHIR

Tujuan dari penyusunan tugas akhir ini adalah:

1. Membuat aplikasi *Cash Bank* dan *General Ledger* suatu perusahaan untuk menggantikan proses pengolahan data manual dan sebagai penyedia informasi terkait *Cash Bank* dan *General Ledger* sebagai usaha peningkatan kinerja perusahaan.
2. Mengimplementasikan pemodelan arsitektur *Service Oriented Analysis and Design* (SOAD) dalam merancang servis dari sistem *Cash Bank* dan *General Ledger*.
3. Mengimplementasikan orkestrasi *web service* pada aplikasi *ERP Accounting* untuk sub domain *Cash Bank* dan *General Ledger*.
4. Mengimplementasikan pendekatan *Workflow* pada aplikasi *ERP Accounting* untuk sub domain *Cash Bank* dan *General Ledger*

7. MANFAAT TUGAS AKHIR

Manfaat yang diharapkan dari tugas akhir ini adalah membangun sebuah aplikasi *Financial Accounting* yang dapat berdiri sendiri dan juga bisa terbagi menjadi beberapa servis, sehingga nantinya dengan mudah di-reuse sesuai dengan domain fungsional lain pada aplikasi *Enterprise Resource Planning* serta penerapan pendekatan *Workflow* untuk menangani proses bisnis *Enterprise Resource Planning* yang dinamis dan bervariasi.

8. TINJAUAN PUSTAKA

a. ERP

ERP / Perencanaan Sumber Daya Perusahaan adalah sistem informasi yang diperuntukkan bagi perusahaan manufaktur maupun jasa yang berperan mengintegrasikan dan mengotomasi proses bisnis yang berhubungan dengan aspek operasi, produksi maupun distribusi perusahaan tersebut. Jadi ERP adalah sebuah terminologi yang diberikan kepada sistem informasi yang mendukung transaksi atau operasi sehari-hari dalam pengelolaan sumber daya perusahaan meliputi dana, manusia, mesin, suku cadang, waktu, material dan kapasitas. Keuntungan penggunaan ERP diantaranya adalah Integrasi data keuangan, standarisasi proses operasi, standarisasi data dan informasi, penurunan inventori dan tenaga kerja, peningkatan servis level dan kontrol keuangan dan penurunan waktu yang dibutuhkan untuk mendapatkan informasi. ^[1]

b. SAP

SAP (*System Application and Product in data processing*) adalah suatu software buatan Jerman yang dikembangkan untuk mendukung suatu organisasi dalam menjalankan kegiatan operasionalnya secara lebih efisien dan efektif. SAP merupakan software Enterprise Resources Planning (ERP), yaitu suatu tools IT dan manajemen untuk membantu perusahaan merencanakan dan melakukan berbagai aktivitas sehari-hari. ^[2]

SAP terdiri dari sejumlah modul aplikasi yang mempunyai kemampuan mendukung semua transaksi yang perlu dilakukan suatu perusahaan dan tiap aplikasi bekerja secara berkaitan satu dengan yang lainnya. Semua modul aplikasi di SAP dapat bekerja secara terintegrasi/terhubung yang satu dengan lainnya.

SAP merupakan software yang banyak dipakai di perusahaan besar untuk mendukung integrasi proses bisnis. Lima tahun terakhir, di perusahaan negara-negara Asia, termasuk Indonesia, sedang gencar-gencarnya mengimplementasikannya. Software buatan Jerman ini telah lama dipakai di perusahaan besar Eropa dan Amerika.

Di Indonesia, banyak perusahaan besar yang telah mengimplementasikan SAP, misalnya Astra International, Toyota Astra Motor, Toyota Motor Manufacturing Indonesia, Bentoel Prima, United Tractor, Daihatsu Motor, Pertamina, Aqua, Telkomsel, Auto 2000, Blue Bird dan masih banyak perusahaan lagi. Ada yang hanya memakai untuk keperluan area tertentu seperti *Financial Accounting* saja atau *Sales and Distributon* saja, namun banyak pula yang mengintegrasikan beberapa modul.

c. Service Oriented Architecture (SOA)

SOA (*service oriented architecture*, arsitektur berorientasi layanan) adalah suatu gaya arsitektur sistem yang membuat dan menggunakan proses bisnis dalam bentuk paket layanan sepanjang siklus hidupnya. SOA juga mendefinisikan dan menentukan arsitektur teknologi informasi (TI) yang dapat menunjang berbagai aplikasi untuk saling bertukar data dan berpartisipasi dalam proses bisnis. Fungsi-fungsi ini tidak terikat dengan sistem operasi dan bahasa pemrograman yang mendasari aplikasi-aplikasi tersebut.

SOA membagi fungsi-fungsi menjadi unit-unit yang berbeda (layanan), yang dapat didistribusikan melalui suatu jaringan dan dikombinasikan serta digunakan ulang untuk membentuk aplikasi bisnis. Layanan-layanan ini saling berkomunikasi dengan mempertukarkan data antar mereka atau dengan mengkoordinasikan aktivitas antara dua atau lebih layanan. Konsep SO sering dianggap didasari atau berkembang dari konsep-konsep yang lebih lama dari komputasi terdistribusi dan pemrograman modular. ^[3]

d. Service Oriented Analysis and Design (SOAD)

Service oriented analysis and design merupakan metode perancangan desain sistem aplikasi secara top-down yang merancang sistem dari level kebutuhan proses bisnis hingga level software komponen. Dalam metode ini, desain aplikasi dibagi menjadi tiga bagian yaitu *conceptual view*, *logical view*, dan *physical view*.

e. Workflow

WF merupakan sebuah teknologi dari Microsoft yang menyediakan sebuah API, *in-process workflow engine*, dan designer untuk mengimplementasikan proses yang berkepanjangan sebagai sebuah *workflow* dalam aplikasi yang berbasis .NET. WF versi terbaru adalah versi 4 dan biasa disebut dengan WF4.

Sebuah *workflow* merupakan serangkaian langkah-langkah atau fase dalam pemrograman. Setiap fase dimodelkan dalam WF sebagai sebuah Activity dan framework .NET menyediakan sebuah library Activity yang mencakup keseluruhan Activity tersebut. Activity-activity tersebut dapat disusun secara visual dalam beberapa workflow menggunakan *Workflow Designer*, yaitu sistem desain yang terdapat dalam Microsoft Visual Studio, atau juga terdapat di aplikasi lainnya.

Sequential workflow merupakan workflow yang bersifat “dapat diprediksi”, dimana alur eksekusi sebuah workflow dapat bercabang, berulang, ataupun menunggu sebuah event eksternal terjadi, namun sequential workflow akan menggunakan himpunan dari Activity maupun aturan lainnya yang telah tersedia untuk terus berjalan, sehingga workflow ini menjadi kontrol dari proses-proses tersebut.

Sebuah state machine workflow merupakan workflow yang digerakkan oleh event-event yang terjadi, yang berarti workflow tersebut bergantung kepada event eksternal hingga selesai. Caranya, pengguna mendefinisikan kondisi-kondisi (state) legal dari workflow tersebut, beserta transisi yang legal antar-state tersebut, dan pergerakan atau transisi tersebut digerakkan oleh event eksternal workflow tersebut. Kesimpulannya, workflow ini mendefinisikan sebuah struktur pemrograman yang dapat diikuti, namun pengendalian berada di luar kendalinya.

Pengguna menggunakan sequential workflow pada saat pengguna tersebut dapat melakukan pengkodean keputusan-keputusan aplikasi di dalam workflow tersebut, dan menggunakan state machine workflow apabila terjadi sebaliknya. Untuk selanjutnya, dalam penggunaan workflow tersebut dapat diatur activity-activity yang diinginkan pengguna dengan menggunakan activity designer, yaitu sebuah base class untuk mengatur activity dalam sebuah workflow.

Selain itu dikenal juga suatu bentuk workflow yang mengkoordinasikan berbagai proses yang melibatkan manusia sebagai pengguna, yang disebut dengan human workflow. Workflow bentuk ini dapat mengotomasi interaksi antarmanusia, bahkan antara manusia dengan sistem, sehingga memungkinkan peran besar manusia dalam sistem itu sendiri. Workflow tipe ini bergantung kepada Activity oleh pengguna (manusia) yang dihitung dalam suatu rentang waktu (harian, mingguan, bahkan bulanan), baik aktifitas terhadap suatu sistem atau aktifitas fisik diluar sistem, sehingga eksekusi workflow ini harus berhenti di suatu titik dan menunggu sebuah event eksternal yang biasanya berupa interaksi pengguna maupun event dalam sistem sendiri untuk bergerak ke fase selanjutnya. ^[4]

f. WCF

WCF atau *Windows Communication Foundation* adalah salah satu teknologi baru dari Microsoft yang memungkinkan aplikasi dalam lingkungan terdistribusi berkomunikasi satu sama lain. WCF adalah model pemrograman lengkap untuk membangun aplikasi berorientasi layanan. Teknologi ini memungkinkan pengembang untuk membangun

solusi aman, handal, dan mendukung transaksi, yang dapat terintegrasi lintas platform serta mampu beroperasi dengan investasi yang ada.

WCF merupakan cara termudah dalam membuat dan mengonsumsi *service* pada platform microsoft maupun .NET. Dengan menggunakan WCF, pengembang dapat fokus pada aplikasi mereka daripada fokus ke protokol komunikasi, karena WCF mengenkapsulasi terhadap teknologi dan pekerjaan teknis, sehingga pengembang bisa lebih produktif. WCF sendiri mendukung berbagai teknologi seperti *Web service*, *Binary .NET to .NET communication*, *Distributed Transaction*, *Queued Messaging*, *RESTful Communication*, serta mendukung *WS-* Specification*.

Karena WCF mendukung komunikasi melalui *Web Service* maka WCF akan dapat saling beroperasi dengan aplikasi platform lain yang mendukung SOAP. Selain itu WCF juga mendukung spesifikasi *WS-** sehingga menyediakan kehandalan, keamanan, dan transaksi ketika berkomunikasi dengan platform lain yang juga mendukung spesifikasi *WS-**. WCF tidak mengharuskan untuk menggunakan protokol HTTP, tetapi juga bisa menggunakan TCP. Pertukaran message pun bisa menggunakan selain SOAP, tetapi bisa juga menggunakan XML sederhana ataupun JSON. ^[5]

g. LINQ

LINQ adalah singkatan dari Language Integrated Query, teknologi ini pertama kali dikenalkan oleh Anders Hejlsberg dalam Microsoft Professional Developers Conference (PDC) 2005. Secara singkat LINQ dapat didefinisikan sebagai sebuah metode yang memudahkan dan menseragamkan cara pengaksesan data. Sehingga dengan menggunakan LINQ developer hanya perlu menggunakan sebuah teknik pengaksesan data saja. Karena jika kita lihat dunia pemrograman saat ini maka umumnya developer perlu menguasai skill yang berbeda-beda untuk setiap media penyimpanan data.

Sebagai contoh ketika sedang berurusan dengan basis data tentunya developer perlu memiliki kemampuan SQL yang mumpuni dan di lain waktu ketika harus berurusan dengan XML developer pun harus memiliki pengetahuan yang cukup tentang cara pengaksesan XML yang tentunya sangat berbeda dengan basis data. Disini lah letak keunggulan LINQ, LINQ hadir untuk menseragamkan cara pengaksesan data tersebut. Sehingga dengan menggunakan LINQ kita hanya cukup menguasai satu saja teknik pengaksesan data untuk berbagai media penyimpanan data (XML, Basis data, dan lainnya)

h. Kontribusi Tugas Akhir Sebelumnya.

Tugas akhir ini merupakan penyempurnaan riset dari tugas akhir sebelumnya, pada tugas akhir yang dulu, tidak terdapat paket *Accounting*, namun berdiri sendiri-sendiri setiap domainnya, seperti fungsional domain *Account Payable*, *Account Receivable*, *Cash Bank*. Dan pada tugas akhir kali ini, penulis bekerja sama dengan Adhrian Chairul Achda untuk menjadikan beberapa fungsional domain tersebut menjadi satu, yaitu domain *Accounting*. Dengan dijadikannya satu paket accounting, maka fleksibilitas aplikasi untuk bisa berdiri sendiri bisa diterapkan. Karena biasanya suatu

perusahaan menginginkan satu modul accounting pada ERP secara utuh. Selain itu, ada penambahan kebutuhan fungsionalitas atau fitur seperti menggunakan standar internasional untuk penulisan *Chart of Account*, *cost of overhead*, dan *white and blue collar payroll*. Selain itu, dengan penggunaan teknologi workflow maka ERP yang dibangun kali ini akan mempermudah dalam masalah fleksibilitas dari fungsionalitas kebutuhan sistem. Dengan kedinamisan aplikasi yang dibangun maka sangat berpengaruh pada kecepatan performa aplikasi dan juga mempermudah para pelaku bisnis yang berlatarbelakang non-IT untuk bisa merubah proses bisnis sesuai kehendaknya.

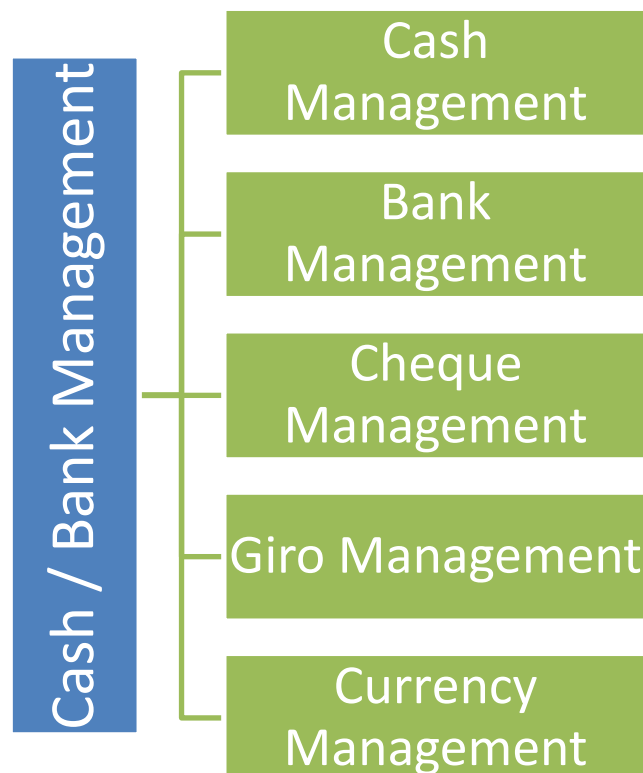
Jadi bisa disimpulkan bahwa tingkat reusability pada tugas akhir sebelumnya hanya sebatas analisa proses bisnis saja. Untuk desain, implementasi dan juga teknologi yang digunakan pada tugas akhir kali sangat berbeda dan diharapkan bisa lebih ringan, akurat dan cepat.

9. RINGKASAN ISI TUGAS AKHIR

a. Deskripsi Aplikasi

- *Cash Bank*

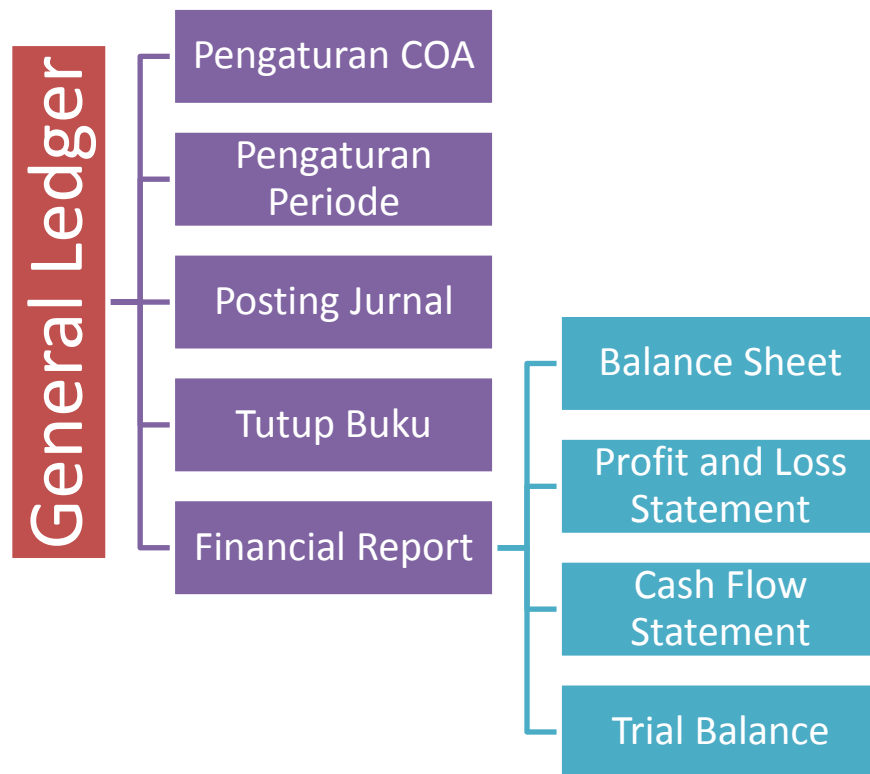
Fungsional domain cash bank secara umum menangani pencatatan transaksi penerimaan dan pengeluaran uang. *Cash bank* melakukan pencatatan transaksi berdasarkan *invoice* yang diterima, kemudian setiap pencatatan tersebut akan diakumulasi dan diproses menjadi sebuah jurnal. Jurnal-jurnal yang dihasilkan cash bank akan dimanfaatkan oleh fungsional domain lain seperti *General Ledger* untuk membuat laporan keuangan.



Gambar 9.1 Proses Bisnis Cash / Bank Management

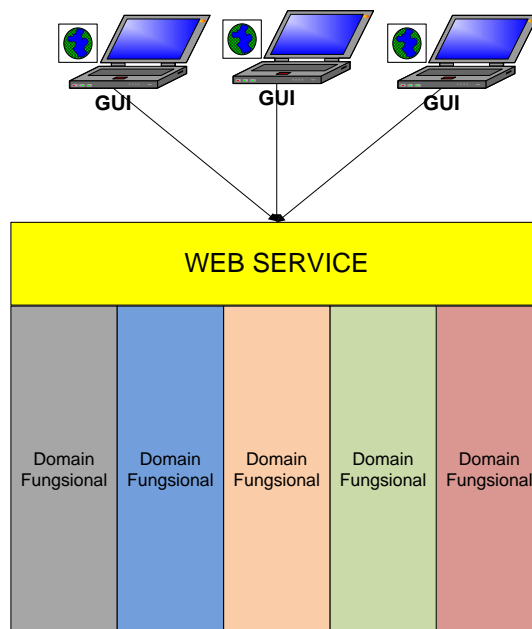
- *General Ledger*

General Ledger atau Buku besar adalah buku utama pencatatan transaksi keuangan yang mengkonsolidasikan masukan dari semua jurnal akuntansi. Buku besar merupakan dasar pembuatan laporan neraca dan laporan laba/rugi. Buku besar dapat memberikan informasi saldo ataupun nilai transaksi untuk setiap kode perkiraan dalam suatu periode akuntansi tertentu.

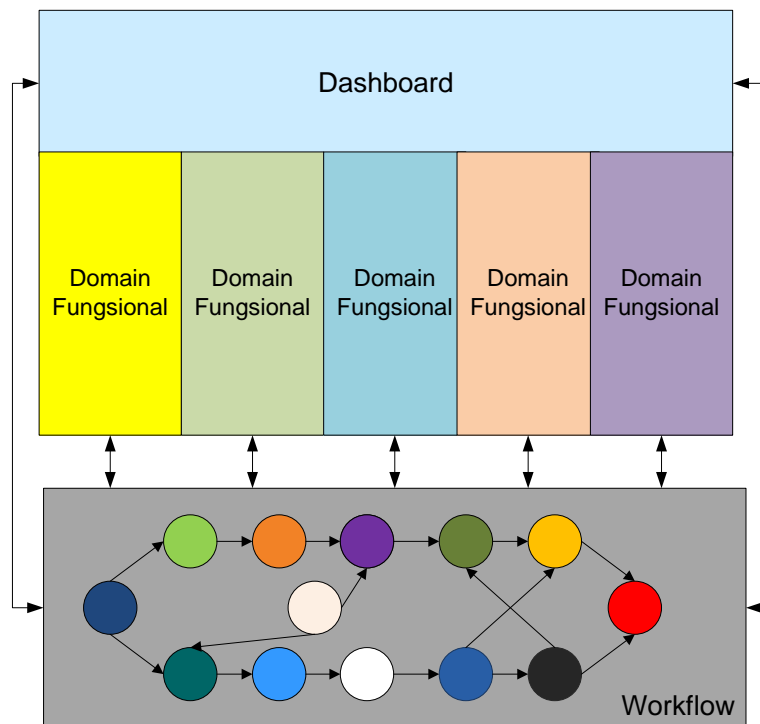


Gambar 9.2 Proses Bisnis General Ledger

b. Desain Sistem



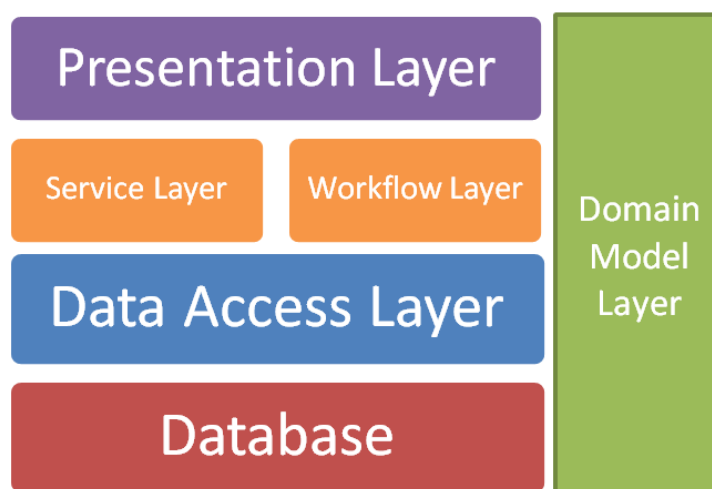
Gambar 9.4 Desain Sistem Berdasarkan Pendekatan Web Service



Gambar 9.5 Desain Sistem Berdasarkan Pendekatan Workflow

Dengan pendekatan workflow seperti gambar diatas, maka kemudahan akan merubah proses bisnis ERP yang dikenal sangat dinamis telah terpenuhi, dengan workflow, maka kita tidak akan susah-susah merubah fungsi dari aplikasi kita, cukup hanya mengganti alur variasi yang ada pada layer workflow. Selain itu, fungsi dari workflow adalah membuat dokumentasi dari perangkat lunak yang dibangun terkesan “hidup”, karena akan terlihat flow / aliran jalannya program dibarengi logic/code-code dalam alirannya.

c. Arsitektur Aplikasi



Gambar 9.6 Arsitektur Aplikasi

10. METODOLOGI

a. Penyusunan proposal tugas akhir

Pada tahap ini penulis menyusun proposal tugas akhir sebagai langkah awal dalam pengerjaan tugas akhir. Dalam pelaksanaan tahap ini penulis menggagas tugas akhir untuk merancang bangun perangkat lunak *Enterprise Resource Planning* Berorientasi *Web Service* Pada Domain Fungsional *Accounting* Menggunakan Framework .NET.

b. Studi literatur

Pada tahap ini dilakukan pengumpulan informasi yang diperlukan untuk perancangan sistem meliputi:

1. Hubungan antar domain fungsional di Enterprise Resource Planning
2. Perencanaan dan pembangunan perangkat lunak
3. Implementasi nHibernate, WCF dan Workflow

Semua informasi tersebut dapat diperoleh dari literatur, paper, maupun buku-buku yang terkait dengan topic tersebut.

c. Analisis dan desain perangkat lunak

Analisis kebutuhan dan perancangan sistem dilakukan untuk merumuskan solusi yang tepat dalam pembuatan aplikasi serta kemungkinan yang dapat dilakukan untuk mengimplementasikan rancangan tersebut. Fase desain meliputi arsitektur perangkat lunak yang digunakan, desain class-class yang terlibat dalam aplikasi, desain basis data, dan lain-lain.

d. Implementasi perangkat lunak

Pada fase ini desain perangkat lunak diwujudkan ke dalam bentuk kode program. Pembangunan aplikasi dilakukan menggunakan platform .NET. Pada fase ini, penulis menggunakan *Microsoft Visual Studio 2010* sebagai alat bantu untuk implementasi pengkodean program.

e. Uji coba dan evaluasi

Pada tahap ini dilakukan uji coba terhadap perangkat lunak yang dibuat. Tujuan uji coba perangkat lunak adalah untuk menemukan kesalahan-kesalahan (*bug*) sedini mungkin sehingga dapat diperbaiki sesegera mungkin.

f. Penyusunan Buku Tugas Akhir

Pada tahap ini dilakukan penyusunan laporan yang menjelaskan dasar teori dan metode yang digunakan dalam tugas akhir ini serta hasil dari implementasi aplikasi perangkat lunak yang telah dibuat. Sistematika penulisan buku tugas akhir secara garis besar adalah sebagai berikut:

1. Pendahuluan
 - a. Latar Belakang
 - b. Rumusan Masalah

- c. Batasan Tugas Akhir
 - d. Tujuan
 - e. Metodologi
 - f. Sistematika Penulisan
2. Tinjauan Pustaka
 3. Desain dan Implementasi
 4. Uji Coba dan Evaluasi
 5. Kesimpulan dan Saran
 6. Daftar Pustaka

11. JADWAL KEGIATAN

Tahapan	2012	2013																								
	Desember	Januari				Februari				Maret				April				Mei				Juni				
Penyusunan Proposal	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>
Studi Literatur	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>
Perancangan sistem	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>
Implementasi	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>
Uji coba dan evaluasi	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>
Penyusunan buku	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>

12. DAFTAR PUSTAKA

- [1] **Kumar, Vinod.** 2004. *Enterprise Resource Planning : Concepts and Practice*. PHI Learning Pvt. Ltd
- [2] **Jones, Peter.** 2009. *Configuring SAP ERP Financials And Controlling*. SYBEX.
- [3] **Erl, Thomas.** SOA: Principles Of Service Design. 2008. PRENTICE HALL
- [4] **Chappel, David.** 2009. *The Workflow Way: Understanding Windows Workflow Foundation*. Microsoft
- [5] **Leroux, Michelle.** 2007. *Learning WCF: A Hands-on Guide*. O'REILLY.

LEMBAR PENGESAHAN

Surabaya, Januari 2013

Menyetujui

Dosen pembimbing I

Dosen pembimbing II

Prof. Drs.Ec.Ir. Riyanarto Sarno,M.Sc,Ph.D

Dwi Sunaryono, S.Kom, M.Kom

NIP. 195908031986011001

NIP. 197205281997021001

Paraf pembimbing 1:

Paraf pembimbing 2:

Tanggal:

Hal 15