

Aplikasi *Database* Karyawan Bagian Marketing dengan Studi Kasus Pada PT.CIC Futures

Radiant Victor Imbar, Mulyadi

Jurusan Sistem Informasi

Fakultas Teknologi Informasi, Universitas Kristen Maranatha

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

email: radiant.vi@eng.maranatha.edu, minx_8@yahoo.com

Abstract

PT.CIC Futures is a periodic broker company and have the right to withdraw and collect people's fund who will invest in type of periodic commerce legally, however until today this particular company still using information system especially for the human capital. Hence, the issues that currently occurred are, irregular human capital's data's, unavailability of monitoring regarding the performance of the employees and unsorted accordingly. Therefore, this company requires an application that are able to help to store employees data's, make organizational structure more organized, and an application that can help to count the employees' commission also evaluate the level of the performance for their employees. This application will be using C# programming language; moreover it will have the access right which it will be use in order to compose employees' data's more neatly.

Keywords: evaluation, monitoring, employee, commission.

1 Pendahuluan

PT.CIC Futures adalah divisi investasi dari CIC Group (Century Intervest Corporation) dengan pusat bisnis berada di Singapura, dan merupakan perusahaan Pialang Berjangka di Indonesia yang berkedudukan di Jakarta. CIC Futures sendiri berdiri pada tanggal 17 Mei 2000. Sesuai dengan Undang – undang No.32 tahun 1997, PT.CIC Futures adalah perusahaan yang berhak melakukan penarikan dan penghimpunan dana masyarakat yang akan berinvestasi di bidang perdagangan berjangka komoditi dengan legal.

Sistem Informasi untuk bagian *business consultant* dan trader pada PT.CIC Futures saat ini masih menggunakan sistem manual, dan tidak tersedianya media penyimpanan data. Sehingga penilaian maupun data kinerja *business consultant* dan trader hanya dibuat pada kertas laporan dan data karyawan tersimpan pada Microsoft Office Excel, oleh karena itu penilaian dan data kinerja para *business consultant* dan trader tidak terlalu akurat dan mudah hilang.

2 Rumusan Masalah

Masalah yang akan dibahas dalam pengembangan aplikasi ini dapat dirumuskan sebagai berikut:

- Bagaimana cara manager dapat melihat dan membuat laporan *employee* dengan tepat dan cepat?
- Bagaimana manager dapat menghitung komisi untuk *employee* dengan akurat dan cepat?

- Bagaimana manager dapat mencari data dengan cepat dan telah terstruktur?
- Bagaimana manager, asisten manager, dan *leader* dapat melihat data *employee* sesuai dengan hak akses yang diberikan?

3 Landasan Teori

3.1 Basis Data^[4]

Basis data, menurut Stephens dan Plew (2000), adalah mekanisme yang digunakan untuk menyimpan informasi atau data. Basis data juga dapat diartikan sebagai kumpulan data yang saling berhubungan satu dengan lainnya yang diorganisir berdasarkan sebuah skema atau struktur tertentu, tersimpan pada *hardware* (komputer) dan menggunakan *software* untuk melakukan manipulasi sebagai penyedia informasi bagi pengguna (*user*). Dengan basisdata, pengguna dapat menyimpan data secara terorganisir. Sehingga informasi yang terdapat dalam data tersebut dengan mudah dapat diambil.

Dari uraian di atas, basisdata data memiliki peranan yang cukup penting. Ada beberapa hal mengapa basisdata diperlukan, yaitu:

- Basisdata merupakan salah satu komponen penting dalam sistem informasi yang merupakan dasar dalam hal menyediakan informasi.
- Basisdata menentukan kualitas informasi, yaitu akurat, tepat pada waktunya dan relevan. Dengan kata lain informasi itu dapat dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya.
- Basisdata dapat mengurangi duplikasi data (data redundancy).
- Basisdata bisa meningkatkan hubungan antar data dengan struktur yang telah ditentukan sehingga memudahkan dalam hal relasi antar komponen basisdata tersebut.

3.2 ERD (*Entity Relationship Diagram*)^[2]

Entity Relationship Diagram adalah relasi antar objek data yang merupakan penggambaran secara sistematis komponen-komponen himpunan entitas dan himpunan relasi. Tujuan utama dari pembuatan *Entity Relationship Diagram (ERD)* adalah untuk menunjukkan himpunan entitas apa saja yang ingin dilibatkan dalam sebuah basis data dan bagaimana hubungan yang terjadi diantara objek-objek tersebut. ERD model terdiri atas tiga komponen utama, yaitu entitas, atribut dan relasi.

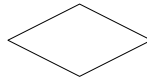
Komponen–komponen dalam *Entity Relationship Diagram*:

- Entitas adalah sebuah barang atau obyek yang dapat dibedakan dari obyek lain. Contoh: buku, barang, pelanggan, mahasiswa, dan lain-lain.



Gambar 10 Entitas

- Relasi adalah asosiasi dua atau lebih entitas.



Gambar 11 Relasi

- Atribut adalah properti yang dimiliki setiap entitas yang akan disimpan datanya.



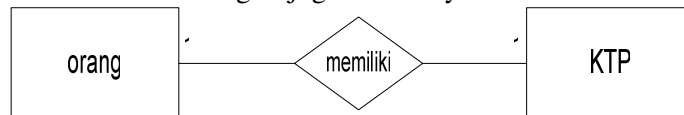
Gambar 12 Atribut

Kardinalitas dalam relationship

Kardinalitas adalah angka yang menunjukkan banyaknya kemunculan suatu obyek terkait dengan kemunculan obyek lain pada suatu relasi. Jenis *Cardinality Ratio*, yaitu :

1. Satu ke satu (*one to one*)

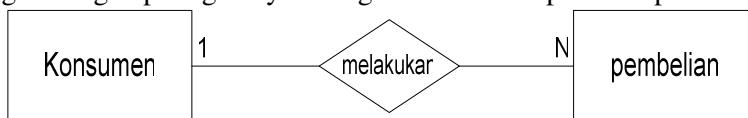
Setiap entitas pada himpunan A berhubungan paling banyak dengan satu entitas pada himpunan entitas B dan begitu juga sebaliknya.



Gambar 13 Kardinalitas One to one

2. Satu ke banyak (*one to many*)

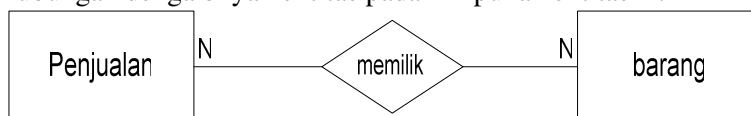
Entitas pada himpunan A dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi tidak sebaliknya, dimana setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan A.



Gambar 14 Kardinalitas One to many

3. Banyak ke banyak (*many to many*)

Entitas pada himpunan A dapat berhubungan dengan banyak entitas pada himpunan entitas B dan demikian juga sebaliknya di mana setiap entitas pada himpunan B dapat berhubungan dengan banyak entitas pada himpunan entitas A.



Gambar 15 Kardinalitas many to many

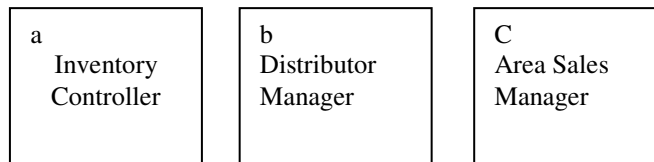
3.3 DFD (*Data Flow Diagram*)^[1]

Data Flow Diagram (DFD) menggunakan notasi atau simbol-simbol. Penggunaan notasi untuk menggambarkan arus data sebelum perancangan program. Tujuan dari DFD adalah untuk membantu memberikan pemahaman terhadap suatu sistem pada semua tingkat kompleksitasnya. Pada tahap analisis, penggunaan DFD ini sangat membantu sekali di dalam komunikasi dengan pemakai sistem untuk memahami sistem secara logika. Penggunaan DFD tidak hanya untuk menggambarkan sistem yang sudah ada melainkan juga untuk sistem yang baru akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir.

DFD termasuk alat yang digunakan pada metodologi pengembangan sistem terstruktur. Selain itu kegunaan DFD adalah digunakan untuk dokumentasi sistem. Beberapa simbol yang digunakan pada DFD untuk maksud mewakili:

- *External Entity* (Kesatuan Luar)

Seperti diuraikan di atas dalam teori sistem, setiap sistem memiliki kesatuan luar atau batas sistem (*sistem boundary*) yang memisahkan sistem dengan lingkungan luarnya. Kesatuan luar dapat disimbolkan dengan notasi kotak. Kesatuan luar tersebut dapat diberi identifikasi dengan huruf kecil diujung kiri atas.



Gambar 16 Notasi Kesatuan Luar pada DFD

- *Data Flow* (Arus Data)

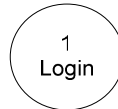
Data flow atau arus data dalam DFD diberikan simbol panah. Arus data mengalir di antara kesatuan luar, proses dan simpanan data. Arus data ini melambangkan atau menunjukkan arus dari data yang dapat berupa masukan untuk sistem atau hasil dari suatu proses untuk selanjutnya disimpan atau digunakan untuk proses berikutnya. Oleh karena arus data dinotasikan sebagai panah, maka identifikasinya diletakkan pada bagian atas atau bawah panah, jika gambar panah mendarat. Sedangkan untuk panah ke bawah atau ke atas identifikasinya diletakkan di samping kanan atau kiri panah.



Gambar 17 Arus data

- *Process* (Proses)

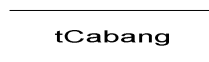
Proses adalah suatu kegiatan yang dilakukan oleh orang atau mesin (komputer) dari hasil arus data yang menuju ke dalam proses untuk dihasilkan arus data yang keluar dari proses. Untuk proses notasinya ditunjukkan dengan bentuk persegi panjang tegak.



Gambar 18 Proses

- **Data Storage (Penyimpanan Data)**

Data storage (penyimpanan data) merupakan notasi dari media simpanan data yang menggambarkan anata lain suatu file atau basisdata pada sistem komputer, suatu arsip atau catatan manual, suatu kotak tempat berkas, suatu tabel acuan manual, atau suatu agenda atau buku. Simbol atau notasi penyimpanan data pada DFD dapat disimbolkan dengan sepasang garis horisontal paralel yang tertutup di salah satu ujungnya.



Gambar 19 Penyimpanan Data

4 Analisis dan Perancangan

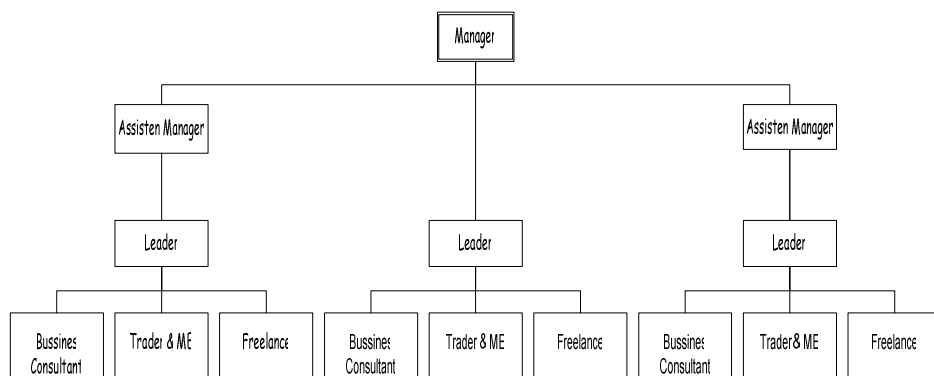
4.1 Proses Bisnis

Pada proses bisnis ini dijelaskan bahwa pertama kali calon *employee* harus mengikuti *training* terlebih dahulu, kemudian setelah selesai di haruskan menandatangani kontrak yang akan menjadi *business consultant* atau *freelance*.

Business consultant dan *freelance* mempunya tugas untuk mencari calon nasabah dengan cara melakukan *telp*, *appointment*, *follow up*. Dan setiap *business consultant* harus memberikan laporan berupa kertas setiap hari nya pada atasan sedangkan *freelance* tidak perlu. Jika *business consultant* atau *freelance* mendapatkan nasabah dengan membuka *account* pada perusahaan maka *business consultant* atau *freelance* akan menjadi *trader*.

Trader sendiri memiliki tugas sendiri, yaitu melakukan *trading* terhadap *account* yang di dapatkan dari *business consultant* atau *freelance* atau didapatkan sendiri. Setiap *trader* akan mendapatkan komisi jika melakukan *trading* setiap bulan nya. Jika *trader* tidak memiliki *account* lagi, maka *trader* harus menjadi *business consultant* atau *freelance* lagi untuk mendapatkan nasabah.

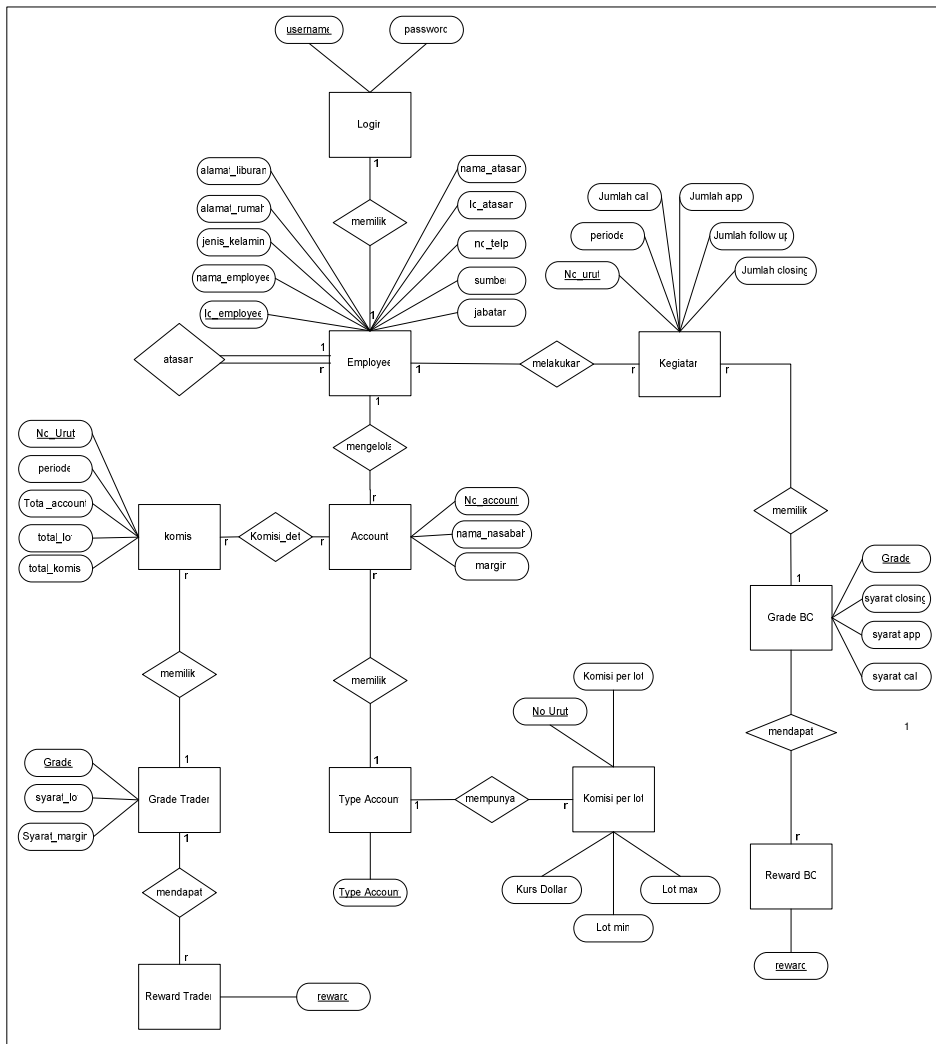
4.2 Skema Data Karyawan



Gambar 11 Skema Data Karyawan

Setiap *manager* hanya memiliki *assisten manager* atau *leader*, setiap *leader* boleh memiliki anak buah *business consultant* dan *trader & ME* maupun *leader*. *Leader* hanya memiliki bawahan *business consultant* dan *trader & ME*.

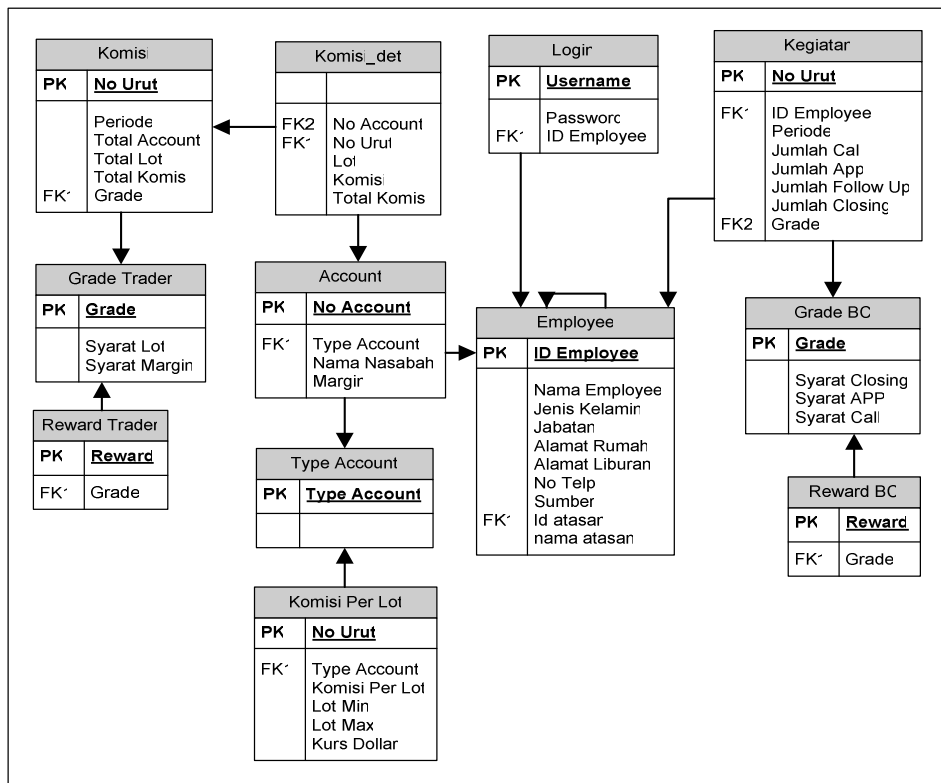
4.3 Entity Relationship Diagram



Gambar 12 Entity Relationship Diagram

Entitas *employee* memiliki relasi *self join* dan memiliki relasi 1 to *many* terhadap entitas *komisi*, *account*, *kegiatan* dan memiliki relasi 1 to 1 terhadap entitas *login*. Entitas *type account* memiliki relasi 1 to *many* terhadap entitas *account* dan entitas *komisi per lot*. Entitas *komisi* memiliki relasi *many* to *many* terhadap entitas *account* dan memiliki relasi *many* to 1 terhadap entitas *grade trader*. *Grade trader* sendiri memiliki relasi 1 to *many* terhadap entitas *reward trader*. Entitas *grade BC* memiliki relasi 1 to *many* terhadap entitas *kegiatan* dan entitas *reward BC*.

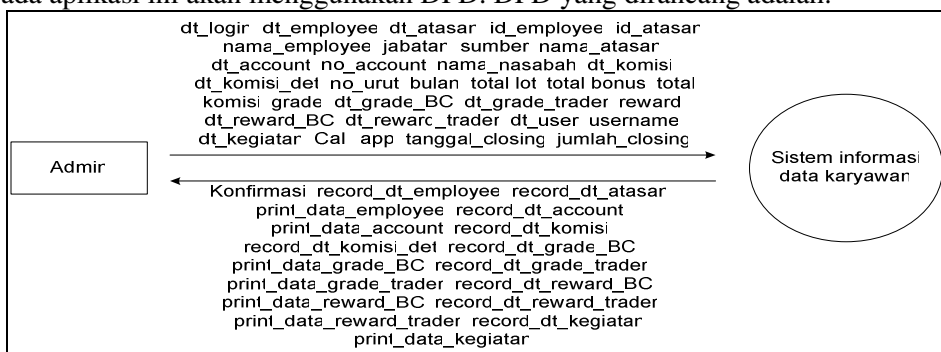
4.4 Transformasi ER kedalam Tabel



Gambar 13 Transformasi ER kedalam table

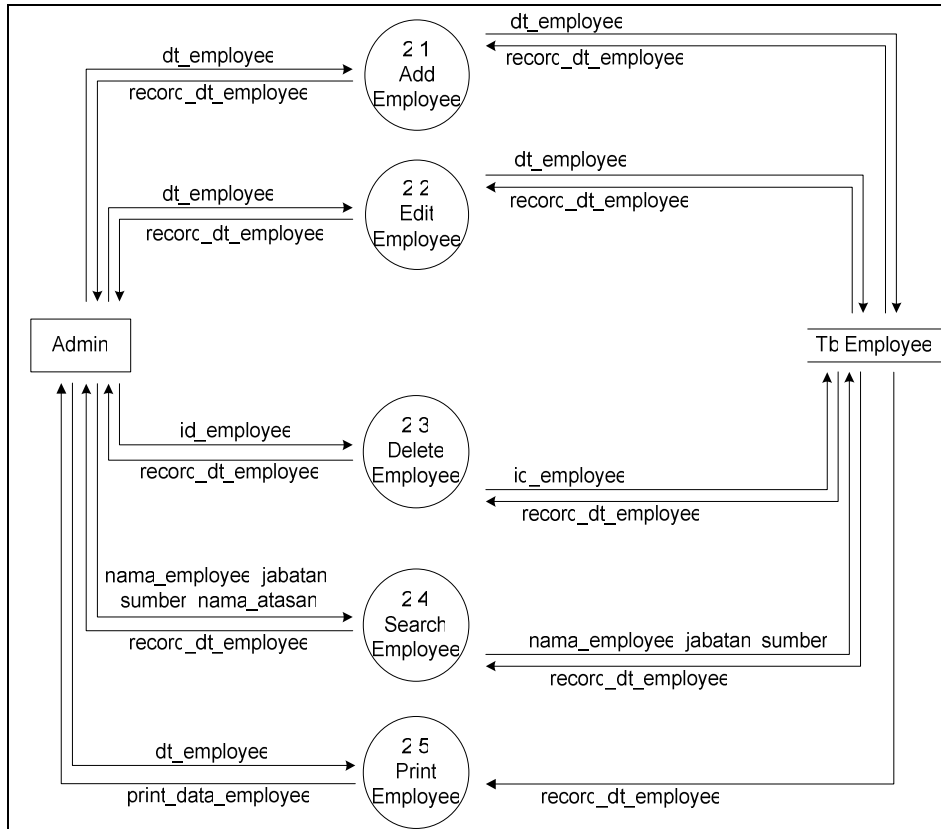
4.5 Data Flow Diagram

Pada aplikasi ini akan menggunakan DFD. DFD yang dirancang adalah:



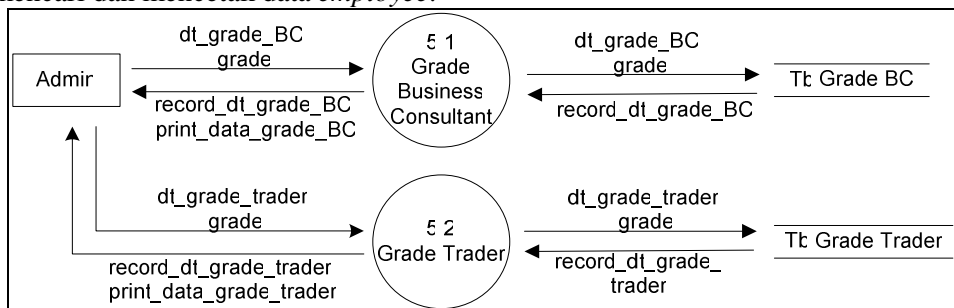
Gambar 14 DFD Level 0

Pada *Data Flow Diagram Level 0* ini dapat dilihat yang dapat mengakses aplikasi ini adalah hanya Admin yang memiliki tugas mengolah data *employee*, data *account*, data *komisi*, data *reward*, data *grade*, data kegiatan dan data *user*.



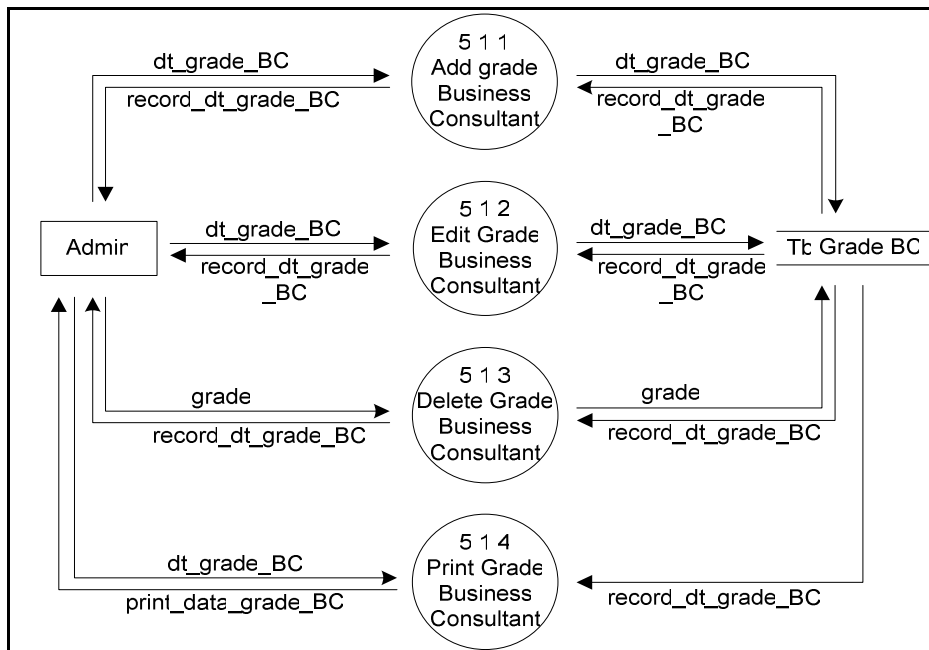
Gambar 206 DFD Level 2 Proses 2 (*Employee*)

Pada DFD level 2 proses 2 ini dijelaskan lebih rinci mengenai proses pengolahan data *employee*. Pada proses ini admin dapat menambah, mengubah, menghapus, mencari dan mencetak data *employee*.



Gambar 17 DFD Level 2 Proses 5 (*Grade*)

Pada DFD level 2 proses 5 ini menjelaskan bahwa *grade* memiliki 2 bagian yaitu: *grade* untuk *business consultant* dan *grade* untuk *trader*. Dan proses ini akan diterangkan lebih lanjut pada proses DFD level 3.



Gambar 21 DFD Level 3 Proses 5.1 (*Grade Business Consultant*)

Pada DFD level 3 proses 5.1 ini menjelaskan lebih lanjut dari proses *grade* untuk *business consultant* yang terdapat pada DFD level 2 proses 5. Pada proses ini admin dapat menambah, mengubah, menghapus dan mencetak data *grade business consultant*.

4.6 Kamus Data

Nama Data	Employee
Deskripsi	Merupakan data yang berisi data employee
Struktur Data	Tabel Employee = @ID Employee + Nama Employee + Jabatan + Jenis Kelamin + Alamat Rumah + Alamat Liburan + No Telp + Sumber + ID Atasan + Nama Atasan ID Employee = {legal-character} Nama Employee = {legal-character} Jabatan = [Manager Asisten Manager Leader <i>Business Consultant</i> <i>Trader & ME</i> <i>Freelance</i>] Jenis Kelamin = [Pria Wanita] Alamat Rumah = {legal-character} Alamat Liburan = {legal-character} No Telp = {0-9} Sumber = [CV MGM Kampus] ID Atasan = {legal-character} Nama Atasan = {legal-character} legal-character = [A-Z a-z 0-9 ' -]

Nama Data	Kegiatan
Deskripsi	Merupakan data yang berisi data Kegiatan
Struktur Data	Tabel Kegiatan = @No Urut + ID Employee+ Periode + Jumlah Call + Jumlah App + Jumlah Follow Up + Jumlah Closing + Grade

	No Urut = {0-9} ID Employee = {legal-character} Periode = [January – December] + 4{0-9} Jumlah Call = {0-9} Jumlah App = {0-9} Jumlah Follow Up = {0-9} Jumlah Closing = {0-9} Grade = [A-Z] *Grade Business Consultant* legal-character = [A-Z a-z 0-9 ' -]
--	--

4.7 PSPEC

Nomor	2.1
Nama	Proses <i>Add Employee</i>
Deskripsi	Proses <i>Input</i> data <i>employee</i>
Input	<i>dt_employee</i>
Output	<i>record_dt_employee</i>
Logika	1. Admin memasukan semua data <i>employee</i> dan data atasan 2. IF semua data <i>employee</i> dan data atasan terisi dengan benar 3. Then data <i>employee</i> dan data atasan sukses bertambah Else admin memasukan ulang data <i>employee</i> dan data atasan

Nomor	2.2
Nama	Proses <i>Edit Employee</i>
Deskripsi	Proses mengubah data <i>employee</i>
Input	<i>dt_employee</i>
Output	<i>record_dt_employee</i>
Logika	1. Admin memilih data <i>employee</i> terlebih dahulu 2. Admin ditampilkan data <i>employee</i> dan data atasan yang telah dipilih 3. Admin mengubah data <i>employee</i> dan data atasan 4. IF data <i>employee</i> dan data atasan yang akan diubah sudah benar

	Then data <i>employee</i> dan data atasan akan berubah Else Admin memilih kembali data <i>employee</i> yang akan diubah
--	--

5 Hasil Tercapai



Gambar 22 Main Form

Pada *form* ini akan ditampilkan pilihan *toolStripMenuItems*, yang terdiri dari *File*, *Komisi*, *Kegiatan*, *Reward & Grade*, *Laporan*. Pada *File* sendiri terdapat *employee*, *account*, *user*, *change user* dan *exit*. Pada *Reward & Grade* terdapat *grade BC* yang berfungsi untuk membuat penilaian terhadap *Business Consultant*, *grade trader & ME* yang berfungsi untuk membuat penilaian terhadap *Trader & Marketing Executive*. Sedangkan *reward BC* berfungsi untuk menampilkan hadiah yang dapat diberikan kepada *Business Consultant*, *reward trader & ME* berfungsi untuk menampilkan hadiah yang dapat diberikan kepada *Trader & Marketing Executive*. Pada *Laporan* terdapat laporan *reward BC*, laporan *reward trader & ME*, laporan transaksi.

ID Employee	Nama Employee	Jabatan	Jenis Kelamin	Alamat Rumah	Alamat Liburan	No Telp	Sumber	ID Atasan	Nama Atasan
AW - Abel	Abel	Asisten Manager	Pria	Bukit Jarian 168		0899009099	AGM	Man - Faizal	Faizal
BC - dadad	dadad	Business Consul...	Pria	kopo		098117717	Kampus	Lead - Jasman	Jasman
BC - didi	didid	Business Consul...	Pria	pungkur		1234567890	AGM	AW - Abel	Abel
Free - beto	beto	Freelance	Wanita	-	-	0	Kampus	AW - Abel	Abel
Lead - Jasman	Jasman	Leader	Pria	Pasteur 68	pagarsih 8	0818901547	C.V	AW - Abel	Abel
Trd - aztech	aztech	Trader & ME	Pria	cimahi		901828	C.V	AW - Abel	Abel
Trd - gomon	gomon	Trader & ME	Wanita	mulya sari 88		022919190	C.V	Lead - Jasman	Jasman
Trd - ika	ika	Trader & ME	Wanita	lembong 10		089908990	Kampus	Lead - Jasman	Jasman

Gambar 23 Halaman Employee

Pada form ini akan ditampilkan *button Add, Edit, Delete, Print* dan *Main Form*. *Button Add* berfungsi untuk menampilkan form *Add Employee*, *button Edit* untuk menampilkan form *Edit Employee*, *button Delete* untuk menghapus data *employee* dengan cara mencari data *employee* yang tersedia dilist kemudian tekan tombol *Delete*. *Print* untuk mencetak seluruh data *employee*. Dan *Main Form* adalah tombol untuk kembali ke *mainform*.

Add Employee

ID Employee : Man - Faizal

Nama Employee : Faizal

Jabatan : Manager

Jenis Kelamin : ☒ Pria ☐ Wanita

Alamat Rumah : babakan jeruk 10

Alamat Liburan :

No Telp :

Sumber :

ID Atasan : -

Nama Atasan : -

Gambar 24 Halaman Add Employee

Pada form ini, *employee* (manager, asisten manager, leader) dapat menginput data *employee* yang berupa ID *Employee* yang didapat dari gabungan nama *employee* dan jabatan, Nama *Employee*, Jabatan, Jenis Kelamin, Alamat rumah, Alamat Liburan, No Telp, Sumber, ID Atasan, Nama Atasan. Jika semua telah terisi

kemudian menekan *button Add* maka data *employee* akan tersimpan pada *database*, jika *button cancel* di tekan maka akan kembali ke tampilan halaman *employee*.

Gambar 25 Halaman *Edit Employee*

Pada *form* ini, *employee* menerima *form* yang sudah terisi data *employee* yang ingin di-*edit*, data ini didapat dari *form employee*. *Employee* tinggal mengubah data yang ingin diubah, kecuali nama dan jabatan kemudian menekan tombol *edit* dan data *employee* yang ada di-*database* akan ter-*edit*, jika tombol *cancel* ditekan maka kembali ke halaman *employee*.

ID Employee	Nama Employee	Jabatan	Jenis Kelamin	Alamat Rumah	Alamat Liburan	No Telp	Sumber	ID Atasan	Nama Atasan
AM - 123	123	Asisten Man	Wanita	a	s	1	MGM	Man - Faizal	Faizal
AM - a	a	Asisten Man	Pria	a		0812233445	MGM	Man - Faizal	Faizal
AM - eror qa?	eror qa?	Asisten Man	Wanita	777	8888	9999	C.V	Man - Faizal	Faizal
Lead - bagas	bagas	Leader	Wanita	seru	susah	0812233445	MGM	AM - a	a
Lead - hrm	hrm	Leader	Pria	ho	ha	081212121	MGM	AM - a	a
Lead - hore	hore	Leader	Wanita	a	s	1	Kampus	AM - a	a
Lead - zzz	zzz	Leader	Pria	z	z	12	MGM	AM - 123	123
Man - Faizal	Faizal	Manager	Pria	babakan jeruk 10		0123456789	C.V	-	-

Gambar 26 Halaman *Print Employee*

Pada *form* ini, hanya akan ditampilkan seluruh data *employee* yang akan di cetak.

6 Kesimpulan dan Saran

6.1 Kesimpulan

Dari pengembangan aplikasi ini, setelah melakukan evaluasi dapat ditarik kesimpulan terhadap tujuan, sebagai berikut:

1. Aplikasi telah dapat menampilkan laporan *employee* dengan cepat dan tepat.

2. Aplikasi telah dapat menghitung komisi dengan tepat sesuai dengan *type account* yang tersedia.
3. Aplikasi dapat mencari data yang telah terstruktur rapi dengan menggunakan hak akses.
4. Aplikasi dapat membantu manager dalam pengecekan dan pencetakan *grade, reward*, laporan para *employee*.

6.2 Saran

Saran-saran yang telah didapat, mengenai aplikasi yang telah dikembangkan ini antara lain adalah:

1. Desain terlalu simple.
2. Aplikasi dapat digunakan secara *mobile* atau via internet.
3. Aplikasi dapat digunakan secara *server – client*.
4. *Error handling* terlalu *simple* dan monoton.

7 Daftar Pustaka

- [1]. Mardiyanto, Dwi. Pemodelan Data Dalam Rekayasa Perangkat Lunak. From: kur2003.if.itb.ac.id/file/SE6162%20ERD.pdf
- [2]. Jogianto HM. (1995), *Analisis dan Perancangan Sistem Informasi*, Yogyakarta: Andi
- [3]. Jaenudin, (2007) Belajar Sendiri .NET dengan Visual C# 2005. Yogyakarta: Andi
- [4]. Fathansyah, Basis Data, Bandung;Informatika
- [5]. Kurniawan, Agus, Risman Adnan, Panji Aryaputra, Norman Sasono, Ali Ahmad Heryana, M. Fathur Rahman, I Wayan Saryada, Adi Wirasta, (2004) Pengenalan Bahasa C#. Retrieved September 12, 2008, from <http://projectotak.net/files/folders/milestone1/entry1.aspx>
- [6]. Modul Rekayasa Perangkat Lunak (2008).
- [7]. Black Box Testing. From: http://www.testingbrain.com/BLACKBOX/BLACK_BOX_Testing.html
- [8]. Alam, M.Agus.J (2005), Cara Mudah Menggunakan Crystal Reports XI. Bandung
- [9]. Kendall & Kendall. (2002) *System Analysis And Desain (Fifth Ed.)*. Upper Saddle River, New Jersey, USA:Prentice-Hall
- [10]. [McI Pol Wes 06] Brett D McLaughlin, Gary Pollice & David West, 2006, “*Head First Object-Oriented Analysis & Design*”. O'REILLY.
- [11]. Deitel, H.M, P.J, Deitel, J.Listfield, T.R.Nieto, C.Yaeger, M.Zlatkina, (2002), *C# How to Program*. Uppersaddle River, New Jersey:Prentice Hall
- [12]. White Box Testing. From: http://www.testingbrain.com/WHITEBOX/WHITE_BOX_Testing.html