

ISSN 0216-4280

JURNAL INFORMATIKA
Volume 6 Nomor 1 Juni 2010

Pelindung :

Rektor Universitas Kristen Maranatha

Penasehat :

Pembantu Rektor Universitas Kristen Maranatha

Pembina :

Dekan Fakultas Teknologi Informasi Universitas Kristen Maranatha

Ketua Tim Redaksi :

Ir. Teddy Marcus Zakaria, MT

Penyunting Ahli :

Dr. Ir. Bambang SP. Abednego

Dr. Richardus Eko Indrajit

Penyunting:

Andi Wahyu Rahardjo Emanuel, BSEE, MSSE

Hapnes Toba, M. Sc

Hendra Bunyamin, S.Si, MT

Tjatur Kandaga, S.Si, MT

Tata Usaha:

Teddy Yusnandar

PENERBIT (PUBLISHER)

Maranatha University Press

ALAMAT PENYUNTING (EDITORIAL ADDRESS)

Sekretariat Jurnal Informatika Universitas Kristen Maranatha

Jurusan Teknik Informatika, Fakultas Teknologi Informasi

Jl. Prof. Drg. Suria Sumantri, MPH, No. 65 Bandung. 40164

Telp (022) 70753665

Fax (022) 2005915

Email: jurnal.informatika@itmaranatha.org

Homepage: <http://www.itmaranatha.org/jurnal.informatika>

Jurnal Informatika UKM terbit sejak 2005 merupakan jurnal ilmiah sebagai bentuk pengabdian dalam hal pengembangan bidang Teknik Informatika dan bidang terkait lainnya.

Jurnal Informatika UKM diterbitkan oleh Jurusan Teknik Informatika Universitas Kristen Maranatha. **Redaksi** mengundang para professional dari dunia usaha, pendidikan dan peneliti untuk menulis mengenai perkembangan ilmu di bidang yang berkaitan dengan **Teknik Informatika**.

Jurnal Informatika UKM diterbitkan 2 (dua) kali dalam 1 tahun pada bulan **Juni** dan **Desember**

Jurnal Informatika
Volume 6 Nomor 1 Juni 2010

DAFTAR ISI

Volume 6 Nomor 1

- | | | |
|----------|---|-----------------|
| 1 | Sistem Pakar Berbasis Web sebagai Alat Bantu Pembelajaran Mahasiswa Kedokteran untuk Penyakit Kanker Darah pada Anak
Meiry Yanti Sihombing, Mewati Ayub | 1 - 10 |
| 2 | Permainan Trading Card Game Magic & Wizard Card Battle
Erico Darmawan Handoyo, Antonius Chandra | 11 - 21 |
| 3 | Aplikasi Pengelolaan PT Makmur Abadi Mandiri dengan Sistem Akuntansi dan DSS dalam Menentukan Pemasok
Diana Trivena Y, Suntono | 23 - 34 |
| 4 | Sistem Pengendalian Lampu dengan Menggunakan <i>Personal Computer</i> (PC) untuk Billing Meja Billiard
Teddy Marcus Zakaria, Henry Kartadinata | 35 - 50 |
| 5 | Perancangan Model Manajemen Pengetahuan menggunakan Model <i>Nonaka Takeuchi</i> (Studi Kasus Administrasi Akademik)
Robby Tan | 51 - 64 |
| 6 | Sistem Informasi Toko Listrik Aryono King dengan Peramalan Stok Barang menggunakan Metode <i>Double Exponential Smoothing</i>
Radiant Victor Imbar, Yon Andreas | 65 - 82 |
| 7 | Pembuatan Perangkat Lunak Aplikasi Pengelolaan Data Kasus Perkara di Kepolisian Resor Kota Bandung Barat
Tiur Gantini, Peter Iman Paskal Mendrofa | 83 - 93 |
| 8 | Konsep dan Perancangan <i>Code-Completion</i> untuk PHP
Tjatur Kandaga, Rinardi Budiadi Sarean | 95 - 103 |

Ucapan Terima Kasih

Redaksi Jurnal Informatika mengucapkan terima kasih yang sebesar – besarnya kepada mitra bestari yang membantu terwujudnya penerbitan Jurnal Informatika Volume 6 Nomor 1 Juni 2010:

1. Danny Manongga (Fakultas Teknologi Informasi Universitas Kristen Satya Wacana)
2. Eko Sedyono (Fakultas Teknologi Informasi Universitas Kristen Satya Wacana)
3. Mewati Ayub (Fakultas Teknologi Informasi Universitas Kristen Maranatha)

Sistem Pakar Berbasis Web sebagai Alat Bantu Pembelajaran Mahasiswa Kedokteran untuk Penyakit Kanker Darah pada Anak

Meiry Yanti Sihombing, Mewati Ayub

Jurusan S1 Teknik Informatika

Fakultas Teknologi Informasi, Universitas Kristen Maranatha

Jl. Prof. Drg. Suria Sumantri no. 65 Bandung 40164

email: i_like_meiry@yahoo.co.id, mewatia@yahoo.com

Abstract

Today, expert system as a part of artificial intelligence has many applications in various fields. This report presents one of expert system applications, namely health problems knowledge base. Specifically, the final project presented here applies expert system approach to the knowledge base of blood cancer disease in children. It is expected to help medical practitioners and students to improve their quality of learning and knowledge. Features of the application are disease definitions, explanations and consultations. There is also a facility to add, delete and update the knowledge base. Knowledge base is implemented using a decision tree, which consists of rules governed by forward chaining inference technique. This application is coded using PHP and MySQL.

Keywords : expert system, knowledge base, blood cancer in children

I. Pendahuluan

Semakin pesatnya kemajuan teknologi informasi khususnya internet, memungkinkan tersedianya layanan melalui situs *web* dalam menyajikan informasi yang cepat dan efisien untuk dapat dipergunakan oleh masyarakat luas. Perkembangan penggunaan internet dapat dipadukan dengan teknologi yang mampu mengadopsi proses dan cara berpikir manusia yaitu bidang *artificial intelligence* (kecerdasan buatan) sehingga menghasilkan sistem pakar berbasis web. Sistem Pakar merupakan salah satu aplikasi dari kecerdasan buatan yang di dalamnya terkandung pengetahuan dan pengalaman yang berasal dari satu atau banyak pakar dalam satu *domain* pengetahuan tertentu sehingga dapat digunakan untuk memecahkan berbagai masalah yang bersifat spesifik [Dur94].

Mahasiswa kedokteran biasanya memperoleh informasi mengenai bahan pembelajaran mereka melalui buku, tetapi seiring dengan berkembangnya teknologi, maka internet dapat dimanfaatkan sebagai bahan untuk mencari informasi. Dalam penelitian ini, akan dibangun suatu sistem pakar mengenai penyakit kanker darah yang merupakan penyakit kanker yang paling banyak diderita oleh anak-anak. Sistem pakar berbasis *web* yang dibangun dapat dijadikan sebagai alat bantu belajar untuk mahasiswa kedokteran. Aplikasi sistem pakar ini berisi fasilitas untuk melihat definisi leukemia, melihat etiologi (penyebab) leukemia, melakukan konsultasi mengenai penderita yang kemudian akan diberikan solusi beserta penjelasannya, melakukan perubahan data pada basis pengetahuan yaitu menambah, menghapus, dan atau mengubah data.

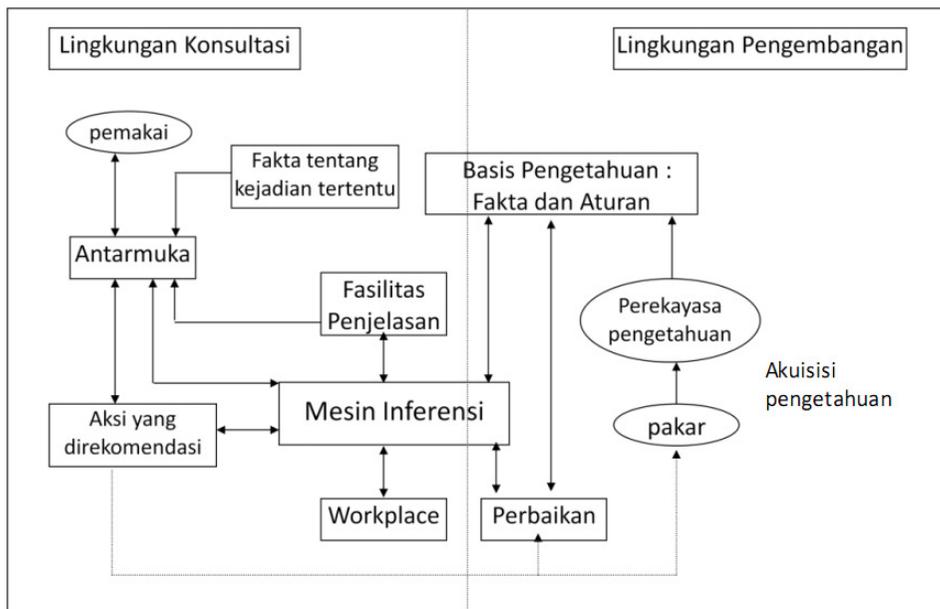
II. Landasan Teori

II.1. Sistem Pakar

Sistem pakar merupakan sistem yang berbasis pengetahuan yang menyediakan solusi dengan mengadopsi kemampuan pakar untuk menyelesaikan masalah dalam suatu domain pengetahuan yang spesifik. Umumnya pengetahuan pada sistem pakar diambil dari seorang atau tim yang merupakan pakar pada bidang tertentu. Dengan sistem pakar ini, orang awam pun dapat dibantu dalam menyelesaikan masalahnya atau hanya sekedar mencari informasi berkualitas yang sebenarnya hanya dapat diperoleh dari para pakar di bidangnya.

Terdapat dua bagian utama dalam sistem pakar, yaitu [Tur92] :

1. Lingkungan pengembangan, digunakan untuk memasukkan pengetahuan ke dalam basis pengetahuan sistem pakar. Lingkungan ini juga dapat digunakan untuk mengubah, menghapus, atau menambah pengetahuan.
2. Lingkungan konsultasi, digunakan oleh pengguna untuk berkonsultasi dengan sistem pakar mengenai masalah yang dihadapinya sehingga pengguna dapat memperoleh solusi untuk permasalahan tersebut.



Gambar 1 Struktur Sistem Pakar [Tur92]

Pada gambar 1 ditunjukkan struktur dari suatu sistem pakar. Dalam struktur tersebut, terdapat tiga komponen utama yang menyusun suatu sistem pakar, yaitu basis pengetahuan, mesin inferensi, dan antarmuka. Basis pengetahuan adalah komponen sistem pakar yang menyimpan pengetahuan dalam suatu domain tertentu, pengetahuan tersebut disimpan dalam bentuk fakta atau aturan (*rules*). Mesin inferensi adalah komponen sistem pakar yang melakukan penalaran

(inferensi) terhadap fakta yang diberikan berdasarkan aturan yang ada dalam basis pengetahuan. Antarmuka merupakan komponen yang menangani interaksi antara sistem pakar dengan pengguna.

Di dalam melakukan penalaran, sistem pakar dapat menggunakan dua cara, yaitu penalaran secara *forward chaining*, atau penalaran secara *backward chaining*. Penalaran *forward chaining* dimulai dengan sekumpulan fakta yang diketahui. Berdasarkan fakta yang diketahui dan aturan yang dimiliki sistem, mesin inferensi akan menurunkan sejumlah fakta baru, hal ini akan dilakukan secara terus menerus sampai tidak ada lagi aturan yang bisa diaktifkan. Penalaran *backward chaining* mencoba membuktikan suatu hipotesis berdasarkan aturan yang ada dengan mengumpulkan sejumlah fakta dari pengguna. Dalam sistem pakar yang dibangun, pendekatan yang dipakai adalah *forward chaining*.

II.2. Definisi Kanker Darah (Leukimia)

Leukimia (kanker darah) adalah jenis penyakit kanker yang menyerang sel-sel darah putih yang diproduksi oleh sumsum tulang [Iss94]. Pada Sumsum tulang di dalam tubuh manusia memproduksi tiga tipe sel darah diantaranya sel darah putih, sel darah merah, platelet. Pada tabel 1 diberikan penjelasan mengenai tipe sel darah.

Tabel 1 Tabel Jenis Sel darah [Iss94]

Gambar sel	Penjelasan
	Sel darah putih : berfungsi sebagai daya tahan tubuh melawan infeksi,
	sel darah merah : berfungsi membawa oksigen kedalam tubuh
	platelet (Keping darah) : bagian kecil sel darah yang membantu proses pembekuan darah.

Pada kondisi normal, sel sel akan tumbuh dan mati sesuai dengan mekanisme yang diatur oleh tubuh sehingga sel yang sudah tua akan mati dan digantikan oleh sel yang muda. Pada penderita kanker, proses ini tidak berjalan sebagaimana mestinya yang mana sel yang sudah tua tidak mati namun tumbuh dengan membabi buta. Pada penderita Leukimia, proses yang tidak normal ini terjadi pada sel darah.

Pada orang normal, pembentukan sel darah terjadi di sumsum tulang. Proses produksi terjadi sedemikian rupa sehingga jumlah sel yang diproduksi sesuai dengan jumlah sel yang akan mati. Pada penderita leukimia, sumsum tulang memproduksi sel darah putih abnormal (sel leukimia). Jumlah sel leukimia yang diproduksi kian hari kian meningkat sehingga mengganggu fungsi darah secara keseluruhan.

Pada kasus leukimia (kanker darah), darah putih diproduksi melebihi yang

sepatutnya. Sel darah putih berasal dari sel stem di sumsum tulang. Leukimia terjadi jika proses pematangan dari stem sel menjadi sel darah putih mengalami gangguan dan menghasilkan perubahan kearah keganasan. Perubahan tersebut seringkali melibatkan penyusunan kembali bagian dari kromosom (bahan genetik sel yang kompleks).

Penyusunan kembali kromosom (translokasi kromosom) tidak terkendali dan menjadi ganas. Pada akhirnya sel-sel ini menguasai sumsum tulang dan menggantikan tempat dari sel-sel yang menghasilkan sel-sel darah yang normal. Kanker ini juga bisa menyusup ke dalam organ lainnya seperti hati, limpa, kelenjar getah bening, ginjal dan otak.

Tipe leukimia dapat dibedakan berdasarkan perkembangan penyakitnya, yaitu [Iss94] :

1. Leukimia Kronis

Pada awal penyakit, sel-sel darah yang abnormal masih dapat mengerjakan pekerjaan mereka, dan orang-orang dengan leukimia kronis mungkin tidak mempunyai gejala-gejala apa saja. Secara perlahan, leukimia kronis memburuk. Ia menyebabkan gejala-gejala ketika jumlah sel-sel leukimia didalam darah meningkat.

2. Leukimia Akut

Sel-sel darah tidak dapat mengerjakan pekerjaan normal mereka. Jumlah sel-sel abnormal meningkat secara cepat. Leukimia akut memburuk secara cepat.

Tipe leukimia juga dapat dikelompokkan berdasarkan jenis sel darah putih yang terpengaruh. Leukimia dapat timbul pada sel-sel lymphoid atau sel-sel *myeloid*. Leukimia yang mempengaruhi sel-sel lymphoid disebut *lymphocytic* leukimia. Leukimia yang mempengaruhi sel-sel *myeloid* disebut *myeloid* leukimia atau *myelogenous leukemia*. Berdasarkan jenis sel darah yang diserang, ada 2 macam leukimia yaitu:

1. Leukimia Limfositik

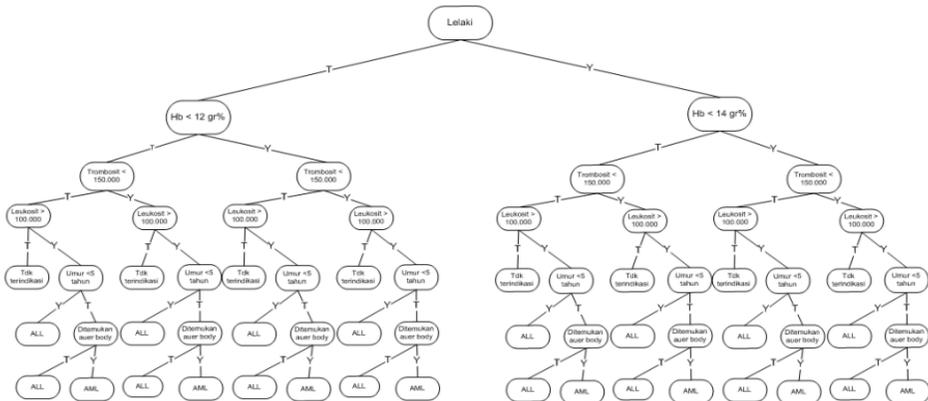
Merupakan tipe leukimia yang paling sering terjadi pada anak-anak. Penyakit ini juga dapat terjadi pada lansia (lanjut usia) yang berumur 65 tahun atau lebih.

2. Leukimia Mielogenosa

Leukimia jenis ini lebih sering terjadi pada dewasa daripada anak-anak. Tipe ini dahulunya disebut leukimia nonlimfositik akut.

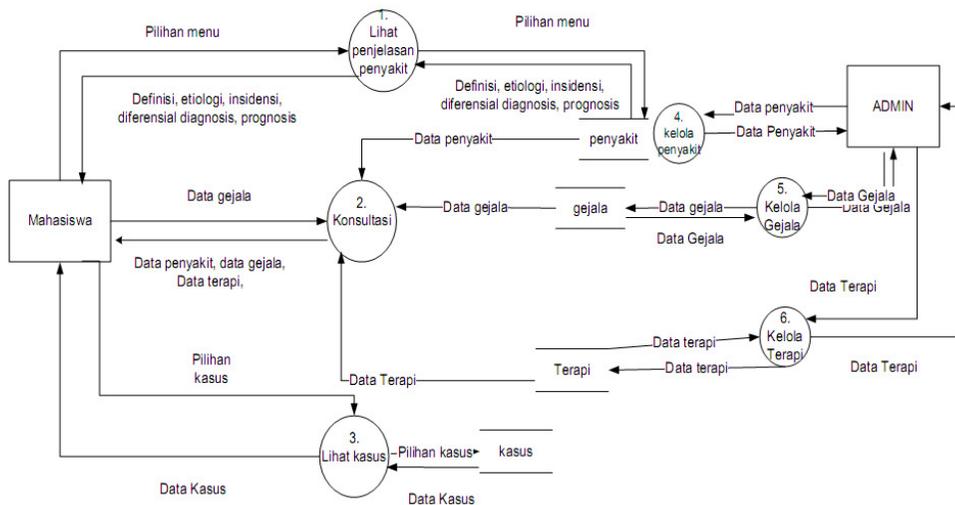
III. Analisis dan Disain

Analisis terhadap domain kanker darah pada anak dimulai dengan mempelajari materi yang berasal dari studi pustaka dari [Iss94] dan [Gar00] dan wawancara dengan seorang dokter sebagai pakar. Hasil akuisisi pengetahuan tersebut dirangkum dalam sebuah pohon keputusan, seperti ditunjukkan pada gambar 2.



Gambar 2 Pohon Keputusan untuk Kanker Darah pada Anak

Proses utama yang terjadi di dalam sistem adalah melihat penjelasan penyakit, konsultasi, melihat kasus penyakit, mengelola penyakit, mengelola gejala, mengelola terapi. Proses penalaran secara *forward chaining* dilakukan di dalam proses konsultasi. Pada gambar 3 ditunjukkan *data flow diagram level 1* untuk sistem pakar yang dibangun.

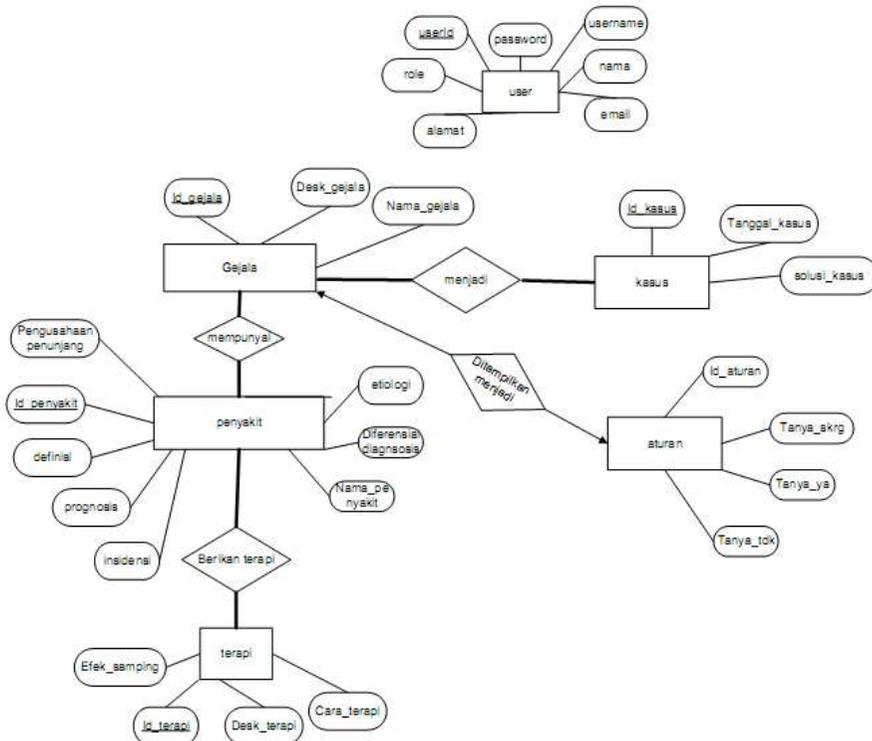


Gambar 3 Data Flow Diagram level 1 untuk Sistem Pakar

Basis pengetahuan yang digunakan disimpan dalam suatu basis data dengan diagram *entity relationship* (ERD) yang ditampilkan pada gambar 4. Adapun tabel yang diturunkan dari ERD pada gambar 4 adalah :

- a. Tabel Gejala(id_gejala,nama_gejala, desk_gejala)
- b. Tabel Penyakit (id_penyakit, nama_penyakit, definisi, etiologi, insidensi, penunjang, prognosis, dif_prognosis)

- c. Tabel Terapi(id_terapi, desk_terapi, cara_terapi, efek_samping)
- d. Tabel Kasus (id_kasus, tanggal, solusi, id_gejala)
- e. Tabel Penyakit_gejala (id_penyakit, id_gejala)
- f. Tabel BerikanTerapi(id_penyakit, id_terapi)
- g. Tabel Gejala_kasus (id_gejala, id_kasus)
- h. Tabel Aturan(id_aturan,tanya_skrng,tanya_ya, tanya_tidak)
- i. Tabel User(userid, namauser, password, nama, alamat, email, role)



Gambar 4 Diagram *Entity Relationship* untuk Sistem Pakar

IV. Implementasi Sistem

Sistem pakar diimplementasikan sebagai suatu situs web yang dibangun menggunakan php dengan mySQL sebagai pengelola basis datanya. Pada tabel 2 ditunjukkan contoh pengujian yang dilakukan pada modul konsultasi.

Tabel 2 Contoh Pengujian Pada Modul Konsultasi

Test	Kasus	Input	Valid Output	Output	Status
1.	Jenis kelamin lelaki	Ya	Memeriksa	Menampilkan solusi penyakit Yaitu Leukimia Limfostik Akut	OK
	Hemoglobin kurang dari 14 gr%	Ya			
	Trombosit < 150.000	Ya			

*Sistem Pakar Berbasis Web sebagai Alat Bantu Pembelajaran Mahasiswa Kedokteran
untuk Penyakit Kanker Darah pada Anak
(Meiry Yanti Sihombing, Mewati Ayub)*

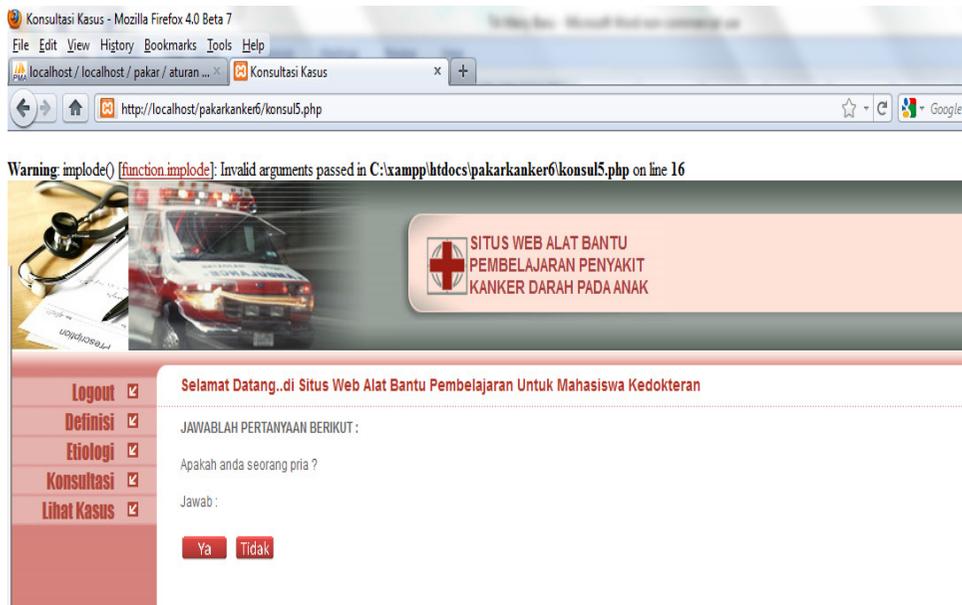
	Leukosit >100.000	Ya			
	Umur < 5 tahun	Ya			
2.	Jenis kelamin lelaki	Ya	Memeriksa	Menampilkan solusi penyakit Yaitu Leukimia Miologenosa Akut	OK
	Hemoglobin kurang dari 14 gr%	Ya			
	Trombosit < 150.000	Ya			
	Leukosit >100.000	Ya			
	Umur < 5 tahun	Tidak			
	Ditemukan <i>auer body</i>	Ya			

Gambar 5 menampilkan halaman antarmuka yang berisi penjelasan mengenai definisi dari kanker darah.



Gambar 5 Antarmuka Halaman Definisi

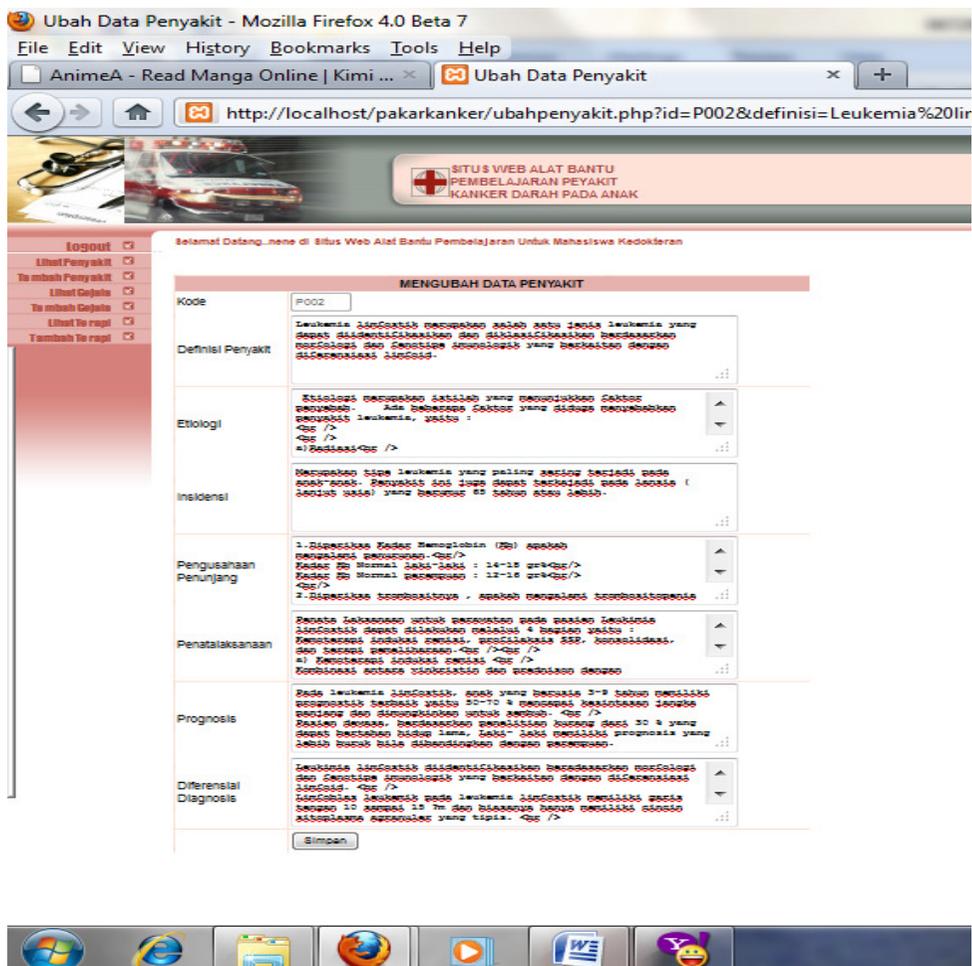
Gambar 6 menampilkan antarmuka konsultasi kasus untuk interaksi antara pengguna dengan sistem, yaitu pengguna menjawab pertanyaan yang diajukan oleh sistem.



Gambar 6 Antarmuka Halaman Konsultasi Kasus

Gambar 7 menampilkan halaman berupa form yang diakses oleh admin untuk mengubah data penyakit.

Sistem Pakar Berbasis Web sebagai Alat Bantu Pembelajaran Mahasiswa Kedokteran untuk Penyakit Kanker Darah pada Anak (Meiry Yanti Sihombing, Mewati Ayub)



Gambar 7 Antarmuka Halaman Ubah Data Penyakit

V. Kesimpulan dan Saran

Beberapa kesimpulan yang dapat ditarik berdasarkan uraian di atas adalah :

1. Sistem pakar berbasis web dapat digunakan untuk membantu mahasiswa kedokteran atau ahli medis sebagai media atau alat bantu pembelajaran mengenai kanker darah pada anak. Pada fitur utama yaitu konsultasi pengguna melalui kasus, sistem akan memberikan solusi mengenai kasus tersebut berupa penjelasan penyakit dan terapi yang harus dilakukan.
2. Sistem dapat melakukan modifikasi data seperti menambah data, mengubah data, dan menghapus data, baik untuk data penyakit, gejala, maupun terapi.
3. Dari segi tampilan antarmuka, sistem menggunakan tampilan yang seragam, dengan letak *link* yang tetap di setiap halamannya. Hal ini dilakukan agar pengguna tidak merasa kesulitan dalam menggunakannya.

Saran-saran untuk penyempurnaan sistem pakar di atas adalah menambahkan

fungsi pemeriksaan *circular reference* pada saat mengubah aturan.

VI. Daftar Pustaka

- [Iss94] Asdie,A. 2000. *Harrison Prinsip-Prinsip Ilmu Penyakit Dalam Edisi 13* (K.Isselbacher,Eugene Braunwald, Jean Wilson, Joseph Martin, Anthony Fauci, Denis Kasper). Jakarta:EGC (Original Published 1994).
- [Dur94] Durkin, J.1994. *Expert Systems Design and Development*. Prentice Hall.
- [Gar00] Garna, H, et.al. 2000. *Pedoman Diagnosis dan Terapi Ilmu Kesehatan Anak*. Bandung: Bagian/SMF Ilmu Kesehatan Anak FKUP/RSHS.
- [Kus03] Kusumadewi, S. 2003. *Artificial Intellegence*. Jakarta:Graha Ilmu.
- [Tur92] Turban, E. 1992. *Expert Systems and Applied Artificial Intelligence*. New York :Macmillan Publishing Company.

Permainan Trading Card Game Magic & Wizard Card Battle

Erico Darmawan Handoyo, Antonius Chandra

Jurusan S1 Teknik Informatika

Fakultas Teknologi Informasi, Universitas Kristen Maranatha

Jl. Prof. Drg. Suria Sumantri no. 65 Bandung 40164

email: khe.wan.xing@gmail.com, st12ik3fr33d0m@yahoo.co.id

Abstract

Trading Card Game is game which prioritize collect cards and use them to set the strategy in the game. Magic & Wizard Card Battle is a Trading Card computer game that developed using programming language Action Script 3.0 and run using runtime Adobe Flash Player. The goal of Magic & Wizard Card Battle game is make enemy point become 0. There are two main features in Magic & Wizard Card Battle game, they are Single player, and multiplayer mode. In single player mode, player compete against CPU, and in multiplayer mode, player can compete against another player to test their skill. Multiplayer mode in this game running using SmartFoxServer PRO, SmartFoxServer PRO which is a multi-platform game server designed to integrated with Adobe Flash.

Keywords: Adobe Flash Player, Action Script 3.0, SmartFoxServer PRO, Flash Game Multiplayer.

1. Pendahuluan

1.1. Latar Belakang

Trading Card Game (TCG) adalah permainan yang mengutamakan mengumpulkan kartu dan menggunakannya untuk mengatur strategi di permainan. Setiap permainan *TCG* memiliki seperangkat peraturan dasar yang menggambarkan tujuan dari para pemain, kategori dari kartu yang digunakan dalam permainan, dan aturan-aturan dasar kartu yang berinteraksi. Setiap kartu akan memiliki teks tambahan yang menjelaskan efek kartu tertentu pada pertandingan.

Pemain memilih kartu yang akan digunakan pada *deck* mereka dari tempat kartu yang tersedia, tidak seperti permainan kartu tradisional seperti poker atau UNO dimana isi *deck* adalah terbatas dan telah ditentukan. Hal ini memungkinkan pemain untuk menyesuaikan strategi *deck* mereka untuk mengambil keuntungan dari interaksi kartu yang menguntungkan dan kombinasi.

Pada saat permainan, pemain melewati tahap untuk memainkan kartu dan melakukan aksi. Urutandan judul untuk setiap tahap berbeda-beda tergantung system dari permainan, tetapi langkah-langkah ini adalah ciri khas:

- Draw Card(s) – Mengambil kartu dari deck.
- Play Card(s) – Menggunakan kartu di tangan untuk berinteraksi.
- Conflict – Cara utama untuk memenangkan permainan.
- Discard Card(s) – Membuang maksimum kartu ditangan.

1.2. Rumusan Masalah

Berdasarkan latar belakang di atas, maka masalah yang muncul adalah:

1. Bagaimana caranya membuat permainan *Trading Card Game* yang cepat dipahami oleh setiap orang.
2. Bagaimana caranya membuat permainan *Trading Card Game* yang dapat membuat setiap pemain bebas berkreasi dalam berstrategi.

1.3. Tujuan

Adapun tujuan dari pembuatan game ini:

1. Membuat *Trading Card Game* bernama Magic & Wizard Card Battle yang mudah dipahami setiap orang.
2. Membuat *Trading Card Game* bernama Magic & Wizard Card Battle yang dapat membuat setiap pemain bebas berkreasi dalam berstrategi.

1.4. Batasan Masalah

Berikut ini adalah batasan-batasan yang diterapkan dalam pembuatan tugas akhir ini:

1. Banyaknya kartu yang ada di permainan ini hanya ada 200 macam.
2. Permainan ini hanya dapat dimainkan user vs AI dan user vs user.

2. Landasan Teori

2.1. Aturan Permainan Magic & Wizard Card Battle

Permainan Magic & Wizard Card Battle ini memiliki beberapa peraturan dasar.

2.1.1. Deck

Deck adalah kumpulan kartu-kartu yang digunakan untuk bertanding, jumlah kartu dalam 1 deck adalah dari 40-50 kartu. Biasanya dalam 1 deck dapat menggunakan 3 buah kartu dengan nama yang sama, tetapi ada beberapa kartu yang hanya dapat digunakan 1 atau 2 buah dengan nama yang sama.

2.1.2. Poin

Poin adalah nilai yang diberikan sebagai nyawa dari pemain. Poin yang diberikan pada awal permainan adalah 4000. Nilai Poin minimum adalah 0 dan maksimum 10000.

2.1.3. Syarat Menang

Dalam permainan Magic & Wizard Card Battle pemain akan menang jika:

1. Poin musuh mencapai angka 0.
2. Musuh kita tidak dapat menarik kartu dari deck pada saat dia harus menarik kartu.

2.2. Action Script 3.0

ActionScript adalah bahasa pemrograman berbasis ECMAScript. ActionScript sering digunakan untuk mengembangkan situs web dan perangkat lunak yang menggunakan platform Adobe Flash Player. ActionScript juga dapat digunakan untuk membangun aplikasi basis data dan aplikasi robotik.

ActionScript pada awalnya dirancang untuk mengendalikan animasi vektor 2D sederhana yang dibuat di Adobe Flash. Pada awalnya difokuskan pada animasi, versi awal konten Flash menawarkan beberapa fitur interaktifitas dan sehingga memiliki kemampuan scripting sangat terbatas. Versi selanjutnya ditambahkan fungsi yang memungkinkan untuk pembuatan game berbasis Web dan RIA (Rich Internet Applications). Sekarang, ActionScript cocok untuk digunakan dalam beberapa aplikasi database, dan dalam robotika dasar.

2.3. SmartFoxServer

SmartFoxServer adalah multi-platform server socket yang dirancang untuk diintegrasikan dengan Adobe Flash, memungkinkan pengembang untuk secara cepat mengembangkan aplikasi multiuser dan game. SmartFoxServer dibuat dengan mengutamakan game multiplayer dan menyediakan tool yang kuat untuk menciptakan berbagai game turn-based dan real-time yang canggih. Benar-benar tidak ada batasan untuk jumlah aplikasi yang dapat dibuat.

SmartFoxServer mendukung semua Flash players mulai dari versi 6 sampai terbaru Flash Player 10 dan menyediakan rich set API untuk ActionScript 2.0, dan 3.0 Java, .Net/Unity3D, Shockwave dan banyak lagi.

2.4. MYSQL

MySQL adalah salah satu jenis database server yang sangat terkenal. Kepopulerannya disebabkan MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses databasenya. Selain itu bersifat free.

MySQL termasuk jenis RDBMS (Relational Database Management System), itulah sebabnya istilah seperti tabel, baris, dan kolom digunakan pada MySQL. Pada MySQL, sebuah database mengandung satu atau sejumlah tabel. Tabel terdiri dari sejumlah baris dan setiap baris mengandung satu atau beberapa kolom.

2.5. ENTITY RELATIONSHIP DIAGRAM

Basis data adalah sekumpulan informasi atau berkas data yang terorganisasi ke dalam tata cara yang khusus dan terkomputerisasi. Basis data juga merupakan himpunan kelompok data yang saling berkaitan satu sama lain, yang diorganisasikan sedemikian rupa, sehingga tidak terdapat duplikasi yang tidak perlu, agar dapat diolah atau dieksplorasi secara cepat dan mudah untuk menghasilkan informasi.

2.6. Adobe Flash

Adobe Flash adalah platform multimedia yang digunakan untuk membuat animasi, video, dan halaman web. Flash sekarang sering digunakan untuk iklan dan game. Flash menggunakan bahasa pemrograman bernama ActionScript. Pada

dasarnya, Flash terdiri dari tiga komponen yaitu : Adobe Flash Player, file format, dan authoring tool/IDE (Integrated Development Environment).

3. Analisis dan Desain Aplikasi

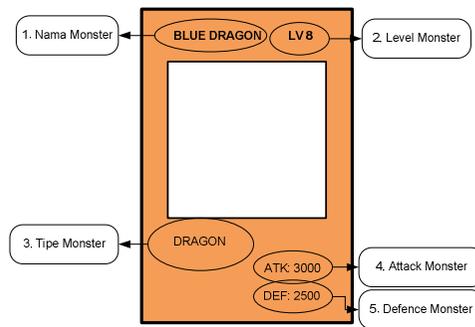
3.1. Rincian Game Magic & Wizard Card Battle

Berikut ini adalah rincian dari game Magic & Wizard Card Battle.

3.1.1. Kartu

Inti dari permainan Magic & Wizard Card Battle adalah kartu. Kartu yang disediakan pada permainan ini adalah 3 macam yang terdiri dari:

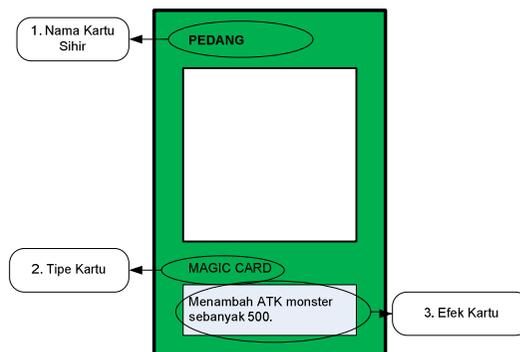
a. Monster Card



Gambar 1. Monster Card

Monster Card adalah kartu yang digunakan untuk bertarung dan mengalahkan musuh.

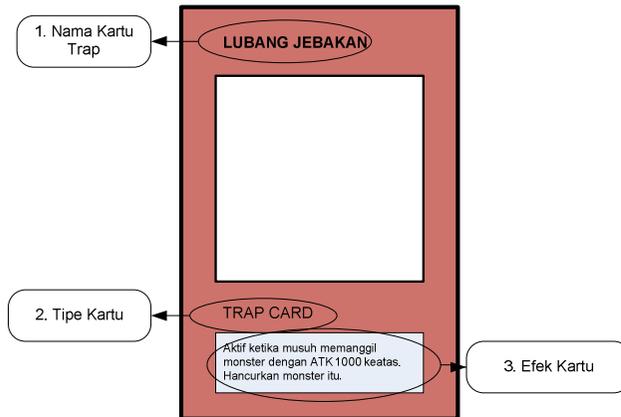
b. Magic Card



Gambar 2. Magic Card

Kartu Sihir adalah kartu yang memiliki efek khusus, Kartu Sihir hanya dapat digunakan pada saat giliran kita saja. Biasanya Kartu Sihir digunakan untuk membantu kita pada saat menyerang.

c. Trap Card

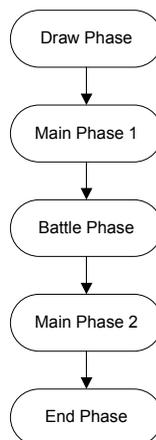


Gambar 3. Trap Card

Kartu Trap adalah kartu yang memiliki efek khusus sama seperti Kartu Sihir, tetapi perbedaan terbesar dari Kartu Trap dan juga Kartu Sihir adalah Kartu Sihir hanya dapat digunakan pada saat giliran kita, tetapi Kartu Trap dapat diaktifkan pada giliran musuh. Kartu Trap biasanya digunakan untuk menghalangi musuh untuk melakukan sesuatu. Untuk mengaktifkan Kartu Trap kita perlu menaruhnya tertutup terlebih dahulu.

3.1.2. Giliran

Cara yang digunakan untuk menentukan giliran pemain yang jalan pertama adalah dengan melakukan suit(gunting, batu, kertas), pemain yang menang suit dapat menentukan siapa yang jalan pertama. Setiap giliran pemain akan melewati beberapa tahap.



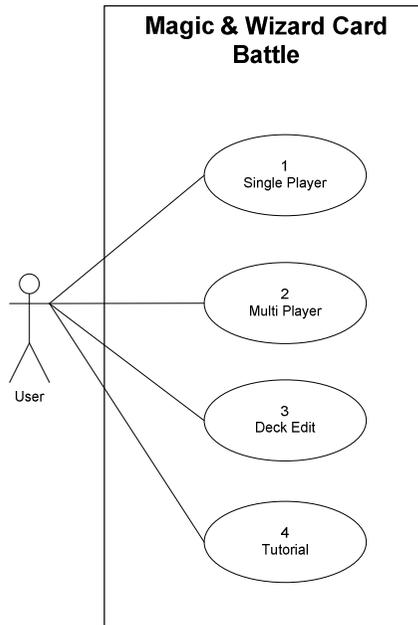
Gambar 4. Tahap pada giliran

3.1.3. Cara mengurangi Poin

Ada 2 cara mengurangi Poin:

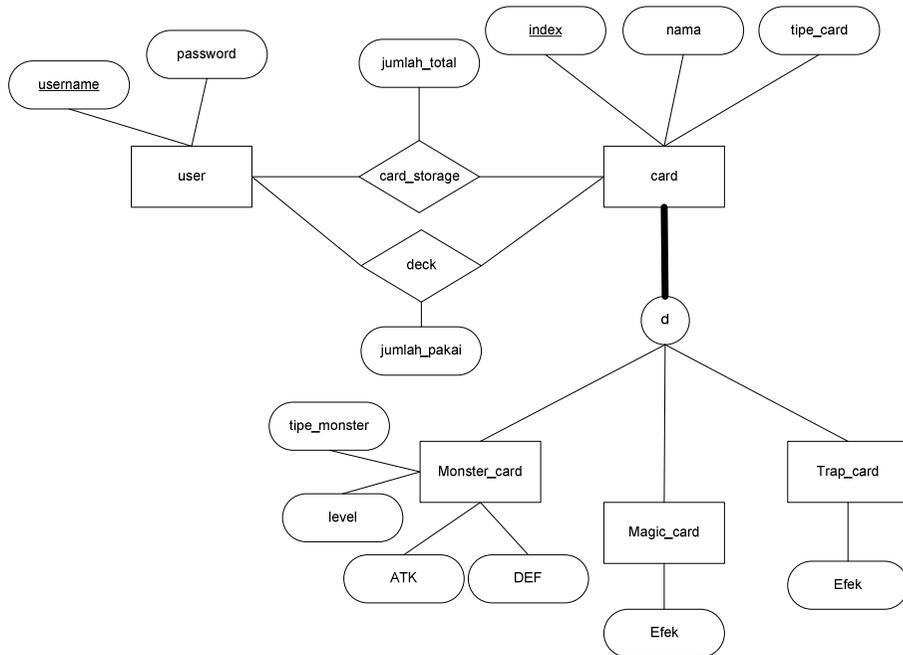
1. Menggunakan kartu sihir atau trap yang memiliki efek mengurangi Poin.
2. Mengurangi Poin dengan Battle Damage.

3.2. Arsitektur Game



Gambar 5. Use Case Diagram

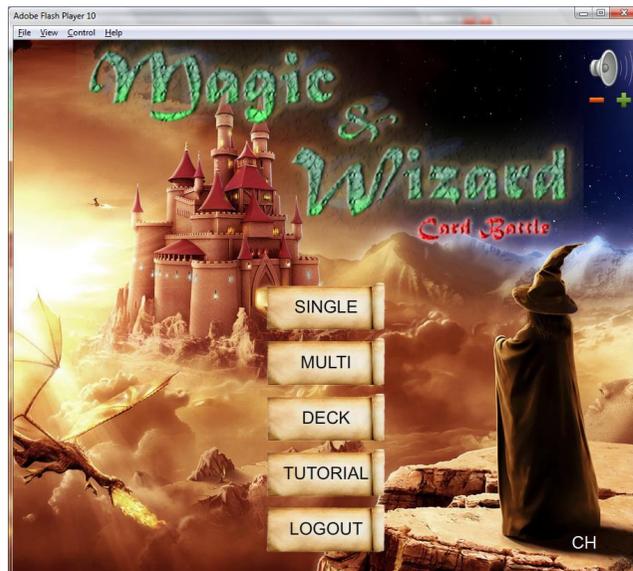
Pada Gambar 5 menunjukkan fitur-fitur yang disediakan pada game Magic & Wizard Card Battle.



Gambar 6. ER-Diagram

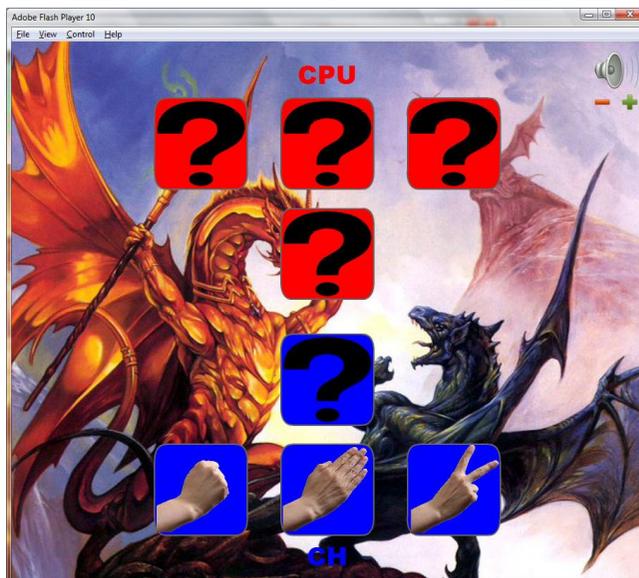
Pada Gambar 6 menunjukkan ER-Diagram untuk database Magic & Wizard Card Battle.

3.3. Desain Aplikasi



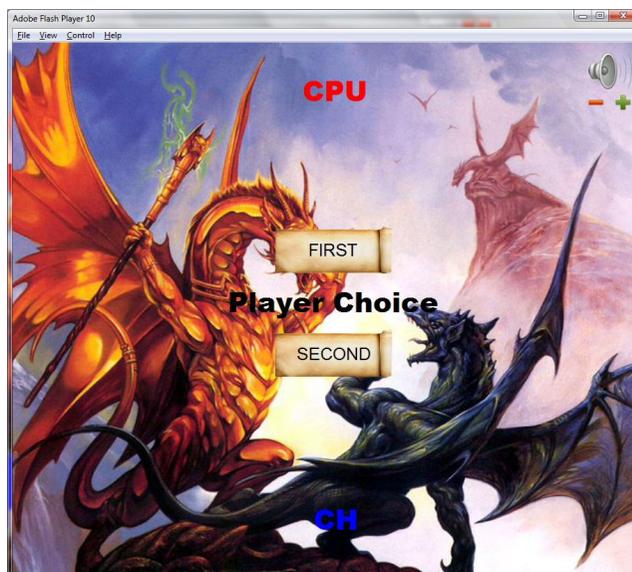
Gambar 7. Main Menu

Gambar 7. Merupakan tampilan pada main menu, pada main menu pemain dapat memilih Single untuk bermain melawan CPU, Multi untuk bermain melawan pemain lain, Deck untuk mengubah deck pemain, Tutorial untuk melihat peraturan dari permainan ini, dan Logout untuk keluar.



Gambar 8. Rock, Paper, Scissors

Gambar 8. Menampilkan tampilan pada saat melakukan suit.



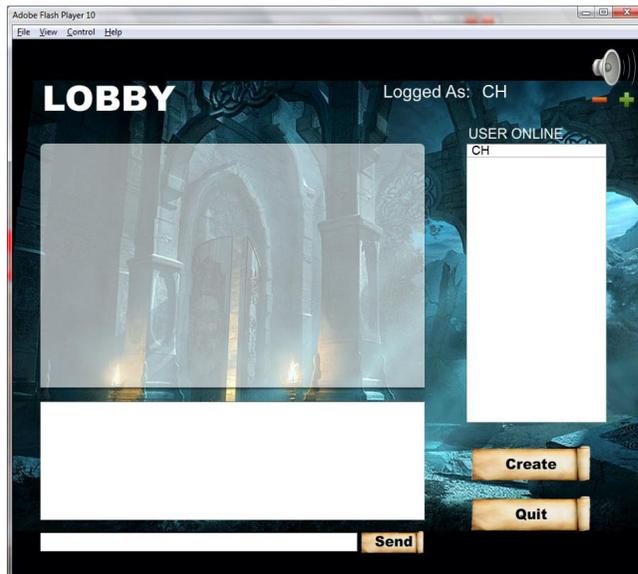
Gambar 9. First / Second

Gambar 9. Menampilkan tampilan pada saat memilih First / Second.



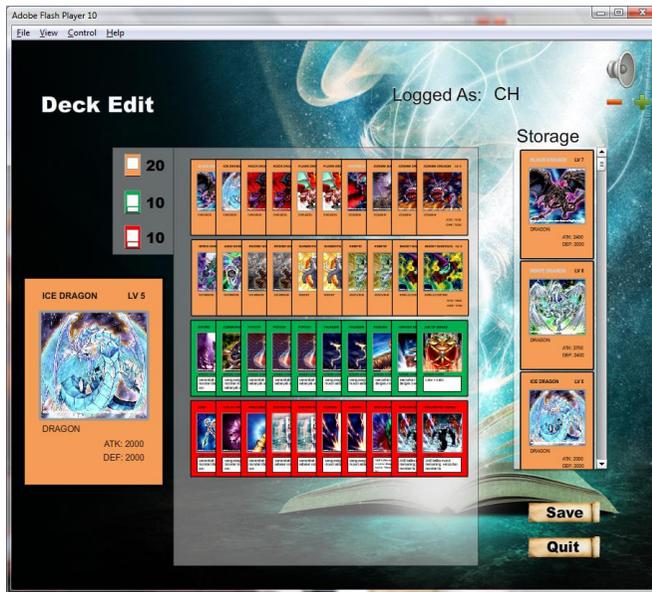
Gambar 10. Arena

Gambar 10. Menampilkan tampilan pada saat pemain bertarung melawan CPU maupun melawan pemain lain.



Gambar 11. Lobby

Gambar 11. Menampilkan tampilan pada saat pemain berada di Lobby, pada saat pemain berada di Lobby, pemain dapat melakukan chatting, Membuat Room untuk bertarung, bergabung dengan Room yang telah dibuat oleh pemain lain.



Gambar 12. Deck Edit

Gambar 12. Menampilkan tampilan pada saat pemain berada di Deck Edit, pada saat pemain berada di Deck Edit, pemain dapat mengeluarkan kartu dari deck, menambah kartu ke deck, dan melakukan save deck.

4. Kesimpulan dan saran

4.1. Kesimpulan

Kesimpulan yang didapatkan berdasarkan hasil pembuatan dan pengujian permainan Magic & Wizard Card Battle adalah:

1. Permainan Magic & Wizard Card Battle cepat dipahami setiap orang.
2. Permainan Magic & Wizard Card Battle membuat pemain bebas berkreasi dalam berstrategi.

4.2. Saran

Permainan Magic & Wizard Card Battle ini masih dapat dikembangkan dengan beberapa hal baru seperti:

1. Menambah jenis kartu baru.
2. Mengganti desain dan tampilan agar lebih menarik.
3. Menambah beberapa peraturan baru.

5. Daftar Pustaka

- [Imb06] Imbar, Radiant Victor, Suteja, Bernard Renaldy. 2006. **Pemrograman Web Commerce dengan Oracle dan ASP**. Bandung: Penerbit Informatika
- [Ros08] Rosenzweig, Gary. **ActionScript 3.0 Game Programming University**; USA: Que Publishing, 2008.

<http://www.smartfoxserver.com/docs/>

Aplikasi Pengelolaan PT Makmur Abadi Mandiri dengan Sistem Akuntansi dan DSS dalam Menentukan Pemasok

Diana Trivena Y, Suntono

Jurusan S1 Teknik Informatika

Fakultas Teknologi Informasi, Universitas Kristen Maranatha

Jl. Prof. Drg. Suria Sumantri no. 65 Bandung 40164

email: diana.ty@eng.maranatha.edu, rythem_mix@yahoo.com

Abstract

“Aplikasi pengelolaan PT MAKMUR ABDI MANDIRI” is an application for perfection of bussines procces that happen with adjustment of market need today. This application build for handle selling, buying, restock inventori that included with Accounting and search fiture. This application supported by Deccision Suport System for choosing supplier that calculates values of criteria and compares it. With this application pleased can help in handling data and make deccision that correct and fleksible. This application build using C# programming language and SQL Server 2008 database.

Key Word : Deccision Suport System, System Accounting

I. Pendahuluan

PT Makmur Abdi mandiri merupakan pabrik dompet dan ikat pinggang yang berlokasi di Kapuk Kamal Muara II, gg Cendana No.7, Jakarta. Kesulitan terjadi di pabrik tersebut karena setiap minggu harus melakukan pencetakan laporan data barang yang jenisnya banyak. Penyisipan data yang ada juga acak-acakan. Pembuatan struk belanja juga mengalami kendala yaitu kesulitan sinkronisasi data harga barang baru yang sebenarnya mengalami kenaikan. Semua data- data tersebut disimpan secara manual di dalam komputer. Selain itu kendalah utamanya yaitu efektifitas dalam pemilihan suplier dari segi kualitas barang, harga barang, dan ketepatan waktu.

Pembuatan sistem informasi dimulai dengan melakukan analisis. Dari hasil analisis dilakukan perancangan sistem akuntansi yang sesuai dengan intergrasi dari modul penjualan dan pembelian serta diharapkan dapat mempermudah dalam mengelola pengimputan data, penyimpanan data, pembelian dan penjualan, data barang , data pemasok, data pelanggan, *order* barang, laba rugi, perubahan modal, aliran kas, dan neraca, serta sistem yang dilengkapi dengan DSS untuk efektifitas dalam pemasokan barang melalui suplier.

Aplikasi ini menangani:

1. Penyimpanan data barang, pembelian, penjualan, pemasok, pelanggan, order barang, laporan laba rugi, neraca, buku besar, dan jurnal umum.

2. Pemilihan pemasok saat melakukan pembelian berdasarkan order pembelian.
3. Pembuatan laporan akuntansi keuangan.

II. Kajian Teori

II.1. Sistem Informasi

“Sistem informasi adalah sekumpulan *hardware*, *software*, *brainware*, prosedur dan atau aturan yang diorganisasikan secara integral untuk mengolah data menjadi informasi yang bermanfaat guna memecahkan masalah dan pengambilan keputusan”(Jogiyanto,2001:13)

Sebuah sistem informasi merupakan suatu kumpulan atau seperangkat komponen yang berhubungan dan mendukung dengan fungsi mengumpulkan, memproses, menyimpan dan mendistribusikan informasi. Hasil dari proses tersebut digunakan pihak manajemen sebagai suatu dasar dalam pembuatan keputusan organisasi. Selain itu, sistem informasi yang baik juga dapat membantu dalam hal penganalisaan dan visualisasi masalah dalam penciptaan produk baru.

II.2. DSS (*Decision Support System*)

Pengambilan keputusan adalah pemilihan beberapa tindakan alternatif yang ada untuk mencapai satu atau beberapa tujuan yang telah ditetapkan (Turban, 2005). Sistem pendukung keputusan adalah suatu sistem berbasis komputer yang menghasilkan berbagai alternatif keputusan untuk membantu manajemen dalam menangani berbagai permasalahan yang terstruktur ataupun tidak terstruktur dengan menggunakan data dan model (McLeod,1998:96).

Menurut Turban (1999), komponen Sistem Pengambilan Keputusan dapat dibangun dari subsistem berikut ini,

1. Subsistem Manajemen Data (*Data Management Subsystem*), meliputi basis data – basis data yang berisi data yang relevan dengan keadaan dan dikelola *software* yang disebut DBMS (*Database Management System*).
2. Subsistem Manajemen Model (*Model Management Subsystem*), berupa sebuah paket *software* yang berisi model-model finansial, statistik, *management science*, atau model kuantitatif, yang menyediakan kemampuan analisa dan *software management* yang sesuai.
3. Subsistem Manajemen Pengetahuan (*Knowledge Management Subsystem*), merupakan subsistem (*optional*) yang dapat mendukung subsistem lain atau berlaku sebagai komponen yang berdiri sendiri (*independent*).
4. Subsistem Antarmuka Pengguna (*User Interface Subsystem*), merupakan subsistem yang dapat dipakai oleh *user*

II.3. Akuntansi

Akuntansi adalah suatu proses mencatat, mengklasifikasi, meringkas, mengolah dan menyajikan data, transaksi serta kejadian yang berhubungan dengan keuangan sehingga dapat digunakan oleh orang yang menggunakannya dengan mudah dimengerti untuk pengambilan suatu keputusan serta tujuan lainnya.

Penerapan akuntansi dipandang dari jenis usahanya dibagi menjadi 3 jenis yaitu:

- Akuntansi pada Perusahaan Jasa.
Perusahaan yang menjual jasa kepada konsumen.
- Akuntansi pada Perusahaan Perdagangan.
Perusahaan yang bergerak pada jual beli produk/ barang.
- Akuntansi pada Perusahaan Manufaktur.
Perusahaan yang memfokuskan diri pada produksi barang.

Pihak- pihak yang membutuhkan informasi Akuntansi yaitu:

1. Pimpinan Perusahaan
2. Pemilik Perusahaan
3. Pekerja/Karyawan
4. Kreditur dan Calon Kreditur
5. Investor dan Calon Investor
6. Pemerintah
7. Pemungut Pajak
8. Kependudukan

Sisklus pembuatan Akuntansi yaitu:

- Pencatatan data ke dalam dokumen sumber/bukti transaksi
- Penjurnalan, yaitu menganalisa dan mencatat transaksi dalam jurnal (buku harian)
- Melakukan posting ke Buku Besar yaitu memindahkan debit dan kredit dari jurnal ke akun Buku Besar.
- Penyusunan Neraca Saldo yaitu menyiapkan Neraca saldo untuk mengecek keseimbangan Buku Besar.
- Membuat ayat jurnal penyesuaian dan memasukkan jumlahnya pada Neraca Saldo.
- Membuat ayat- ayat penutup yaitu menjurnal dan memindahbukukan ayat-ayat penutup.
- Menyusun laporan keuangan yaitu laporan laba-rugi, perubahan modal dan neraca

Pada dasarnya proses akuntansi akan membuat output laporan rugi laba, laporan buku besar, dan laporan neraca pada suatu perusahaan atau organisasi lainnya. Pada suatu laporan akuntansi harus mencantumkan nama perusahaan, nama laporan, dan tanggal penyusunan atau jangka waktu laporan tersebut untuk memudahkan orang lain memahaminya. Laporan dapat bersifat periodik dan ada juga yang bersifat suatu waktu tertentu saja.

II.4. Basis Data

Basis data dapat diartikan sebagai suatu bentuk penyimpanan informasi yang terpusat agar data-data atau informasi yang ada di dalamnya mudah dicari, dikelola dan digunakan kembali. Basis data dapat kita anggap sebuah lemari arsip, dimana arsip yang ada di dalamnya dapat kita pergunakan lagi dan dengan mudah

kita cari lagi karena biasanya sudah diurutkan berdasarkan urutan nama arsipnya. Basis data terdiri dari tabel–tabel yang berisi data-data, dan dalam tabel–tabel tersebut terdiri atas sebuah baris dan sebuah kolom. Baris dalam basis data disebut juga dengan *record* dan kolom dalam Basis data disebut juga field. Setiap *record* dalam tabel merepresentasikan masing-masing data dan setiap kolom dalam tabel merepresentasikan jenis data yang ada dalam masing-masing kolom tersebut. Model basis data menyatakan hubungan antar rekaman (*record*) yang tersimpan dalam basis data. Ada beberapa model basis data, yaitu : 7

1. *Entity – Relationship Model*
2. *Relational Model*
3. *Object – oriented Model*
4. *Semi – structured Model*
5. *Network Model*
6. *Hierarchical Model*

II.5. AHP(*Analytical Hierarchy Process*)

Menurut Turban (2005), *Analytical Hierarchy Process* (AHP) adalah suatu metode analisis dan sintesis yang dapat membantu proses Pengambilan Keputusan. AHP merupakan alat pengambil keputusan yang *powerful* dan fleksibel, yang dapat membantu dalam menetapkan prioritas-prioritas dan membuat keputusan di mana aspek-aspek kualitatif dan kuantitatif terlibat dan keduanya harus dipertimbangkan.

Dengan mereduksi faktor-faktor yang kompleks menjadi rangkaian “*one on one comparisons*” dan kemudian mensintesa hasil-hasilnya, maka AHP tidak hanya membantu orang dalam memilih keputusan yang tepat, tetapi juga dapat memberikan pemikiran/alasan yang jelas dan tepat. AHP sangat cocok dan flexibel digunakan untuk menentukan keputusan yang menolong seorang decision maker untuk mengambil keputusan yang kualitatif dan kuantitatif berdasarkan segala aspek yang dimilikinya. Kelebihan lain dari AHP adalah dapat memberikan gambaran yang jelas dan rasional kepada decision maker tentang keputusan yang dihasilkan.

Jenis-jenis AHP :

- *Single-criteria*
Pilih satu alternatif dengan satukriteria, Pengambilan keputusan yang melibatkan satu/lebih alternatif dengan satu kriteria.
- *Multi-criteria*
Pengambilan keputusan yang melibatkan satu/lebih alternatif dengan lebih dari satu kriteria Pilih satu alternatif dengan banyak kriteria.

III. Analisis dan Rancangan Sistem

III.1. Proses Bisnis

Proses transaksi dalam perusahaan ini mencakup proses transaksi pembelian dan transaksi penjualan. Proses transaksi pembelian adalah perusahaan melakukan proses pembelian barang kepada *supplier* untuk memenuhi kebutuhan dagangnya. Sedangkan poses transaksi penjualan adalah proses transaksi yang dilakukan oleh perusahaan untuk melakukan kegiatan menjual barang kepada konsumen.

Dalam proses transaksi pembelian, *user* dapat melakukan pemesanan barang. Setelah itu, *user* akan memilih suplier. Dari proses pemilihan suplier, ada 4 hal yang dititik beratkan sebagai penentu keputusan pemilihan suplier yaitu:

- Kualitas barang
- Harga barang
- Ketepatan pengiriman
- Status suplier

Sebelum melakukan pembelian, *user* terlebih dahulu mengatur bobot penilaian. Bobot penilaian terdiri dari presentase dan nilai bobot. Presentase antara 0 sampai 100 % dan. setelah itu sistem secara otomatis akan mengkalkulasi nilai bobot berdasarkan presentase yang *user input*. Setelah itu *user* memilih barang apa sajah yang akan dibeli dan sistem akan menentukan pemasok yang tepat.

Nilai bobot terdiri dari 4 kriteria yaitu: kualitas barang, harga barang, ketepatan waktu pengiriman, dan status pemasok. Dari keempat kriteria tersebut didapat rumus perhitungan yaitu:

$$\text{Nilai} = (\text{Nilai Kualitas Barang} * \text{Bobot Kualitas Barang}) + (\text{Nilai harga Barang} * \text{Bobot Harga Barang}) + (\text{Nilai Ketepatan Waktu Pemasok} * \text{Bobot Waktu Ketepatan Pemasok}) + (\text{Nilai Status Pemasok} * \text{Bobot Status Pemasok})$$

Contoh Penilaian Pemasok:

Contoh Penilaian Untuk Pemasok P01 dengan Kode Barang B01

Jumlah Barang Rusak B01	= 3 buah
Jumlah Barang B01	= 100 buah
Harga Barang B01	= Rp 5000
Harga Barang Termurah B01	= Rp 4500
Status Pemasok P01	= Tidak Ada
Toleransi Status	= 20 (Tidak Ada)
Lama keterlambatan Pemasok P01	= 2 hari
Toleransi Keterlambatan	= 5/hari
Nilai Kualitas P01- B01	= $(100-3)/100 = 0,97$
Nilai Harga P01- B01	= $4500/5000 = 0,9$
Nilai Ketepatan P01-B01	= $(100-(2*5))/100 = 0,9$
Nilai Status P01-B01	= $(100-20)/100 = 0,8$

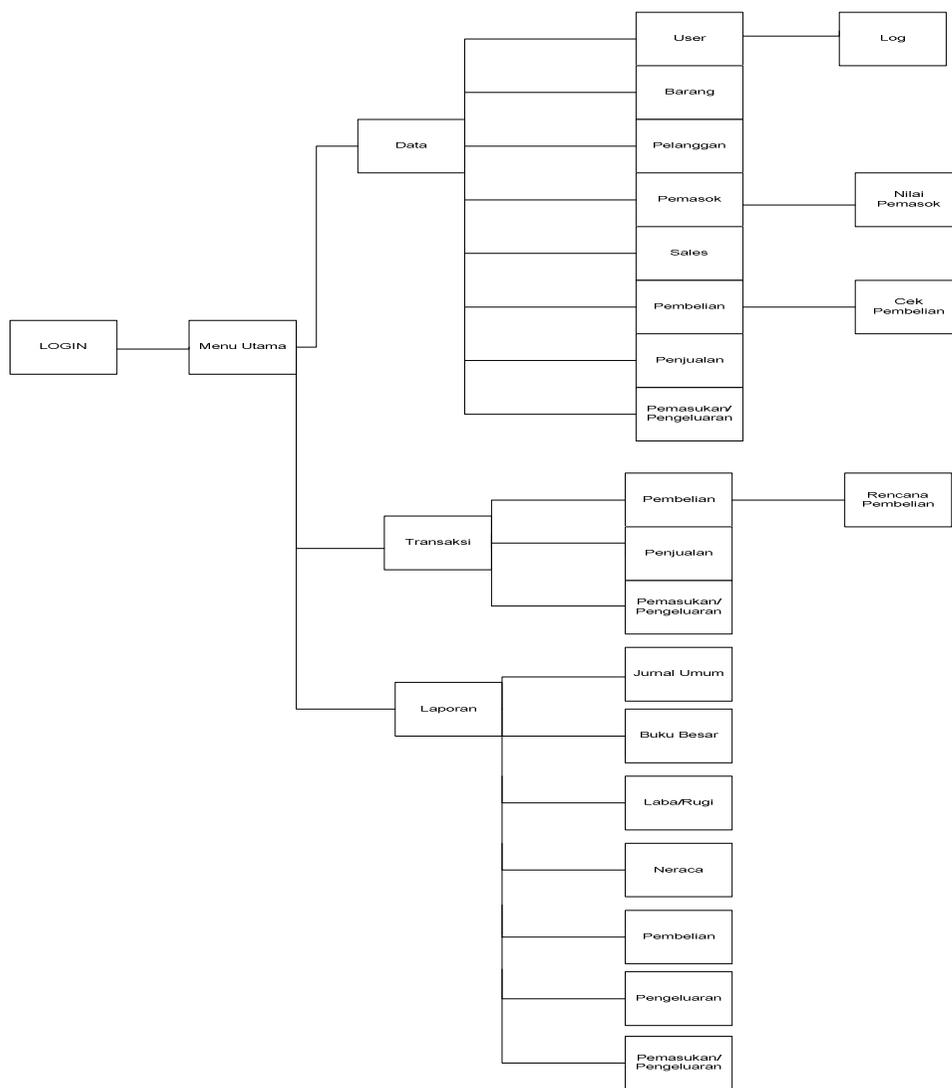
BOBOT Kualitas	= 40%	= 0,4
BOBOT Harga	= 30%	= 0,3
BOBOT Ketepatan	= 20%	= 0,2
BOBOT Status	= 10%	= 0,1

Tabel 1 Penilaian Pemasok Perbarang

Id pemasok	Kd barang	kualitas	harga	ketepatan	Status	Nilai
P-01	B-01	0,97 * 0,4	0,9 * 0,3	0,9 * 0,2	0,8 * 0,1	0,97 * 0,4 + 0,9 * 0,3 + 0,9 * 0,2 + 0,8 * 0,1
P-01	B-01	0,388	0,27	0,18	0,08	0,918

III.2. Perancangan Antarmuka Pengguna

Bagian ini menjelaskan kerangka dari sistem secara garis besar.



Gambar 1 Peta Aplikasi

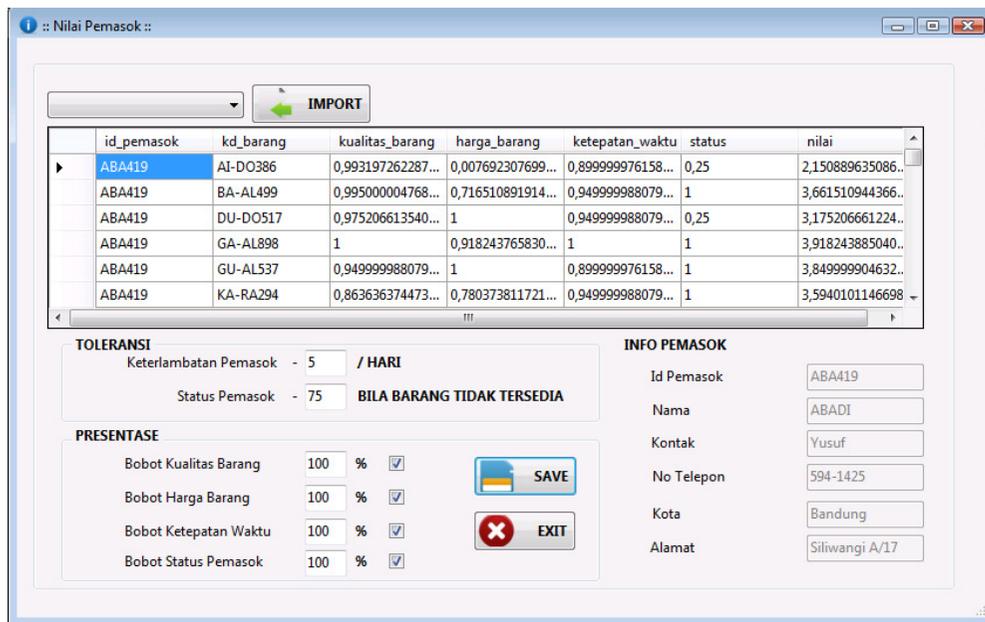
IV. Hasil Penelitian

Bagian ini akan dijelaskan mengenai hasil dari pembuatan aplikasi sistem informasi berbasis C# beserta print screen *user interface* aplikasi. Pada Form Login, pengguna dapat masuk sebagai *user* atau *admin*. Apabila pengguna adalah *admin*, pengguna mempunyai hak penuh dalam menggunakan fitur didalam aplikasi ini, sedangkan *user* hanya dapat menjalankan menu transaksi saja.



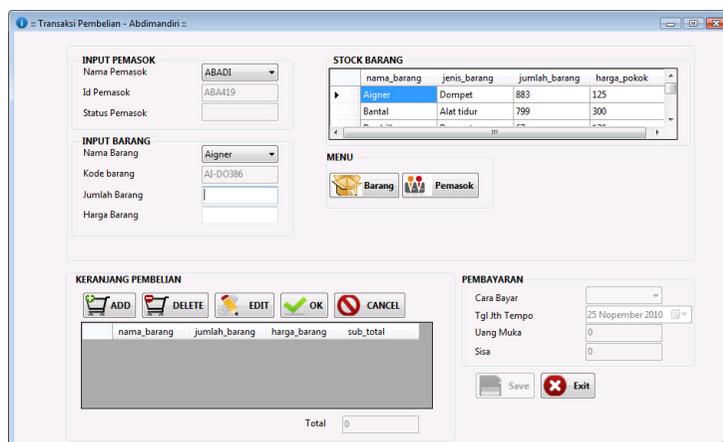
Gambar 2 Tampilan Menu Utama

Pada gambar 2 Form Main Admin, admin dapat mengakses Menu data, transaksi, dan laporan, sedangkan *User* hanya dapat mengakses menu transaksi.



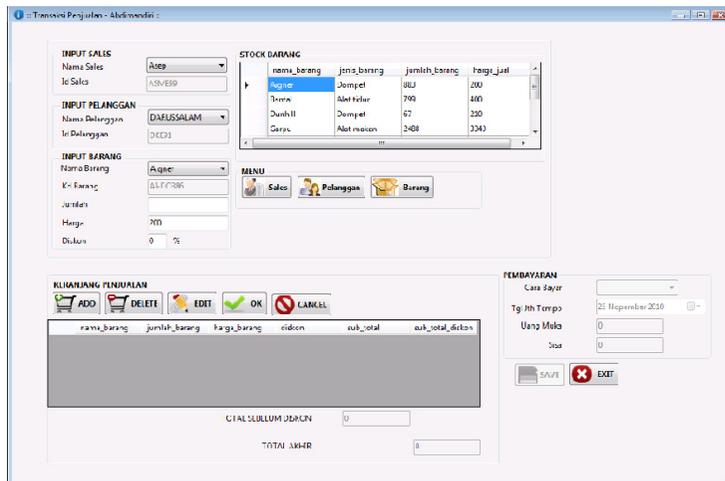
Gambar 3 Tampilan Menu Nilai Pemasok

Pada gambar 3 Menu Nilai Pemasok, *Admin* dapat mengubah aturan penilaian yang dibagi menjadi dua bagian yaitu TOLERANSI dan PRESENTASE. Pada bagian TOLERANSI, *Admin* mengatur pengurangan untuk pelanggaran yang dilakukan. Pada bagian PRESENTASE, *Admin* mengatur kriteria apa saja yang ingin digunakan untuk menilai pemasok, dan seberapa besar nilai yang diinginkan pada masing- masing kriteria dengan mengatur angka atau dengan mematikan *checkboxlist*, maka kriteria tersebut tidak dikalkulasi. Setelah selesai mengkonfigurasi, *Admin* dapat menekan tombol simpan untuk memperbaharui nilai pemasok. Tombol *Export* digunakan untuk menampilkan table nilai pemasok. Untuk keluar dari menu nilai pemasok *admin* dapat menekan tombol *EXIT*



Gambar 4 Tampilan Menu Transaksi Pembelian

Pada gambar 4 menu Transaksi Pembelian, untuk memilih pemasok *Admin* dapat memilih pemasok pada bagian *INPUT PEMASOK*, pemilihan pemasok didukung oleh perhitungan dari nilai pemasok yang telah diinputkan pada menu nilai pemasok terhadap kode barang tertentu. Penentuan pemasok pada transaksi pembelian ini dilengkapi dengan DSS berdasarkan kriteria *TOLERANSI* dan *PRESENTASE*.



Gambar 5 Tampilan Menu Transaksi Penjualan

Pada gambar 5 menu Transaksi Penjualan, untuk memilih sales *Admin* dapat memilih sales pada bagian *INPUT SALES*, untuk memilih pelanggan *Admin* dapat memilih pelanggan pada bagian *INPUT PELANGGAN*, untuk memilih barang *Admin* dapat memilih barang pada bagian *INPUT BARANG*, untuk memasukan barang ke keranjang penjualan, *Admin* dapat menekan tombol *ADD* untuk menghapus barang yang tidak diinginkan di keranjang pembelian, untuk mengubah data barang di keranjang pembelian *Admin* dapat menekan Tombol *EDIT* setelah menginput perubahan di *INPUT BARANG*.

Setelah yakin dengan apa yang akan dijual, untuk mengaktifkan *PEMBAYARAN*, *Admin* harus menekan tombol *OK*, untuk batal dengan *PEMBAYARAN*, *Admin* dapat menekan tombol *CANCEL*. Tombol *SAVE* digunakan untuk menyimpan data penjualan.

Print Screen dibawah merupakan menu untuk mencetak laporan- laporan berdasarkan tanggal. Tombol *PRINT* digunakan untuk mencetak laporan. Tombol *VIEW* digunakan untuk menampilkan data pada *datagrid*.

PT Makmur Abdimandiri
JI Kamal Muara II 7, Kapuk, Cengkareng

BUKU BESAR
PERIODE : 01-Jan-2011 - 12-Jan-2011

ket akun	tanggal	Debit	Kredit
pembelian no tran 64	12/01/2011 0:00:	0	1.221
penjualan no tran 70	12/01/2011 0:00:	105.000	0
pembelian no tran 65	12/01/2011 0:00:	0	1.000
beban gaji	01/01/2011 0:00:	0	1.000
pembelian no tran 66	12/01/2011 0:00:	0	12.321
bayar utang no tran 65	12/01/2011 0:00:	0	10.200
penjualan no tran 71	12/01/2011 0:00:	6.646	0
		111,646	25,742
	Balance	85.904	

Gambar 6 Laporan Buku Besar

Pada gambar 6 Laporan buku besar menampilkan transaksi pada periode waktu tertentu sesuai dengan pilihan periode waktu yang dipilih oleh pengguna



Gambar 7 Penentuan Periode Laporan Neraca

Pada gambar 7 penentuan periode laporan neraca, pengguna terlebih dahulu menentukan periode waktu untuk menampilkan laporan neraca. Penentuan periode waktu dengan menentukan tanggal awal dan tanggal akhir transaksi yang akan ditampilkan.

NERACA
PT Makmur Abdimandiri
 Jl Kamal Muara II 7, Kapuk, Cengkareng
 021 5552437 - 021 6195272 - 021 6195275
 PERIODE : 01-Jan-2011 - 12-Jan-2011

Nama Akun	Trial Balance		Adjustment		Trial Adjustment		Income Statement		Balance Sheet	
	Debit	Kredit	Debit	Kredit	Debit	Kredit	Debit	Kredit	Debit	Kredit
Kas	162.694	0	0	0	162.694	0	0	0	162.694	0
Inventori	0	63.694	0	0	0	63.694	0	0	0	63.694
Utang	0	0	0	0	0	0	0	0	0	0
Piutang	0	0	0	0	0	0	0	0	0	0
Pemasukan	0	100.000	0	0	0	100.000	0	100.000	0	0
Pengeluaran	1.000	0	0	0	1.000	0	0	0	1.000	0
TOTAL	163.694	163.694	0	0	163.694	163.694	0	100.000	163.694	63.694
							100.000	0	0	100.000
							100.000	100.000	163.694	163.694

Gambar 8 Laporan Neraca

Pada gambar 8 laporan neraca, laporan ini akan tampil setelah pengguna memilih periode waktu pada gambar 7 penentuan periode laporan neraca.

The screenshot shows a window with the title "Laporan Laba/Rugi - Abd...". It contains two date selection fields: "Tanggal Awal" set to "12 Januari 2011" and "Tanggal Akhir" set to "12 Januari 2011". Below these fields is a "PRINT" button with a printer icon.

Gambar 9 Penentuan Periode Laporan Laba/Rugi

Pada gambar 9 penentuan periode laporan laba/rugi, pengguna terlebih dahulu menentukan periode waktu untuk menampilkan laporan laba/rugi. Penentuan periode waktu dengan menentukan tanggal awal dan tanggal akhir transaksi yang akan ditampilkan.

LAPORAN LABA RUGI		
PT Makmur Abdimandiri		
Jl. Kamal Muara II 7, Kapuk, Cengkareng		
021 5552437 - 021 6195272 - 021 6195275		
PERIODE : 12-Jan-2011 - 12-Jan-2011		
Profit		
Inventori Penjualan	111.646	
Inventori Pembelian	<u>24.742</u>	
		86.904
Pendapatan		
Pemasukan	0	
Piutang	<u>0</u>	
		0
Pembayaran		
Pengeluaran	0	
Utang	<u>0</u>	
		0
		<u>86.904</u>

Gambar 10 Laporan Laba/Rugi

Pada gambar 10 laporan laba/rugi, laporan ini akan tampil setelah pengguna memilih periode waktu pada gambar 9 penentuan periode laporan laba/rugi.

V. Simpulan dan Saran

V.1. Simpulan

Pada bagian laporan sebelumnya dijelaskan seluruh hasil analisis mengenai pembelian, penjualan, penentuan pemasok, dan akuntansi keuangan, maka dapat ditarik beberapa kesimpulan yang berkaitan dengan pembahasan yang dilakukan yaitu:

1. Aplikasi yang dibangun memiliki fungsi untuk mendukung jalannya usaha yang dimiliki oleh pemilik PT makmur Abdimandiri
2. Menu yang ada pada aplikasi dapat langsung dimengerti kegunaannya oleh *user*.
3. Dengan adanya aplikasi yang dibangun pekerjaan dapat dikerjakan lebih mudah
4. Data-data yang dihasilkan oleh aplikasi sangat berguna bagi perusahaan.
5. Laporan Keuangan yang dihasilkan oleh aplikasi sangat berguna bagi perusahaan.
6. Fungsi DSS pemilihan pemasok sangat berguna bagi perusahaan.

V.2. Saran

Aplikasi Pengelolaan PT MAKMUR ADBIMANDIRI dengan Sistem Akuntansi dan Dss digunakan untuk mendukung usaha yang sedang berjalan,

namun untuk dapat lebih mendukung usaha yang dijalankan maka terdapat beberapa saran yang diberikan:

1. Tampilan awal pada aplikasi lebih dikembangkan supaya lebih menarik.
2. Menu untuk mengatur penilaian pemasok terasa sedikit memusingkan

VI. Daftar Pustaka

- [Har05] Hartini Deliana, Lisye Mareta Cahya, Kikis Sabrina Kaisariza.(2009). Database dengan SQL Server 2005. Binus University
- [Har04] Harianto, Bambang,Ir.,MT.(2004). Sistem Manajemen Basis Data.Informatika Bandung
- [Jog01] Jogiyanto H.M.(2001), Analisis & Disain Sistem Informasi : Pendekatan Terstruktur Teori Dan Praktek Aplikasi Bisnis, Edisi. Kedua, Andi Offset Yogyakarta
- [Mal00] Mallach, Efrem G. (2000). Decision Support And Data Warehouse System. The McGraw-Hill Companies, Inc. : Singapore
- [McL98] McLeod. (1998). The Red Deer DecisionSupport System. Craigiebuckler, Aberdeen.
- [Mun07] Munawar.(2007).Pemodelan Visual dengan UML.Graha Ilmu, Bandung
- [Mun02] Munir, Rinaldi. (2002). Algoritma & Pemrograman (Edisi 2). CV.Informatika, Bandung.
- [Nug02] Nugroho, Adi.(2002).Analisi dan Perancangan Sistem Informasi dengan Metodologi Berorientasi Objek. Informatika, Bandung
- [Sal96] Salusu, J. 1996 Pengambilan Keputusan Stratejik Untuk Organisasi Publik dan Organisasi Non Profit. PT.GRASINDO, Jakarta

Sistem Pengendalian Lampu dengan Menggunakan *Personal Computer* (PC) untuk *Billing* Meja Billiard

Teddy Marcus Zakaria, Henry Kartadinata

Jurusan S1 Teknik Informatika

Fakultas Teknologi Informasi, Universitas Kristen Maranatha

Jl. Prof. Drg. Suria Sumantri no. 65 Bandung 40164

email :teddy.mz@maranatha.edu, henta090@gmail.com

Abstract

The development of electronics technology in particular has penetrated almost all aspects of life. The various electrical equipment influences in various places. If the various electrical equipment is controlled manually must be very inconvenient, moreover if the distance between each other apart. Based on these issues we can use controlling various electrical equipment with the existing computer in the company. In this case, an example of the electrical control of the lamp, the lamp control is used for billing system on billiard table lamp. With the light control system on the billiard table billing much ease is found, for example, we do not need to turn on the lights manually billiard table but only control over the computer or a PC to turn it on. The billiard table lamp is controlled by the billing system. In this way, unused tables will not light, so the transaction can not be manipulated by the employee because the light was not turned on manually.

Keywords: Billing System, billiard table lamp, lighting control system.

1. Pendahuluan

Perkembangan teknologi khususnya elektronika telah merambah hampir ke semua aspek kehidupan. Seperti halnya beragam peralatan listrik yang tersebar diberbagai tempat. Apabila beragam peralatan listrik tersebut dikendalikan secara manual dalam situasi dan kondisi tertentu tentunya sangat merepotkan. Apalagi jika jarak antara satu dengan yang lainnya berjauhan. Berdasarkan masalah tersebut kita dapat memanfaatkan sebuah alat pengendali untuk mengendalikan beragam peralatan listrik melalui komputer yang ada di perusahaan. Sebagai contoh, pengendalian alat listrik berupa lampu. Pengendalian lampu ini digunakan untuk sistem *Billing* pada lampu meja billiard. Dengan adanya sistem pengendalian lampu pada *Billing* meja billiard maka banyak kemudahan yang didapatkan, contohnya kita tidak perlu menyalakan lampu meja billiard secara *manual* tapi hanya mengendalikan lewat komputer atau PC untuk menyalakannya.

Dalam sistem pengendalian lampu meja billiard ini dapat mempermudah pekerjaan di tempat billiard karena lampu meja billiard dapat menyala dan mati secara otomatis dikendalikan oleh komputer atau PC. Dengan cara ini meja yang tidak terpakai tidak akan menyala sehingga transaksi tidak dapat dimanipulasi oleh karyawan karena lampu tidak dinyalakan secara *manual*.

2. Rumusan Masalah dan Tujuan Penelitian

Tujuan dari pembuatan aplikasi Sistem pengendalian lampu dengan menggunakan personal computer (PC) untuk *billing* meja billiard adalah:

- Untuk mempermudah pekerjaan di tempat billiard.
- Untuk melakukan pencatatan transaksi sesuai dengan perhitungannya sehingga daya dan transaksi tidak dapat dimanipulasi.

3. Landasan Teori

Sistem pengendalian lampu dengan menggunakan personal computer (PC) untuk *billing* meja billiard terdiri dari beberapa perangkat lunak dan perangkat keras yaitu:

Perangkat Lunak

a) Borland Delphi 7

Borland Delphi menggunakan bahasa pemrograman Pascal. Bahasa pemrograman ini dipakai pertama kali pada *software* pemrograman bernama Turbo Pascal. Pada akhirnya muncullah *software* yang bernama berbasis *Window* yaitu *Borland Delphi*. Yang digunakan dalam penelitian ini adalah Borland Delphi 7. Versi ini dipilih karena memiliki fitur (fasilitas), kinerja yang bagus dengan spesifikasi hardware minimum. Disamping itu Borland Delphi cukup populer dan banyak digunakan para pemrogram.

Untuk berkomunikasi dengan *port* LPT, akan digunakan *external library* yaitu 'inout32.dll'. File ini akan dipanggil menggunakan prosedur dan fungsi.

b) Database MS Access 2007

Microsoft Access dapat menggunakan data yang disimpan di dalam format Microsoft Access. Para pengguna/*programmer* yang telah berpengalaman dapat menggunakannya untuk mengembangkan perangkat lunak aplikasi yang kompleks, sementara para *programmer* yang kurang berpengalaman dapat menggunakannya karena *Microsoft Access* merupakan program yang telah diatur sedemikian rupa agar para penggunanya baik dari para *programmer* yang handal atau tidak. Access juga menawarkan teknik-teknik pemrograman berorientasi objek.

c) UserPort 1.0

UserPort digunakan untuk membuka akses ke *port paralel* Windows 32 bit. Win32 tidak dapat digunakan untuk mengakses *port paralel* secara langsung. Oleh karena itu anda memerlukan *software driver* tambahan agar akses ke *port paralel* dapat dilakukan. Sebagai catatan, apabila sistem operasi anda memakai *Window* 16 bit, misalnya *Windows* 95 atau *Win* 98, Anda tidak perlu menginstalasi *software* ini. Pada *Windows* 16 bit, *port paralel* dapat diakses langsung oleh *software*. UserPort dibuat oleh Thomas Franzon, yang melisensi *software* buatannya ini sebagai freeware. UserPort merupakan *software driver* yang digunakan untuk membuka koneksi *port paralel* agar dapat diakses langsung melalui *software* yang akan dibuat.

Perangkat Keras

a) DB-25 Male dan Female

Pada perancangan perangkat keras, *pararel port* digunakan sebagai pengirim sinyal data ke rangkaian antarmuka dan pada rangkaian antar muka tersebut akan diubah menjadi sinyal listrik bertenaga 12 Volt DC.



Gambar 1 DB-25 Female



Gambar 2 DB-25 Male

b) LED Dioda

LED dioda dimanfaatkan sebagai penyearah arus listrik, yaitu piranti elektronik yang mengubah arus atau tegangan bolak-balik (AC) menjadi arus tegangan searah (DC). Jenis dioda LED mampu menghasilkan cahaya apabila pada dioda tersebut bekerja arus listrik dengan arah *forward bias*/ bias arus maju. Arus listrik juga akan bekerja hanya pada arus bias maju.



Gambar 3 LED

c) Transistor

Alat semikonduktor yang dipakai sebagai penguat, sebagai sirkuit pemutus dan penyambung (*switching*), stabilisasi tegangan, modulasi sinyal atau sebagai fungsi lainnya.



Gambar 4 Transistor

d) Resistor

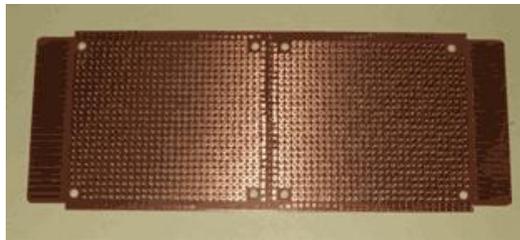
Komponen elektronik yang dibuat untuk mempunyai hambatan listrik yang tidak dipengaruhi oleh arus listrik yang melewatinya. Biasanya resistor juga dibuat untuk tidak dipengaruhi oleh temperatur atau faktor lainnya. Resistor bisa mempunyai hambatan yang tetap atau berubah-ubah (variabel).



Gambar 5 Resistor

e) PCB (Printed Circuit Board)

Suatu papan yang berfungsi sebagai tempat terpasang dan tersambungny berbagai komponen elektronik.

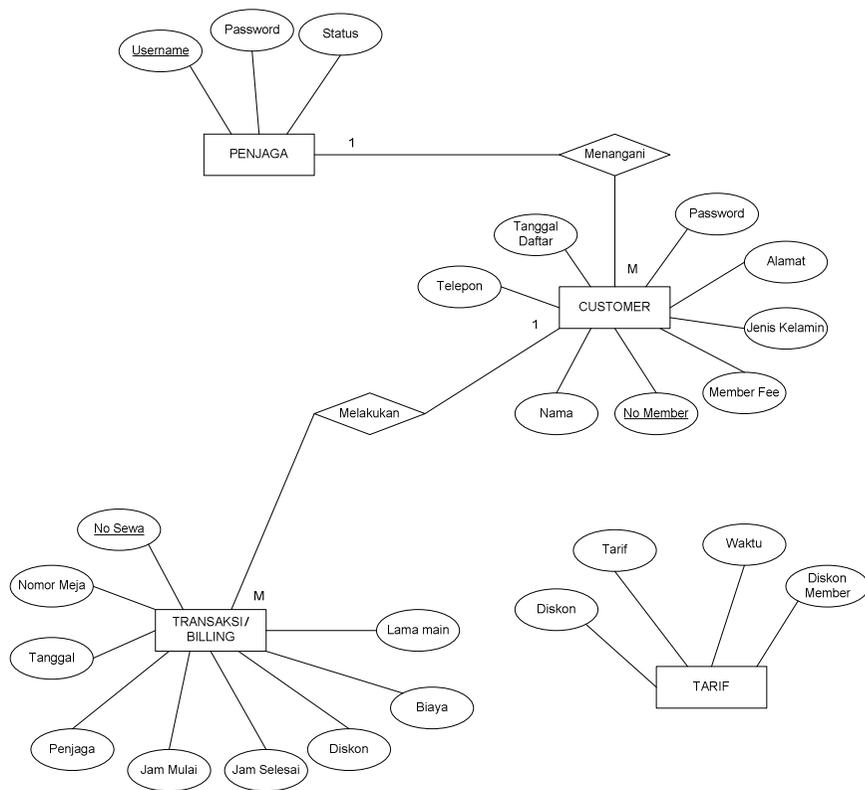


Gambar 6 PCB (Printed Circuit Board)

4. Desain Sistem

Berikut adalah ER-Diagram untuk aplikasi:

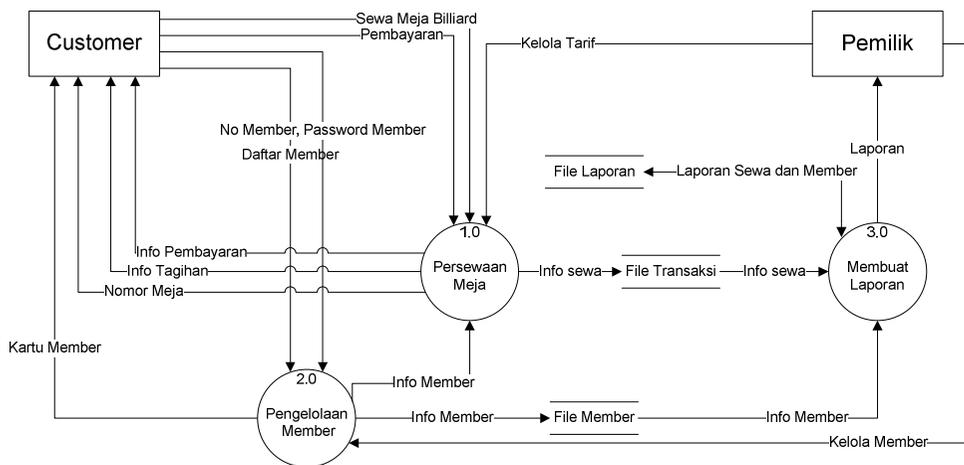
a) ER-Diagram



Gambar 7 ER-Diagram

Entitas Penjaga adalah operator yang menangani *customer*. Entitas Customer adalah pelanggan yang menyewa meja billiard, sebagai kunci utamanya adalah No.Member. Satu customer cukup menggunakan 1 No.member, karena setiap pembuatan customer baru diharuskan membayar Member Fee (cukup 1x pembayaran untuk selamanya). Entitas Transaksi digunakan untuk mencatat pemakaian (penyewaan) meja billiard dan pembayarannya. Tarif pemakaian per jam terdapat pada Entitas Tarif. Entitas Tarif dapat disesuaikan bila ada kenaikan biaya sewa perjam.

b) DFD Level 1



Gambar 8 DFD Level 1

Penjelasan DFD level 1 Sistem *Billing* Billiard :

Pada proses level 1 Sistem *Billing* Billiard memiliki 3 proses utama, proses tersebut adalah :

1. Persewaan Meja

Pada proses ini terdapat dua entitas yaitu entitas *customer* dan entitas pemilik. Pada entitas *customer* terdapat proses menyewa meja billiard, pada proses ini diberikan meja billiard untuk digunakan sesuai yang diinginkan. Proses info tagihan sewa meja billiard diberikan kepada *customer* berdasarkan perhitungan lama waktu menyewa meja billiard. Pada entitas pemilik terdapat proses mengelola tarif yaitu memasukkan tarif untuk sewa meja. Dari tagihan sewa maka *customer* melakukan pembayaran dan proses bukti pembayaran akan diberikan kepada *customer*. Proses persewaan meja akan mencatat semua transaksi yang terjadi. Dari proses catatan yang diambil maka akan menghasilkan informasi sewa yang dicatat dalam file transaksi yang akan diberikan kepada proses membuat laporan.

2. Pengelolaan Member

Pada proses ini terdapat 2 entitas yaitu entitas *customer* dan entitas pemilik. Pada entitas *customer* dapat mendaftar menjadi *member* atau anggota, pada proses ini diberikan kartu *member* yang terdapat nomor anggotanya yang diberikan kepada *customer*. Dari entitas pemilik dapat mengelola data *member* termasuk pencatatan data *member* yang baru mendaftar. Dari proses pengelolaan data *member* maka akan tercatat dalam file *member* yang akan diberikan kepada proses membuat laporan. Proses *member* ini juga berfungsi untuk memberikan diskon khusus untuk yang menyewa meja billiard.

3. Laporan

Pada proses membuat laporan terdapat 1 entitas yaitu entitas pemilik. Pada proses membuat laporan ini terdapat catatan-catatan transaksi dan data *member* yang diambil dari file transaksi dan file *member*. Proses membuat laporan ini menghasilkan laporan yang akan diberikan kepada pemilik.

5. Hasil Implementasi

a) Mengendalikan Lampu

Fitur mengendalikan lampu merupakan fitur penting dari aplikasi sistem pengendalian lampu menggunakan komputer sehingga *device* yang berupa perangkat elektronik *switch* lampu pun sama pentingnya. Untuk menghubungkan perangkat elektronik *switch* lampu dengan komputer melalui *portprinter* LPT / DB-25 agar saling berkomunikasi membutuhkan sebuah fungsi *library* dalam bentuk file DLL yaitu file “Inpout32.dll” dengan pemanggilan *function*. Berikut realisasi *code* baca/tulis dari menghubungkan perangkat elektronik *switch* dengan komputer :

```
Procedure Out32 (Addr:word;Data:byte);stdcall;external  
'inpout32.dll';
```

Procedure Out32 digunakan untuk menulis data dari *port* dengan menggunakan fungsi *library* dari “Inpout32.dll”. Penulisan data ke *port* LPT menggunakan prosedur ini.

```
Function Inp32 (Addr:word):byte;stdcall;external  
'inpout32.dll';
```

Function Inp32 digunakan untuk membaca data dari *port* dengan menggunakan fungsi *library* dari “Inpout32.dll”. Untuk membaca data dari *port* LPT digunakan fungsi Inp32.

Berikut potongan program pengecekan *port* LPT “

```
tmp:=Inp32($378);  
if checkbox1.checked then  
Out32($378,tmp or $1)  
Else Out32($378,tmp and $fe);
```

Alamat dari *port* LPT adalah \$378 dengan fungsi membaca *port* Inp32 dan disimpan dalam *var*tmp : byte. Kemudian untuk menuliskan data dari komputer melalui *port* dengan fungsi menulis *port* Out32 dengan alamat *port* LPT \$378 dan alamat dalam *port* dengan penulisan *hex* yaitu \$1

Untuk pengiriman bit menggunakan perhitungan *binary* yaitu dengan angka 0 dan 1. Ini merupakan instruksi-instruksi komputer yang dimengerti oleh mesin. Berikut contoh perhitungan *biner* dengan *hexadecimal* :

Tabel 1 Bilangan *biner* dan *Hex Decimal*

Biner	Hex Decimal	Biner	Hex Decimal
0000	0	1000	8
0001	1	1001	9
0010	2	1010	a
0011	3	1011	b
0100	4	1100	c
0101	5	1101	d
0110	6	1110	e
0111	7	1111	f

Saat ingin menyalakan lampu nomor 1 maka dalam bilangan biner yaitu 0000 0001 jika dalam bilangan *hex* yang ditulis pada program adalah gabungan antara bilangan *hex* 0 dan 1 dan jika ingin mematikan lampu nomor satu dalam bilangan biner yaitu 1111 1110 yaitu gabungan antara bilangan *hex* f dan e.

b) Menghitung tarif

Algoritma Menghitung tarif

```

PROGRAM Hitung tarif
{Membaca waktu awal dan waktu akhir pemakaian W1(hh:mm:ss)
dan W2(hh:mm:ss). Asumsi W2.hh >= W1.hh. Kemudian menghitung
lama pemakaian, yaitu W2 - W1 = W3 dan menghitung tarif
sesuai lama pemakaian)
DEKLARASI
Type Jam :record < hh:integer, {0..23}
           mm:integer, {0..59}
           ss:integer {0..59} >
W1 : Jam    {Jam mulai pemakaian}
W2 : Jam    {Jam akhir pemakaian}
W3 : Jam    {Lama pemakaian}
Biaya, Persen, Persen member, Diskon jam, tarif :Real
Biaya member , Diskon member, Diskon: Real
ALGORITMA
Read (W1.hh, W1.mm, W1.ss)
Read (W1.hh, W2.mm, W2.ss)
Read (Member)
Read (Tarif), Read (diskon member), Read (Persen Member),
(Persen)

If W2.ss>= W1.ss then
    W3.ss ← W2.ss - W1.ss
Else
    W3.ss ← (W2.ss + 60) - W1.ss
    W2.mm ← W2.mm - 1
Endif
If W2.mm >= W1.mm then
    W3.mm ← W2.mm - W1.mm
Else
    W3.mm ← (W2.mm + 60) - W1.mm
    
```

```
W2.hh ← W2.hh - 1
Endif
W3.hh ← W2.hh - W1.hh
Write (W3.hh ,W3.mm, W3.ss)
If Member = true then
    Diskon member ← (((W3.hh * 60) + W3.mm) * Tarif ) +
    Tarif)
    * persen member
    Biaya member ← (((W3.hh * 60) + W3.mm) * Tarif ) +
    Tarif) - Diskon member
Else If ((W3.hh > Diskon jam) and (Persen > 0)) then
    Diskon ← (((W3.hh * 60) + W3.mm) * Tarif ) + Tarif) *
    persen
    Biaya ← (((W3.hh * 60) + W3.mm) * Tarif ) + Tarif) - Diskon
Else
    Biaya ← (((W3.hh * 60) + W3.mm) * Tarif ) +
    Tarif)
EndIf
EndIf
Write (Diskon member)
Write (Biaya member)
Write (Diskon)
Write (Biaya)
```

Code Program menghitung tarif

```
DecodeTime (StrToTime (FBilling.ESelesai2.Text) -
StrToTime (FBilling.EMulai2.Text), Jam2, Menit2, Detik2,
MDetik2);
FBilling.ELama2.Text := IntToStr(Jam2) + ':' +
IntToStr(Menit2) + ':' + IntToStr(Detik2);
Persen := StrToFloat (FBilling.EDiskon.Text)*0.01;
PersenMember := StrToFloat (FBilling.EDiskonMember.Text)*0.01;

if FBilling.EMember2.Text <> '' then
begin
FBilling.LBDiskon2.Caption := FormatFloat('#', (((Jam2 * 60)
+ Menit2) * StrToInt (FBilling.EdTarif.Text)) +
StrToInt (FBilling.EdTarif.Text))*PersenMember);
FBilling.LBBiaya2.Caption := FormatFloat('#', (((Jam2 * 60) +
Menit2) * StrToInt (FBilling.EdTarif.Text)) +
StrToInt (FBilling.EdTarif.Text)) -
StrToInt (FBilling.LBDiskon2.Caption));
end
else
begin
if (Detik2 > StrToInt (FBilling.EDiskonJam.Text)) and
(StrToFloat (FBilling.EDiskon.Text)>0) then
begin
FBilling.LBDiskon2.Caption := FormatFloat('#', (((Jam2 * 60)
+ Menit2) * StrToInt (FBilling.EdTarif.Text)) +
StrToInt (FBilling.EdTarif.Text))*Persen);
```

```
FBilling.LBbiaya2.Caption := FormatFloat('#', (((Jam2 * 60) +
Menit2) * StrToInt (FBilling.EdTarif.Text)) +
StrToInt (FBilling.LBDiskon2.Caption));
end
else
begin
FBilling.LBDiskon2.Caption := '0';
FBilling.LBbiaya2.Caption := FormatFloat('#', (((Jam2 * 60) +
Menit2) * StrToInt (FBilling.EdTarif.Text)) +
StrToInt (FBilling.LBDiskon2.Caption));
end;
end;
```

c) Member

Algoritma Member

```
PROGRAM member
{Memasukkan data-data member untuk digunakan dalam penggunaan
member}
DEKLARASI
Nomor Member :integer
Nama, alamat, jenis kelamin, password :string
Telepon :integer
Tanggal Daftar :date
ALGORITMA
1   Buka koneksi ke database member
2   Tampilkan Form member
3   Aktifkan tombol cari, edit, tambah, batal, hapus,
laporan
4   Masukkan (Nomor Member, Nama, Alamat, Jenis kelamin,
Telepon)
5   Masukkan (Tanggal Daftar, Password)
6   If Nomor Member diinput then
7       Cari data Nomor Member pada tabel member
8       Tampilkan data - data member
9   End if
10  Masukkan data - data member
11  If ditekan tombol Simpan then
12      If textbox ada yang kosong then
13          Tampilkan pesan "Lengkapi form dengan
lengkap"
14      Else if
15          if Nomor Member sudah ada then
16              Tampilkan pesan "Nomor Member sudah ada"
17          Else if
18              If proses simpan error then
19                  Tampilkan pesan error database
20              Else if
21                  Simpan data ke tabel Member
22                  kembali ke nomor 2
```

```
23         End if
24     End if
25 End if
26 End if
27 If ditekan tombol batal then
28     Batalkan inputan pada form member
29     Kembali ke nomor 2
30 End if
31 Masukkan nomor member
32 If ditekan tombol cari then
33     if textbox nomor member kosong then
34         Tampilkan pesan "Masukkan nomor member
35         yang ingin dicari"
36     Else
37         Ambil data dengan nomor member pada tabel
38         member
39     End if
40 End if
41 If ditekan tombol hapus then
42     Hapus data pada tabel member
43     Kembali ke nomor 2
44 End if
45 Ambil data dengan nomor member pada tabel member
46 If ditekan tombol edit then
47     If textbox ada yang kosong then
48         Tampilkan pesan "Lengkapi form dengan
49         lengkap"
50     Else
51         Update data pada tabel kelurahan
52         Kembali ke nomor 2
53     End if
54 End if
```

Coding Memasukan data member

```
if ((ENomorMember.Text='') or (ENama.Text='') or
(EAlamat.Text='') or
(ETelepon.Text='') or (EPassword.Text='') or
(EPembayaran.Text=''))then
showmessage('Lengkapi Form dengan Lengkap!')
else if TbMember.Locate('Nomor_Member',ENomorMember.Text,[])
then
showmessage('Nomor member sudah ada!')
else
begin
TbMember.Refresh;
TbMember.Append;
//TbMember['Nomor_Member'] := StrToInt(ENomorMember.Text);
TbMember['Nama'] := ENama.Text;
TbMember['Alamat'] := EAlamat.Text;
if RBPria.Checked = true then TbMember['Jenis_Kelamin'] :=
'Pria'
```

```
else TbMember['Jenis_Kelamin'] := 'Wanita';  
TbMember['Telepon'] := ETelepon.Text;  
TbMember['Password'] := Encrypt(EPassword.Text);  
TbMember['Tanggal_Daftar'] := {MEDaftar.Text :=  
}DateToStr(DateTimePicker1.Date);  
TbMember['Member_Fee'] := EPembayaran.Text;  
TbMember.Post;  
FormShow(Sender);
```

d) Contoh Implementasi



Gambar 9 Form Utama

Pada form utama ini terdapat tombol-tombol yaitu menu untuk menjalankan sistem *billing* meja billiard.

	Mulai	Selesai	Lama	Member	Diskon	Biaya	Hapus
1	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/> <input type="text"/>	Rp	Rp	
2	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/> <input type="text"/>	Rp	Rp	
3	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/> <input type="text"/>	Rp	Rp	
4	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/> <input type="text"/>	Rp	Rp	
5	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/> <input type="text"/>	Rp	Rp	
6	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/> <input type="text"/>	Rp	Rp	
7	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/> <input type="text"/>	Rp	Rp	
8	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/> <input type="text"/>	Rp	Rp	

Tarif per Menit : Member : %
Diskon : % Lebih dari : Jam
Penjaga : Tanggal : 11/19/2009

Buttons: **Set Member** **Set Diskon** **Set Tarif** **Timer**

Status: Hint Waktu : 2:22:52 PM

Gambar 10 Form *Billing*

Form ini muncul ketika tombol *billing* pada program utama ditekan. Fungsi dari form ini adalah untuk tampilan penghitungan penggunaan meja billiard.

Nomor_Member	Nama	Jenis_Kelamin	Telepon	Alamat	Password
44	Henry	Pria	08562220377	Adibrata 12	«²³»«²³»
45	karta	Pria	08562220377	Adibrata 12	«²³»«²³»
46	Nez	Wanita	08562220377	Adibrata 12	«²³»«²³»
49	Agnes	Wanita	08562220377	Adibrata 12	«²³»«²³»
50	Adelia	Wanita	08562220377	Adibrata 12	«²³»«²³»

Gambar 11 Form Member

No_Sewa	Nomor_Meja	Tanggal	Penjaga	Jam_Mulai	Jam_Selesai	Lama_Main	Mem
2	7	11/15/2009		5:16:55 PM	5:16:55 PM	0:0:0	
3	7	11/29/2009	0325090	7:15:30 PM	7:15:45 PM	0:0:15	
4	5	11/29/2009	0325090	7:16:12 PM	7:17:01 PM	0:0:49	
5	4	12/4/2009	0325090	8:40:31 PM	8:40:44 PM	0:0:13	
6	1	12/7/2009	0325090	7:29:00 PM	7:30:01 PM	0:1:1	
7	1	12/10/2009	0325090	7:22:18 PM	7:22:18 PM	0:0:0	

Gambar 12 Form Laporan

Gambar ini adalah form laporan. Form ini muncul ketika tombol laporan pada form utama ditekan. Fungsi dari form ini adalah untuk melihat data-data transaksi pemakaian meja billiard.

The screenshot shows a software window titled "FTimer" with a close button in the top right corner. The main content area has a yellow header "Timer Billing". Below this, there are two orange headers: "Timer Menyalakan Lampu Meja" and "Timer Mematikan Lampu Meja". Each of these sections contains a grid of checkboxes for tables 1 through 8, with a digital time display showing "17:17". At the bottom, there are two buttons: "Aktifkan" and "Non Aktifkan". A status bar at the very bottom shows "Klik disini untuk mengaktifkan timer" and "Waktu : 10:45:04 PM".

Gambar 13 Form Timer

6. Kesimpulan dan Saran

a) Kesimpulan

Dari hasil evaluasi, secara umum aplikasi ini sangat membantu pekerjaan operator di tempat penyewaan meja billiard. Dengan sistem pengendalian lampu berbasis komputer ini, setiap lampu meja billiard dapat dikendalikan lewat komputer. Waktu pemakaian meja billiard dapat dicatat ke dalam basisdata, sehingga memudahkan operator untuk menghitung penerimaan biaya sewa dari pelanggan. Hal ini menghindari manipulasi waktu penyewaan meja yang kerap terjadi jika pengendalian lampu meja billiard dilakukan secara manual.

b) Saran

Operator meja billiard harus melakukan pemeriksaan berkala terhadap lampu meja billiard karena aplikasi ini tidak dapat mendeteksi lampu yang rusak. Penggunaan UPS pada sistem komputer sangat disarankan karena saat perhitungan biaya penyewaan, komputer harus tetap menyala untuk menghindari kerusakan basisdata.

7. Daftar Pustaka

- [Kur06] Dede Kurnia.2006. KendaliPerangkatListrikmelaluiPort Printer, from <http://www.teknisoft.or.id>
- [Der08] Dersky. 2008. Koneksi Delphi Access, from <http://dersky.files.wordpress.com/2008/02/koneksi-delphi-access.pdf>
- [Kad01] Kadir, Abdul. 2001. Pemrograman Database Menggunakan Delphi. Salemba Infotek.
- [Wah09] Komputer, Wahana. 2009. Aplikasi Cerdas Menggunakan Delphi. Andi Offset.
- [Lut02] Lutfi. 2002. Akses Peralatan Luar lewat ParallelPrinterPort, from <http://www.klik-kanan.com/akses-peralatan-luar-lewat-parallel-printer-port.htm>
- [Mal06] Malik, Jamaludin. 2006. Tip & Trik Unik Delphi Lanjutan. Andi Offset.
- [Muh04] Muhamad Ali. 2004. Mengakses data base pada delphi, from <http://www.ilmukomputer.com/2003Mengakses-database-pada-Delphi>.
- [Set05] Setiawan, Yudha. 2005. Tip Delphi . Andi Offset.
- [Ted05] Teddy Marcus, Agus Prijono, Josef Widiadhi. 2005. Pemrograman Delphi dengan ADOExpress. Informatika
- [Ted04] Teddy Marcus. 2004. Delphi Developer dan SQL Server. Informatika
- [Fra02] Thomas Franzon. 2002. UserPort 1.0, <http://www.brothersoft.com/userport-122645.html>

Perancangan Model Manajemen Pengetahuan menggunakan Model *Nonaka Takeuchi* (Studi Kasus Administrasi Akademik)

Robby Tan

Jurusan S1 Teknik Informatika

Fakultas Teknologi Informasi, Universitas Kristen Maranatha

Jl. Prof. Drg. Suria Sumantri no. 65 Bandung 40164

email: red_phoenix_zero@yahoo.com

Abstract

Education quality is one of the main factors in executing education process. This value can be measured from the quality of the educators, curriculum, books, and many more. On the other hand, there is other determining factor that is equally important, which is the service of the place of education. In order to create good service, there should be a standard of service and equality of information from all the stakeholders. With the increase of the knowledge and information, the service should be faster and the decision process should be more effective. Meanwhile, not all of the knowledge and information inside an organization is in explicit forms, and still many tacit knowledge and information. Author proposes the design model of knowledge management so that the tacit knowledges can be collected and transform into explicit forms. The design model will use the model from Nonaka Takeuchi, in which the creation of knowledge is sourced from each individuals and then is transmitted and externalized, collected and transformed into a new knowledge.

Keywords: knowledge management, knowledge management portal model, SOP

1. Pendahuluan

Saat ini teknologi informasi sudah dimanfaatkan dan berkembang dalam segala aspek kehidupan manusia. Hal itu juga berarti dalam bidang akademik atau pendidikan. Dalam bidang pendidikan, teknologi informasi itu terwujud dalam sebuah sistem yang bernama sistem informasi akademik.

Sistem informasi akademik kini sudah banyak dikembangkan oleh banyak perguruan tinggi untuk menunjang kebutuhan akan informasi akademik dan meningkatkan pelayanan bagi mahasiswa dan seluruh civitas akademika. Dalam sistem informasi tersebut terdapat beberapa bagian yang dapat dibahas secara luas. Di sini penulis mengkhususkan pada bagian sistem informasi akademik.

Sistem informasi akademik mencakup pelayanan kepada mahasiswa dan kepada dosen, khususnya kepada sivitas akademika. Pelayanan ini mencakup apa saja yang harus dilakukan dalam sebuah kegiatan, dokumen-dokumen yang dipakai, dan bagaimana pertanggungjawaban kegiatan-kegiatan tersebut. Untuk memenuhi standar mutu sebuah pelayanan yang baik maka diperlukan sebuah *Standard Operating Procedures (SOP)*.

Untuk membuat sebuah SOP yang baik diperlukan berbagai macam pengetahuan yang terdapat dalam sistem tersebut. Saat ini prosedur-prosedur yang ada sudah dibuat akan tetapi belum dibakukan menjadi bentuk SOP. Selain itu juga masih banyak pengalaman-pengalaman dan pengambilan keputusan yang sifatnya masih berupa *tacit knowledge* yang berarti pengetahuan tersebut masih bersifat personal dan masih terdapat di pikiran individu-individu yang terlibat dalam sistem

tersebut. Untuk dapat menghasilkan SOP yang baik yang sesuai dengan standar mutu administrasi akademik setiap perguruan tinggi maka *tacit knowledge* yang berupa pengalaman-pengalaman individu tersebut akan disatukan bersama dengan prosedur-prosedur sehingga menjadi sebuah *explicit knowledge* yang dapat diketahui oleh semua pihak sehingga dapat menjadi aset yang berharga bagi organisasi dan dapat dikembangkan lebih luas lagi sehingga tercipta sebuah standar kerja yang baik.

Secara umum, SOP adalah sebuah penetapan tertulis mengenai apa yang harus dilakukan, kapan, di mana, dan oleh siapa, beserta dengan cara yang paling efektif. SOP diperlukan karena seringkali banyak organisasi yang melakukan kerja dengan prosedur kerja yang tidak jelas dan tidak transparan. Atau mungkin saja sebuah organisasi sudah mempunyai sebuah prosedur kerja yang jelas, namun prosedur kerja tersebut mungkin saja berbeda-beda (tidak standar) meski untuk pelayanan tersebut menghasilkan *output* yang sama.

Dengan prosedur kerja yang berbeda-beda, hal tersebut dapat mempersulit bagi mahasiswa maupun dosen untuk lebih memahami prosedur yang benar secara utuh. Atau dapat saja terjadi peraturan-peraturan non tertulis yang masih dimiliki oleh individu, sehingga memungkinkan terjadinya sebuah pengambilan keputusan yang salah karena pengetahuan tersebut masih bersifat *personal*. Di sisi lain, meskipun *output* yang dihasilkan sama, tetapi mutu pelayanan mungkin saja berbeda. Dapat saja terjadi bahwa seseorang mendapatkan informasi yang berbeda dari orang-orang yang terlibat dalam organisasi sehingga dapat menimbulkan kesalahpahaman mengenai pelayanan tersebut. Atau mungkin saja dengan birokrasi yang terlalu berbelit-belit, tujuan akhir dari pelayanan tersebut tidak tersampaikan kepada mahasiswa.

2. Landasan Teori

Secara umum, *knowledge* dalam padanan kata bahasa Indonesia berarti pengetahuan. Akan tetapi pengertian lebih luas dari pengetahuan yaitu penggunaan secara menyeluruh dari informasi dan data yang diselaraskan dengan kemampuan potensial, kompetensi, ide/pikiran, komitmen dan motivasi seseorang. Atau dengan kata lain pengetahuan merupakan pemahaman manusia terhadap sesuatu yang telah didapatkan melalui proses pembelajaran dan pengalamannya.

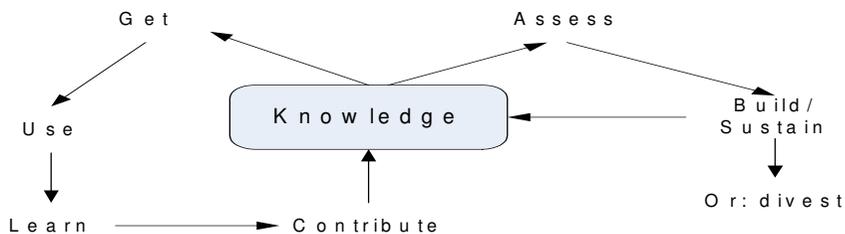
Saat ini, pengetahuan dipandang sebagai komoditas dari aset intelektual yang mempunyai komoditas yang berbeda dengan komoditas-komoditas lainnya. Hal-hal yang membedakan pengetahuan dengan komoditas-komoditas lainnya adalah:

- a. Penggunaan dari pengetahuan tidak akan menghabiskan pengetahuan itu sendiri.
- b. Penyampaian pengetahuan kepada pihak lain tidak akan membuat pemberi pengetahuan kehilangan pengetahuan tersebut.
- c. Kemampuan untuk menggunakan pengetahuan masih sangat sedikit sedangkan pengetahuan sendiri sangat luas.
- d. Kebanyakan dari pengetahuan berharga dari sebuah organisasi berakhir sama seperti berakhirnya hari.

Pengetahuan terbagi menjadi dua jenis yaitu *tacit knowledge* dan *explicit knowledge*. *Tacit knowledge* merupakan pengetahuan dari para pakar baik individu

maupun masyarakat beserta dengan pengalaman-pengalaman. *Tacit knowledge* sangat sulit untuk dibagikan kepada orang lain sehingga muncullah sebuah ungkapan “Kita lebih tahu dari apa yang dapat kita katakan”. Sedangkan *explicit knowledge* merupakan sesuatu yang dapat diekspresikan dengan kata-kata maupun angka serta dapat disampaikan dalam bentuk ilmiah, spesifik, manual, dan sebagainya. Sebagai contoh pembandingan tacit dan explicit knowledge yaitu: seorang anak dapat menemukan gambar ayah atau ibunya di dalam sebuah foto yang berisi puluhan orang. Akan tetapi, jika anak tersebut sudah menerangkan ciri-ciri dari orang tuanya belum tentu orang lain dapat menemukannya.

Siklus manajemen pengetahuan merupakan sebuah fase yang menjelaskan penangkapan (*capture*), penciptaan (*creation*), kodifikasi (*codification*), penyebarluasan (*sharing*), pengaksesan (*accessing*), aplikasi, dan penggunaan kembali pengetahuan yang berada dalam sebuah organisasi. Sintesis dari pendekatan yang akan dijabarkan akan dibangun menjadi kerangka kerja dalam mengubah sebuah informasi menjadi aset pengetahuan yang berharga untuk organisasi.



Gambar 1. Siklus Manajemen Pengetahuan Bukowitz and Williams

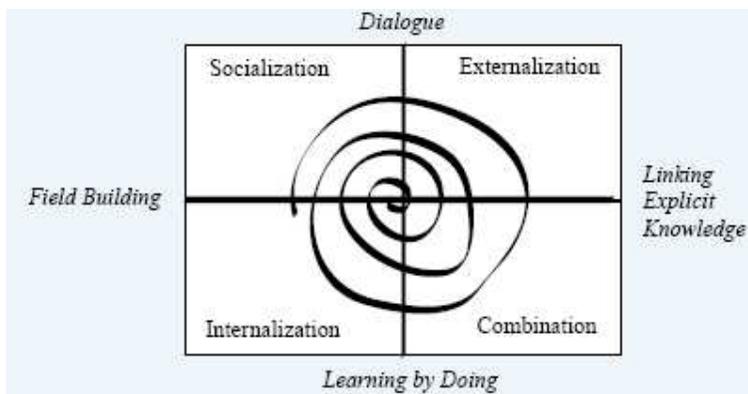
Siklus manajemen pengetahuan Bukowitz dan Williams menjelaskan sebuah kerangka kerja dari proses manajemen pengetahuan yaitu bagaimana organisasi menghasilkan, memelihara, dan menyebarkan suatu strategi yang benar untuk menciptakan nilai terhadap pengetahuan yang sudah ada. Berikut adalah penjelasan dari masing-masing proses pada siklus ini:

- Get* adalah proses mencari informasi yang diperlukan untuk pengambilan keputusan, pemecahan masalah atau untuk inovasi.
- Use* adalah bagaimana menggunakan informasi untuk berinovasi (baik individual maupun kelompok)
- Learn* adalah bagaimana organisasi dapat belajar dari pengalamannya, baik dari kesuksesan (*best practice*) maupun dari kegagalan (*lesson learned*) untuk menciptakan keunggulan persaingan (*competitive advantage*).
- Contribute* adalah memberikan pengetahuan yang diperoleh dari hasil pembelajaran (*learning*) untuk individu lainnya.
- Assess* adalah evaluasi dari manusianya (kompetensi), pelanggan (hubungan pelanggan), modal perusahaan (dasar-dasar pengetahuan, proses bisnis, infrastruktur teknologi, nilai, norma, budaya) dan modal intelektual (hubungan antara manusia, pelanggan dan modal organisasi).

- f. *Build and sustain* adalah untuk meyakinkan bahwa modal intelektual perusahaan di masa yang akan datang akan membawa perusahaan tetap bertahan dan bersaing.
- g. *Divest* adalah tempat pembuangan pengetahuan yang sudah tidak terpakai lagi (tidak bernilai).

Untuk mendapatkan sebuah pengetahuan maka perlu diperhatikan dasar dari pengetahuan itu. Dasar dari pengetahuan adalah data yang diolah menjadi informasi kemudian informasi-informasi tersebut diolah kembali menjadi pengetahuan. Data adalah sekumpulan ciri-ciri, fakta atau kejadian. Informasi adalah sebuah pesan yang biasanya dalam bentuk dokumen atau komunikasi yang dapat dilihat atau didengar. Pengetahuan merupakan gabungan pengalaman, nilai, dan informasi kontekstual, pandangan para ahli yang menyediakan suatu kerangka kerja untuk mengevaluasi dan menggabungkan pengalaman-pengalaman baru dan informasi.

Menurut Nonaka dan Takeuchi, penciptaan pengetahuan selalu dimulai dari individu. Pengetahuan tersebut dikumpulkan dan kemudian dibakukan dalam sebuah perusahaan sehingga dapat menjadi pengetahuan bagi orang lain. Dalam model ini terdapat empat model konversi pengetahuan yaitu:



Gambar 2. Model Konversi Nonaka Takeuchi

- a. *Tacit knowledge* ke *tacit knowledge* disebut dengan proses sosialisasi. Sosialisasi meliputi kegiatan berbagi *tacit knowledge* antar individu. Istilah sosialisasi digunakan karena *tacit knowledge* disebarkan melalui kegiatan bersama seperti tinggal bersama, meluangkan waktu bersama dan bukan melalui tulisan atau instruksi verbal. Dengan demikian, dalam kasus tertentu *tacit knowledge* hanya bisa disebarkan jika seseorang merasa bebas untuk menjadi seseorang yang lebih besar yang memiliki pengetahuan tacit dari orang lain.
- b. *Tacit knowledge* ke *explicit knowledge* disebut dengan proses eksternalisasi. Eksternalisasi membutuhkan penyajian *tacit knowledge* ke dalam bentuk yang lebih umum sehingga dapat dipahami oleh orang lain. Pada tahap eksternalisasi ini, individu memiliki komitmen terhadap sebuah kelompok dan menjadi satu dengan kelompok tersebut. Dalam prakteknya,

eksternalisasi didukung oleh dua faktor kunci. Pertama, artikulasi *tacit knowledge* seperti dialog. Kedua yaitu menterjemahkan *tacit knowledge* dari para ahli ke dalam bentuk yang dapat dipahami seperti dokumen, manual dan sebagainya.

- c. *Explicit knowledge* ke *explicit knowledge* disebut dengan proses kombinasi.

Kombinasi meliputi konversi *explicit knowledge* ke dalam bentuk himpunan *explicit knowledge* yang lebih kompleks. Dalam prakteknya, fase kombinasi tergantung kepada tiga proses yaitu penangkapan dan integrasi *explicit knowledge* baru termasuk pengumpulan data eksternal dari dalam atau dari luar institusi kemudian mengkombinasikan kedua data tersebut. Kedua, penyebarluasan *explicit knowledge* tersebut melalui presentasi atau pertemuan langsung. Ketiga, pengolahan kembali *explicit knowledge* sehingga lebih mudah dimanfaatkan kembali, misal menjadi dokumen rencana, laporan, manual, dan sebagainya.

- d. *Explicit knowledge* ke *tacit knowledge* disebut dengan proses internalisasi. Internalisasi pengetahuan baru merupakan konversi dari *explicit knowledge* ke dalam *tacit knowledge* organisasi. Individu harus mengidentifikasi pengetahuan yang relevan dengan kebutuhannya di dalam pengetahuan organisasi tersebut. Dalam prakteknya, internalisasi dapat dilakukan dalam dua dimensi. Pertama yaitu dengan penerapan *explicit knowledge* dalam tindakan atau praktek langsung. Kedua yaitu dengan penguasaan *explicit knowledge* melalui simulasi, eksperimen, atau belajar sambil bekerja.

Untuk pengklasifikasian pengetahuan-pengetahuan yang didapat, maka digunakan salah satu cara yaitu dengan menggunakan taksonomi pengetahuan. Taksonomi adalah klasifikasi sistem yang mendasar yang memungkinkan untuk mendeskripsikan konsep dan hubungan antara konsep-konsep tersebut. Semakin tinggi tempat dari konsep tersebut, maka konsep tersebut akan semakin luas atau umum. Dan semakin ke bawah, maka konsep tersebut akan semakin mendetail dan jelas.

Setelah dikodifikasi, maka dapat dibentuklah kumpulan-kumpulan pengetahuan tersebut dan disatukan menjadi sebuah SOP. SOP adalah penetapan tertulis mengenai apa yang harus dilakukan, kapan, di mana, oleh siapa, bagaimana cara melakukan, apa saja yang diperlukan, dan lain-lain yang semuanya itu merupakan prosedur kerja yang harus ditaati dan dilakukan. Saat ini banyak perusahaan yang tidak mempunyai SOP, yang mengakibatkan banyak pekerjaan yang tidak terlaksana dengan baik, yang bersangkutan tidak bertanggung jawab, ada kelalaian kerja, kesimpangsiuran, kesalahan, yang mengakibatkan resiko kerugian besar bagi perusahaan/ organisasi.

SOP merupakan sebuah set instruksi-instruksi yang tertulis yang mendokumentasi rutinitas atau aktivitas yang berulang yang dilakukan oleh sebuah organisasi. Pembuatan dan penggunaan dari SOP merupakan salah satu bagian dari kualitas sebuah sistem. SOP bukan hanya merupakan pedoman prosedur kerja rutin yang harus dilaksanakan, tetapi SOP juga berfungsi untuk mengevaluasi pekerjaan yang telah dilakukan, apakah pekerjaan tersebut telah dikerjakan dengan baik atau

tidak, kendala apa yang dihadapi, mengapa kendala tersebut terjadi sehingga dapat diambil keputusan yang tepat melalui SOP.

3. Metodologi Penelitian

Fokus penelitian ini yaitu memodelkan manajemen pengetahuan pada bagian akademik sehingga pengetahuan-pengetahuan yang ada dapat digunakan ulang sebagai sebuah alat untuk mencapai keunggulan kompetitif. Pengelolaan ini mencakup penangkapan *tacit knowledge* yang masih ada di individu-individu yang terlibat dan mengolahnya menjadi sebuah *explicit knowledge* yang berbentuk *knowledge base management* yang dapat dipakai dan dipercaya sebagai sebuah prosedur kerja yang baku beserta dengan aturan-aturan yang dipakai ataupun pengalaman-pengalaman yang selama ini sudah terkumpul. Pengetahuan-pengetahuan yang sudah terkumpul tersebut kemudian dibagikan kepada setiap individu untuk digunakan kembali (*reuse*) dan untuk dikembangkan agar pengetahuan dalam perusahaan agar tercapai sebuah proses perkembangan yang terus-menerus (*continuous improvement*).

Penelitian dilakukan dengan melakukan terlebih dahulu studi pustaka mengenai hal-hal yang berhubungan dengan manajemen pengetahuan dan SOP. Kemudian dilanjutkan dengan menggali kondisi yang sudah ada saat ini sehingga dimungkinkan mendapatkan bentuk dasar dari sistem manajemen pengetahuan yang diinginkan. Kemudian dibentuk sebuah *road map* dari manajemen pengetahuan ini. Setelah itu barulah dilakukan tahap *knowledge capture* dilakukan dengan menggunakan dua buah metode yaitu dengan wawancara dengan para ahli, cerita dan pembelajaran dengan observasi. Tahap berikutnya dilanjutkan dengan membagi pengetahuan-pengetahuan tersebut dengan menggunakan taksonomi pengetahuan, pembuatan *sociogram*, dan model dari portal manajemen pengetahuan.

4. Hasil Penelitian

Dari hasil analisa kondisi didapatkan bahwa pola kerja yang selama ini dilakukan mengacu kepada pola rutin. Untuk penjadwalan kuliah, permintaan kesediaan mengajar, perkuliahan, ujian dan hal-hal lain dilakukan sesuai dengan kalender akademik yang ditentukan setiap semester. Terkadang dalam sebuah proses yang di dalamnya mencakup bermacam-macam prosedur terdapat prosedur yang terlewat. Hal tersebut dapat disebabkan karena kurangnya pengetahuan atau pembelajaran yang diberikan. Ataupun juga terdapat beberapa kegiatan minor yang tidak tercantum pada kalender akademik sehingga kegiatan tersebut masih bersifat *tacit* karena masih terdapat dalam memori individu masing-masing yang menyebabkan kegiatan tersebut mempunyai potensi untuk terlewat karena belum dibuat secara utuh. Karena terdapat sebuah kegiatan yang tidak dilaksanakan, maka potensi timbulnya masalah semakin besar. Selain itu pencarian dokumen yang memakan waktu yang lama karena tempat penyimpanan yang belum tersusun secara benar dan terkadang terdapat pengetahuan atau pengalaman yang belum dimiliki oleh orang-orang yang terlibat. Pola ini mempunyai masalah jika pada suatu saat orang yang bertanggung jawab untuk tugas tersebut tidak dapat hadir, terkena halangan lainnya atau terjadi pergantian orang dalam masa jabatannya. Dapat saja terjadi bahwa sebuah pekerjaan tidak dapat tuntas karena prosedur kerja yang tidak jelas, kealpaan pengetahuan atau terjadinya sebuah kesalahan

pengambilan keputusan karena tidak adanya pengetahuan yang sama di antara orang-orang yang saat ini terlibat dalam bagian atau proses tersebut.

Karena itu dibutuhkan sebuah sistem yang mempunyai kemampuan untuk menyimpan pengetahuan dan berbagi pengetahuan. Seluruh pengetahuan tersebut dapat diakses ataupun diberikan meskipun orang-orang yang mengetahui pengetahuan tersebut tidak berada di tempat. Setiap orang berhak untuk memasukkan ide ataupun pengetahuan mengenai pengetahuan ini terutama jika terjadi kasus-kasus atau masalah yang baru, terjadi perubahan kebijakan akademik sehingga pengetahuan organisasi akan terus berkembang sesuai dengan kondisi yang diharapkan. Selain itu juga dimaksudkan agar tercipta sebuah standar pelayanan yang sama, prosedur yang sama, dan pengetahuan yang sama di antara para anggota yang terlibat dalam organisasi.

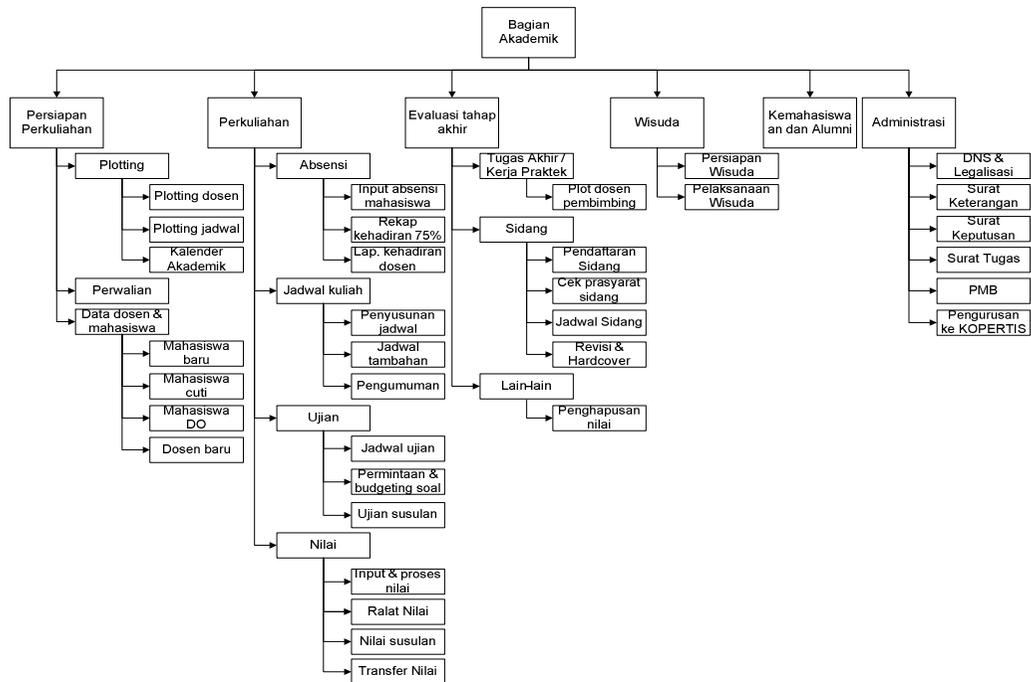
Untuk dapat melaksanakan hal di atas, maka *tacit knowledge* yang masih berada individu-individu tersebut dapat diambil dan dapat disatukan dalam sebuah sistem *knowledge base management*. Keunggulan pertama dari sistem ini yaitu bahwa pengetahuan yang sudah didapatkan organisasi tidak akan hilang karena tidak disimpan dalam bentuk *tacit knowledge*. Kedua bahwa semua orang dapat dapat memanfaatkan pengetahuan tersebut agar tercipta sebuah standar prosedur kerja yang sama meskipun orang yang mengetahui pengetahuan tersebut tidak berada di tempat. Ketiga yaitu memungkinkan untuk terjadinya proses pembaharuan pengetahuan sesuai dengan kebijakan yang berlaku untuk saat ini atau berdasarkan kasus-kasus yang sering terjadi. Selain itu juga dapat menghindari keterlambatan pekerjaan rutin, mengatur peraturan-peraturan yang sama untuk semua prosedur kerja, dan memberikan pengetahuan yang sama bagi seluruh staf dan bagian-bagian lain yang terlibat. Dan jika pengetahuan-pengetahuan tersebut sudah disatukan sehingga menjadi sebuah *knowledge base management* maka pengambilan keputusan dapat dilakukan dengan cepat dan benar berdasarkan pengetahuan yang ada ataupun dari masalah yang pernah terjadi.

Dalam proses mendapatkan pengetahuan tersebut maka penulis melakukan wawancara dengan seluruh pihak yang terkait dengan proses-proses yang ada sehingga penulis mendapatkan gambaran umum tentang proses yang dilakukan beserta dengan penjelasan dari proses yang dilakukan dan dokumen-dokumen yang terkait. Kemudian dilakukan dengan proses observasi terhadap setiap proses yang dilakukan sehingga alur proses yang tadi sudah didapatkan dari hasil wawancara dapat dilengkapi kembali. Observasi terhadap proses-proses ini perlu untuk dilakukan karena pada saat dilakukan wawancara, mungkin saja terdapat *tacit knowledge* yang secara tidak sadar dilakukan pada saat melakukan sebuah proses. Dengan demikian dikehendaki agar seluruh proses dapat diketahui dengan mendalam dan tepat.

Saat ini, seluruh dokumen-dokumen yang terkait sudah dimasukkan dalam sebuah map/ rak, diberi label dan jumlahnya diperiksa secara berkala. Akan tetapi, terkadang ketika jumlah permintaan banyak, maka formulir (dokumen) tersebut dapat habis dan diperlukan waktu untuk memperbanyak dokumen yang dapat menyebabkan keterlambatan pemrosesan.

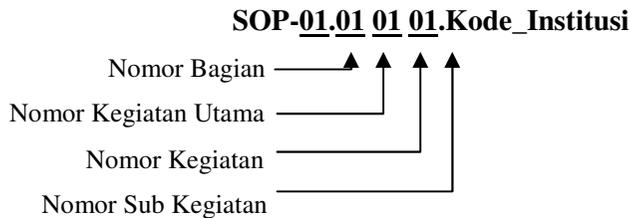
Pembentukan klasifikasi pengetahuan didasarkan atas hasil wawancara dan observasi. Pengetahuan-pengetahuan tersebut dikelompokkan sesuai dengan

pekerjaan utama yang ada. Berikut adalah contoh klasifikasi pengetahuan yang dibentuk:



Gambar 3. Rancangan Umum Klasifikasi Pengetahuan Bagian Akademik

Untuk tahap kodifikasi pengetahuan, dilakukan sesuai dengan pembagian dari taksonomi pengetahuan yang sudah dibentuk. Secara umum, pembagian kode itu adalah sebagai berikut:



Berikut disertakan pula contoh rancangan SOP:

1. Rancangan cover SOP

Kode: SOP-01.0301.Kode_Instansi	Revisi : 00	Tanggal Berlaku: 1 Agustus 2009
Pembuat :		
<p>STANDARD OPERATING PROCEDURES BAGIAN ADMINISTRASI AKADEMIK TUGAS AKHIR/KERJA PRAKTEK</p>		
Nama Institusi Alamat Institusi	Kunjungi situs kami di www.alamatwebinstitusi.ac.id	

2. Rancangan lembar revisi dan persetujuan SOP

Revisi

<i>Tanggal Revisi:</i>	<i>Tanggal Revisi Selanjutnya:</i>
------------------------	------------------------------------

<i>No.</i>	<i>Tanggal Revisi</i>	<i>Perubahan</i>	<i>Penulis</i>

Persetujuan

Dokumen ini memerlukan persetujuan dari:

<i>Nama</i>	<i>Jabatan</i>

Distribusi

Dokumen ini telah didistribusikan kepada:

<i>Nama</i>	<i>Jabatan</i>

3. Rancangan lembar revisi dan persetujuan SOP

1. Definisi

Merupakan titik akhir dari masa perkuliahan di mana mahasiswa diharuskan untuk membuat tugas akhir atau kerja praktek sebagai salah satu syarat kelulusan baik untuk Program Sarjana maupun Diploma 3.

2. Tujuan

Memberikan dosen pembimbing bagi mahasiswa yang baru pertama kali mengambil tugas akhir (untuk program Sarjana) dan Kerja Praktek (untuk program Diploma 3), memvalidasi judul tugas akhir dan kerja praktek yang akan disusun agar tidak terjadi duplikasi, dan mengatur untuk bimbingan bersama bagi para mahasiswa yang baru mengambil tugas akhir.

3. Pelaksana

Penanggung jawab dari pekerjaan ini adalah Bagian Administrasi dan Akademik / Koordinator Tugas Akhir. Pihak lain yang terlibat adalah Ketua Jurusan, dosen dan mahasiswa yang mengambil tugas akhir atau kerja praktek.

4. Standar Waktu Pelaksanaan

<i>No.</i>	<i>Proses</i>	<i>Standar waktu pelaksanaan</i>
<i>1</i>	<i>Penentuan dosen pembimbing TA/ KP</i>	<i>Pengumuman dosen pembimbing dilakukan setelah mahasiswa memasukkan judul TA / KP yang kemudian akan dievaluasi oleh Ketua Jurusan / Koordinator Tugas Akhir.</i>

5. Prosedur

Prosedur ini merupakan prosedur yang dilakukan untuk menentukan dosen pembimbing bagi setiap mahasiswa yang mengambil tugas akhir maupun kerja praktek.

PENENTUAN DOSEN PEMBIMBING TA / KP (SOP-01.030101.Kode_Instansi)

1. Prosedur 1
2. Prosedur 2
3. Prosedur 3
4. dst.

Setelah penyusunan SOP selesai, langkah selanjutnya adalah untuk memodelkan alur informasi yang terdapat pada bagian administrasi akademik menggunakan *sociogram*. Pada bagian ini, terdapat beberapa orang yang terlibat yaitu bagian administrasi akademik, dosen, mahasiswa, dan pejabat struktural (Rekrorat, Dekan, Pembantu Dekan, Ketua dan Sekretaris Jurusan). Untuk pembentukan *sociogram* ini, perlu memperhatikan beberapa hal yaitu:

1. Struktur organisasi
2. Relasi di antara anggota-anggota dalam organisasi

Dalam pembuatan *sociogram* terdapat hal yang perlu diperhatikan yaitu struktur organisasi. Akan tetapi *sociogram* ini bukanlah struktur organisasi di mana struktur organisasi memperlihatkan hubungan formal melainkan merupakan pemetaan dan alat ukur hubungan, arus yang terjadi antara setiap individu, kelompok, organisasi yang memproses pengetahuan.

Dalam sebuah organisasi akademik, tidaklah mungkin untuk menghindari informasi yang berasal dari struktur organisasi karena keputusan biasanya diambil pada rapat atau pemegang keputusan barulah dibagikan kepada unit-unit kerja yang berada di bawahnya. Dari struktur organisasi akan didapatkan alur informasi secara formal di mana informasi biasanya mengalir dari pucuk pimpinan tertinggi kemudian mengalir ke unit-unit yang ada di bawahnya. Akan tetapi terkadang informasi tersebut juga dapat tersebar tanpa melalui hubungan secara formal karena terdapat relasi/ kedekatan di antara anggota-anggota dalam organisasi.

Tahap selanjutnya adalah pembentukan portal manajemen pengetahuan. Pada fase ini akan dibentuk sebuah tim yang bertugas mengatur pengetahuan-pengetahuan yang ada dan sebagai pelaksana pengembangan portal manajemen pengetahuan ini. Untuk pengembangan portal tentu saja diperlukan pihak pimpinan, pejabat struktural, tim ahli dan *end user*. Untuk tim ahli yang perlu dilibatkan adalah para personel yang menguasai masalah pada bagian tersebut, pejabat struktural, dan juga dapat melibatkan *end user*. Selain itu juga perlu diperhatikan sarana penyebaran dari portal ini. Pembangunan model dari portal manajemen pengetahuan ini adalah berbasis web. Karena itu tentu diperlukan sarana dan prasarana yang memadai. Yang termasuk dalam hal ini adalah sumber daya perangkat keras, perangkat lunak, dan juga sumber daya manusia.

Portal manajemen pengetahuan ini akan disimpan pada sebuah server sebagai basis dari penempatan portal manajemen pengetahuan ini. Portal ini akan dapat diakses oleh semua pengguna (karyawan, pejabat struktural, pimpinan, dll) melalui komputer yang terhubung dengan jaringan intranet. Desain model ini mengacu kepada tujuan dari manajemen pengetahuan di mana sebuah pengetahuan ditangkap, dikodifikasi dan kemudian dibagikan atau disebarluaskan kepada pengguna. Selain itu juga diperlukan sarana untuk berbagi pikiran atau informasi sehingga timbul sebuah interaksi, dialog dan *sharing* pengetahuan. Hal ini harus dapat dipenuhi oleh model portal manajemen pengetahuan ini.

Pada tampilan awal dari model portal manajemen ini hanya terdapat menu *home*, form untuk *login*, dan menu untuk forum beserta tampilan untuk artikel yang sering dibaca dan kalender. Untuk mengakses pengetahuan yang terdapat dalam model portal manajemen pengetahuan ini, setiap user diwajibkan untuk login

terlebih dahulu. Hal ini dimaksudkan agar hanya user yang telah terdaftar yang berhak untuk menggunakan dan mengetahui pengetahuan dan prosedur yang terdapat dalam *knowledge center* ini.

Setiap user baru dapat mengisi form registrasi yang terdapat pada portal ini (create an account) kemudian dapat memberitahukan kepada administrator portal ini alamat e-mail dan user yang dimasukkan agar *account* untuk menggunakan portal manajemen pengetahuan ini dapat diaktifkan. Setiap user yang diberikan kepadanya adalah hak akses untuk membaca artikel dan menulis ke dalam forum.

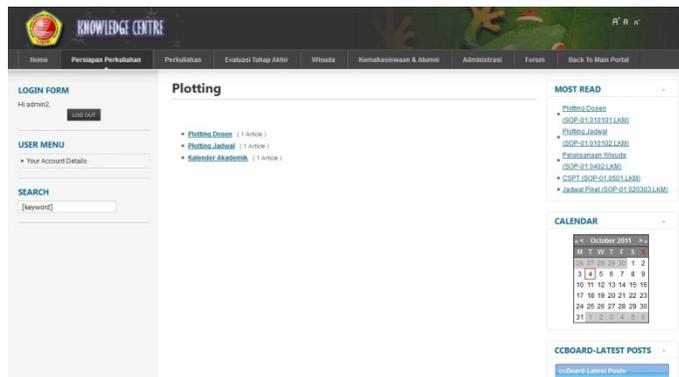
Setelah pengguna memasukkan *username* dan *password* yang telah didaftarkan, maka menu yang muncul akan bertambah. Menu-menu yang muncul tersebut merupakan gambaran dari taksonomi pengetahuan di Bagian Akademik (Gambar 3). Untuk dapat melihat tingkatan dari taksonomi tersebut cukup arahkan *mouse* ke menu dan sub-menu yang akan dibuka. Ataupun jika pengguna masih bingung tentang artikel yang harus dicari dapat menggunakan form pencarian (*search*) yang akan muncul setelah pengguna melakukan proses *login*. Pada model portal ini juga terdapat sebuah forum yang digunakan untuk saling berbagi pengetahuan mengenai prosedur atau proses yang ada.

Untuk dapat menulis atau mengubah artikel ini, akses hanya diberikan kepada kepala bagian dari masing-masing kepala bagian. Hal ini untuk mencegah perubahan pengetahuan secara sembarangan dan menjaga keutuhan dari pengetahuan yang telah ada. Selain kepada kepala bagian, hak akses untuk mengubah atau menulis artikel tersebut juga akan diberikan kepada tim manajemen pengetahuan yang akan dibentuk. Setiap pengetahuan yang dimasukan dalam model portal manajemen pengetahuan ini juga dibatasi aksesnya dalam arti hanya user yang telah terdaftar yang dapat melihat pengetahuan-pengetahuan tersebut. Untuk lebih jelasnya, hak akses setiap user dapat dilihat dalam tabel di bawah ini:

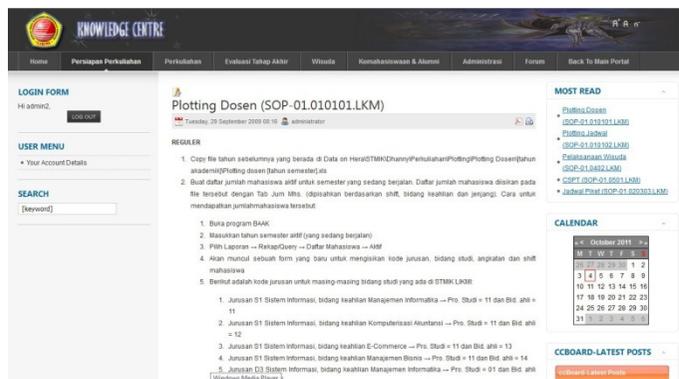
Tabel 1. Hak Akses User

<i>Jenis User</i>	<i>Group</i>	<i>Hak Akses</i>
<i>Super Administrator</i>	<i>Back-End</i>	<i>Akses penuh dan dapat menghapus seluruh artikel (pengetahuan) dan pengguna yang ada</i>
<i>Administrator</i>	<i>Back-End</i>	<i>Akses penuh dan dapat menghapus artikel tetapi tidak dapat menghapus jenis user super administrator</i>
<i>Author</i>	<i>Front-End</i>	<i>Akses diberikan untuk mengubah dan menambahkan artikel</i>
<i>Registered</i>	<i>Front-End</i>	<i>Akses yang diberikan hanya dapat membaca artikel</i>

Berikut adalah beberapa cuplikan implementasi yang sudah dilakukan:



Gambar 4. Tampilan Halaman Plotting



Gambar 5. Tampilan SOP Plotting Dosen

Implementasi selanjutnya dari portal ini akan ditambahkan fungsi kalender yang berfungsi untuk mencatat kegiatan-kegiatan akademik dari institusi. Kalender ini pun sudah mempunyai fungsi untuk membagi jenis-jenis kegiatan dan mengirimkan notifikasi kepada setiap anggotanya begitu memasuki kalender kegiatan yang baru. Selain itu akan diberikan sebuah forum sebagai sarana untuk berbagi pikiran dan informasi berdasarkan kasus-kasus yang pernah terjadi atau saat ini sedang terjadi. Jika tim manajemen pengetahuan mendapatkan pengetahuan baru, maka dapat diadakan pertemuan untuk membahas pengetahuan yang baru tersebut. Jika ternyata pengetahuan yang baru tersebut lebih efektif, maka dapat diadakan perubahan pengetahuan (SOP).

5. Kesimpulan

Berdasarkan hasil penyusunan model *knowledge management system* khususnya pada Bagian Administrasi Akademik maka dapat disimpulkan beberapa hal berikut:

1. Pengetahuan akan dapat dibuat lebih terorganisasi dengan membuat sistem ini dan membuat alur proses yang jelas terhadap proses-proses yang ada.
2. Pengumpulan *tacit* dan *explicit knowledge* dapat dilakukan dengan memasukkan informasi atau pengetahuan yang baru pada forum dan

kemudian dapat dibahas dengan lebih detail jika benar-benar ingin dimasukkan ke dalam KMS.

3. Pencarian prosedur atau dokumen dapat dilakukan pada KMS ini sehingga akan meningkatkan waktu pelayanan dan ketepatan dalam melaksanakan prosedur.
4. Terdapat beberapa faktor teknis di dalam melaksanakan prosedur ini yang sulit untuk diungkapkan dalam menyusun KMS ini. Faktor teknis ini antara lain dapat berupa kedekatan dengan dosen atau mahasiswa, kemampuan negosiasi, kemampuan dalam menjelaskan sesuatu, kecerdasan linguistik.

Model ini dapat disesuaikan sesuai dengan kebutuhan dari masing-masing institusi.

6. Daftar Pustaka

- [Cab03] Caballero, Rich, Jim Cook, Chad Fisher, Doug Gantt, Bill Miller, *Developing Standar Operating Procedures in Wildland fire Management*, Leadership Toolbox Reference SOP Workbook, 2003
- [Dal05] Dalkir, Kimiz, *Knowledge Management in Theory and Practice*, Elsevier, Oxford, 2005
- [Dav98] Davenport, Thomas H & Prusak, *Working Knowledge: How Organizations Manage What They Know*. Boston: Harvard Business School Press, 1998
- [Kla99] Klasson, Kirk, *Managing Knowledge for Advantage: Content and Collaboration Technologies*, The Cambridge Information Network Journal vol. 1, no. 1, 1999
- [San05] Sanchez, Ron, *Knowledge Management and Organizational Learning: Fundamental Concept for Theory and Practice*, Lund Institute of Economic Research Working Paper Series, 2005
- [San07] Sangkala, *Knowledge Management*, PT RajaGrafindo Persada, 2007
- [Sha01] Sharp, Alec and Patrick McDermott, *Workflow Modeling Tools For Process Improvement and Application Development*, Artech House, 2001
- [Tar03] Targowski, Andrew S., *Electronic Enterprise : Strategy and Architecture*, IRM Press., 2003
- [Twi99] Twiwana, Amrit, *The Knowledge Management Toolkit*, Prentice Hall, 1999

Sistem Informasi Toko Listrik Aryono King dengan Peramalan Stok Barang menggunakan Metode *Double Exponential Smoothing*

Radiant Victor Imbar, Yon Andreas

Jurusan S1 Sistem Informasi

Fakultas Teknologi Informasi, Universitas Kristen Maranatha

Jl. Prof. Drg. Suria Sumantri No. 65 Bandung. 40164

email : radiant.vi@eng.maranatha.edu, oyon87@yahoo.com

Abstract

At this time the development of information technology grow faster. Electrical Stores Aryono King makes an information system to support their sales system. This application include the sales process, purchasing, supply, and make financial report. The advantage of this application can do forecasting future stock preparation based on the sale of items before, using a methods of Double Exponential Smoothing. With this information system is expected to help electrical shop to be more efficient in terms of data processing and effective in providing stock of goods each month.

Keyword : financial report, forecasting, purchasing, sales, supply.

1. PENDAHULUAN

1.1 Latar Belakang

Banyak perusahaan atau toko yang sudah menggunakan komputer yang dilengkapi sistem informasi yang bertujuan untuk melakukan proses penyimpanan datanya.

Toko listrik Aryono King yang saat ini masih melakukan pengaturan bisnisnya secara manual merasakan kelemahan sistem manual dalam menjalankan bisnisnya. Banyaknya barang dan harga menjadi kendala dalam melakukan proses bisnisnya yang berakibat kesalahan harga dan pengecekan stok barang. Dari masalah diatas muncul pemikiran untuk membuat sebuah sistem informasi Toko Listrik yang dapat membantu dalam menjalani bisnisnya sehari – hari. Sistem informasi yang dapat melakukan proses penjualan, pembelian, mengatur data barang dan peramalan stok barang mendatang.

Diharapkan dengan adanya sistem informasi ini, akan membuat data – data barang terkomputerisasi dengan baik.. Sistem informasi yang dibuat akan dilengkapi dengan sistem peramalan stok barang diharapkan dapat menambah kinerja dan pelayanan terhadap para pelanggan dalam hal penyajian barang.

Sistem Informasi yang dibuat dapat menangani :

1. Melakukan proses pembelian barang secara terkomputerisasi.
2. Melakukan proses penjualan barang secara terkomputerisasi.

3. Membuat pengaturan stok barang, harga (satuan, grosir, dan beli) dari setiap barang yang ada dan pengecekan barang yang harus di retur.
4. Membuat laporan keuangan berdasarkan semua transaksi yang terjadi.
5. Melakukan peramalan stok barang bulan berikutnya dengan menggunakan metode *Double Exponential Smoothing*.

1.2 Rumusan Masalah

Adapun rumusan – rumusan masalah yang akan dibahas antara lain :

1. Bagaimana membuat Sistem Informasi toko listrik yang dapat melakukan proses pembelian barang ?
2. Bagaimana membuat Sistem Informasi toko listrik yang dapat melakukan proses penjualan barang ?
3. Bagaimana membuat Sistem Informasi untuk mengatur barang mulai dari stok, harga dan barang retur ?
4. Bagaimana membuat Sistem Informasi yang dapat membuat laporan keuangan berdasarkan transaksi yang terjadi ?
5. Bagaimana membuat sistem peramalan stok barang, sehingga tidak terjadi penumpukan stok barang?

1.3 Tujuan Penelitian

Sesuai dengan rumusan masalah pada nomor 1.2, maka tujuan penelitian ini adalah :

1. Melakukan proses pembelian barang secara terkomputerisasi.
2. Melakukan proses penjualan barang secara terkomputerisasi.
3. Membuat pengaturan stok barang, harga (satuan, grosir, dan beli) dari setiap barang yang ada dan pengecekan barang yang harus di retur.
4. Membuat laporan keuangan berdasarkan semua transaksi yang terjadi.
5. Melakukan peramalan stok barang bulan berikutnya dengan menggunakan metode *Double Exponential Smoothing*.

1.4 Batasan Masalah

Batasan – batasan Sistem Informasi

1. Sistem Informasi ini tidak dapat di akses oleh orang luar, aplikasi bersifat *private* yang hanya dapat digunakan pada toko listrik Aryono King.
2. Sistem Informasi ini digunakan untuk melakukan proses bisnis.
3. Sistem dapat melakukan proses penjualan dan pembelian untuk toko listrik AryonoKing
4. Sistem dapat melakukan proses penyimpanan data barang yang rusak untuk ditukarkan ke *supplier*
5. Sistem dapat melakukan proses peramalan stok barang yang akan datang.

2. LANDASAN TEORI

2.1 Proses Bisnis

Proses bisnis dapat didefinisikan juga sebagai kelompok - kelompok dari keputusan-keputusan yang terkait dan kegiatan - kegiatan yang dibutuhkan untuk mengelola sumberdaya - sumberdaya bisnis (IBM-BSP, 1984). Sementara itu, Menurut Manganeli & Klein (1994), proses bisnis didefinisikan sebagai:

“*Interrelated series of activities that convert business input into business output*”. Masukkan dapat berupa material, peralatan, objek terukur lainnya, ataupun berbagai macam informasi yang kemudian diubah menjadi sejumlah keluaran yang diperlukan oleh penerima.

2.2 Sistem Informasi

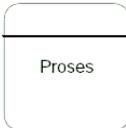
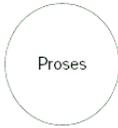
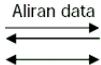
Sistem informasi adalah aplikasi komputer untuk mendukung operasi dari suatu organisasi: Operasi, instalasi, dan perawatan komputer, perangkat lunak, dan data. Sistem informasi adalah satu kesatuan data olahan yang terintegrasi dan saling melengkapi yang menghasilkan output baik dalam bentuk gambar, suara maupun tulisan.

2.3 Basis Data

Sebuah basis data adalah sebuah struktur yang umumnya dikategorikan dalam 2 hal, yaitu sebuah basisdata *flat* dan sebuah basisdata relasional. MySQL merupakan sebuah *database* relasional. Pada *database* yang memiliki struktur relasional, terdapat tabel-tabel yang menyimpan data. Setiap tabel terdiri dari kolom dan baris. Sebuah kolom mendefinisikan jenis informasi yang akan disimpan (misalnya nama, umur, berat, tinggi). Jika kolom mendefinisikan jenis informasi yang akan disimpan, maka sebuah baris adalah data aktual yang disimpan. Setiap baris dari tabel adalah masukan dari tabel tersebut dan berisi nilai-nilai untuk setiap kolom tabel tersebut. (contoh masukannya adalah Yon Andreas umur 23 tahun, 50 kg dan 170cm).

2.4 Data Flow Diagram (DFD)

DFD adalah suatu model logika data atau proses yang dibuat untuk menggambarkan dari mana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut (Hartini, 2008).

Gane/Sarson	Yourdon/De Marco	Keterangan
		Entitas eksternal, dapat berupa orang/unit terkait yang berinteraksi dengan sistem tetapi diluar sistem
		Orang, unit yang mempergunakan atau melakukan transformasi data. Komponen fisik tidak diidentifikasi.
		Aliran data dengan arah khusus dari sumber ke tujuan
		Penyimpanan data atau tempat data direfer oleh proses.

Gambar 1 Simbol DFD (Hartini, 2008)

2.5 Entity Relationship Diagram(ERD)

ERD adalah model konseptual yang mendeskripsikan hubungan antara penyimpanan. ERD digunakan untuk memodelkan struktur data dan hubungan antar data. Dengan ERD, model dapat diuji dengan mengabaikan proses yang dilakukan (Hartini, 2008). ERD pertama kali dideskripsikan oleh Peter Chen yang dibuat sebagai bagian dari perangkat lunak CASE. Notasi yang digunakan dalam ERD dapat dilihat pada Tabel di bawah ini.

Dalam ERD terdapat istilah entitas lemah (*Weak Entity*) dan entitas kuat (*Strong Entity*). Entitas kuat adalah entitas yang memiliki atribut kunci (*Key*), entitas yang mandiri, yang keberadaannya tidak bergantung pada keberadaan entitas yang lainnya. Instansiasi entitas kuat selalu memiliki karakteristik yang unik disebut *identifier* (sebuah atribut tunggal atau gabungan atribut-atribut yang secara unik dapat digunakan untuk membedakannya dari entitas kuat yang lain) (Hartini, 2008).

2.6 Decision Support System (DSS)

Decision Support System (DSS) atau Sistem Penunjang Keputusan (SPK) muncul pertama kali pada awal tahun 1970-an oleh Scott-Morton. Mereka mendefinisikan SPK sebagai suatu sistem interaktif berbasis komputer yang dapat membantu para pengambil keputusan dalam menggunakan data dan model untuk memecahkan persoalan yang bersifat tidak terstruktur.

Dari definisi tersebut, dapat diindikasikan empat karakteristik utama dari SPK, yaitu (Marimin, 2006) :

1. SPK menggabungkan data dan model menjadi satu bagian.
2. SPK dirancang untuk membantu para manajer (pengambil keputusan) dalam proses pengambilan keputusan dari masalah yang bersifat semi structural (atau tidak terstruktur).
3. SPK lebih cenderung dipandang sebagai penunjang penilaian manajer dan sama sekali bukan untuk menggantikannya.
4. Teknik SPK dikembangkan untuk meningkatkan efektifitas dari pengambil keputusan.

Definisi lain dari SPK menurut Minch dan Burns dalam Eriyanto (1998) adalah konsep spesifik sistem yang menghubungkan komputerisasi informasi dengan para pengambil keputusan sebagai pemakainya. Karakteristik pokok yang melandasi teknik SPK adalah :

1. Interaksi langsung antara komputer dengan pengambil keputusan.
2. Dukungan menyeluruh dari keputusan bertahap ganda.
3. Suatu sintesa dari konsep yang diambil dari berbagai bidang, antara lain ilmu komputer, psikologi, intelegensia buatan, ilmu sistem dan ilmu manajemen.
4. Mempunyai kemampuan adaptif terhadap perubahan kondisi dan kemampuan berevolusi menuju sistem yang lebih bermanfaat.

Secara umum, SPK terdiri dari tiga komponen, yaitu (Marimin, 2006) :

1. Manajemen Data. Termasuk di dalamnya adalah database yang berisi data yang berhubungan dengan sistem yang diolah menggunakan perangkat lunak yang disebut sistem manajemen basis data.
2. Manajemen Model. Yaitu paket perangkat lunak yang terdiri dari model financial, statistikalm ilmu manajemen, atau model kuantitatif lain yang menyediakan kemampuan sistem analisis.
3. Subsistem dialog. Yaitu subsistem yang menghubungkan pengguna dengan perintah perintah dalam SPK.

2.7 Forecasting

Forecasting adalah peramalan apa yang akan terjadi pada masa yang akan datang, sedangkan rencana merupakan penentuan apa yang akan dilakukan pada awaktu yang akan datang. Dengan sendirinya terjadi perbedaan antara *forecast* dengan rencana (pangestu, 1986).

Forecast adalah peramalanan apa yang akan terjadi, tapi belum tentu bisa dilaksanakan oleh perusahaan. *Forecasting* bertujuan untuk mendapatkan peramalan atau prediksi yang bisa meminimumkan kesalahan dalam meramal yang biasanya diukur dengan *mean square error*, *mean absolute error* (Pangestu, 1986).

Metode Double Exponential Smoothing

Metode ini merupakan model linier yang dikemukakan oleh Brown. Didalam metode *Double Exponential Soothing* dilakukan proses *smoothing* dua kali , sebagai berikut :

$$S'_t = \alpha X_t + (1 - \alpha)S'_{t-1}$$

$$S''_t = \alpha S'_t + (1 - \alpha)S''_{t-1}$$

$$a_t = 2S'_t - S''_t$$

$$b_t = \frac{\alpha}{1 - \alpha} (S'_t - S''_t)$$

$$S_{t+m} = a_t + b_t m$$

Keterangan :

S_{t+m} = Nilai ramalan untuk *m periode* ke depan

m = Jarak periode yang akan diramalkan

X_t = Nilai actual periode ke-t

S'_t = Nilai *Smoothing period* ke-t

α = Konstanta *Smoothing* (1/n)

Contoh Soal :

Tabel 1 Tabel Permintaan Barang (Pangestu,1986:26)

Minggu	Permintaan Barang
1	120
2	125
3	129
4	124
5	130

Akan dicari ramalan minggu ke-6 dengan $\alpha = 0,2$

$$S'_t = \alpha \cdot X_t + (1 - \alpha)S'_{t-1}$$

$$S'_1 = 120$$

$$S'_2 = (0,2)125 + (0,8)120 = 121$$

$$S'_3 = (0,2)129 + (0,8)121 = 122,60$$

$$S'_4 = (0,2)124 + (0,8)122,60 = 122,88$$

$$S'_5 = (0,2)130 + (0,8)122,88 = 124,30$$

$$S''_t = \alpha \cdot S'_t + (1 - \alpha)S''_{t-1}$$

$$S''_1 = 120$$

$$S''_2 = (0,2)121 + (0,8)120 = 120,2$$

$$S''_3 = (0,2)122,60 + (0,8)120,2 = 120,68$$

$$S''_4 = (0,2)122,88 + (0,8)120,68 = 121,12$$

$$S''_5 = (0,2)124,30 + (0,8)121,12 = 121,76$$

$$a_t = 2S'_t - S''_t$$

$$a_1 = 2(120) - 120 = 120$$

$$a_2 = 2(121) - 120,2 = 121,80$$

$$a_3 = 2(122,60) - 120,68 = 124,52$$

$$a_4 = 2(122,88) - 121,12 = 124,64$$

$$a_5 = 2(124,30) - 121,76 = 126,84$$

$$b_t = \frac{\alpha}{1 - \alpha} (S'_t - S''_t)$$

$$b_1 = 0$$

$$b_2 = \frac{0,2}{0,8} (121 - 120,2) = 0,20$$

$$b_3 = \frac{0,2}{0,8} (122,60 - 120,68) = 0,48$$

$$b_4 = \frac{0,2}{0,8} (122,88 - 121,12) = 0,44$$

$$b_5 = \frac{0,2}{0,8} (124,30 - 121,76) = 0,64$$

$$S_{t+m} = a_t + b_t \cdot m, m=1$$

$$S_6 = a_5 + b_5$$

$$= 126,84 + 0,64$$

$$= 127,48$$

Jadi ramalan penjualan tahun ke-6 adalah 127.48

2.7 Pengertian Akuntansi

Secara umum, akuntansi dapat didefinisikan sebagai sistem informasi yang menghasilkan laporan kepada pihak-pihak yang berkepentingan mengenai aktivitas ekonomi dan kondisi perusahaan (Farahmita, Amanugrahini, Hendrawan, 2008:10).

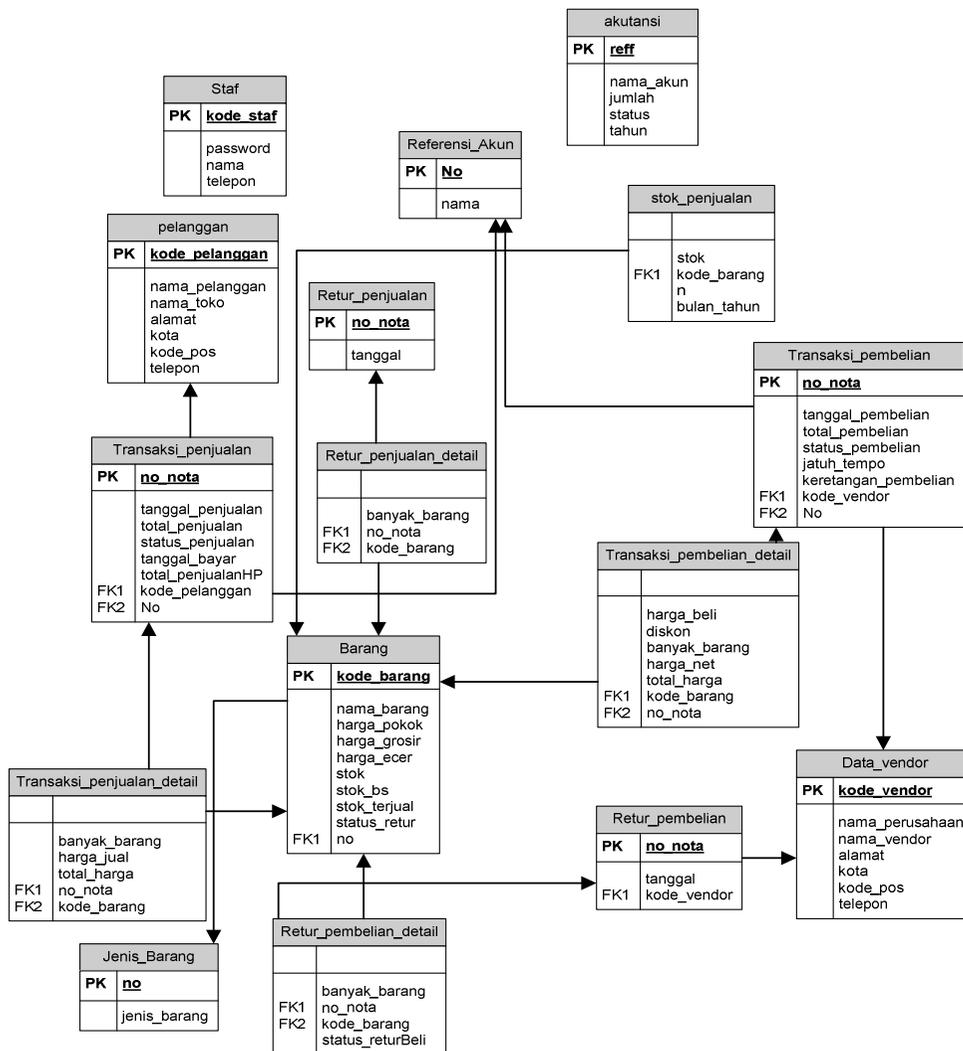
Akuntansi dapat didefinisikan berdasarkan dua aspek penting yaitu:

1. Penekanan pada aspek fungsi yaitu pada penggunaan informasi akuntansi. Berdasarkan aspek fungsi akuntansi didefinisikan sebagai suatu disiplin ilmu yang menyajikan informasi yang penting untuk melakukan suatu tindakan yang efisien dan mengevaluasi suatu aktivitas dari organisasi. Informasi tersebut penting untuk perencanaan yang efektif, pengawasan dan pembuatan keputusan oleh manajemen serta memberikan pertanggung jawaban organisasi kepada investor, kreditor, pemerintah dan lainnya.
2. Penekanan pada aspek aktivitas dari orang yang melaksanakan proses akuntansi. Dalam aspek ini orang yang melaksanakan proses akuntansi harus :
 - Mengidentifikasi data yang relevan dalam pembuatan keputusan.
 - Memproses atau menganalisa data yang relevan.
 - Mengubah data menjadi informasi yang dapat digunakan untuk pembuatan keputusan.

3. ANALISIS DAN RANCANGAN SISTEM

3.1 Tabel Relasi

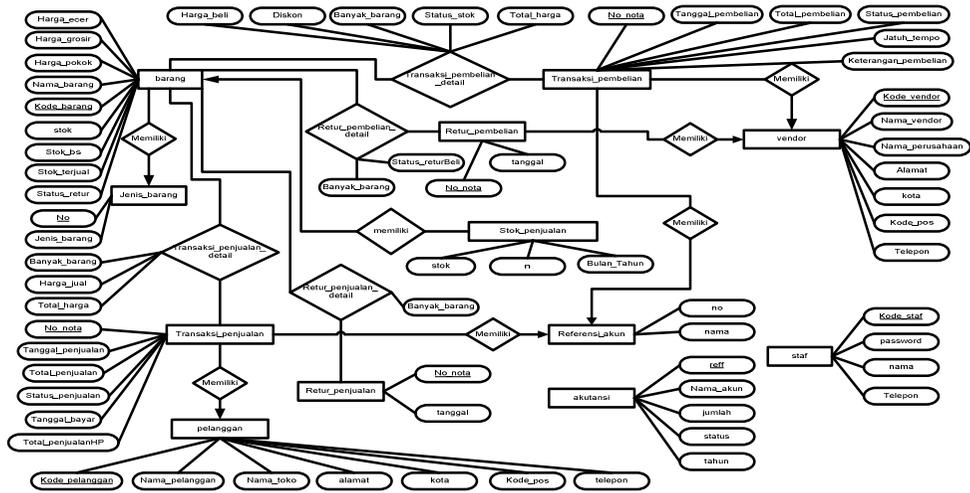
Pada gambar 2 adalah gambar tabel relasi atau hubungan antar tabel yang satu dengan yang lainnya. Dalam sistem informasi ini terdapat 15 tabel yang saling berelasi satu dengan yang lainnya. Untuk lebih jelasnya dapat dilihat pada gambar 2.



Gambar 2 Tabel Relasi

3.2 Entity Relationan Diagram (ERD)

Gambar 3 adalah gambar perancangan desain ERD dari sistem informasi toko listrik AryonoKing.

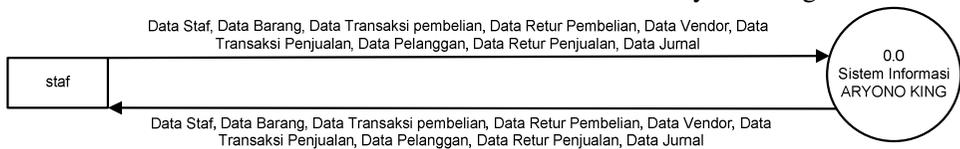


Gambar 3 ERD

3.3 DATA FLOW DIAGRAM (DFD)

3.3.1 DFD Level 0

DFD Level 0 untuk sistem informasi toko listrik Aryono King.



Gambar 4 DFD Level 0

Keterangan :

- Data Barang = kode_barang + nama_barang + harga_pokok + harga_grosir + harga_ecer + stok + stok_bs + stok_terjual + status_retur + no + jenis_barang
- Data Transaksi Pembelian = no_nota + tanggal_pembelian + total_pembelian + status-pembelian + jatuh_tempo + keterangan_pembelian + harga_beli + diskon + banyak_barang + harga_net + total_harga + kode_barang + kode_vendor + no
- Data Retur Pembelian = no_nota + tanggal + status_retur + banyak_barang + kode_barang + kode_vendor
- Data Vendor = kode_vendor + nama_perusahaan + nama_vendor + alamat + kota + kode_pos + telepon
- Data Transaksi Penjualan = no_nota + tanggal_penjualan + total_penjualan + status_penjualan + tanggal_bayar + total_penjualanHP + banyak_barang + harga_jual + total_harga + kode_barang + kode_pelanggan + no
- Data Pelanggan = kode_pelanggan + nama_pelanggan + nama_toko + alamat + kota + kode_pos + telepon

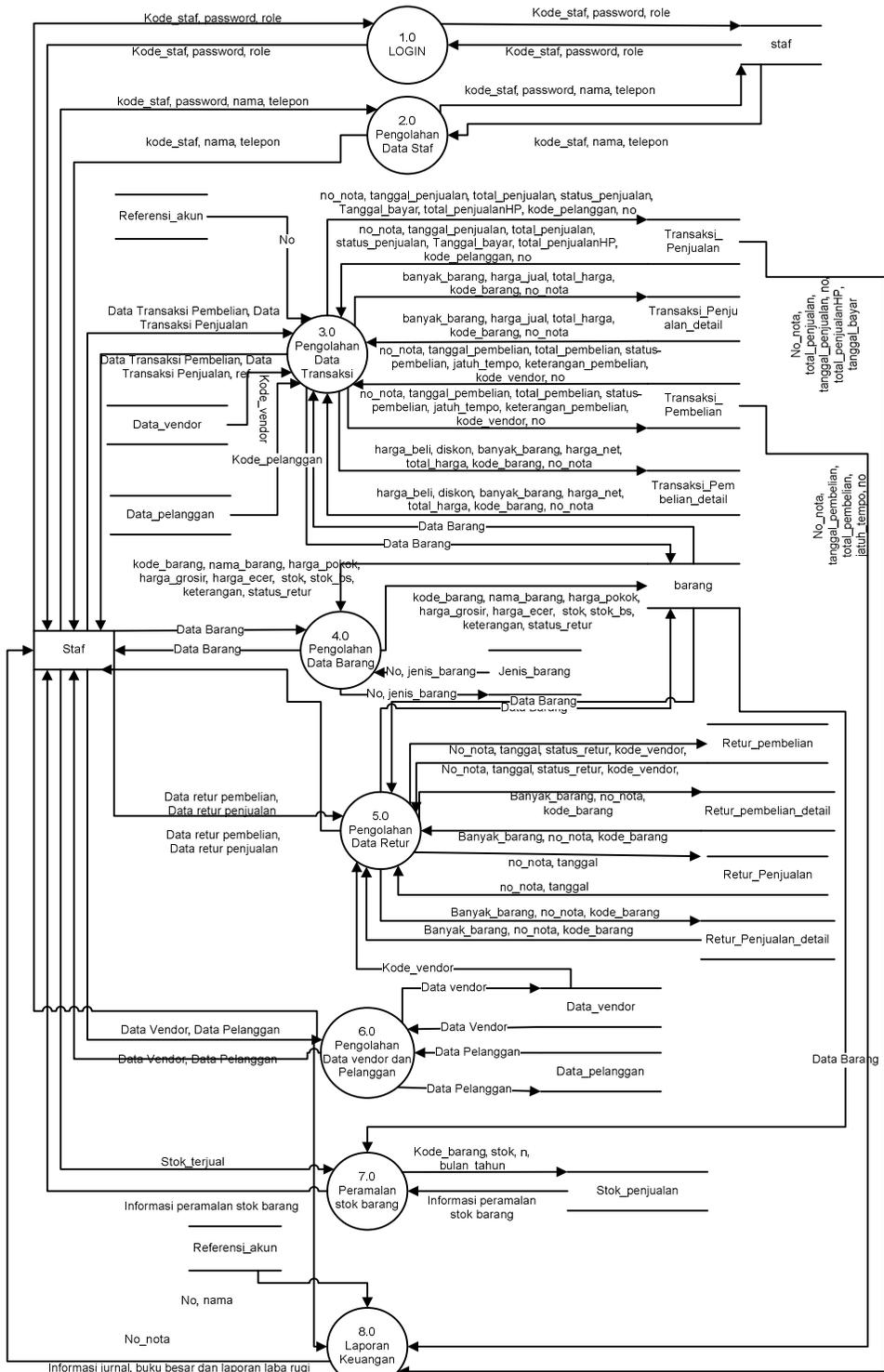
- Data Retur Penjualan = no_nota + tanggal + banyak_barang + kode_barang
- Data Jurnal = No + nama
- Data staf = kode_staf + password + nama + telepon.

3.3.2 DFD Level 1

Keterangan Gambar :

- Data Transaksi Pembelian = no_nota + tanggal_pembelian + total_pembelian + status_pembelian + jatuh_tempo + keterangan_pembelian + harga_beli + diskon + banyak_barang + harga_net + total_harga + kode_barang + kode_vendor + no
- Data Transaksi Penjualan = no_nota + tanggal_penjualan + total_penjualan + status_penjualan + tanggal_bayar + total_penjualanHP + banyak_barang + harga_jual + total_harga + kode_barang + kode_pelanggan + no
- Data Barang = kode_barang + nama_barang + harga_pokok + harga_grosir + harga_ecer + stok + stok_bs + stok_terjual + status_retur + no + jenis_barang
- Data Retur Pembelian = no_nota + tanggal + status_retur + banyak_barang + kode_barang + kode_vendor
- Data Retur Penjualan = no_nota + tanggal + banyak_barang + kode_barang
- Data Vendor = kode_vendor + nama_perusahaan + nama_vendor + alamat + kota + kode_pos + telepon
- Data Pelanggan = kode_pelanggan + nama_pelanggan + nama_toko + alamat + kota + kode_pos + telepon
- Data Bulanan = Kode_penjualanBulanan + bulan + tahun + total_penjualan + kode_barang

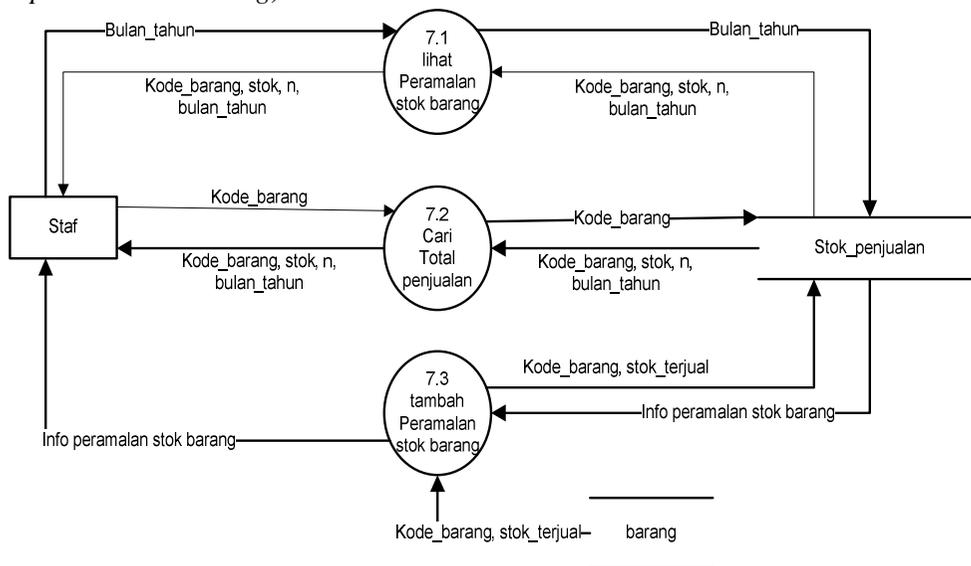
Sistem Informasi Toko Listrik Aryono King dengan Peramalan Stok Barang menggunakan Metode Double Exponential Smoothing (Radiant Victor Imbar, Yon Andreas)



Gambar 5 DFD Level 1

3.3.3 DFD Level 2 Proses 7.0 Peramalan Stok Barang

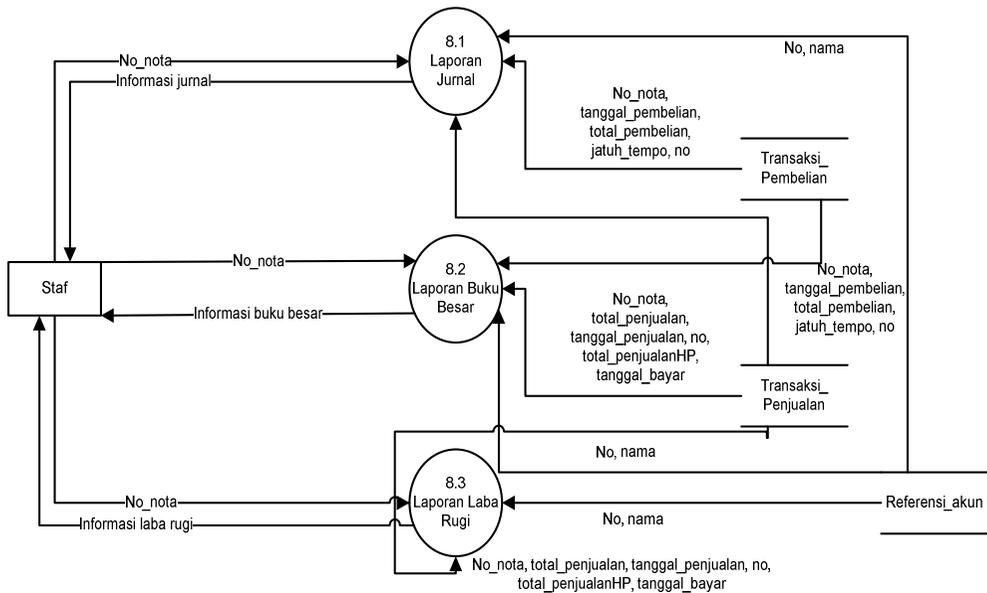
Turunan dari DFD Level 1 proses 7.0 proses peramalan stok barang, terdapat 3 proses yaitu lihat peramalan stock barang, cari total penjualanm tambah peramalan stok barang. Peramalan stok barang terjadi setiap 1 bulan sekali, total penjualan per-barang akan diakumulasikan dengan rumus peramalan (*Double exponential smoothing*).



Gambar 6 DFD Level 2 Proses 7.0

3.3.4 DFD Level 2 Proses 8.0 Laporan Keuangan

Gambar 7 adalah gambar DFD level 2 turunan dari proses 8.0 Laporan keuangan. Pada DFD ini menjelaskan proses yang terjadi dalam laporan keuangan / jurnal.

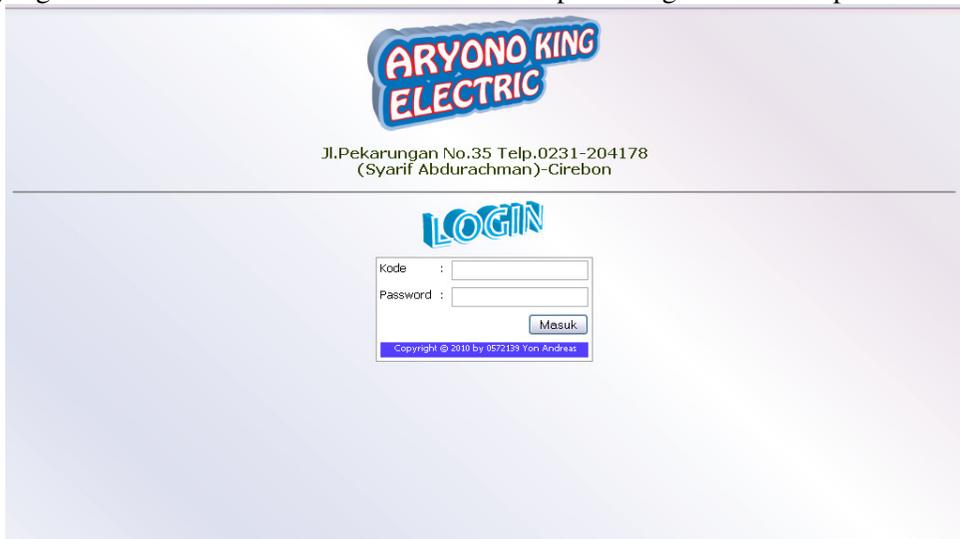


Gambar 7 DFD Level 2 Proses 8.0

4. HASIL PENELITIAN

4.1 Tampilan Login

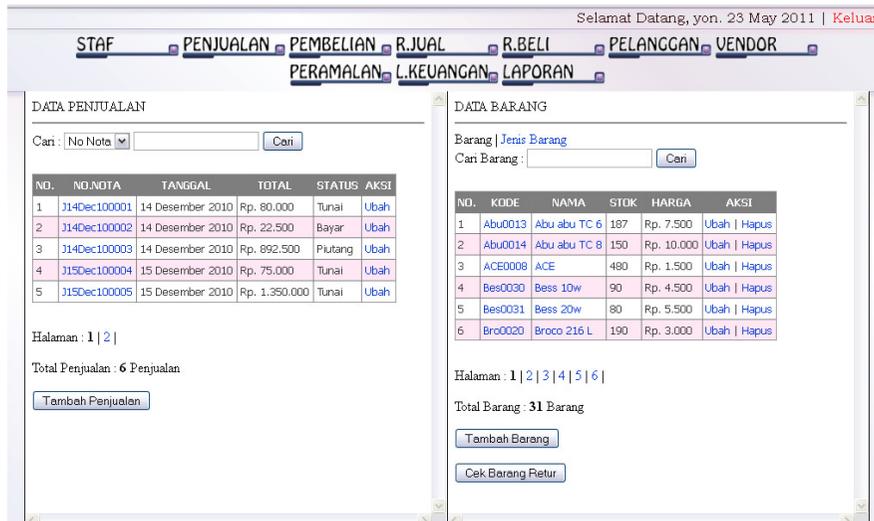
Tampilan pada awal aplikasi adalah tampilan login *User* yang ingin masuk ke dalam sistem informasi harus memasukan kode dan *password* ke dalam *textbox* yang tersedia. Tombol masuk untuk melakukan proses login ke dalam aplikasi.



Gambar 8 Tampilan Login

4.2 Tampilan Awal

Tampilan awal dari sistem informasi toko listrik Aryono King setelah user melakukan proses login. Pada bagian atas terdapat 9 *menu* yang digunakan untuk melakukan proses bisnis dari toko Aryono King. Di bawah *menu* terdapat 2 kolom kerja, kolom pertama untuk melakukan semua kegiatan dari ke 9 menu dan kolom kedua hanya untuk Data Barang(semua kegiatan yang berhubungan dengan barang).



Gambar 9 Tampilan Setelah Login

4.3 Tampilan Peramalan Stok Barang

Gambar 10 adalah tampilan dari peramalan stok barang, semua data barang beserta peramalan stoknya ada pada tampilan ini user dapat melakukan beberapa proses :

- Cari peramalan Barang berdasarkan kode barang (Tombol Cari)
- Lihat detail Barang (Link Nama)
- Liat Detail Peramalan (Link Aksi Detail)
- Memasukan stok penjualan Per bulan, user hanya dapat memasukan stok penjualan per bulan pada saat tanggal 28-31 (Tombol Masukan Stok Penjualan Bulan ini)

NO.	KODE BARANG	NAMA	PERAMALAN STOK	AKSI
1	Abu0013	Abu abu TC 6	Data Tidak Cukup	Detail
2	Abu0014	Abu abu TC 8	8	Detail
3	ACE0008	ACE	10	Detail
4	Bes0030	Bess 10w	0	Detail
5	Bes0031	Bess 20w	34	Detail
6	Bro0020	Broco 216 L	10	Detail
7	Chi0001	Chiyoda 9w	0	Detail
8	Chi0006	Chiyoda 5w Hyper(merah)	10	Detail

Gambar 10 Tampilan Peramalan Stok Barang

4.4 Tampilan Laporan Keuangan

Tampilan di bawah ini adalah tampilan dari Laporan Keuangan yang terdiri dari :

- **Jurnal Umum**
Jurnal Umum berdasarkan transaksi penjualan dan pembelian yang masuk setiap harinya.
- **Buku Besar**
Buku Besar berdasarkan dari transaksi penjualan dan pembelian yang terjadi setiap hari
- **Laporan Rugi / Laba**
Laporan Rugi / Laba dibuat setiap akhir tahun (tutup tahun). Gunanya untuk mengetahui keuntungan toko dalam 1 periode (1 tahun) dan sebagai acuan laporan keuangan tahun berikutnya.

Jurnal Umum

Berikut ini adalah contoh tampilan dari salah satu Jurnal Umum. Jurnal Umum berdasarkan transaksi penjualan dan pembelian yang masuk setiap harinya.

**LAPORAN KEUANGAN
JURNAL UMUM PENJUALAN**

Cari No Nota:

NO.	NO.NOTA	TANGGAL	NAMA	REFF	DEBIT	KREDIT
1	J14Dec100001	14 Desember 2010	Kas	110	Rp. 80.000	
			Penjualan	410		Rp. 80.000
			Harga Pokok Penjualan	510	Rp. 60.770	
			Persediaan Barang Dagang	115		Rp. 60.770
2	J14Dec100002	14 Desember 2010	Piutang	112	Rp. 22.500	
			Penjualan	410		Rp. 22.500
			Harga Pokok Penjualan	510	Rp. 13.500	
			Persediaan Barang Dagang	115		Rp. 13.500
		15 Desember 2010	Kas	110	Rp. 22.500	
			Piutang	112		Rp. 22.500
3	J14Dec100003	14 Desember 2010	Piutang	112	Rp. 892.500	
			Penjualan	410		Rp. 892.500
			Harga Pokok Penjualan	510	Rp. 783.750	
			Persediaan Barang Dagang	115		Rp. 783.750

Halaman : 1 |

Gambar 11 Laporan Keuangan Jurnal Umum

Buku Besar

Berikut ini adalah contoh tampilan dari salah satu buku besar. Buku Besar berdasarkan dari transaksi penjualan dan pembelian yang terjadi setiap hari.

**LAPORAN KEUANGAN
BUKU BESAR KAS**

Cari No Nota:

[Penjualan](#) | [Pembelian](#)

Reff : 110 **Kas Penjualan** Debit, Rp. 1.111.000

NO.	TANGGAL	NO.NOTA	DEBIT	KREDIT
1	14 Desember 2010	J14Dec100001	Rp. 80.000	
2	15 Desember 2010	J14Dec100002	Rp. 22.500	
3	15 Desember 2010	J15Dec100004	Rp. 75.000	
4	15 Desember 2010	J15Dec100005	Rp. 1.350.000	
5	18 April 2011	J18Apr110006	Rp. 75.000	
Total :			DEBIT Rp. 1.602.500	

Halaman : 1 |

Total Buku Besar : 5 Buku Besar

Gambar 12 Laporan Keuangan Buku Besar

4.4.3 Laporan Rugi / Laba

Berikut ini adalah contoh tampilan dari laporan Rugi/ Laba. Laporan Rugi / Laba dibuat setiap akhir tahun (tutup tahun). Gunanya

untuk mengetahui keuntungan toko dalam 1 periode (1 tahun) dan sebagai acuan laporan keuangan tahun berikutnya.

**ARYONO KING
LAPORAN LABA RUGI
2010**

KETERANGAN	DEBIT	KREDIT
Pendapatan :		
Penjualan		Rp. 2.495.000
Total Pendapatan		Rp. 2.495.000
Beban :		
Harga Pokok Penjualan	Rp. 1.960.520	
Total Beban		Rp. 1.960.520
LABA		Rp. 534.480

Gambar 13 Tampilan Laporan Rugi / Laba

5. SIMPULAN DAN SARAN

5.1 Simpulan

Berdasarkan hasil dari penelitian yang telah selesai dikerjakan maka terdapat beberapa kesimpulan yaitu :

1. Dengan adanya Sistem Informasi Toko Listrik ini dapat membuat sebuah proses transaksi pembelian barang dengan lebih praktis dan dapat dengan mudah dalam hal pencarian data transaksi pembelian.
2. Dengan adanya Sistem Informasi Toko Listrik ini dapat membuat sebuah proses transaksi penjualan barang dengan lebih praktis dan dapat dengan mudah dalam hal pencarian data transaksi penjualan.
3. Sistem yang dapat mengolah data barang lebih efisien dan mudah dalam pencarian data barang, stok barang, dan harga barang sehingga dapat terhindar dari kehilangan stok barang dan salah harga.
4. Fitur retur barang membuat pendataan stok barang lebih nyata dan tidak berantakan, mempermudah juga dalam pengembalian barang retur kepada pihak vendor karena sudah terdata.
5. Fitur Laporan Keuangan dapat menjadi pertimbangan bagi pihak toko untuk lebih dapat memaksimalkan pendapatan per tahunnya dan membuat perhitungan keuangan dan barang yang lebih jelas.
6. Dengan adanya Peramalan Stok barang untuk 1 bulan ke depan dapat membuat pihak toko lebih mudah dalam melakukan proses pembelian barang atau pengadaan stok barang, sehingga dapat menghindari penumpukan barang di gudang.

5.2 Saran

Adapun beberapa saran agar sistem informasi ini dapat bekerja lebih maksimal :

1. Pengguna Sistem Informasi harus teliti dalam hal pemasukan barang ke sebuah transaksi agar tidak terjadi kesalahan sebelum transaksi berakhir.

2. Harus dilakukan maintenance berkala dan pengecekan rutin dengan cara pemeriksaan kembali apakah data – data yang sudah dimasukan sudah benar atau ada kesalahan, melakukan update aplikasi jika ada *bug* atau penambahan fitur baru yang dirasa penting
3. Untuk mendapatkan peramalan stok barang yang akurat harus memiliki banyak data penjualan barang per-bulannya.

6. DAFTAR PUSTAKA

- [Fad09] Fadli.(2009). “Belajar MySQL Database”.Retrieved May 14, 2010, from <http://fadli.web.id/belajar-mysql-database.aspx>.
- [Far08] Farahmita, A. Amanugrahini. Hendrawan, T. (2008).*Awarren Reeve Fees accounting – Pengantar Akutansi* (Warren, Carl S. Reeve, James M. Fees, Philip E).Singapore :*Cengage Learning. (Original work published 2005)*.
- [Har08] Hartini,(2008). “Analisis Dengan Diagram Aliran Data (DFD)”. *Roeper Riview*, 8. Retrieved July 16, 2010,form : UNSRI database.
- [Har08] Hartini, (2008). “Analisis Dengan Diagram Keterhubungan ENTITAS (*Entity Relationship diagram/ERD*)”. *Roeper Riview*, 9. Retrieved July 16, 2010, form :UNSRI database.
- [Kri08] Kristanto, A.(2008). *Perancangan Sistem Informasi dan Aplikasinya*, Yogyakarta : Gava Media.
- [Mar06] Marimin, (2006), *Teknik dan Aplikasi Pengambilan Keputusan Kriteria Majemuk*.Jakarta : Grasindo.
- [Rog01] Roger, T. B. (2001). *Management Process Business - Profiting from process*. USA : Sams Publishing.
- [Sub86] Subagyo, P. (1986). *Forecasting Konsep dan Aplikasi*.BPFE.Yogyakarta : Yogyakarta.
- [Sut95] Sutedja, R. B. (1995), *Mudah dan Cepat Menguasai Pemograman Web*. Yogyakarta : Andi Offset.

Pembuatan Perangkat Lunak Aplikasi Pengelolaan Data Kasus Perkara di Kepolisian Resor Kota Bandung Barat

Tiur Gantini, Peter Iman Paskal Mendrofa

Jurusan S1 Teknik Informatika

Fakultas Teknologi Informasi, Universitas Kristen Maranatha

Jl. Prof. Drg. Suria Sumantri No. 65 Bandung 40164

email:tiur.gantini@gmail.com, peter_mendrofa@hotmail.com

Abstract

Criminal cases is one of the most common cases occur in the society environment. This makes the Criminal Investigation Unit be the police performance benchmarks. However, in the practise of daily tasks, these unit still do more manual data management. This thing can reduce the effectiveness and efficiency of work. The design and implementation of data management of criminal cases into a software application is expected to assist the Criminal Investigation Unit in theirs tasks enforcement. Datas used in the designing and implementing of these applications comes from the West Bandung Police Resort, supported by other references obtained from the study of literature and the internet. The application is built using object-oriented programming language C# with .NET framework and the DBMS (Database Management System) using SQL Server. With this application, Criminal Investigation Unit, especially the Administration unit will be able to more easily make the management of criminal data cases as well as preparing monthly reports.

Keywords: Criminal Investigation Unit, West Bandung Police Resort, data management of criminal cases, C#, .NET Framework, DBMS, SQL Server

1. Pendahuluan

Kemajuan teknologi informasi dewasa ini sangat mempengaruhi aktivitas manusia. Hal ini ditandai dengan penggunaan teknologi komputer di hampir semua bidang, di sini komputer berperan sebagai alat yang dapat digunakan untuk membantu agar pekerjaan dapat diselesaikan dengan mudah dan cepat. Salah satu bentuk penerapannya dalam kehidupan sehari-hari saat ini adalah sistem informasi yang memungkinkan organisasi melakukan penyimpanan, pengaksesan, dan pengolahan data dengan lebih mudah. Tidak seperti dulu, pengolahan data menjadi informasi masih dilakukan secara manual. Pengolahan data dengan metode seperti ini tidak dapat menjamin integritas dari data itu sendiri. [5] [6]

Satuan Reserse Kriminal (SatResKrim) merupakan salah satu unit terpenting dalam organisasi kepolisian. SatResKrim merupakan “ujung tombak” kepolisian dalam hal penanganan kasus-kasus kriminal. Kinerja kepolisian bahkan sering diukur oleh masyarakat melalui keberhasilan SatResKrim dalam menangani kasus-kasus kriminal. Salah satu tugas SatResKrim adalah mengelola data kasus kriminal yang jumlahnya sangat banyak. Pengelolaan data yang baik sangat dibutuhkan demi menjamin bahwa data-data tersebut dapat digunakan untuk menghasilkan informasi.

Hingga saat ini, beberapa unit di kepolisian – termasuk SatResKrim – masih melakukan pengelolaan data secara manual, meskipun data yang dimiliki dan harus

dikelola sangat banyak dan kompleks. Hal inilah yang membuat penyusun merancang dan mengimplementasikan sistem informasi untuk SatResKrim. Penyusun berharap bahwa hasil perancangan dan implementasi, dapat bermanfaat untuk membantu SatResKrim dalam pelaksanaan tugasnya dimasa mendatang.

2. Rumusan Masalah dan Tujuan Pembahasan

Tujuan dari pembuatan perangkat lunak aplikasi pengelolaan data kasus perkara di Poresta Bandung Barat adalah:

- a. Menciptakan aplikasi yang mempermudah pengelolaan data anggota kepolisian di SatResKrim, data masyarakat di wilayah hukum Polresta Bandung Barat, data kasus-kasus perkara dan Berita Acara Pemeriksaan (BAP).
- b. Optimalisasi pencarian terhadap data kasus kriminal tertentu.
- c. Menciptakan perangkat lunak aplikasi untuk menampilkan datad dan rekapitulasi kasus kriminal di wilayah hukum Polresta Bandung Barat yang dibutuhkan dalam pembuatan laporan bulanan.

3. Landasan Teori dan Rancangan Perangkat Lunak

3.1 Flowchart

Flowchart adalah penyajian yang sistematis tentang proses dan logika dari kegiatan penanganan informasi, atau penggambaran secara grafik langkah-langkah dan urutan prosedur dari suatu program. *Flowchart* membantu analis dan *programmer* untuk memecahkan masalah ke dalam segmen-segmen yang lebih kecil dan membantu dalam menganalisis alternatif-alternatif lain selama pengembangan. [1]

3.2 Basis Data [7]

Basis data (*database*) adalah pengorganisasian sekumpulan data yang saling terkait sehingga memudahkan aktivitas untuk memperoleh informasi. Untuk mengelola basis data, diperlukan perangkat lunak yang disebut DBMS (*Database Management System*). DBMS adalah perangkat lunak sistem yang memungkinkan para pemakai membuat, memelihara, mengontrol, serta mengakses basis data dengan cara yang praktis dan efisien.

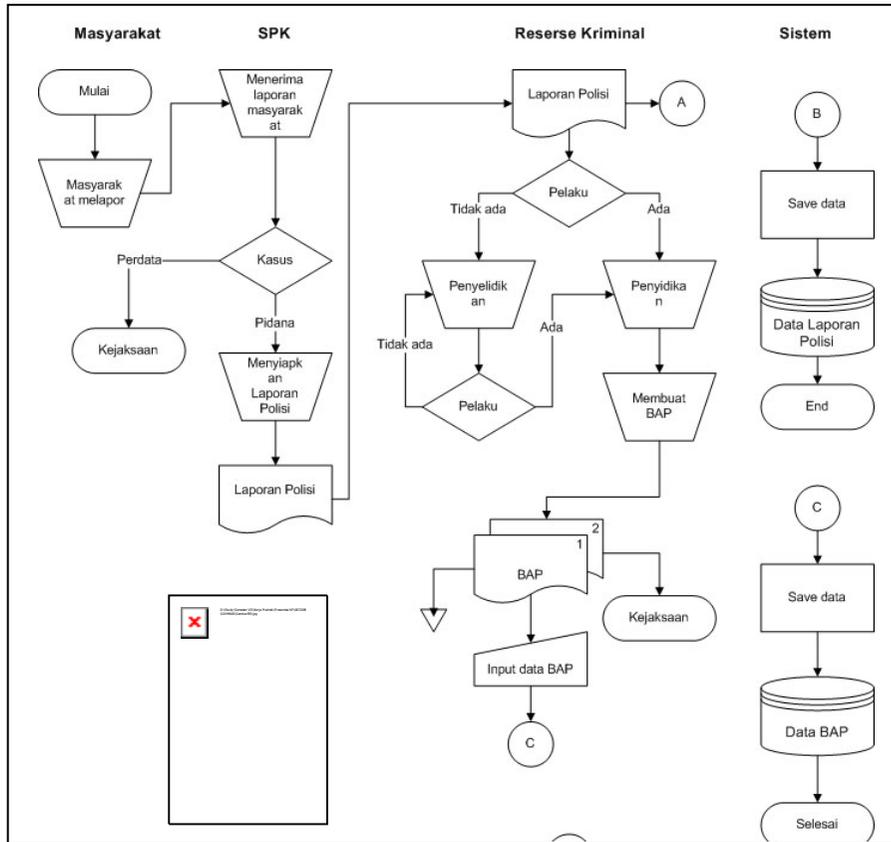
Entity Relationship Diagram (ERD) merupakan suatu metode pemodelan untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data (entitas) yang mempunyai hubungan (relasi) satu sama lain. ERD digunakan untuk memodelkan struktur data dan hubungan antar data.

SQL (*Structured Query Language*) adalah bahasa yang digunakan untuk mengakses basis data yang tergolong relasional. Standar SQL mula-mula didefinisikan oleh ISO (*International Standards Organization*) dan ANSI (*the American National Standards Institute*) yang dikenal dengan sebutan SQL86.

Secara umum, SQL terdiri dari dua bahasa, yaitu *Data Definition Language* (DDL) dan *Data Manipulation Language* (DML). DDL digunakan untuk mendefinisikan, mengubah, serta menghapus basis data dan objek-objek yang diperlukan untuk basis data, misalnya tabel, *view*, *user*, dan sebagainya. Sedangkan DML digunakan untuk memanipulasi data yang ada dalam tabel.

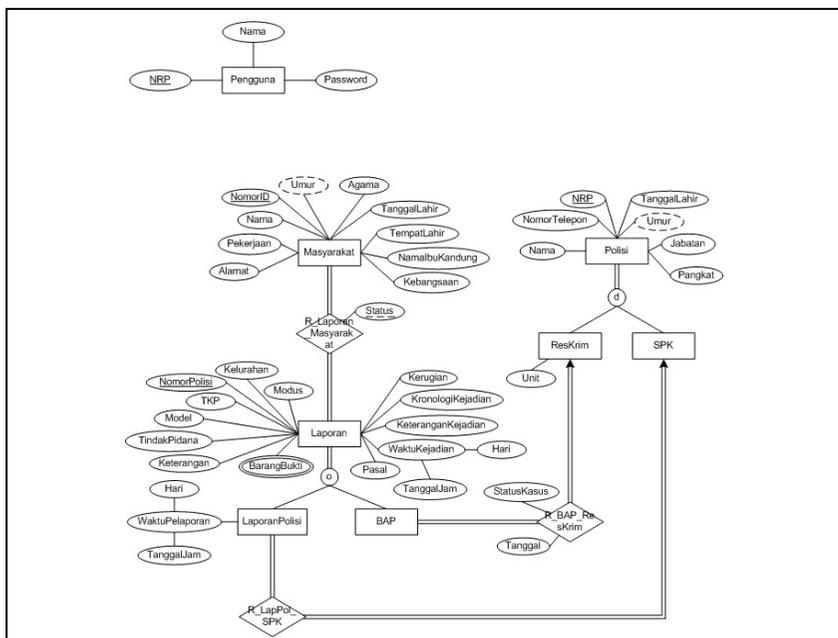
Microsoft SQL Server adalah sebuah sistem manajemen basis data relasional (RDBMS) produk Microsoft dan Sysbase. Umumnya SQL Server digunakan di

dunia bisnis yang memiliki basis data berskala kecil sampai dengan menengah, tetapi kemudian berkembang dengan digunakannya SQL Server pada basis data berskala besar. [4]



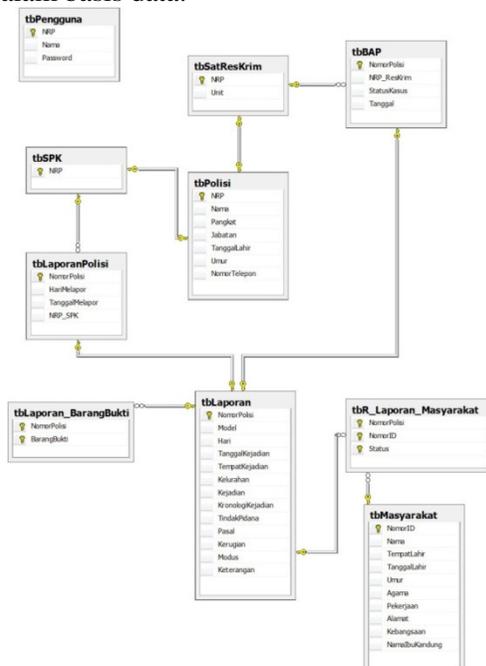
Gambar 1. Flowchart

Gambar 1 ini adalah *flowchart* dari sistem yang akan dibangun. *Flowchart* ini menggambarkan proses penanganan kasus kriminal oleh SatResKrim.



Gambar 2. ER Diagram

Gambar 2 adalah ER Diagram untuk menggambarkan desain database dari sistem yang akan dibangun. Sedangkan Gambar 3 adalah penerapan ER Diagram ke dalam Relasi Diagram di dalam mSQL Server yang menggambarkan hubungan antar entitas yang ada di dalam basis data.



Gambar 3. Penerapan ER Diagram ke dalam Relasi Diagram

3.3 Unified Modeling Language(UML) [2]

Unified Modeling Language (UML) adalah keluarga notasi grafis yang didukung oleh meta-model tunggal untuk membantu mendeskripsikan dan mendesain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman berorientasi objek (OO).

Use Case Diagram

Use case adalah teknik untuk merekam persyaratan fungsional sebuah sistem. Use case mendeskripsikan interaksi tipikal antara para pengguna sistem dengan sistem itu sendiri, dengan memberi sebuah narasi tentang bagaimana sistem tersebut digunakan.



Gambar 4. Use case diagram

Gambar 4 adalah gambaran diagram use case diagram dari perangkat lunak aplikasi yang akan dibangun. Perangkat lunak didesain dengan menggunakan UML karena perangkat lunak dikembangkan dengan pemrograman berorientasi objek C#.

3.4 C#

C# (dibaca: *C Sharp*) merupakan bahasa pemrograman berorientasi objek yang dikembangkan oleh Microsoft sebagai bagian dari .NET Framework. Bahasa pemrograman ini dibuat berdasarkan bahasa C++ yang telah dikombinasikan dengan aspek-aspek maupun fitur-fitur yang terdapat pada bahasa-bahasa pemrograman lainnya seperti Java, Delphi, Visual Basic, dan lain-lain dengan beberapa penyederhanaan. [3]

4. Pengembangan Perangkat Lunak

4.1. Form Utama

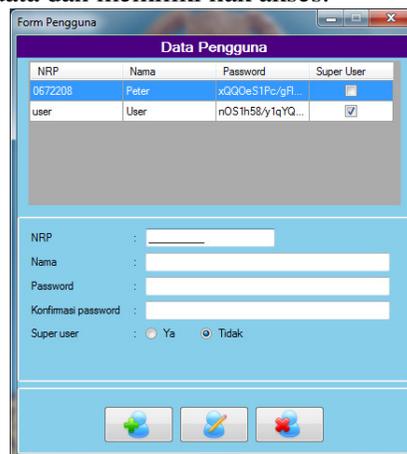
Ketika membuka aplikasi, tampilan yang pertama kali muncul adalah tampilan *form* utama seperti terlihat pada Gambar 4. Sebagian besar menu dalam keadaan tidak aktif dan tidak dapat diakses oleh pengguna. Untuk dapat mengaksesnya, pengguna harus *login* terlebih dahulu.



Gambar 4. Form utama

4.2. Form Pengguna

Gambar 5 adalah tampilan *form* pengguna. *Form* ini digunakan untuk manajemen data pengguna. Perangkat lunak hanya dapat digunakan oleh pengguna yang terdaftar di basis data dan memiliki hak akses.



Gambar 5. Form pengguna

4.3. Form Polisi

Gambar 6 adalah tampilan *form* polisi. *Form* ini dibagi menjadi dua tab, yaitu SatResKrim dan SPK. Tab SatResKrim untuk mengelola data anggota polisi di bagian Satuan Reserse Kriminal, sedangkan SPK untuk mengelola data anggota polisi di bagian Sentral Pelayanan Khusus.

NRP	Nama	Pangkat	Jabatan	Tanggal Lahir	Umur	Nomor Telepon
0672002	Wil Smith	Besku	Anggota	17/10/1987	22	
0672133	Glenn Griffin	Besku	Anggota	10/09/1988	21	

Gambar 6. Form Polisi

4.4. Form Masyarakat

Gambar 7 adalah tampilan *form* masyarakat. *Form* ini digunakan untuk mengelola data masyarakat di wilayah hukum Polresta Bandung Barat. Penambahan data anggota masyarakat hanya boleh dilakukan ketika pengguna membuat Laporan Polisi.

Nomor ID	Nama	Tempat Lahir	Tanggal Lahir	Umur	Agama	Pekerjaan
M-00001	Dicky Iqra N		10/09/1987	22	Islam	Mahasiswa
M-00002	Agus Riyanto		07/07/1975	34	Islam	Swasta
M-00003	Ricky Martin	Bandung	11/11/1992	17	Islam	Pelajar

Gambar 7. Form Masyarakat

4.5. Form Laporan Polisi

Gambar 8 adalah form laporan polisi. Form ini menampilkan semua data laporan yang pernah disampaikan masyarakat dan diproses oleh SatResKrim.

Nomor Polisi	Hari	Tanggal Kejadian	Tempat Kejadian	Tindak Pidana	Pasal	Keterangan	Nama Pelapor
734	Senin	04/05/2009 6:30	J Setabuchi Gg	CM-2	363 KUHP	Sesesa	Dicky Iqra N
735	Senin	04/05/2009 3:00	J Sedehana / J...	CM-2	363 KUHP	Ludik	Agus Riyar

Gambar 8. Form Laporan polisi

4.6. Form Pelapor/Korban/Tersangka/Saksi

Gambar 9 adalah tampilan form pelapor/korban/tersangka/saksi. Form ini menampilkan daftar pelapor/korban/tersangka/saksi dari kasus tertentu yang telah dipilih terlebih dahulu dari form laporan polisi. Melalui form ini, pengguna dapat menambahkan, mengedit, dan menghapus data pelapor, korban, tersangka, ataupun saksi dari kasus yang telah dipilih tersebut.

Nama	Tempat Lahir	Tanggal Lahir	Umur	Agama
Dicky Iqra N		10/09/1987	22	Islam

Gambar 9. Form Pelapor/korban/tersangka/saksi

4.7. Form Berita Acara Pemeriksaan (BAP)

Gambar 11 adalah *Form* BAP akan ditampilkan ketika aplikasi menyimpan data BAP yang telah ditambahkan oleh pengguna. Untuk BAP, pengguna hanya diperkenankan untuk mengedit dan menambah saja.

Gambar 10. Form berita acara pemeriksaan

4.8. Form Data Kriminal Per Periode

Gambar 12 adalah form yang menampilkan data kriminal yang ditangani oleh SatResKrim Polresta Bandung Barat. Data ditampilkan berdasarkan periode tertentu. Periode ditentukan oleh pengguna.

JENIS KEJADIAN	TGL / JAM KEJADIAN	TEMPAT KEJADIAN PERKARA	KELURAHAN	NO / TGL LP	IDENTITAS PELAPOR	IDENTITAS TERSANGKA	KERUGIAN	MODUS	KET
CM-2 363 KUHP	04-05-2009 06:30	Jl Setiabudhi Gg. Sarbini / Pemukiman		734A 04-05-2009	Dicky Igra N, 22, Mahasiswa, Kp Sinargalih Kab. Garut		0,00		Selesai
CM-2 363 KUHP	04-05-2009 06:30	Jl Setiabudhi Gg. Sarbini / Pemukiman		734A 04-05-2009	Brad Pitt, 18, Mahasiswa, Jl Surya Sumantri No 1		0,00		Selesai
CM-2 363 KUHP	04-05-2009 03:00	Jl Sederhana / Jalan Umum	Pasteur / Sukajadi	739B 04-05-2009	Agus Riyanto, 34, Swasta, Cibarengkok Bandung		8000000,00	Kunci palsu	Lidik
Penggelapan 372 KUHP	05-05-2009 10:00	Jl Surya Sumantri / Kantor	Sukawarna / Sukajadi	762B 05-05-2009	Ricky Martin, 18, Pelajar, Jl Surya Sumantri No. 2	Ricky Martin, 18, Pelajar, Jl Surya Sumantri No. 2	85000000,00	Sewa mobil	Lidik

Gambar 11. Form data kriminal per periode

4.9. Form Rekapitulasi Pola Lokasi

Gambar 13 adalah form untuk menampilkan rekapitulasi pola lokasi kejadian kriminal. Rekapitulasi didasarkan pada tempat kejadian perkara setiap kasus yang dikelompokkan berdasarkan jenis kejadiannya.

The screenshot shows a software window titled "Form Rekapitulasi Pola Lokasi". At the top, it displays "Bulan: Mei" and "Tahun: 2009". The main area is a grid with six panels, each representing a different location type. The top-left panel, "Pemukiman", shows one incident of type "CM-2 363 KUHP". The top-middle panel, "Tempat Umum", also shows one incident of the same type. The top-right panel, "Tempat Ramai", is empty. The bottom-left panel, "Sekolah", is empty. The bottom-middle panel, "Kantor", is empty. The bottom-right panel, "Lain-Lain", is empty. Each panel has a "Jumlah:" label at the bottom indicating the count of incidents.

Gambar 12 Form rekapitulasi pola lokasi

4.10. Form Rekapitulasi Jenis Kejadian

Gambar 14 adalah form untuk menampilkan rekapitulasi jenis kejadian yang dilaporkan oleh masyarakat. Aplikasi juga akan menampilkan jenis kejadian dengan jumlah kasus paling banyak, jumlah kasus yang paling sedikit, dan jumlah total kasus.

The screenshot shows a software window titled "Form Rekapitulasi Jenis Kejadian". It features a table with two columns: "Jenis Kejadian" and "Jumlah". The table lists "CM-2 363 KUHP" with a count of 3 and "Penggelapan 37..." with a count of 1. Below the table, there are summary statistics: "Jenis Kejadian maksimum: 3", "Jenis Kejadian minimum: 1", and "Jumlah: 4".

Gambar 13 Form rekapitulasi jenis kejadian

5. Kesimpulan dan Saran

5.1. Kesimpulan

Berdasarkan pengamatan selama perancangan, pengembangan, dan implementasi perangkat lunak, maka perangkat lunak aplikasi pengelolaan data kasus perkara Kepolisian Resor Kota Bandung Barat terbukti dapat membantu unit SatResKrim dalam mengelola data kasus perkara yang ditangani. Berikut ini adalah beberapa kemampuan dari perangkat lunak aplikasi.

- a. Menghindari redundansi data anggota SatResKrim yang menangani kasus kriminal, sebagai akibat dari pencatatan secara berulang-ulang setiap ada laporan dari masyarakat.
- b. Menyimpan data anggota masyarakat yang pernah terlibat dalam kasus kriminal tertentu, sehingga pihak kepolisian memiliki *history* data anggota masyarakat yang dapat digunakan sebagai referensi jika ada kasus-kasus lain yang berhubungan dengan anggota masyarakat tersebut dimasa mendatang.

- c. Mengurangi penggunaan kertas sebagai media pengarsipan data, sehingga ketika berkas-berkas kasus yang ditangani harus berpindah tempat sesuai dengan proses penyelidikan dan penyidikan, pengendalian terhadap data tetap dapat dilakukan secara maksimal.
- d. Sinkronisasi data yang lebih baik antara Laporan Polisi, dengan Berkas Acara Pemeriksaan (BAP) yang dibuat sesudahnya. Penggunaan DBMS dapat menjamin bahwa perubahan data pada Laporan Polisi juga akan memperbaharui BAP secara otomatis, begitu juga sebaliknya.
- e. Optimalisasi metode pencarian, di mana pengguna tidak perlu lagi mencari data kasus tertentu dari sekumpulan berkas yang sering membutuhkan waktu lebih lama, melainkan dengan menggunakan perangkat lunak yang memungkinkan pencarian menjadi jauh lebih cepat.
- f. *Report-report* yang dibuat secara otomatis oleh perangkat lunak. *Report-report* ini dapat digunakan untuk pengendalian tingkat kriminalitas di wilayah hukum Polresta Bandung Barat. Selain itu, *report-report* tersebut juga dapat digunakan untuk pengendalian terhadap kinerja pihak kepolisian dalam menangani kasus-kasus kriminal.

5.2. Saran

Berikut ini adalah beberapa hal yang dapat menjadi masukan untuk pengembangan sistem dimasa mendatang.

- a. Modul untuk pengelolaan terhadap data-data kasus kriminal dari tiap-tiap kepolisian sektor yang bernaung di bawah Polresta Bandung Barat.
- b. *Report-report* dari setiap unit di Polresta Bandung Barat untuk melengkapi Laporan Bulanan Unit SatResKrim dan Polresta Bandung Barat secara keseluruhan.
- c. Pengembangan perangkat lunak aplikasi agar mencakup semua unit di Polresta Bandung Barat. Sehingga sistem akses terhadap *resource* dan data dapat lebih terintegrasi dan membantu efektivitas kinerja anggota kepolisian di masing-masing unit.

6. Daftar Pustaka

- [1] Universitas Gunadarma Staffsite. 2009. *Flowchart*. [Tanggal akses 27 Agustus 2009]. [<http://febriani.staff.gunadarma.ac.id>].
- [2] Fowler, Martin. 2003. *UML Distilled (3rd edition)*. Yogyakarta: Penerbit Andi. (Original English edition 2004).
- [3] Hartanto, Budi. 2008. *Memahami Visual C# .NET Secara Mudah*. Yogyakarta: Penerbit Andi.
- [4] Imbar, Radiant Victor, Suteja, Bernard Renaldy. 2006. *Pemrograman Web Commerce dengan Oracle dan ASP*. Bandung: Penerbit Informatika.
- [5] Kadir, Abdul. 2002. *Pengenalan Sistem Informasi*. Yogyakarta: Penerbit Andi.
- [6] Kroenke, David M. 1992. *Management Information Systems*. Watsonville: Mitchell McGraw-Hill.
- [7] McFadden, Fred R., Hoffer, Jeffrey A., Prescott, Mary B. 1999. *Modern Database Management*. 5th Edition, Addison Wesley.

Konsep dan Perancangan *Code-Completion* untuk PHP

Tjatur Kandaga, Rinardi Budiadi Sarean

Jurusan S1 Teknik Informatika

Fakultas Teknologi Informasi, Universitas Kristen Maranatha

Jl. Prof. Drg. Suria Sumantri no. 65 Bandung 40164

email: tjatur.kandaga@itmaranatha.org, rinardi_1518_sarean@hotmail.com

Abstract

In this information age, the development of web application is very important. Almost all corporations have one or more websites to support their business. Even now, individuals have a personal webpages. PHP is one of programming language able to create web application, PHP stands for Hypertext Preprocessor. This language capable of building web application from a simple one to a relatively complex one. To be able to create a web application rapidly, programmer need good tools. Code completion is one of tools/features that programmers find very useful. A code completion tool can complete the code according to the keywords, and standard library of a particular programming language. This paper provide an example of building code completion tool in PHP. Besides keywords and standard libraries, the tool built can read another source code in the same project as it's source of words list as well. This code completion feature is part of a project that has been successfully built an integrated development environment for PHP language.

Keyword : Code Completion, PHP, Web

1. Pendahuluan

Bahasa pemrograman *PHP* adalah salah satu bahasa pemrograman untuk aplikasi web yang terkenal. Bahasa ini dieksekusi oleh server web setiap kali ada permintaan. Banyak sekali keuntungan menggunakan *PHP*, antara lain, penggunaan sintak pemrograman bahasa *C* yang sudah terkenal, tidak adanya tipe tipe data untuk variabel yang digunakan, banyaknya fungsi yang disediakan untuk ekspansi dengan menggunakan sistem lain. Selain itu juga *PHP* merupakan bahasa pemrograman yang sifatnya *open source*. Artinya bahasa ini boleh dipakai oleh siapapun juga, dan untuk apapun tanpa harus mengeluarkan biaya untuk lisensinya. Sifat *open source* juga, menjadikan bahasa ini boleh dimodifikasi untuk memenuhi kebutuhan pengguna.

Karena sifatnya yang dieksekusi oleh web server, maka tidak dibutuhkan *compiler* khusus untuk editor *PHP*. Banyak sekali *editor php* yang tersebar di dunia ini, mulai dari yang *open source* sampai yang profesional. Akan tetapi hampir semua editor yang ada, kurang akan fitur yang sangat penting untuk sebuah editor yang bagus yaitu *code completion* (melengkapi kode secara otomatis).

Tulisan ini akan membahas mengenai salah satu konsep untuk merancang sebuah *code completion* untuk *PHP* dan sekaligus membahas perancangannya yang direpresentasikan melalui diagram agar dapat dikembangkan menggunakan bahasa pemrograman yang ada.

2. PHP (Hypertext Preprocessor)

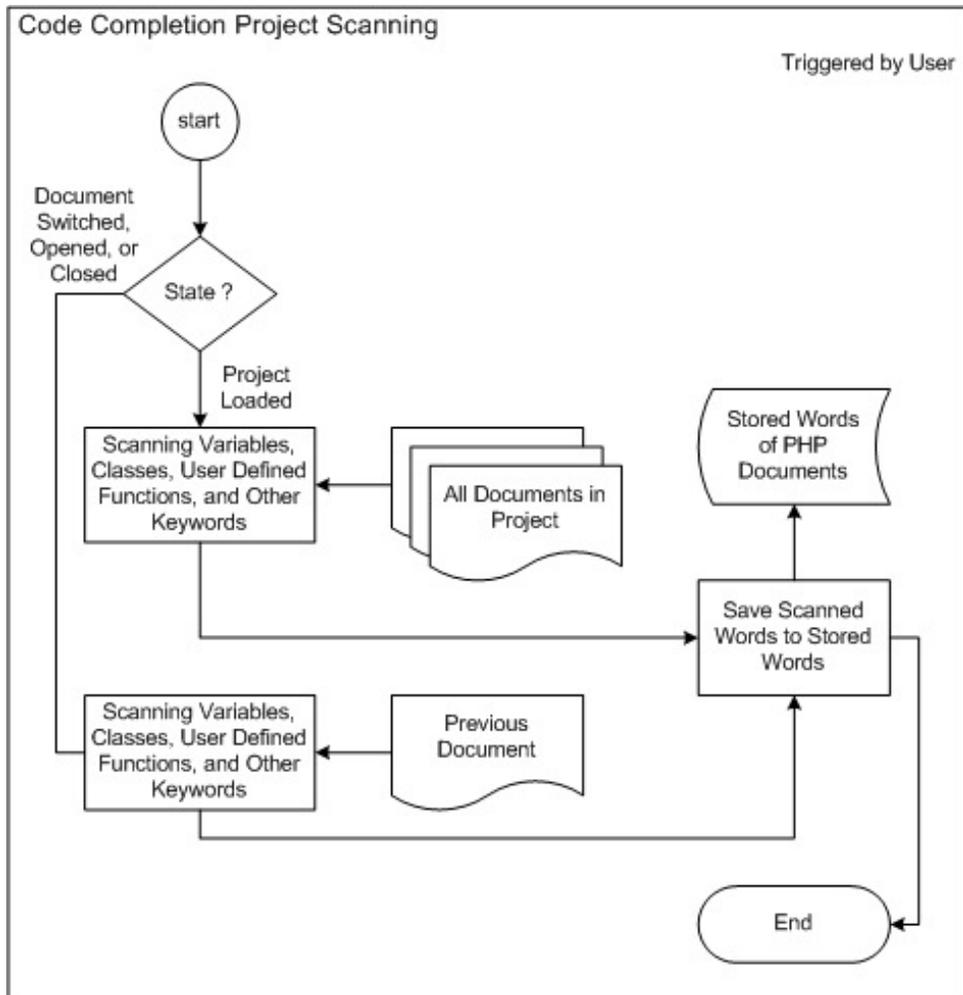
PHP, singkatan dari *Hypertext Preprocessor*, merupakan sebuah *scripting language* yang digunakan untuk menghasilkan halaman *web*. *PHP* bermula dari sebuah *scripting* yang dikembangkan oleh *Rasmus Lerdorf* pada tahun 1995¹. Dari sumber tersebut, dijelaskan lebih lanjut, bahwa kemudian dari bahasa tersebut dikembangkan menjadi *PHP (PHP 3.0)* oleh *Andi Gutmans* dan *Zeev Suraski* pada tahun 1997. Bentuk ini merupakan bentuk yang mirip seperti digunakan sekarang ini. Kemudian pada tahun 1998, Mereka berdua mulai mengembang *core* dari mesin *PHP* yang dikenal dengan nama *Zend Engine*. Setahun kemudian versi *4.0* dari *PHP* menggunakan mesin ini sebagai intinya. Kemudian setelah pengembangan yang cukup lama, pada akhirnya diperkenalkanlah *PHP 5.0* dengan *Zend Engine II*. Mesin ini mempunyai fitur tambahan seperti pemrograman berorientasi objek dan banyak fungsi serta fitur lainnya.

3. Konsep Code Completion yang Dibahas

Code completion merupakan salah satu kemampuan wajib yang harus dimiliki oleh *editor* bahasa pemrograman apapun. Bila diterjemahkan, *code completion* berarti melengkapi kode. Kode disini adalah kode bahasa pemrograman. Bila didefinisikan, *code completion* berarti sebuah fitur yang mampu memprediksi kata-kata (kode) dengan masukan satu atau beberapa karakter saja dari pengguna, dan memunculkan hasilnya sehingga pengguna dapat memilih mana yang akan dipakai, lalu memasukannya dalam dokumen. Fitur ini akan berjalan dengan otomatis menunggu masukan dari pengguna.

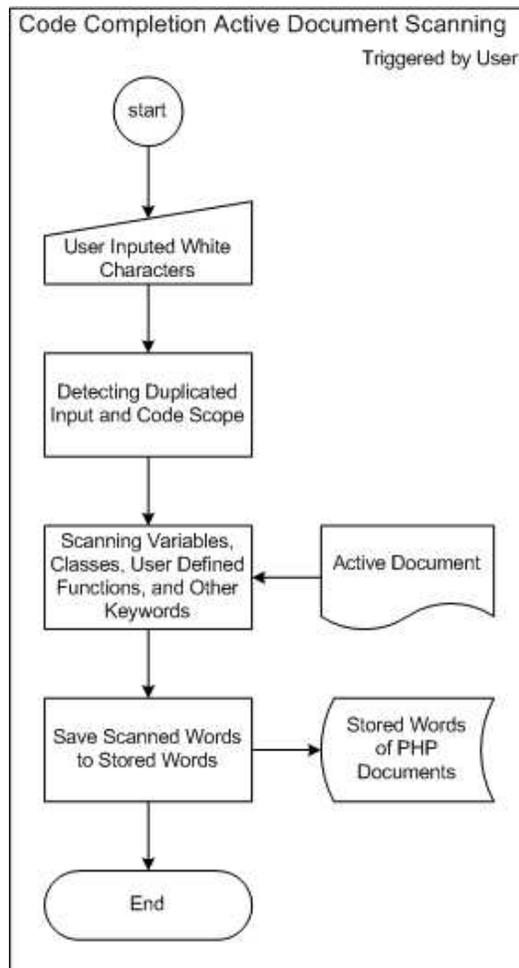
Aplikasi ini menerapkan *code completion* ini sebagai fitur yang pasti akan mempermudah pengguna dalam mengetikkan kode *PHP*. Dibawah ini beberapa diagram alir dan penjelasannya dari cara kerja *code completion* yang akan diterapkan dalam aplikasi.

¹ <http://id2.php.net/manual/en/history.php.php>



Gambar 1 Code Completion Project Scanning

Diagram diatas merupakan diagram yang menunjukkan kerja *code completion* dalam men-*scan* kata kata kunci dalam level proyek. Bisa dilihat, rutinitas ini akan berjalan bila ada proyek yang dibuka, atau pengguna pindah kerja dari satu dokumen ke dokumen lainnya. Berlaku pula bila pengguna menutup atau membuka dokumen lain yang berada dalam satu proyek. Setelah keadaan tersebut, rutin ini akan memulai memeriksa dokumen – dokumen dan memasukan semua variabel, kelas, fungsi, maupun kata-kata kunci lainnya. Setelah itu memasukan semuanya kedalam penyimpanan kata sementara.



Gambar 2 Code Completion Active Document Scanning

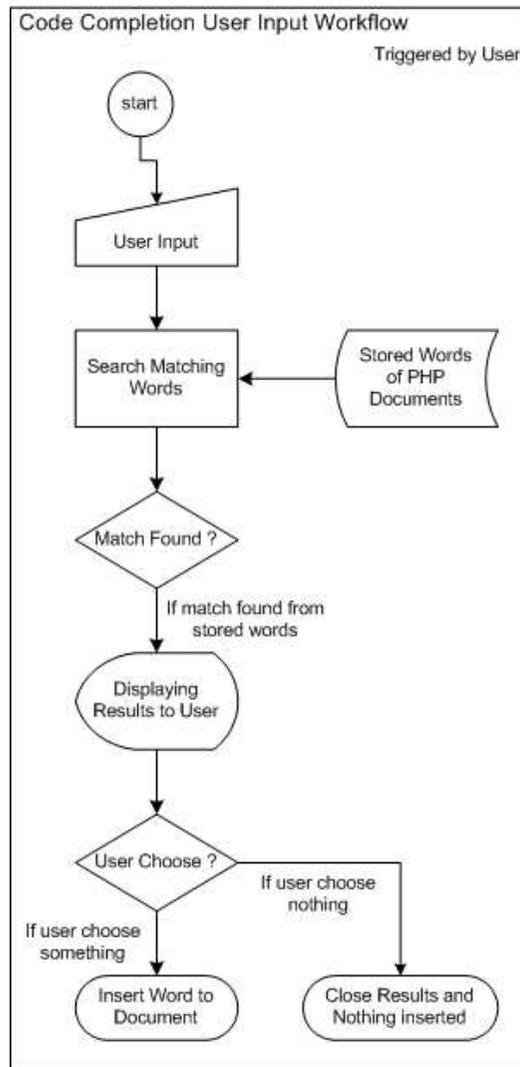
Diagram diatas merupakan diagram yang menunjukkan cara kerja *code completion* dalam men-*scan* dokumen yang sedang aktif. Bila dilihat, rutin ini sebagian memiliki kesamaan fungsi dengan diagram yang sebelumnya. Hanya saja, rutinitas dimulai ketika pengguna menekan salah satu *white characters*. Kemudian rutin akan mengecek apakah penekanan tombol tersebut merupakan duplikat dari yang sebelumnya atau bukan. Dan masih dalam satu fungsi yang sama, rutin juga akan mengecek keberadaan kursor, apakah dalam lingkup suatu kelas, fungsi atau global. Setelah itu baru fungsi-fungsi setelahnya sama dengan fungsi pada diagram sebelumnya.

Untuk men-*scan* dokumen, digunakan teknik *parsing* sederhana. *Parsing* merupakan teknik untuk menganalisa sekumpulan karakter atau kalimat dengan tujuan untuk mengasosiasikannya dengan suatu grup berdasarkan sintaks tertentu². *Parsing* untuk *code completion* ini menggunakan *regular expression* untuk

² <http://dictionary.reference.com/search?q=parse&x=0&y=0>

membantu mencari *token* atau kata kunci yang diinginkan. *Regular Expression* adalah sekumpulan karakter dengan aturan tertentu untuk mendeskripsikan pola pencarian³. Untuk aplikasi yang akan dikembangkan, hanya akan dicari kata-kata kunci yang mewakili kelas *PHP*, fungsi *PHP*, variabel *PHP*, dan konstanta *PHP*. *Parser* ini tidak mempedulikan urutan dalam eksekusi kode *PHP*, karena *parser* ini hanya akan mengambil informasi dari kode *PHP* tersebut. Oleh karena itu yang terpenting adalah, informasi kepemilikan (*ownership*) bukan urutan atau *valid*-nya sebuah kata-kunci.

Parser akan berjalan untuk men-*scan* isi dokumen pada saat pengguna mengetikkan kode *PHP*. Kemudian hasilnya akan disimpan.



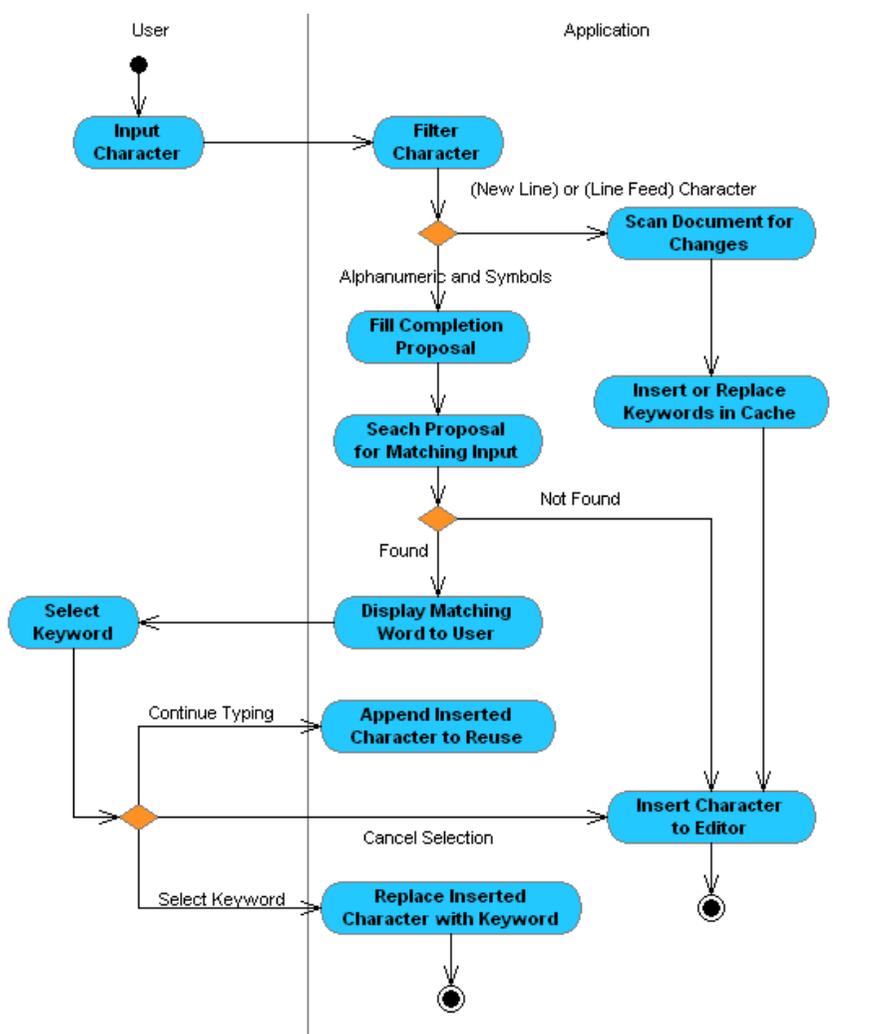
Gambar 3 Code Completion User Input Workflow

³ <http://regular-expression.info/>

Diagram ini menunjukkan cara kerja *code completion* bila pengguna memasukan karakter-karakter kedalam dokumen. Minimal karakter yang dibutuhkan adalah satu dan bukan *white characters*. Setelah itu, rutin akan mengambil karakter-karakter yang diketikan dan mencocokkannya dengan data penyimpanan kata sementara lalu mengambil semua kata yang mirip dengan yang diketikan oleh pengguna. Setelah itu, rutin ini akan menyuruh aplikasi untuk mengeluarkan kata-kata tadi untuk kemudian dipilih oleh pengguna. Bila pengguna memilih kata yang terdapat dalam hasil pencarian tadi, maka rutin akan langsung mengganti kata yang diketikan tadi dengan yang dipilih oleh pengguna. Bila pengguna tidak memilih apa-apa (dengan menekan tombol keluar atau *white characters*) maka rutin ini tidak akan melakukan perubahan pada dokumen.

4. Diagram Aktivitas

Dibawah ini merupakan diagram aktivitas untuk penggunaan konsep *code completion* ini kedalam aplikasi.



Gambar 4 Diagram Aktivitas

Aktivitas ini merupakan contoh penerapan konsep *code completion* diatas untuk aplikasi *editor PHP*. Kegiatan dimulai dari pengguna mengetikkan kode-kode program. Setiap pengguna memasukan satu karakter, aplikasi akan melakukan pengecekan terhadap karakter yang dimasukan tersebut. Bila karakter merupakan *white-space* (*tab*, *return*, *space*), maka aplikasi akan melakukan *scanning* pada dokumen yang aktif. Kemudian memasukan hasilnya kedalam penyimpanan sementara. Bila karakter yang dimasukan selain *white-space*, maka aplikasi langsung mencari kata kunci yang sesuai dengan karakter tersebut. Bila ditemukan, aplikasi akan memunculkan jendela kecil untuk memberikan pilihan kepada pengguna, apa yang akan diketik sebenarnya. Bila pengguna memilih, maka karakter yang dimasukan, akan diganti dengan pilihan pengguna. Bila pengguna membatalkan pilihan, maka jendela tersebut akan menutup. Dan yang terakhir, bila pengguna melanjutkan pengetikan, maka karakter-karakter tersebut akan dicari lagi oleh aplikasi untuk ditampilkan kembali yang cocok-nya.

5. Contoh Implementasi

Sesuai dengan perancangan yang dijelaskan didalam bab III pada landasan teori, untuk menopang fitur *code-completion*, dibutuhkan sebuah scanner/parser untuk mencari kata-kata kunci yang dibutuhkan oleh *completion proposal*. Parser yang dikembangkan menggunakan teknik parsing sederhana, yaitu hanya mencari dan menyimpan. Parser untuk aplikasi ini dikembangkan dengan bentuk sebuah *thread*. Sehingga kerja parser tidak mengganggu kerja aplikasi. Parser ini juga menggunakan *regular expression* untuk mencari kata-kata kunci yang diinginkan.

Berikut dibawah ini merupakan pola *regular expression* yang digunakan dalam parser aplikasi ini.

Tabel 1 Pola Regular Expression

▶ Kata Kunci	▶ Pola Regular Expression
▶ Require	▶ (?<=require)([_once\(\s]+")([\a-zA-Z0-9/_]+)
▶ Include	▶ (?<=include)([_once\(\s]+")([\a-zA-Z0-9/_]+)
▶ Konstanta	▶ (?<=define\()([\S]+)(?=))
▶ Variabel	▶ (?<=\\$)([_A-Za-z0-9]+)
▶ Fungsi	▶ (?<=function)([_A-Za-z0-9]+)(?=\()
▶ Kelas	▶ (?<=class)([_A-Za-z0-9]+)

Pola-Pola tersebut dipakai untuk metoda-metoda yang dipakai untuk mencari dan menyimpan kata-kata kunci sesuai dengan pola masing-masing. Berikut merupakan contoh kode program untuk fungsi yang merekap semua fungsi untuk mencari kata-kata kunci tersebut. Kode program dibuat menggunakan *Codegear Delphi 2007* untuk aplikasi *desktop* dan menggunakan *multithread*.

```
1 function TDocumentParseParserThread.ParsePHPDocument: TPHPDocument;
```

```
2 var I,C,SO,EO : Integer;
3   S : String;
4   L : TScopeStorageItems;
5 begin
6   Result := TPHPDocument.Create;
7   TraceDirectives(Result.Directives);
8   {Parse PHP Section from Document.}
9   TracePHPSection(Result.Sections);
10  if (Result.Sections.Count > 0) then
11    for I := 0 to Result.Sections.Count - 1 do
12      begin
13        {First, scan for classes within every section.}
14        S := Result.Sections.Items[I].Info.Content;
15        SO := Result.Sections.Items[I].Info.StartOffset;
16        TraceClass(S,SO,Result.Classes);
17        {Second, scan for functions within every section, outside the class(es) scope.}
18        if (Result.Classes.Count > 0) then
19          begin{if there're function(s) within this class...}
20            for C := 0 to Result.Classes.Count do
21              begin
22                {Define offset based on function offset and class offset.}
23                if (C = 0) then SO := Result.Sections.Items[I].Info.StartOffset
24                  else SO := Result.Classes.Items[C-1].Info.EndPosition;
25                if (C = Result.Classes.Count) then EO :=
26                  Result.Sections.Items[I].Info.EndOffset
27                  else EO := Result.Classes.Items[C].Info.NameOffset;
28                {Copy string from start offset to end offset as scope.}
29                S := Copy(FLines.Text,SO + 1,EO - SO);
30                {Trace Function within the scope.}
31                TraceFunction(S,SO,"",Result.Functions);
32                end;{end for C := 0...}
33              end else begin
34                {if there's no classes within this section, use previous scope, then scan for
35                functions.}
36                TraceFunction(S,SO,"",Result.Functions);
37                end; { end if (Result.Classes... else... }
38                {Third, scan for variables within every section, outside the scope of class(es)
39                and function(s).}
40                S := Result.Sections.Items[I].Info.Content;
41                SO := Result.Sections.Items[I].Info.StartOffset;
42                L := Scoper(Result);
43                if (L.Count > 0) then
44                  begin
45                    for C := 0 to L.Count do
46                      begin
47                        {Define offset based on function offset and class offset.}
48                        if (C = 0) then SO := Result.Sections.Items[I].Info.StartOffset
49                          else SO := L.Items[C-1].EndOffset;
50                        if (C = L.Count) then EO := Result.Sections.Items[I].Info.EndOffset
51                          else EO := L.Items[C].StartOffset;
52                        {Copy string from start offset to end offset as scope.}
53                        S := Copy(FLines.Text,SO + 1,EO - SO);
54                        {Trace Variables within the scope.}
55                        TraceVariable(S,SO,otNoOwner,"",Result.Variables);
56                        end;{end for C := 0...}
57                      end else begin
58                        {if there's no classes or function within this section, use previous scope, then
59                        scan for variables.}
60                        TraceVariable(S,SO,otNoOwner,"",Result.Variables);
```

```
57     end;
58     L.Free;
59     {Fourth, scan for any inclusions within this section.Both include and require.}
60     S := Result.Sections.Items[[]].Info.Content;
61     SO := Result.Sections.Items[[]].Info.StartOffset;
62     EnumeratePHPInclusion(S,SO,Result.Inclusions);
63     {Fifth, scan for any defined constants within this section.}
64     EnumeratePHPConstant(S,SO,Result.Constants);
65     end; {end for I := 0...}
66 end;
```

Penjelasan dari cuplikan kode diatas adalah sebagai berikut.

- Baris 6 : Membuat objek untuk hasil *parse*.
- Baris 7 : Mengambil kata penunjuk didalam kode.
- Baris 9 : Mengambil segmen *PHP* dalam dokumen.
- Baris 11 : Memproses segmen yang terambil satu per satu.
- Baris 16 : Ambil kelas *PHP*.
- Baris 18-35 : Proses mengambil fungsi *PHP* diluar kelas.
- Baris 37-57 : Proses mengambil variabel *PHP* diluar kelas.
- Baris 62 : Mengambil *include* dan *require* dari segmen.
- Baris 64 : Mengambil konstanta *PHP* dari segmen.

6. Simpulan

Berdasarkan project yang sudah dilakukan dapat diambil beberapa simpulan, diantaranya:

1. Fitur *code completion* dapat membantu *programmer* dalam meningkatkan kenyamanan dan produktivitas ketika membuat program.
2. Dapat dibuat *code completion tool* yang mengikutsertakan kode program yang sebelumnya telah dibuat oleh pemrogram sebagai sumber kata yang dikenali, selain *keywords* dan pustaka standar dari bahasa program tujuan.
3. *Parsing* dan *regular expression* merupakan elemen yang sangat penting dalam sebuah *code completion tool*.

7. Daftar Pustaka

1. PHP: History of PHP – Manual. Retrieved July 6th 2009 from <http://id2.php.net/manual/en/history.php.php>
2. Parse Definition | Definition of Parse at Dictionary.com. Retrieved July 6th 2009 from <http://dictionary.reference.com/search?q=parse&x=0&y=0>
3. Regular-Expressions.info - Regex Tutorial, Examples and Reference - Regexp Patterns. Retrieved July 6th 2009 from <http://regular-expression.info/>

PEDOMAN PENULISAN ARTIKEL

Jurnal Informatika UKM menerima karya tulis:

1. Dalam bentuk hasil penelitian, tinjauan pustaka, dan laporan kasus dalam bidang ilmu yang berhubungan dengan Teknologi Informasi.
2. Belum pernah dipublikasikan dalam jurnal ilmiah manapun. Bila pernah dipresentasikan, sertakan keterangan acara, tempat, dan tanggalnya.
3. Ditulis dalam bahasa Indonesia atau bahasa Inggris.

Sistematika yang ditetapkan untuk tiap kategori karya-karya tulis tersebut adalah:

1. Hasil Penelitian

Hasil penelitian terdiri atas judul, penulis, abstrak berbahasa Indonesia untuk artikel berbahasa Inggris atau abstrak berbahasa Inggris untuk artikel berbahasa Indonesia (masing-masing terdiri atas 150-200 kata), disertai kata kuncinya. Pendahuluan, metoda, pembahasan, simpulan, dan saran, serta daftar pustaka (merujuk sekurang-kurangnya 3 [tiga] pustaka terbaru).

2. Tinjauan Pustaka

Naskah hasil studi literatur terdiri atas judul dan penulis. Pendahuluan (disertai pokok-pokok ide kemajuan pengetahuan terakhir sehubungan dengan masalah yang digali). Permasalahan mencakup rangkuman sistematik dari berbagai narasumber. Pembahasan memuat ulasan dan sintesis ide. Simpulan dan saran disajikan sebelum daftar pustaka. Tinjauan pustaka merujuk pada sekurang-kurangnya 3 (tiga) sumber pustaka terbaru.

3. Laporan Kasus

Naskah laporan kasus terdiri atas judul, abstrak berbahasa Indonesia untuk teks artikel berbahasa Inggris atau abstrak berbahasa Inggris untuk teks artikel berbahasa Indonesia (50-100 kata) disertai kata kuncinya, pendahuluan (disertai karakteristik lokasi, gambaran umum budaya yang relevan, dll), masalah, pembahasan, dan resume atau simpulan.

Tatacara penulisan naskah:

- a. Artikel diketik rapi dengan menggunakan Microsoft Word, dikirim dalam disket beserta print-outnya. Jenis huruf yang digunakan adalah *times new roman* ukuran 11. Panjang artikel berkisar 10 – 11 halaman, ukuran kertas B5, satu spasi. Judul ditulis di tengah-tengah ukuran 13.
- b. Artikel ditulis dalam bahasa Indonesia atau bahasa Inggris yang baik dan benar. Abstrak ditulis miring (*italic*) ukuran huruf 10. Panjang gambar dan foto harus dalam bentuk jadi dengan resolusi gambar yang memadai (jelas dan nyaman dilihat), serta dalam ukuran yang sesuai dengan format jurnal ilmiah, dan dalam bentuk disket.
- c. Daftar pustaka ditulis alfabetis sesuai dengan nama akhir (tanpa gelar akademik) baik penulis asing maupun penulis Indonesia, berisi maksimal 15 (lima belas) penulis yang dirujuk, font ukuran 10.

Contoh format daftar pustaka:

Buku:

[Ald04] Aldrich, C. (2004). *Simulations and the future of learning: An Innovative (and Perhaps Revolutionary) Approach to e-Learning*. Pfeiffer, San Francisco: USA.

Jurnal:

[Bac05] Backus, G. & Amlin, J. (2005). *Using gaming simulation to understand deregulation dynamics*. *Simulation & Gaming* Thousand Oaks: Mar 2005. Vol. 36, Iss. 1, p. 45-57.

Internet:

[Hol02] Holub, Allen I. (2002). *Programming Java Threads in The Real World*. *Java World*
Available: <http://www.javaworld.com/javaworld/jw-09-1998/jw-09-threads.html>. Accessed: 01/07/2002.

- d. Penulis mencantumkan institusi asal dan alamat korespondensi lengkap. Penulis yang artikelnya dimuat akan mendapat imbalan / honor peserta beserta 2 eksemplar jurnal ilmiah.
- e. Kepastian pemuatan atau penolakan akan diberitahukan secara tertulis. Artikel yang tidak dimuat akan dikembalikan. Redaksi jurnal ilmiah berhak melakukan penyuntingan.

Keterangan lain yang diperlukan dapat diperoleh dengan menghubungi redaksi melalui:

Sekretariat Jurnal Informatika Universitas Kristen Maranatha

Jurusan Teknik Informatika, Fakultas Teknologi Informasi

Jl. Prof. Drg. Suria Sumantri, MPH, No. 65 Bandung. 40164

Telp (022) 70753665

Fax (022) 2005915

Email: jurnal.informatika@itmaranatha.org atau teddy.mz@maranatha.edu

Homepage: <http://www.itmaranatha.org/jurnal.informatika>

FORMULIR BERLANGGANAN

1. Nama :

2. Alamat :

3. Telepon/HP :

4. Email :

Menyatakan untuk berlangganan Jurnal Informatika mulai Edisi :
..... dan bersedia membayar biaya cetak dan ongkos kirim
sebesar Rp. 50.000 (/eks).

Biaya akan dikirim ke rek. **282-153-3939, BCA Maranatha Bandung** a/n
Radiant Victor Imbar

Pemohon :

(.....)

- Formulir Berlangganan dan Bukti Transfer dapat dikirim lewat pos/faks/email ke :
 - Universitas Kristen Maranatha
 - Fakultas Teknologi Informasi (FIT)
 - Alamat : Jl. Suria Sumantri 65 Bandung – 40164
 - Faks : +62-022- 2005915
 - Email : jurnal.informatika@itmaranatha.org atau teddy.mz@maranatha.edu