

PENGEMBANGAN APLIKASI ANALISIS VENDOR PERFORMANCE DI PT. SUZUKI INDOMOBIL MOTOR

Ifan Junaedi

Teknik Informatika- STMIK Jayakarta

ifan@stmik.jayakarta.ac.id

ABSTRAK

Dalam perusahaan *manufacturing*, rantai pasok adalah salah satu faktor penentu keberhasilan perusahaan. Pemilihan bahan baku yang berkualitas akan menjamin kualitas produk yang dihasilkan. Selain itu, ketersediaan bahan baku yang diperlukan dalam proses produksi menjadi hal yang mutlak untuk dipenuhi. Jika *supplier* terlambat mengirimkan bahan baku yang diperlukan, maka proses produksi bisa terhenti dan mengakibatkan tidak tercapainya target produksi. Target yang tidak tercapai tersebut harus dipenuhi dengan cara *overtime* atau lembur. Perusahaan harus mengeluarkan biaya produksi tambahan demi mencapai jumlah produksi yang dibutuhkan. Hal tersebut merupakan kerugian bagi perusahaan.

Metode penelitian yang digunakan dalam membangun Aplikasi Analisis Vendor Performance ini adalah berdasarkan konsep analisa dan perancangan sistem informasi, dimana tahapan awal adalah melakukan investigasi terhadap masalah dan prosedur yang berjalan, termasuk analisa masalah. Tahapan selanjutnya adalah perancangan sistem yaitu merancang sistem yang diusulkan dengan menggunakan diagram-diagram pendukung sampai dengan implementasi.

Hasil yang diharapkan dalam penelitian ini adalah merancang aplikasi untuk sistem analisis vendor performance untuk memperoleh gambaran mengenai audit vendor performance disebuah perusahaan *manufacturing* dan memberikan gambaran mengenai konsep pemeliharaan kualitas produk atau *quality assurance* dari sisi pemilihan vendor, studi kasus pada PT. Suzuki Indomobil motor, selain itu penggunaan database memungkinkan untuk melakukan penyimpanan data dalam jumlah yang banyak sehingga akan lebih mudah, efektif dan efisien.

Kata Kunci : *sistem informasi, aplikasi analisis vendor, analisa dan perancangan*

I. PENDAHULUAN

A. Latar Bclakang Masalah

Dalam perusahaan *manufacturing*, rantai pasok adalah salah satu faktor penentu keberhasilan perusahaan. Pemilihan bahan baku yang berkualitas akan menjamin kualitas produk yang dihasilkan. Selain itu, ketersediaan bahan baku yang diperlukan dalam proses produksi menjadi hal yang mutlak untuk dipenuhi. Jika *supplier* terlambat mengirimkan bahan baku yang diperlukan, maka proses produksi bisa terhenti dan mengakibatkan tidak tercapainya target produksi. Target yang tidak tercapai tersebut harus dipenuhi dengan cara *overtime* atau lembur. Perusahaan harus mengeluarkan biaya produksi tambahan demi mencapai jumlah produksi yang dibutuhkan. Hal tersebut merupakan kerugian bagi perusahaan.

Untuk mendapatkan bahan baku berkualitas yang sesuai dengan kebutuhan produksi serta tidak terjadinya keterlambatan

dalam pengiriman bahan baku tersebut, maka pemilihan *supplier* menjadi satu hal yang harus dilakukan. Perusahaan akan menerima penawaran penyediaan bahan baku produksi dari banyak *supplier* atau *vendor*. Perusahaan harus memiliki standar baku pemilihan *supplier* sebelum memutuskan untuk menerima penawaran tersebut. Setelah menjadi mitra tetap, perusahaan secara berkala melakukan penilaian terhadap kinerja *supplier*. Proses penilaian tersebut disebut dengan audit *vendor performance*. Hasil dari audit tersebut akan menjadi pertimbangan bagian *procurement* dalam melakukan pembelian bahan baku produksi. Tujuan akhir dari pemilihan *vendor* yang sesuai adalah untuk menjaga kualitas produk yang dihasilkan dengan selektif memilih kualitas bahan baku yang akan digunakan.

Pada PT Suzuki Indomobil Motor, proses audit tersebut masih dilakukan secara manual. Jumlah *supplier* yang diaudit setiap

bulannya menjadikan proses pelaporan hasil *audit* tersebut berlangsung lama.

B. Identifikasi Masalah

Masalah yang dapat diidentifikasi dalam pembangunan aplikasi analisis *vendor performance* adalah sebagai berikut:

1. Bagaimana membuat aplikasi penilaian *performance vendor* untuk *supplier* baru dan *supplier* yang sudah menjadi mitra tetap, serta mengintegrasikannya dengan sistem yang sudah ada.
2. Bagaimana mengolah data hasil audit kedalam laporan yang bisa langsung diakses setelah hasil audit diinput. Laporan yang dibuat merupakan laporan yang bersifat *analytic* hingga mampu membuat peringkat *vendor* terbaik setiap periodenya.

C. Perumusan Masalah

Dari identifikasi masalah diatas, penulis merumuskan permasalahan yang akan dibahas dalam tugas akhir ini adalah bagaimana pembangunan aplikasi analisis *vendor performance* pada PT. Suzuki Indomobil Motor?

D. Pembatasan Masalah

Pembatasan masalah pada penelitian sebagai berikut:

1. Pada PT. Suzuki Indomobil Motor, aplikasi yang digunakan adalah Oracle Application E-Business suite release 10.5.9 dengan *back-end database* oracle 10g. Seluruh *custom application* dibuat dengan menggunakan oracle form dan report 6i. Karena kebutuhan spesifikasi *hardware* dan *software legacy sistem* yang ada sangat tinggi, maka untuk pembuatan aplikasi analisis *vendor performance*, perlu dilakukan penyesuaian karena tidak memungkinkannya proses implementasi dengan memakai aplikasi yang sama dengan *legacy system* yang ada. Oleh karena itu, pada penelitian ini, *tools* yang digunakan untuk pembangunan aplikasi adalah Oracle Form developer 10g. Untuk pembuatan laporan, menggunakan Oracle Reports developer 10g. Sedangkan

database yang digunakan adalah Oracle Database 10g Express Edition yang bersifat *free* dan membutuhkan spesifikasi *hardware* yang lebih kecil, namun fitur-fitur yang tersedia hampir sama dengan *database* Oracle 10g. Sedangkan *database editor* menggunakan *tools sql-developer*.

2. Dalam tahap analisis dan desain, pemodelan sistem menggunakan metoda *data flow* atau metoda terstruktur. *Tools* yang digunakan adalah power designer.
3. Dalam tugas akhir ini tidak dibahas bagaimana mengintegrasikan aplikasi yang dibuat dengan Oracle application E-Business Suite.

E. Tujuan Penelitian Dan Manfaat Penelitian

Tujuan pengembangan aplikasi analisis *vendor performance* tersebut adalah untuk memudahkan proses audit *vendor* serta pembuatan laporan hasil audit pada FT Suzuki Indomobil Motor. Sedangkan manfaat dari pembangunan aplikasi tersebut adalah

1. Memperoleh gambaran mengenai audit *vendor performance* di sebuah perusahaan *manufacturing* dengan studi kasus pada PT. Suzuki Indomobil Motor.
2. Memberikan gambaran kepada pembaca mengenai konsep pemeliharaan kualitas produk atau *quality assurance* dari sisi pemilihan *vendor*, sehingga bisa dikembangkan mcnjadi bahan penelitian lain yang lebih mendalam.

F. Metode Penelitian

Dalam proses pengumpulan data, penulis menggunakan beberapa metode penelitian yang merupakan pengamatan objek secara langsung. Adapun metode-metode yang digunakan yaitu :

1. Pengamatan (*Observation*)
Penelitian secara langsung pada perusahaan dan terlibat langsung pada proses implementasi aplikasi analisis *vendor performance* di PT Suzuki Indomobil Motor.
2. Wawancara (*Interview*)
Mengajukan pertanyaan-pertanyaan yang mengarah pada ljuan penulisan, pada pihak yang terkait yang memiliki informasi yang dibutuhkan dalam penyusunan skripsi ini.

3. Studi *literatur* (Kepustakaan)
Mempelajari buku-buku, situs internet,
serta forum diskusi internet.

II. LANDASAN TEORI

A. Analisis Vendor Performance

1. Pengertian Analisis

Definisi analisis menurut Kamus Bahasa Indonesia adalah penguraian suatu pokok atas berbagai bagiannya dan penelaahan bagian itu sendiri serta hubungan antar bagian untuk memperoleh pengertian yang tepat dan pemahaman arti keseluruhan.

2. Vendor Performance

Analisis *vendor performance* adalah aktifitas yang dilakukan untuk memantau dan meningkatkan kualitas produk berdasarkan pemilihan *vendor*. Analisis *vendor performance* bisa dilakukan dari sisi produk dan *service* yang disediakan oleh *vendor*. Dari perspektif internal audit, kegiatan ini disebut audit *vendor performance*. Audit *vendor performance* bisa dilakukan dengan dua cara, yaitu dengan memilih produk atau *service* yang dibutuhkan, atau dengan memilih *vendor* yang terlebih dahulu. Poin-poin yang akan diaudit disesuaikan dengan apa yang dibutuhkan oleh perusahaan.

Tahapan yang lazim dilakukan pada audit *vendor performance* adalah:

1. Memilih *vendor* yang memiliki poin bagus sesuai dengan kebutuhan bahan barang atau jasa yang diperlukan dari *vendor* tersebut
2. Lakukan konfirmasi jika terdapat poin-poin yang dianggap buruk pada *vendor* yang terpilih
3. Temukan dan diskusikan solusi dari poin-poin *minus* tersebut
4. Hilangkan *vendor* yang memiliki nilai buruk setelah pengecekan selesai.

B. Sistem Informasi

1. Pengertian Pembangunan

Definisi Pembangunan menurut Kamus Bahasa Indonesia Edisi Ketiga (2002:103) adalah Proses, cara, perbuatan

membangun.

2. Pengertian Sistem

Sistem adalah kumpulan komponen atau subsistem yang dirancang untuk mencapai suatu tujuan (Tata Sutabri, 2005:9). Sedangkan menurut pendapat Me Leod, Jr (2004:9) sistem adalah sekelompok elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan.

Menurut Kamus Besar Bahasa Indonesia Edisi Ketiga (2002:1077) sistem adalah perangkat unsur yang saling berkaitan sehingga membentuk suatu totalitas.

3. Pengertian Sistem Informasi

Tata Sutabri (2005: 42) memberikan definisi mengenai sistem informasi adalah sebagai berikut:

Sistem Informasi adalah Suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan.

4. Analisis Sistem Informasi

Menurut Jeffrey L. Whitten, et.al diterjemahkan dalam bahasa Indonesia oleh Tim Penerjemah Andi (2004: 176) analisis sistem adalah sebuah teknik pemecahan masalah yang menguraikan sebuah sistem menjadi bagian-bagian komponen dengan tujuan mempelajari seberapa bagus bagian-bagian komponen tersebut bekerja dan berinteraksi untuk meraih tujuan mereka.

5. Perancangan Sistem Informasi

Definisi perancangan sistem informasi menurut Jeffrey L. Whitten, et.al yang diterjemahkan oleh Tim Penerjemah Andi (2004: 176) adalah sebagai berikut:

Perancangan sistem adalah sebuah teknik pemecahan masalah yang saling melengkapi (dengan analisis sistem) yang merangkai kembali bagian-bagian komponen menjadi sebuah sistem yang lengkap. Hal ini melibatkan

penambahan, penghapusan dan perubahan bagian-bagian relatif pada sistem aslinya.

6. Implementasi Sistem Informasi

Jeffrey L. Whitlen, cl.al (1994:726) mengemukakan bahwa :

System Implementation is the contruction of the new system and delivery of that system into production (meaning day to day operation). Unfortunately, systems development is a common synonym..

Penulis dapat menjelaskan bahwa implementasi sistem atau implementasi sistem informasi adalah kontruksi dari sebuah sistem yang baru dan mengantarkan sistem informasi menjadi sebuah produk. Implementasi sistem disebut juga pengembangan sistem.

7. Decision Support System

Decission Support System adalah sebuah sistem yang memberikan dukungan kepada seorang manajer, atau kepada sekelompok manajer yang relatif kecil yang bekerja sebagai tim pemecah masalah, dalam memecahkan masalah semi terstruktur dengan memberikan informasi atau saran mengenai keputusan tertentu. Informasi tersebut diberikan oleh laporan berkala, laporan khusus, maupun *output* dari model matematis. Model tersebut juga mempunyai kemampuan untuk memberikan saran dalam tingkat yang bervariasi.

Tahap-lahap dalam pengambilan keputusan antara lain adalah :

1. kegiatan intelijen
2. kegiatan merancang
3. kegiatan memilih dan menelaah.

Kegiatan intelijen ini merupakan kegiatan mengamati lingkungan untuk mengetahui kondisi-kondisi yang perlu diperbaiki. Kegiatan ini merupakan tahapan dalam perkembangan cara berfikir. Untuk melakukan kegiatan intelijen ini diperlukan sebuah sistem informasi. dimana informasi yang diperlukan ini didapatkan dari kondisi internal maupun eksternal sehingga seorang manajer dapat mengambil sebuah keputusan dengan tepat.

Kegiatan merancang merupakan sebuah kegiatan untuk menemukan, mengembangkan dan menganalisis berbagai alternatif tindakan yang mungkin untuk dilakukan. Tahap perancangan ini meliputi pengembangan dan mengevaluasi serangkaian kegiatan alternatif. Pertimbangan-pertimbangan utama telah diperkenalkan oleh Simon untuk melakukan tahapan ini, apakah situasi keputusan ini terprogram atau tidak. Sedangkan kegiatan memilih dan menelaah ini digunakan untuk memilih satu rangkaian tindakan tertentu dari beberapa yang tersedia dan melakukan penilaian terhadap tindakan yang telah dipilih.

8. Perangkat Pemodelan Sistem

a. Data Flow Diagram

Data Flow Diagram atau disingkat DFD merupakan model proses yang digunakan untuk menggambarkan aliran data melalui sebuah sistem dan tugas atau pengolahan yang dilakukan oleh sistem. Menurut Roger S. Pressman yang diterjemahkan oleh L.N. Harnaningrum (2002:353) DFD adalah sebuah teknik grails yang menggambarkan aliran informasi dan transformasi yang diciptakan pada saat data bergerak dari input menjadi output. Fungsi dari DFD adalah:

1. Memberikan indikasi bagaimana data ditransformasikan pada saat data bergerak melalui sistem
 2. Untuk menggambarkan fungsi-fungsi yang mentransformasi aliran data.
- Masih menurut Roger S. Pressman yang diterjemahkan oleh L.N. Harnaningrum (2002:353)

:

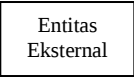
DFD dapat digunakan untuk menyajikan sebuah sistem atau perangkat lunak pada setiap abstraksi. DFD tingkat 0 disebut juga *context diagram*,

menggambarkan seluruh elemen sistem sebagai proses tunggal dengan data input dan output yang digambarkan dengan anak panah. Proses detail dari sebuah sistem dapat diturunkan menjadi *bubble* proses yang lebih rinci di tingkat DFD yang lebih detail. Sebagai contoh DFD tingkat 0 dapat berisi lima atau enam *bubble* proses dengan anak panah yang saling berhubungan pada DFD tingkat 1. Setiap proses yang direpresentasikan pada tingkat 1 merupakan subfungsi dari seluruh sistem yang digambarkan dalam *context diagram*.

Notasi dasar yang digunakan dalam DFD dijelaskan dalam tabel berikut:

Tabel 2.1 Tabel notasi dasar dalam *data flow diagram*

(sumber : Roger S. Pressman, 2002: 365)

No	Notasi	Fungsi
1		Prosedur atau consumer informasi yang ada di luar batasan sistem untuk dimodelkan
2		Transfer informasi (fungsi) yang ada dalam batas sistem untuk dimodelkan
3		Menunjukkan aliran data
4		Penyimpanan data yang disimpan untuk digunakan

b. Data Dictionary

Pengertian *data dictionary* atau kamus data menurut

Roger S. Pressman yang diterjemahkan oleh L.N. Harnaningrum (2002:353) adalah penyimpanan yang berisi deskripsi dari semua objek data yang dikonsumsi atau diproduksi oleh perangkat lunak. Sedangkan menurut pendapat Edward Yourdon (1988:189) :

Data dictionary is an organized listing of all the data elements that are pertinent to the system with precise, rigorous definitions so that both user and systems analyst will have a common understanding of all input, outputs, components of stores and intermediate calculations.

Kamus data adalah daftar organisasi dari semua elemen data yang ada dalam sistem secara lengkap dengan definisi yang baku sehingga user dan analis sistem akan memiliki pengertian sama untuk *input*, *output*, komponen penyimpanan dan perhitungannya. Kamus data diperoleh berdasarkan hasil perancangan model sistem (DFD).

c. Proses Spesification

Proses specification adalah deskripsi setiap fungsi yang disajikan pada DFD (Roger S. Pressman. 2002: 354). Sedangkan menurut Edward Yourdon (1989:203) yaitu: *Pro cess specification, the description of what's happening inside each bottom-level, primitive bubble in a dataflow diagram.*

Penulis menyimpulkan spesifikasi proses adalah deskripsi dari pada apa yang terjadi di dalam setiap tingkat terbawah yang ada di dalam *data flow diagram (primitive bubble)*.

d. Entity Relational Diagram

Menurut Roger S. Pressman yang diterjemahkan oleh L.N. Harnaningrum (2002:360)

entity relational diagram adalah notasi yang digunakan untuk melakukan aktifitas pemodelan data. ERD menggambarkan hubungan antara objek data. Sedangkan menurut pendapat Jeffrey L. Whitter et.al yang diterjemahkan oleh Tim Penerjemah Andi (2004: 281), ERD adalah model data yang menggunakan beberapa notasi untuk menggambarkan data dalam konteks entitas dan hubungan yang dideskripsikan oleh data tersebut.

9. Perancangan Basis Data

a. Pengertian Basis Data

Menurut Fathansyah (1999:2) Basis Data dapat didefinisikan dalam sejumlah sudut pandang seperti :

1. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
2. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redundansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan.
3. Kumpulan file/tabel/arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

b. Sistem Basis Data

Pengertian sistem basis data menurut Fatansyah (1999:9) adalah sebagai berikut:

Sistem basis data merupakan sistem yang terdiri atas kumpulan file (tabel) yang saling berhubungan (dalam sebuah basis data di sebuah sistem komputer) dan sekumpulan program (DBMS/ *DataBase Management System*) yang memungkinkan beberapa pemakai dan/atau program lain untuk mengakses dan memanipulasi file

(tabel-tabel) tersebut.

c. Relasional Basis Data

Fatansyah (1999:20) mengemukakan bahwa :

Pada Model Relasional, basis data akan tersebar (dipilah-pilah) kedalam berbagai tabel 2 dimensi. Setiap tabel selalu terdiri atas lajur mendatar yang disebut dengan baris data (*Row/Record*) dan lajur vertikal yang biasa disebut dengan kolom (*Column/Field*). di setiap pertemuan baris data dan kolom itulah, item - item data (satuan data terkecil) akan ditempatkan.

Hingga saat ini, relasional basis data merupakan model basis data yang paling banyak diterapkan/digunakan. Kemudahan dalam penerapan dan kemampuannya dalam mengakomodasi berbagai kebutuhan pengelolaan basis data yang ada di dunia nyata merupakan alasan mengapa model ini lebih populer untuk diterapkan.

C. Tinjauan Perangkat Lunak

1. Sejarah Oracle

Perusahaan Oracle didirikan pada tahun 1977 oleh tiga orang *programmer*, Bob Miner, Ed Oates dan Larry Ellison yang menjabat sebagai CEO (*Chief Executive Officer*). Dari ketiga *programmer* tersebut, Larry adalah yang "nomor 3" dalam *programming*. Larry lebih sering (dan lebih senang) mengerjakan pemasaran, dan *drop out* dari kuliah karena keasyikannya di Oracle. Perusahaan ini berkonsentrasi pada pembuatan *database server* di *mainframe*. Kisah sukses Oracle Corp terkait dengan sejarah dan teori *database* relasional.

Larry Ellison penemu Software Development Laboratories Tahun 1977. Tahun 1979 SDE dirubah menjadi RSI memperkenalkan produk Oracle Versi 2 sebagai awal produk komersial *relational database system*. Versi ini tidak mendukung transaksi tapi menerapkan

basic SQL untuk *query* dan *join*. RSI tidak pernah meluncurkan versi 1 sementara versi 2 di anggap sebagai trik marketing. Tahun 1983 RSI merubah namanya menjadi oracle corporation.

2. Perkembangan fitur produk-produk Oracle

Perkembangan produk yang dikeluarkan oleh Oracle corporation cukup pesat. Saat ini, oracle merupakan perusahaan yang memiliki teknologi *database* relasional terancang.

1. 1979 Oracle 2 merupakan oracle pertama yang dikeluarkan. Hanya mempunyai basic SQL *function*, *query* dan *join* saja.
2. 1984 Oracle Corporation released Oracle version 4
3. 1985 Oracle DBMS mulai mendukung model *client-server*, Dengan tersedianya teknologi *networking* di era 80 an oracle versi 5.0 mendukung distribusi *query*
4. 1989 Oracle Corporation memasuki pasar produk aplikasi dan dengan mulai dkembangkannya produk ERP- oracle financials dengan dukungan konsep RDBMS. Oracle RDBMS versi 6 dengan mendukung PL/SQL, *row-level locking* dan *hot backups*.
5. 1992 Oracle version 7 dikeluarkan dengan mendukung *referential integrity*, *stored procedures* and *triggers*
6. 1997 Oracle Corporation meluncurkan versi 8, yang mensupport *object-oriented development* dan *multimedia applications*.
7. 1999 Oracle9i muncul untuk mengantisipasi era Internet (akhirannya pada oracle 8i menunjukan arti "internet"). Oracle 8i *database* memasukkan konsep *Java virtual machine* (JVM).
8. 2001 Oracle9i muncul dengan 400 fitur baru termasuk fasilitas membaca dan menulis dokument XML 9i juga dilengkapi dengan opsi

untuk *Real Application Cluster* (RAC), *computer cluster database*, sebagai pengganti konsep paralel *server*

9. 2003, Oracle Corporation meluncurkan Oracle Database 10g. Huruf g memberikan tekanan untuk kebutuhan marketing "grid computing ready"
10. 2005 Oracle Database 10.2.0.1 — juga dikenal sebagai oracle 10g release 2 (10gR2).

3. Oracle database 10g

Oracle 10g merupakan *database server* yang berteknologi *grid computing*. Teknologi tersebut merupakan pengembangan dari RAC (*Real Application Cluster*). Salah satu fungsi dari RAC adalah memberikan perlindungan terhadap kelangsungan data dalam perusahaan sehingga apabila terjadi *crash* pada salah satu *server database*, maka tidak akan mempengaruhi kinerja perusahaan. Hal ini disebabkan karena teknologi RAC memungkinkan untuk membuat beberapa *database server* menjadi seolah-olah satu *database server*, sehingga apabila ada *database server* yang *down*, kinerja *database server* tersebut akan diambil alih oleh server-server yang lain.

Menurut Nugroho (2008) server basis data oracle 10g memuat *database* dan *instance* oracle. *Instance* oracle sendiri merupakan sejumlah proses yang diperlukan agar sistem *database* dapat berfungsi dengan baik. *Instance* oracle mengalokasikan dirinya dalam sejumlah penyangga atau *buffer memory* yang dikenal sebagai *System Global Area* atau SGA dan proses-proses latar belakang atau *background process* yang menangani pekerjaan-pekerjaan yang terkait dengan basis data. Hubungan antara *database* dengan *instance* dapat dianalogikan seperti CPU dan *Hardisk*. CPU merupakan *instance* dan *Hardisk*nya adalah *database*.

Singkatnya, RAC memungkinkan satu *instance* mengelola

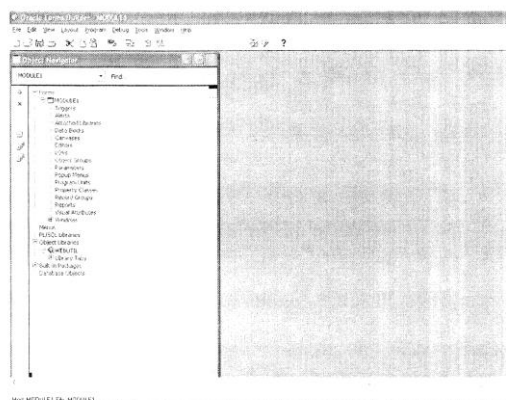
banyak *database*. Pada teknologi *grid computing* yang diadaptasi oracle 10g, mampu mengelola banyak *instance* dan banyak *database*.

4. Oracle Developer suite 10g

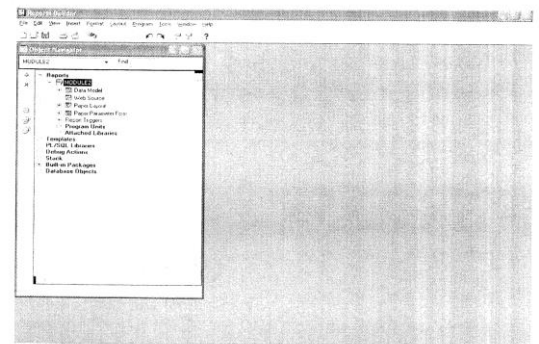
Versi terbaru dari oracle developer adalah oracle developer suite 10g. Versi ini masih *compatible* dengan form 6i. Oracle Forms Developer dan Oracle Application Server Forms Services telah diupgrade untuk menyederhanakan *development* dan *deployment* kedalam aplikasi *web*. Produk yang dikeluarkan oleh oracle yang dikategorikan sebagai oracle developer suite 10g adalah :

1. Oracle Jdeveloper
2. Oracle Form
3. Oracle designer
4. Oracle reports
5. Oracle developer suite software configuration manager
6. Oracle discoverer
7. Oracle business intelligent beans.

Pada penelitian ini, perangkat yang akan digunakan untuk membangun aplikasi analisis *vendor performance* adalah Oracle Form dan Oracle report. Gambar 2.1 dan 2.2 merupakan *environment* dalam Oracle Form dan report 10g.



Gambar 2.1
Environment pada oracle form builder
10g



Gambar 2.2 *Environment*, pada oracle report
bidder 10g

5. Oracle SQL dan PL/SQL

PL/SQL (*Procedural*

Language/ Structure Query Language) adalah suatu blok yang berisi skrip-skrip bahasa prosedural. Oracle menyediakan sebuah fleksibilitas yang tinggi untuk menuliskan skrip program yang tidak terdapat. dalam SQL. PL/SQL memadukan kekuatan manipulasi data dari SQL dengan kekuatan proses dari bahasa prosedural. Kita bisa mengontrol aliran program dengan pernyataan IF dan LOOP. Seperti bahasa prosedural lainnya, PL/SQL mencakup deklarasi variable, defmisi fungsi, dan penanganan error. PL/SQL memudahkan kita untuk memecahkan permasalahan yang sulit kedalam kode program yang mudah dimengerti, dan dapat digunakan kembali untuk aplikasi lain. Ketika sebuah permasalahan bisa diselesaikan dengan menggunakan SQL murni, maka kita bisa langsung memasukkan perintah SQL tersebut kedalam PL/SQL program. Tipe data PL/SQL dapat menyesuaikan dengan tipe kolom dalam SQL.

PL/SQL memiliki performa transaksional yang tinggi dan memiliki banyak kelebihan, yaitu:

1. Integrasi dengan SQL
2. Performa yang lebih baik
3. Produktifitas yang lebih tinggi
4. *Portable*, dapat dipindahkan, *dicopy* dengan mudah
5. Keamanan yang tinggi
6. Akses terhadap *database package*
7. *Support* untuk *Object-Oriented*

Programming

8. *Support* pembangunan aplikasi dan halaman berbasis web

III. OBJEK PENELITIAN DAN ANALISIS SISTEM BERJALAN

A. Objek Penelitian

1. Sejarah PT. Suzuki Indomobil Motor

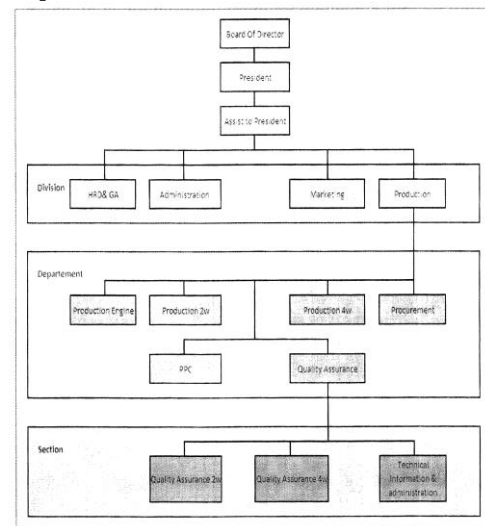
Pada tahun 1970, dibawah bendera PT. Indohero Steel & Engineering Corporation, Suzuki group memperkenalkan produk roda dua. Pada tahun 1976, dibawah kepemimpinan Soebronto Laras, Group Suzuki memulai produksi kendaraan roda 4 pertamanya Pick Up ST 10 dan mobil penumpang Suzuki Fronte. Pada perkembangannya, Suzuki banyak mengeluarkan produk yang sesuai dengan kebutuhan masyarakat. Pada 1999, untuk meningkatkan kepercayaan *international*, Suzuki mendapatkan sertifikat ISO, dan meluncurkan tipe baru 1.000 cc Suzuki Karimun, kendaraan *city car* yang efisien. Suzuki Karimun juga mendapatkan penghargaan sebagai kendaraan terfavorit di Jepang.

Dengan munculnya *trend* baru kendaraan baru MPV, Suzuki mengeluarkan produk handalnya yaitu Suzuki Aerio yang mendapat sambutan hangat di masyarakat. Pada tahun 2004 Suzuki mendapatkan penghargaan menjadi *Mother Plan Automotive* di kawasan Asia Tenggara dengan memproduksi kendaraan tipe MPV yang berkelas *international*, dengan nama APV (*Asean Project Vehicle*) 1.500 c. Selain memenuhi permintaan domestik, APV juga diekspor ke beberapa negara lain. Pada tahun yang sama Suzuki meluncurkan kendaraan roda dua 4 tak 125 cc yang handal di kelasnya itu SHOGUN 125 dan Suzuki SMASH SR 110 yang gesit dan irit. Dan di penghujung tahun yang sama juga diluncurkan sepeda motor bertipe *hyper underbone* berteknologi tinggi yaitu SATRIA FU 150.

3. Struktur Organisasi

PT. Suzuki Indomobil Motor yang berada di bawah Suzuki Motor Corporation Jepang, merupakan perusahaan *manufacturing* yang memproduksi kendaraan roda 2, roda 4 dan juga mesinnya. Bersama dengan PT. Suzuki Indomobil Sales sebagai *sale distributor*, mengembangkan penjualannya ke seluruh *dealer* di Indonesia, juga untuk pasar internasional.

Pada PT. Suzuki Indomobil Motor, terdapat 4 divisi perusahaan, yaitu HRD dan GA, *administration*, *marketing* dan *production*. Pada divisi *production*, terdapat 6 departemen yang berkaitan langsung dengan seluruh kegiatan produksi.



Gambar 3.2

Struktur Organisasi PT. Suzuki Indomobil Motor

4. Analisis Vendor Performance pada PT. Suzuki Indomobil Motor

Analisis *vendor performance* pada PT. Suzuki Indomobil Motor dilakukan secara berkala. Setiap tahun, bagian *Quality Assurance* mengeluarkan laporan audit dan daftar peringkat prestasi *vendor* atau *vendor rank progress*. Namun, proses audit sendiri dilakukan setiap bulan. Audit *vendor* baru dilakukan jika

bagian produksi memerlukan komponen atau *part* baru sesuai dengan kebutuhan produksi.

Bagian *Quality Assurance* melakukan audit dengan kriteria standar sesuai dengan manajemen mutu yang diterapkan pada PT. Suzuki Indomobil Motor. Selain itu, bagian ini juga menerima daftar *claim* dari bagian *Part Inspection* juga dari *dealer-dealer* yang diteruskan pada bagian *Service*. *Claim* tersebut akan ditindak lanjuti oleh *vendor*, dan bagian *Quality Assurance* akan memberikan penilaian pada solusi yang diberikan oleh *vendor*.

a. Audit Vendor Baru

Berdasarkan *Form No QA . 062 Rev.02* mengenai standar penilaian audit *vendor bam, point* yang dijadikan penilaian adalah sebagai berikut:

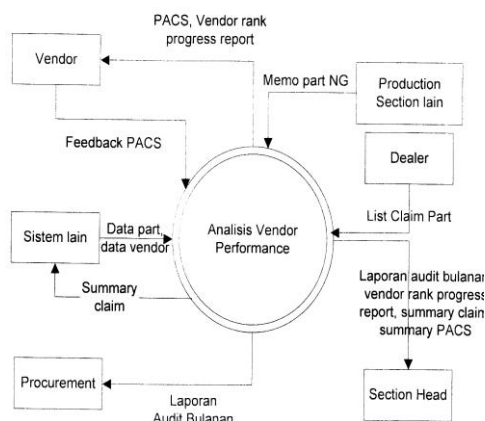
Rumus untuk menentukan nilai audit, adalah sebagai berikut ;

$$\text{Nilai audit} = \frac{\text{Total Nilai}}{\text{Target}} \times 100$$

b. Audit Vendor Regular

Berdasarkan *Form No. QA. 059. rev. 5* tentang audit *vendor* regular pada PT. Suzuki Indomobil Motor, item penilaian pada proses audit *vendor* regular adalah sebagai berikut :

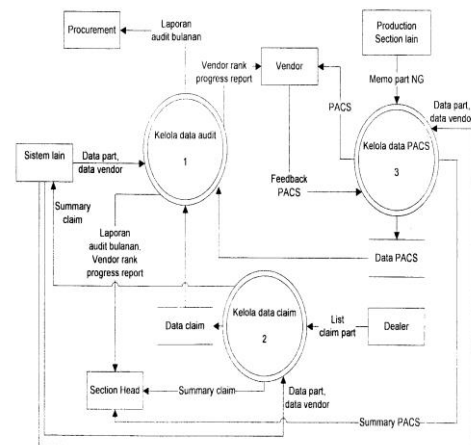
5. Context Diagram



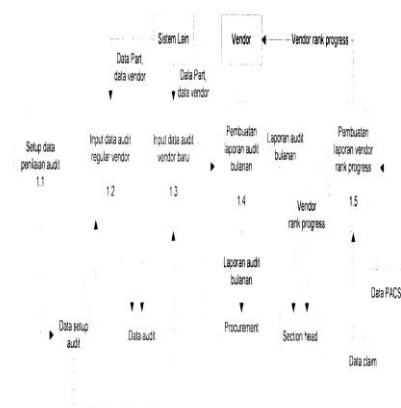
Gambar 3.4

Context diagram

(4) Data Flow Diagram (a). DFD Level 1

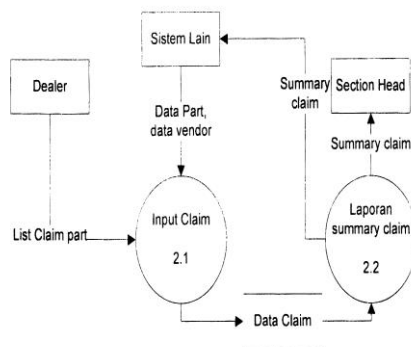


(b). DFD Level 2 Proses 1 Kelola data audit



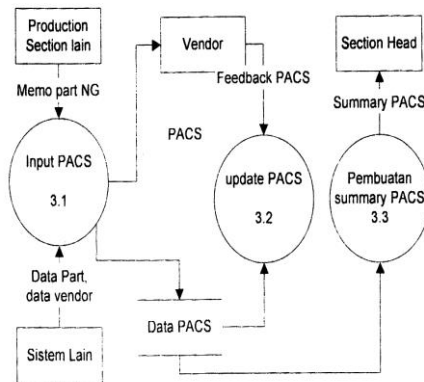
Gambar 3.6
DFD Level 2 proses 1

(c). DFD Level 2 Proses 2 Kelola data claim



Gambar 3.7
DFD Level 2 proses 2

(d). **DFD level 2 Proses 3 Kelola data PACS**



Gambar 3.8
DFD Level 2 proses 3

(5). **Data Dictionary**

Data dictionary dibuat berdasarkan aliran data yang terdapat dalam DFD. Aliran data yang masuk adalah *item-item* data yang berhubungan dan yang diperlukan bagi sistem analisis *vendor performance*. Data yang mengalir bisa didapat dari sistem lain, atau dari entitas luar. Data yang berkaitan dengan sistem lain tidak akan seluruhnya digunakan bagi sistem ini, hanya *item-item* data tertentu yang diperlukan bagi sistem yang akan dicantumkan dalam *data dictionary*.

1. Data setup audit= @audit_jtem id + audit item + vendor type + total_std_point { @question_item_id + question_item + standar_point }

2. Data audit = @audit_id + nomor_audit + vendor_type + vendor_name + auditor 1 + auditor2 + auditor3 + periode + part_name + tanggal_evaluasi + type_part + RTP + date + batas_tanggal_audit { @audit_line_id + Item_audit + point_result { @audit_dtl_id + question_item + penemuan + standar_point + hasil + tanggal_improvement1 + point_improvement1 + tanggal_improvement2 + point_improvement2 } } { @audit_minus_id + minus_item + minus_point }
3. Data claim = @claim_id + nomor_claim + vendor_id + issue_by + tanggal_claim + claim_type + sold_average + claim_period + demerit_point + claim_period_to + grand_total { @claim_line_id + part_name + quantity + model + problem }
4. Data PACS = @pacs_id + PACS_number + issue by + tanggal_pacs + vendor + model + class_part + defect_qty + demerit_point + problem_content + evaluation_point + part_number
5. Data Vendor (dari si stem lain) = @vendor_id + vendor_name + vendor_code + start_date_active + end_date_active
6. Data Parts (dari sistem lain) = @part_id + part_name + part_description + uom + part_type + part_status

(6). **Process Specification**

(a). **Proses 1.1 setup data penilaian audit**

Begin
Open File data_setup_audit
While data_setup_audit ada
Rekam data_setup_audit
End While
Close file data_setup_audit
End

(b). **Proses 1.2 input data audit regular vendor**

Begin

- Open File data_vendor*
Open File data_setup_audit
Open File data_part
While datajvendor ada
Open file data_audit
Rekam data_audit
Close file data_audit
End While
Close file data_vendor
Close file data_setup_audit
Close file data_part
End
- (c). **Proses 1.3 input data audit new vendor**
Begin
Open File datajvendor
Open File data_setup_audit
Open File data_part
While data_vendor ada
Open file data_audit
Rekam data_audit
Close file data_audit
End While
Close file data_vendor
Close file data_setup_audit
Close file data_part
End
- (d). **Proses 1.4 Pembuatan iaporan audit bulanan**
Begin
Open File data_setup_audit
Open File data vendor
Open File data audit
Open File data_part
While data_audit ada
display data_audit
print data_audit
End While
Close file data_setup audit
Close File data_vendor
Close File data_audit
Close file data_part
End
- (e). **Proses 1.5 Pembuatan laporan vendor rank progress**
Begin
Open Pile data_vendor
Open File data_part
Open File data_claim
Open File data_pacs
Calculate nilai_vendor
Order by nilai_vendor
- While data_vendor ada*
display data_claim, data_pacs
calculate vendor rank
print vendor rank
End While
Close File data_vendor
Close file data_part
Close File data_claim
Close File data_pcs
End
- (f). **Proses 2.1 input data claim**
Begin
Open File data_vendor
Open File data_part
Open File data_claim
While data_claim ada
Rekam data_claim
End While
Close file data_part
Close File data_vendor
Close File data_claim
End
- (g). **Proses 2.2 Pembuatan laporan summary claim**
Begin
Open File data_vendor
Open File data_part
Open File data_claim
While data_claim ada
display data_claim
print data_claim
End While
Close file data_part
Close File data_vendor
Close File data_claim
End
- (h). **Proses 3.1 input data PACS**
Begin
Open File data_vendor
Open File data_part
Open File data_PACS
While data_PACS ada
Rekam data_PACS
End While
Close file data_part
Close File data_vendor
Close File data_PACS
End
- (i). **Proses 3.2 update PACS**
Begin
Open File data_vendor

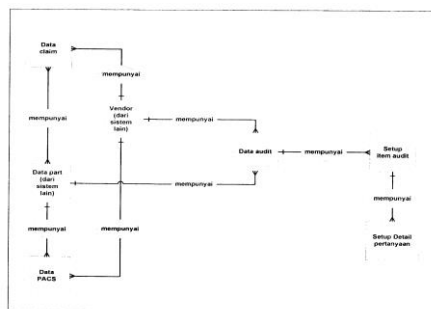
Open File data_PACS
While data_vendor ada
update data_PACS
End While
Close File data_vendor
Close File data_PACS
End

(j). Proses 3.3 Pembuatan I summary PACS

Begin
Open File data_vendor
Open File data_part
Open File data_PACS
While data_PACS ada
display data_PACS
print data_PACS
End While
Close file data_part
Close File data_vendor
Close File data_PACS
End

b. Pemodelan Data Konseptual
Pemodelan

konseptual menggunakan diagram ER. Untuk mempermudah penggambaran diagram ER, tidak semua atribut ditampilkan di setiap entitas. Atribut yang ditampilkan adalah atribut kunci dari setiap entitas. Keterangan lengkap mengenai atribut dari entitas-entitas akan dijelaskan lebih rinci dalam keterangan terpisah.



Gambar 3.9

Conceptual Entity Relational Diagram

Aplikasi audit vendor performance memerlukan data dari sistem lain yang telah ada. Data vendor diperoleh dari sistem pengadaan barang atau purchasing. Sedangkan daiapart berasal dari sistem

pengelolaan barang atau inventory. Keterangan mengenai atribut setiap entitas adalah sebagai berikut:

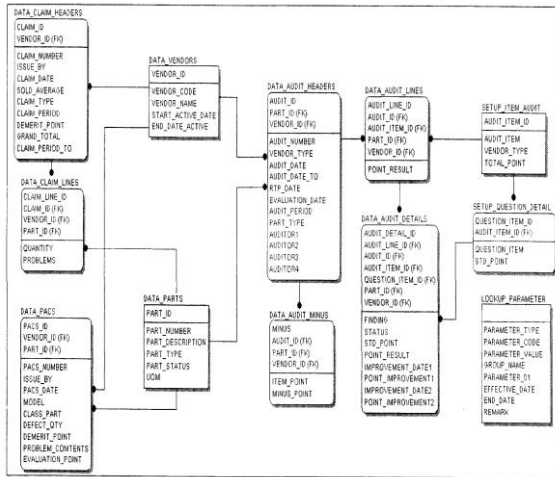
1. Data audit: audit_id, nomor_audit, vendor type, vendor name, auditor 1, auditor2, auditors, periode, part_id, part name, tanggal evaluasi, type_part, RTF date, batas_tanggal_audit, audit_line_id, Item_audit, point_result, audit_dtl_id, question_item, penemuan, standar_point, hasil, tanggal_Improvement1, point_Improvement1, tanggal_improvement2, point improvement2, audit_minus_id, minus_item, minus_point
2. Setup item audit: audit_item id, audit_item, total_std point, vendor_type.
3. Setup detail pertanyaan: audit item_id, question_item_id, question_item, standar_point.
4. Data PACS: pacs_id, pacs_number, issue_by, tanggal_pacs, vendor_id, model, class_part, defect_qty, demerit_point, problem_content, evaluation_point, part_id
5. Data Claim: claim_id, claim_number, vendor_id, issue_by, tanggal_claim, claim type, sold_average, claim_period, demerit_point, claim_period_to, grand_total, claim_line_id, part_id, quantity, model, problem
6. Data vendor (dari si stem lain): vendor_id, vendor_name, vendor_code , start_date active_end_date_active
7. Data part (dari sistem lain): part_id, part_name, part_description, uom, part_type, part_status.

IV PERANCANGAN DAN IMLEMENTASI

Berdasarkan *Framework for Aplication of Software Technique (FAST)*, proses perancangan dan implementasi meliputi *phase physical design and integration*, *phase construction and testing*, *phase installation and elivery* serta *phase maintenance*. Pada tahap ini, *logical design* yang telah dibuat di bab sebelumnya, dikonversi kedalam bentuk fisik berupa *database* tabel dan aplikasi.

A. Physical Design and Integration
1. Perancangan Basis Data

a. Skema Relasi



gambar 4.1

Skema relasi aplikasi analisis vendor performance

Perancangan Antar muka (Interface)

a. Form Setup Audit Item

Gambar 4.2

Perancangan form setup audit

Gambar 4.2

Perancangan form setup audit question

b. Form Setup Data Master

Gambar 4.4

Perancangan form setup data master

Gambar 4.5

Perancangan form audit transaction

Perancangan form entry PACS

e. Form Entry Claim

Gambar 4.6
Perancangan form audit transaction minus point

Gambar 4.9
Perancangan form entry claim

3. Perancangan Laporan

a. Laporan Audit Vendor Regular

Gambar 4.7
Perancangan form audit transaction detail
d. Form Entry PACS

Gambar 4.10
Perancangan laporan audit regular

b. Laporan Audit Vendor Baru

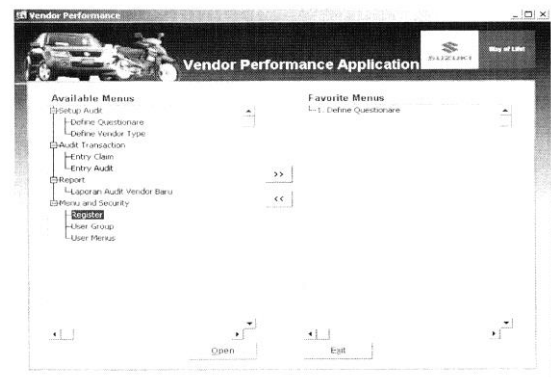
Gambar 4.8

Gambar 4.11

Perancangan laporan audit *vendor* baru
e. Laporan Summary Audit

Gambar 4.12

Perancangan laporan *summary* audit



Gambar 4.17
Menu utama

d. Laporan Summary PACS

Gambar 4.13

Perancangan laporan *summary* PACS

c. Form Setup Audit Item

Audit Item	Total Point
Keperluan manajemen	35
Wewenang dan politik pengaturan untuk manajer quality assurance	35
Pelaksanaan quality audit	35
Standarisasi	35
Kontrol produk awal	6
Kontrol proses	6
Kontrol inspeksi	6
Pengawasan penyimpangan mutu	9
Kontrol supplier	9
Kontrol peralatan diat ukur	6

Buttons: Question, Save, Close

Gambar 4.18
Form *define* audit

Antar Muka

a. Form Login

Gambar 4.16
Form Login

b. Menu Utama

Gambar 4.19

Form define audit question

Gambar 4.26
Laporan audit vendor baru

d. Form Audit Transaction

Gambar 4.20
Form audit transaction

c. Laporan Summary Audit

Gambar 4.27
Laporan summary audit

3. Laporan

a. Laporan Audit Vendor Regular

Gambar 4.25
Laporan audit vendor regular

d. Laporan Summary PACS

Gambar 4.28
Laporan Summary PACS

b. Laporan Audit Vendor Baru

e. Laporan Summary Claim

2. Dalam pemlaian *vendor rank*, terdapat banyak *variable* penilaian yang sering berubah, sehingga aplikasi akan mengalami perubahan sejalan dengan sistem penilaian yang berlaku.

REFERENSI

- [1] Anonym 2007.
<http://anakbinus.blogsome.eom/2007/12/3/paper-management-support-system/>
- [2] Iswahyudi, Catur. Mengukur Kualitas Website dan Layanan TI. 2010. <http://catur.dosen.akprind.ac.id/2010/09/01/mengukur-kualitas-website-dan-layanan-ti/>
- [3] L. Hutabarat, Bernaridho. *Pemrograman Oracle PL/SQL*. 2002. Yogyakarta: Penerbit Andi Yogyakarta
- [4] L. Whitten, Jeffrey, Lonnie D. Bentley, Kevin D. Dittman Diterjemahkan oleh Tim Penerjemah Andi. 2004. *Met ode Desain & Analisis Sistem*. Yogyakarta: Penerbit Andi.
- [5] Lorentz, Diana, Cathy Shea, Simon Watt. *Oracle Database SQL Quick Reference IOg Release 2*. 2005 http://download.oracle.com/docs/cd/B19306_01/server.102/b14195.pdf
- [6] Lorentz, Diana. *Oracle Database SQL Reference IOg Release 2*. 2005.
http://download.oracle.com/docs/cd/B19306_01/server.102/b14200.pdf
- [7] Nugroho, Adi. *Menjadi Administrator Basis Data Oracle IOg*. 2008. Bandung: INFORMATIKA Bandung.
- [8] S. Pressman, Ph. D, Roger Diterjemahkan oleh L.N. Harnaningrum. 2002. *Rekayasa Perangkat Lunak*. Yogyakarta: Penerbit Andi Yogyakarta.