KARYA ILMIAH

**SISTEM INFORMASI PEMETAAN PENJUALAN RETAIL
DENGAN GOOGLE MAP BERBASIS WEB
PADA PT. MATAHARI DEPARTMENT STORE. Tbk**

Oleh:

**AKIM MANAOR HARA PARDEDE, ST., M.Kom
NIDN. 0124037904**

STMIK KAPUTAMA

**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER KAPUTAMA
STMIK KAPUTAMA
BINJAI
2013**

ABSTRAK

Perkembangan teknologi dan informasi melalui media internet saat ini sangat pesat. Penyampaian informasi yang cepat, akurat dan relevan saat ini sudah menjadi kebutuhan pokok bagi perusahaan. Kemajuan teknologi memacu perusahaan membuat pembahasan agar dapat mempermudah dalam penyampaian informasi yang dibutuhkan.

Dengan adanya sistem baru dengan media GOOGLE MAP berbasis web ini maka diharapkan dapat memberikan berbagai kemudahan serta kecepatan akses para manager masing-masing cabang dalam menyajikan informasi tentang laporan penjualan bulanan PT. Matahari Department Store.Tbk yang tersebar diseluruh indonesia.

Dalam rancangan sistem ini, penulis menggunakan Visual Studio 2010 sebagai perancangan User Interface (tatap muka) berbasis Web dengan editor ASP.NET dan Microsoft SQL Server 2008 sebagai media pembuatan database. Dengan adanya komponen pengembangan diatas diharapkan dapat mempermudah proses pengolahan laporan penjualan bulanan masing-masing cabang PT. Matahari Department Store.Tbk. Dan apabila proses diatas berjalan dengan baik, maka akan diperoleh sistem informasi sesuai yang diharapkan. Untuk lebih jelas tentang sistem informasi ini pembaca dapat melihatnya pada Karya Ilmiah yang telah penulis rancang.

Kata Kunci : Sistem Informasi, Laporan, Penjualan

ABSTRACT

The development of information technology and media through the internet is very fast. Rapid delivery of information, accurate and relevant today has become a staple for the company. Advances in technology spur companies to make the discussion in order to facilitate the delivery of information required.

With the new system with a web-based media GOOGLE MAP is then expected to provide the convenience and speed of access to the manager of each branch in presenting information about the monthly sales reports PT. Matahari Department Store.Tbk spread throughout Indonesia.

In the design of this system, the author uses Visual Studio 2010 as the design of User Interface (face to face) with a Web-based editor ASP.NET and Microsoft SQL Server 2008 database as a media creation. With the development of the above components is expected to facilitate the processing of monthly sales reports each branch PT. Matahari Department Store.Tbk. And if the above process goes well, it will obtain the information system as expected. For more information about the system's readers can see the project design has been the author.

Keywords: System Information, Reports, Sales

KATA PENGANTAR

Puji Syukur Penulis ucapkan kepada Tuhan Yang Maha Esa atas segala limpahan dan karunia-Nya sehingga penulis dapat menyelesaikan Karya Ilmiah ini dengan judul “**Sistem Informasi Pemetaan Penjualan Retail Dengan Google Map Berbasis Web Pada Matahari Department Store.Tbk**” Adapun pihak-pihak yang banyak membantu penulis, dengan kerendahan hati penulis mengucapkan terima kasih kepada :

1. Bapak Devi Yendrianof, S.Kom., M.Si. selaku Ketua STMIK KAPUTAMA Binjai.
2. Seluruh kerabat dan pihak-pihak yang telah memberikan motivasi dan kerjasamanya yang baik terhadap penulis.
3. Teristimewa istri tercinta Yessika P L Tobing, SKM.

Penulis menyadari bahwa Laporan Karya Ilmiah ini masih jauh dari kesempurnaan karena terbatasnya ilmu yang penulis miliki. Saran dan kritikan yang sifatnya membangun sangat diharapkan. Semoga Laporan Karya Ilmiah ini ada manfaatnya. Akhir kata penulis mengucapkan terima kasih.

Binjai, Juli 2013

Penulis

Akim Manaor Hara Pardede, ST., M.Kom
NIDN. 0124037904

DAFTAR ISI

ABSTRAK	i
KATA PENGANTAR	iii
RIWAYAT HIDUP	vi
DAFTAR ISI	vii
DAFTAR TABEL	ix
DAFTAR GAMBAR	x
DAFTAR LAMPIRAN	xi
 BAB I PENDAHULUAN	 1
1.1 Latar Belakang	1
1.2 Identifikasi Masalah	2
1.3 Batasan Masalah	2
1.4 Rumusan Masalah	3
1.5 Tujuan Penelitian	4
1.6 Manfaat Penelitian	4
1.7 Sistematika Penulisan	5
 BAB II LANDASAN TEORI	 8
2.1 Pengertian Sistem.....	8
2.2 Pengertian Informasi	9
2.3 Pengertian Sistem Informasi	9
2.4 Pengertian Pemetaan	10
2.5 Pengertian Penjualan.....	12
2.6 Pengertian Retail	13
2.7 Google Map	13
2.8 Web	14
2.9 Kamus Data.....	14
2.10 Database	15
2.11 Data Flow Diagram (DFD)	17
2.12 Flowchart Dokumen/Flowmap.....	18
2.13 SQL Server 2008.....	21
2.14 ASP.NET.....	21
2.15 Visual Studio 2010.....	22
 BAB III ANALISIS SISTEM BERJALAN	 26
3.1 Tinjauan Perusahaan	26
3.1.1 Sejarah Perusahaan	26
3.1.2 Struktur Organisasi dan Fungsi	30
3.2 Prosedur Sistem Berjalan	38
3.3 Diagram Alir Data Sistem Berjalan	38
3.4 Kamus Data Sistem Berjalan	40
3.5 Spesifikasi Sistem Berjalan.....	40

3.5.1	Spesifikasi Bentuk Dokumen Masukkan	40
3.5.2	Spesifikasi Bentuk Dokumen Keluaran	41
3.5.3	Spesifikasi File	41
3.5.4	Struktur Kode	43
3.5.3	Spesifikasi Program	43
3.6	Permasalahan.....	44
3.7	Alternatif Pemecahan Masalah	44
BAB IV	RANCANGAN SISTEM USULAN	45
4.1	Prosedur Sistem Usulan	45
4.2	Diagram Alir Data Sistem Berjalan	46
4.2.1	Diagram Konteks Sistem Usulan	46
4.2.2	Data Flow Diagram Level 0 Sistem Usulan	47
4.3	Kamus Data Sistem Usulan	48
4.4	Spesifikasi Sistem Usulan	48
4.4.1	Spesifikasi Rancangan Dokumen Masukkan	49
4.4.2	Spesifikasi Rancangan Dokumen Keluaran	51
4.4.3	Normalisasi File	52
4.4.4	Spesifikasi File	54
4.4.5	Struktur Kode	58
4.4.6	Spesifikasi Program	60
4.5	Hasil Program	60
4.5.1	Form Login	60
4.5.2	Form Menu Utama	61
4.5.3	Form Input Data Toko	62
4.5.4	Form Input Data Store Manager	62
4.5.5	Form Input Data Penjualan	63
4.5.6	Form Input User Management	64
4.5.7	Laporan Data Toko	64
4.5.8	Laporan Data Store Manager	65
4.5.9	Laporan Data Penjualan	65
4.5.10	Laporan Data User Management	66
4.6	Spesifikasi Sistem Komputer	66
4.6.1	Umum	66
4.6.2	Spesifikasi Perangkat Keras	67
4.6.3	Spesifikasi Perangkat Lunak	67
4.7	Jadwal Implementasi	67
BAB V	KESIMPULAN DAN SARAN	69
5.1	Kesimpulan	69
5.2	Saran	70

DAFTAR TABEL

Tabel I.1	Rencana Kegiatan	7
Tabel II.1	Simbol-Simbol Kamus Data	15
Tabel II.2	Simbol-Simbol Data Flow Diagram (DFD)	18
Tabel II.3	Simbol-Simbol Dalam Flowchart Document/Flowmap	20
Tabel III.3	Flowmap Sistem Berjalan Pada Matahari	39
Tabel III.2	Tabel Data Toko	42
Tabel III.3	Tabel Data Barang	42
Tabel III.4	Tabel Data Penjualan	42
Tabel IV.1	Tabel Data Store Manager	54
Tabel IV.2	Tabel Data Toko	55
Tabel IV.3	Tabel Data Penjualan	56

DAFTAR GAMBAR

Gambar II.1	Form Designer	22
Gambar II.2	Toolbars	23
Gambar II.3	Menu Bar	23
Gambar II.4	Tittle Bar	24
Gambar II.5	ToolBox	24
Gambar II.6	Solution Explorer	25
Gambar II.7	Properties	25
Gambar III.1	Struktur Organisasi PT. Matahari Department Store.Tbk	30
Gambar III.2	Dokumen Masukkan Penjualan.....	40
Gambar III.3	Dokumen Keluaran Laporan Penjualan	41
Gambar IV.1	Diagram Konteks Sistem Usulan	46
Gambar IV.2	Data Flow Diagram Level 0 Sistem Usulan	47
Gambar IV.3	Rancangan Form Input Data Store Manager	49
Gambar IV.4	Rancangan Form Input Data Toko	50
Gambar IV.5	Rancangan Form Input Data Penjualan	51
Gambar IV.6	Rancangan Laporan Data Toko, Store Manager dan Penjualan	52
Gambar IV.7	Rancangan Relasi Antar Tabel	57
Gambar IV.8	Rancangan Spesifikasi Program	60
Gambar IV.9	Tampilan Form Login	61
Gambar IV.10	Tampilan Form Menu Utama	61
Gambar IV.11	Tampilan Form Input Data Toko	62
Gambar IV.12	Tampilan Form Input Data Store Manager	63
Gambar IV.13	Tampilan Form Input Data Penjualan	63
Gambar IV.14	Tampilan Form User Management	64
Gambar IV.15	Tampilan Laporan Data Toko	64
Gambar IV.16	Tampilan Laporan Data Store Manager	65
Gambar IV.17	Tampilan Laporan Data Penjualan.....	65
Gambar IV.18	Tampilan Laporan Data User Management	66

DAFTAR LAMPIRAN

Lampiran 1	: Surat Keterangan Dosen Pembimbing
Lampiran 2	: Surat Pernyataan Bersedia Membimbing
Lampiran 3	: Lembar Persetujuan Test Program
Lampiran 4	: Berita Acara Test Program
Lampiran 5	: Kartu Bimbingan Untuk Pembimbing I
Lampiran 6	: Kartu Bimbingan Untuk Pembimbing II
Lampiran 7	: Listing Program

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pada era globalisasi dan transformasi saat ini menyebabkan perubahan yang sangat pesat dengan tingkat persaingan yang ketat. Oleh karena itu setiap organisasi diuntut untuk melakukan kegiatan operasionalnya secara efektif dan efisien supaya dapat mempertahankan eksistensinya. Organisasi yang memiliki kebutuhan akan data dan informasi, sudah seharusnya mempersiapkan sistem pengolahan data yang baik, agar dapat mencapai tujuan yang sesuai dengan yang di rencanakan. Pada masa lalu informasi merupakan barang yang mahal, sehingga penyebarannya berjalan dengan lambat. Akibatnya banyak kegiatan operasional organisasi yang tertunda untuk di laksanakan . informasi yang berkualitas yaitu informasi yang akurat, relevan, dan tepat waktu. Berbeda dengan masa lalu, sekarang informasi sudah sangat mudah di dapat dan pada saat ini penyampaian informasi yang baik menjadi faktor pendukung utama dalam kesuksesan sebuah organisasi.

PT. Matahari Department Store.Tbk merupakan salah satu perusahaan yang bergerak di bidang retail, perusahaan ini menjual barang-barang fashion, kosmetik, elektronik, mainan anak-anak dan alat-alat rumah tangga, maka dipandang perlu suatu sistem laporan penjualan yang baik. Hal ini bertujuan untuk mengetahui secara sistematis dalam penjualan masing-masing toko yang tersebar di seluruh indonesia, dan dengan hasil laporan tersebut pimpinan cabang dapat mengetahui *Achievement* penjualan dari target yang telah ditentukan oleh perusahaan pada masing-masing toko melalui sistem pemetaan dengan GOOGLE MAP.

Berdasarkan uraian yang telah di jelaskan di atas, untuk dapat membantu kelancaran operasi kerja pada PT. Matahari Department Store.Tbk, maka penulis membahas suatu sistem pemetaan penjualan dengan judul **“SISTEM INFORMASI PEMETAAN PENJUALAN RETAIL DENGAN GOOGLE MAP BERBASIS WEB PADA PT.MATAHARI DEPARTMENT STORE.TBK”**.

1.2 Identifikasi Masalah

Dengan memperhatikan uraian pada latar belakang di atas, maka penulis dapat mengidentifikasi masalah-masalah dalam penelitian ini, yaitu:

1. Bagaimanana implementasi pemetaan penjualan seluruh toko dengan memanfaatkan GOOGLE MAP ?
2. Bagaimana agar data yang tersaji dapat diakses dan dipahami dengan mudah oleh para manager masing-masing toko sebagai bahan pertimbangan keputusan untuk meningkatkan penjualan pada masing-masing toko?
3. Bagaimana untuk membandingkan level penjualan pada toko yang satu dengan toko yang lain sehingga dapat mengimplementasikan strategi bisnis untuk meningkatkan penjualan melebihi toko yang lainnya?

1.3 Batasan Masalah

Batasan masalah merupakan hal yang paling penting untuk ditentukan terlebih dahulu sebelum sampai pada tahap pembahasan selanjutnya. Sehubungan dengan keterbatasan waktu, biaya, serta kemampuan yang penulis miliki serta untuk meghindari kesimpang siuran dalam penulisan agar sesuai dengan judul dan latar

belakang yang dijelaskan di atas. Maka penulis membatasi pembahasan materi pada hal-hal sebagai berikut:

1. Penelitian dilakukan di PT.Matahari Department Store.Tbk binjai supermall.
2. Data yang akan di teliti mencakup data penjualan, data toko, dan data store manager saja.
3. Perancangan User Interface (tatap muka) dirancang berbasis web dengan Visual Studio 2010 dengan editor ASP.NET dan bahasa yang dipakai adalah Visual Basic dengan database SQL SERVER 2008.

1.4 Rumusan Masalah

Berdasarkan uraian yang telah dipaparkan oleh penulis dalam identifikasi masalah dan agar pembahasan tidak menyimpang dari judul Karya Ilmiah, maka penulis merumuskan permasalahan yang akan dibahas adalah:

1. Bagaimana merancang sistem informasi pemetaan penjualan retail pada PT.Matahari Department Store.Tbk?
2. Bagaimana proses pengolahan data laporan penjualan dapat dilakukan dengan baik dan mudah?
3. Bagaimana merancang sebuah sistem yang dapat mempermudah dalam informasi laporan penjualan pada masing-masing toko PT.Matahari Department Store.Tbk?

1.5 Tujuan Penelitian

Adapun tujuan dari penulisan materi penelitian ini adalah untuk merancang sistem informasi pemetaan penjualan retail dengan GOOGLE MAP berbasis web menggunakan Visual Basic pada PT.Matahari Department Store.Tbk.

1.6 Manfaat Penelitian

Sesuai dengan tujuan dilakukanya penelitian pada, maka manfaat yang diharapkan dapat diperoleh adalah:

1. Membantu mempermudah dalam hal penyajian dan pelaporan data penjualan pada masing-masing toko.
2. Dapat mengetahui *Achievement* cabang toko lain menggunakan sistem pemetaan GOOGLE MAP.

1.7 Sistematika Penulisan

Untuk memberikan gambaran yang lebih mudah terhadap penulisan Karya Ilmiah ini maka secara sistematika dapat dijelaskan isi dari Karya Ilmiah berikut ini:

BAB I : PENDAHULUAN

Menjelaskan secara umum tentang kegiatan penelitian yang dilakukan meliputi latar belakang, identifikasi masalah, batasan masalah, rumusan masalah, tujuan penelitian, manfaat penelitian dan sistematika penulisan.

BAB II : LANDASAN TEORI

Menguraikan tentang berbagai teori yang berkaitan dengan permasalahan yang penulis kemukakan, terdiri dari pengertian sistem, pengertian informasi, pengertian sistem informasi, pengertian pemetaan, pengertian penjualan, pengertian retail, google map, web, kamus data, database, data flow diagram (DFD), flowchart dokumen/flowmap, SQL server 2008, ASP.NET, visual studio 2010.

BAB III : ANALISIS SISTEM BERJALAN

Bab ini berisi tentang tinjauan perusahaan, sejarah perusahaan, struktur organisasi dan fungsi perusahaan, prosedur sistem berjalan, diagram alir data sistem berjalan, kamus data sistem berjalan, spesifikasi sistem berjalan, permasalahan dan alternatif pemecahan masalah.

BAB IV : RANCANGAN SISTEM USULAN

Bab ini berisi tentang prosedur sistem usulan, diagram alir data sistem usulan, kamus data sistem usulan, spesifikasi sistem usulan, spesifikasi sistem komputer dan jadwal implementasi.

BAB V : KESIMPULAN DAN SARAN

Bab ini berisi tentang kesimpulan berupa point-point dari hasil pembahasan pada bab-bab sebelumnya dan saran-saran yang dianggap berguna untuk perbaikan sistem yang sedang berjalan saat ini.

BAB II

LANDASAN TEORI

2.1 Pengertian Sistem

Secara umum, sistem dapat didefinisikan sebagai sekumpulan objek-objek yang saling berelasi dan berintraksi secara hubungan antar objek bisa dilihat sebagai satu kesatuan yang dirancang untuk mencapai suatu tujuan.

Menurut Schott (Hanif Al Fatta,2007:4), menyatakan bahwa “Sistem terdiri dari unsur-unsur seperti masukkan (*input*), pengolahan (*processing*), serta keluaran (*output*)”.

Menurut Murdick dan Ross (Hanif Al Fatta,2007:3), menyatakan bahwa “Sistem sebagai seperangkat elemen yang digabungkan satu dengan yang lainnya untuk suatu tujuan bersama”.

Menurut Davis (Al-Bahra Bin Ladjamudin,2005:3), menyatakan bahwa “Sistem sebagai bagian-bagian yang saling berkaitan yang beroperasi bersama-sama untuk mencapai beberapa sasaran atau maksud”.

Dari beberapa pengertian di atas diambil kesimpulan bahwa sistem adalah kumpulan atau kelompok unsur yang saling terkait untuk mencapai tujuan. Jika dalam sebuah sistem terdapat elemen yang tidak memberikan manfaat dalam mencapai tujuan yang sama, maka elemen tersebut dapat dipastikan bukanlah bagian dari sistem.

2.2 Pengertian Informasi

Menurut Hanif Al Fatta (2007:9), menyatakan bahwa “Informasi adalah data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat dalam pengambilan keputusan saat ini atau mendatang.

Menurut Raymond McLeod (Al-Bahra Bin Ladjamudin, 2005:9), menyatakan bahwa “Informasi sebagai data yang telah diolah menjadi bentuk yang lebih berarti bagi penerimanya”.

Menurut Gordon. B. Davis (Al- Bahra Bin Ladjamudin,2005:8), menyatakan bahwa “Informasi sebagai data yang telah diolah menjadi bentuk yang lebih berarti dan berguna bagi penerimanya untuk mengambil keputusan masa kini maupun yang akan datang”.

Berdasarkan beberapa pendapat yang telah dikemukakan diatas dapat ditarik kesimpulan bahwa informasi adalah data yang sudah diolah, dibentuk, atau dimanipulasi sesuai dengan keperluan tertentu untuk digunakan dalam proses pengambilan keputusan.

2.3 Pengertian Sistem Informasi

Sistem informasi berasal dari dua kata yaitu kata sistem dan informasi yang telah dijelaskan sebelumnya. Sistem informasi (*Information System*) adalah sekumpulan komponen yang saling berhubungan, mengumpulkan atau mendapatkan, memproses, menyimpan dan mendistribusikan informasi untuk menunjang pengambilan keputusan dan pengawasan dalam suatu organisasi serta membantu manager dalam mengambil keputusan (Kanneth,2008).

Menurut Al-Bahra Bin Ladjamudin (2005:13), menyatakan bahwa “Sekumpulan prosedur organisasi yang pada saat dilaksanakan akan memberikan informasi bagi pengambil keputusan dan untuk mengendalikan organisasi”.

Telah diketahui bahwa informasi merupakan hal yang sangat penting bagi manajemen dalam mengambil keputusan. Informasi dapat diperoleh dari sistem informasi (*Information System*) atau disebut juga dengan *processing system* atau *information generating systems*.

Sistem informasi menerima masukan data dan informasi, mengelola data tersebut sesuai dengan data intruksi dan mengeluarkan hasilnya. Model dasar sistemnya masukan, pengolahan, keluaran adalah cocok bagi kasus sistem pengolahan informasi yang paling sederhana dimana masukan tiba pada saat bersamaan tetapi hal ini jarang terjadi.

2.4 Pengertian Pemetaan

Dalam kamus bahasa Indonesia pemetaan atau visualisasi adalah pengungkapan suatu gagasan atau perasaan dengan menggunakan gambar, tulisan, peta, dan grafik. Sementara itu Spasser (1997:78), menyatakan bahwa “peta adalah alat relasi (*relational tools*) yang menyediakan informasi antar hubungan entitas yang dipetakan.” Definisi pemetaan yang dirumuskan dalam kamus bahasa Indonesia menekankan ungkapan perasaan dalam bentuk gambar, tulisan, peta, dan grafik. Definisi ini menekankan produk atau *output* dari peta. Sedangkan Spasser lebih menekankan proses kegiatan pemetaan. Kedua pendapat ini tidak berbeda melainkan saling melengkapi, karena sebuah produk atau *output* pemetaan dihasilkan melalui

proses. Sehingga dapat dinyatakan bahwa pemetaan merupakan sebuah proses yang memungkinkan seseorang mengenali elemen pengetahuan serta konfigurasi, dinamika, ketergantungan timbal balik dan interaksinya. Pemetaan pengetahuan digunakan untuk keperluan manajemen teknologi, mencakup definisi program penelitian, keputusan menyangkut aktivitas yang berkaitan dengan teknologi, disain, struktur berbasis pengetahuan serta pemrograman pendidikan dan pelatihan. *Output* dari kegiatan pemetaan adalah gambar, tulisan, peta, dan grafik yang menunjukkan hubungan antar elemen pengetahuan.

Menurut Chen dalam Ristiyono (2008:21), menyatakan bahwa “peta ilmu pengetahuan menggambarkan suatu hubungan ruang antara batas penelitian dalam bidang kegiatan yang signifikan, juga dimana bidang penelitian itu didistribusikan serta dapat memberikan makna dari hubungan tersebut”. Peta ilmu pengetahuan dapat menggambarkan dan memberikan makna dari hubungan ruang antara batas penelitian yang bidang kegiatannya signifikan dan bidang kegiatan tersebut dapat didistribusikan. Peta ilmu pengetahuan tidak hanya merupakan suatu alat yang praktis untuk menyampaikan informasi mengenai aktivitas ilmiah, tetapi juga dapat dijadikan sebagai suatu dasar untuk mengkaji atau memahami aktivitas ilmiah dengan menggambarannya secara tersusun dan terstruktur. Visualisasi ilmu pengetahuan dapat diwujudkan dalam bentuk peta, sehingga muncullah bidang pemetaan ilmu pengetahuan atau *knowledge mapping*. Pemetaan ilmu pengetahuan dapat dilakukan berdasarkan beberapa cara yang terkait erat dengan subjek dokumen.

Menurut Sulistyo-Basuki (2002:1), menyatakan bahwa “pemetaan pengetahuan dapat dilakukan dengan bentuk pemetaan kronologis, pemetaan berbasis

co-word, pemetaan kognitif dan pemetaan”. Dari pendapat Sulisty-Basuki tersebut dapat diketahui pemetaan pengetahuan terdiri dari 4 (empat) bentuk yakni kronologis, berbasis *co-word*, kognitif dan konseptual.

2.5 Pengertian Penjualan

Aktivitas penjualan merupakan pendapatan utama perusahaan karena jika aktivitas penjualan produk maupun jasa tidak dikelola dengan baik maka secara langsung dapat merugikan perusahaan. Hal ini dapat disebabkan karena sasaran penjualan yang diharapkan tidak tercapai dan pendapatan pun akan berkurang.

Menurut Henry Simamora (2000:24), dalam buku “Akuntansi Basis Pengambilan Keputusan Bisnis” menyatakan bahwa, “Penjualan adalah pendapatan lazim dalam perusahaan dan merupakan jumlah kotor yang dibebankan kepada pelanggan atas barang dan jasa”.

Menurut Henry Simamora (2000:24), dalam buku “Akuntansi Basis Pengambilan Keputusan Bisnis” menyatakan bahwa: “Penjualan adalah pendapatan lazim dalam perusahaan dan merupakan jumlah kotor yang dibebankan kepada pelanggan atas barang dan jasa”.

Menurut Chairul Marom (2002:28), dalam buku “Sistem Akuntansi Perusahaan Dagang” menyatakan bahwa :“Penjualan artinya penjualan barang dagangan sebagai usaha pokok perusahaan yang biasanya dilakukan secara teratur”.

Berdasarkan pengertian di atas, maka dapat disimpulkan bahwa penjualan adalah persetujuan kedua belah pihak antara penjual dan pembeli, dimana penjual

menawarkan suatu produk dengan harapan pembeli dapat menyerahkan sejumlah uang sebagai alat ukur produk tersebut sebesar harga jual yang telah disepakati.

2.6 Pengertian retail

Retail berasal dari bahasa Perancis yaitu "Retailer" yang berarti "Memotong menjadi kecil kecil" (Risch, 2010).

Dalam kamus Bahasa Inggris - Indonesia, Retail bisa juga di artikan sebagai "Eceran"

Menurut Kotler (2007:592), mendefinisikan usaha eceran (*retailing*) adalah semua kegiatan yang melibatkan penjualan barang dan jasa secara langsung kepada konsumen akhir untuk penggunaan pribadi bukan untuk bisnis.

Menurut Hendri Maruf (2006:7), menyatakan bahwa “Retail adalah kegiatan usaha menjual barang/jasa kepada perorangan untuk keperluan diri sendiri, keluarga, atau rumah tangga. Kegiatan tersebut mencakup penjualan barang dan jasa kepada pengguna yang bervariasi mulai dari mobil, pakaian, makanan, hingga tiket bioskop”.

Dengan demikian pengertian retail dapat disimpulkan merupakan penjualan dari sejumlah kecil komoditas kepada konsumen.

2.7 Google Map

Menurut pengertian tentang *Google Map* dalam (<http://blog.xinthinx.us/2010/06/pengertian-google-maps-api.html>), merupakan suatu peta dunia yang dapat

digunakan untuk melihat suatu daerah. Dengan kata lain, *Google Map* merupakan suatu peta yang dapat dilihat dengan menggunakan suatu *browser web*.

2.8 Web

Menurut pengertian tentang Web dalam (<http://starfirm.blogspot.com/2012/09/pengertian-web.html>), merupakan suatu media penyebaran informasi melalui internet. Fasilitas web memberikan izin kepada pengakses untuk melakukan akses dalam berinteraksi melalui foto, teks, video, suara, animasi serta grafik. Web secara fisik adalah kumpulan komputer pribadi, web browser, koneksi ke ISP, komputer server, router dan [switch](#) yang dimanfaatkan untuk mengalirkan informasi yang kemudian akan menjadi wahana pertama bagi berbagai pihak terkait.

2.9 Kamus Data

Menurut Tata Surbani (2007:172), mendefinisikan Kamus Data (*Data Dictionary*) sebagai katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Maka dapat diambil kesimpulan bahwa kamus data digunakan untuk membantu dalam pembuatan sistem agar dapat dimengerti aplikasi secara detail serta mengorganisasikan semua elemen data yang digunakan dalam sistem. Ada beberapa hal yang perlu diperhatikan dalam pembuatan kamus data diantaranya adalah sebagai berikut :

1. Apa semua aliran data dan penyimpanan dalam DFD sudah didefinisikan dalam kamus data.
2. Apakah semua komponen elemen data sudah didefinisikan dengan baik.

3. Apakah semua notasi atau simbol yang digunakan pada kamus data sudah di koreksi.
4. Apakah elemen-elemen data dalam kamus data tidak menjelaskan sesuai dalam DFD/*entity relational*.

Adapun simbol-simbol yang digunakan dalam kamus data diantaranya:

Tabel II.1. Simbol-Simbol Kamus Data

SIMBOL	KETERANGAN
=	Terdiri dari
+	Dan
()	Pilih(boleh ya atau tidak)
{ }	Inters atau pengulangan proses
[]	Pilih salah satu pilihan
	Pemisah pilihan didalam tanda []
*	Kerangan atau catatan
@	Petunjuk (key field)

2.10 Database

Database merupakan *representasi* kumpulan fakta yang saling berhubungan disimpan secara bersama sedemikian rupa dan tanpa pengulangan (*redudensi*) yang tidak perlu, untuk memenuhi berbagai kebutuhan. Data perlu disimpan dalam database untuk keperluan penyediaan informasi lebih lanjut. Data didalam database perlu diorganisasikan sedemikian rupa, agar informasi yang dihasilkan berkualitas.

Menurut Irmansyah (2003:2), “Database merupakan kumpulan dari item dari data yang saling berhubungan satu dengan yang lainnya yang diorganisasikan berdasarkan sebuah skema atau struktur tertentu, tersimpan di *hardware* komputer dan dengan *software* untuk melakukan manipulasi untuk kegunaan tertentu”.

Organisasi database yang baik juga berguna untuk efisiensi kapasitas penyimpanannya. Prinsip utama database adalah pengaturan data dengan tujuan utama fleksibilitas dan kecepatan pada saat pengambilan data kembali. Adapun beberapa ciri dari database diantaranya adalah sebagai berikut:

1. Efisiensi meliputi kecepatan, ukuran dan ketepatan.
2. Data dalam jumlah besar.
3. Berbagi pakai (dipakai bersama-sama/*sharebility*)
4. Mengurangi bahkan menghilangkan terjadinya duplikasi dan ketidak konsistenan data.

Menurut Abdul Kadir dan Tera Ch.Triwahyuni (2005:484), menyatakan bahwa Basis Data adalah “suatu pengorganisasian sekumpulan data yang saling terkait sehingga memudahkan aktifitas untuk memperoleh informasi”. Sebagai contoh, basis data akademis, mengandung table-tabel yang berhubungan dengan data mahasiswa, dan jurusan, data mata kuliah, dan data yang diperoleh oleh mahasiswa.

Database mempunyai beberapa kriteria penting yang harus dipenuhi yaitu sebagai berikut :

1. Berorientasi pada data (*Data Oriented*) dan bukan berorientasi pada program (*Program Oriented*) yang akan menggunakannya.

2. Data yang digunakan oleh pemakai yang berbeda-beda atau beberapa program aplikasi tanpa perlu mengubah basis data.
3. Data dapat berkembang dengan mudah dan baik volume maupun strukturnya.
4. Data yang ada dapat memenuhi kebutuhan sistem-sistem baru secara mudah.
5. Dapat digunakan dengan cara yang berbeda-beda.
6. Kerangkapan data (*Data Redudancy*) minimal.

2.11 Data Flow Diagram (DFD)

Data Flow Diagram (DFD) adalah suatu model logika atau proses yang digunakan untuk menggambarkan atau membuat model dari asal dan kemana tujuan data yang dikeluarkan dari sistem yang dibuat, dimana data disimpan, proses apa saja yang dihasilkan dari data tersebut dan interaksi antara data yang tersimpan serta proses yang dikenakan pada proses tersebut. Dengan menggunakan Data Flow Diagram (DFD) sistem dapat digambarkan mulai dari level yang paling tinggi hingga diuraikan ke level yang lebih rendah.

Menurut Tata Surbani (2004:163), Data Flow Diagram didefinisikan sebagai suatu jaringan yang menggambarkan suatu sistem komputerisasi, manipulasi atau gabungan dari keduanya yang penggambarannya disusun dalam bentuk kumpulan komponen sistem yang saling berhubungan sesuai dengan aturan mainnya.

Ada beberapa simbol yang digunakan dalam pembuatan atau penggambaran sistem dengan menggunakan Data Flow Diagram (DFD) diantaranya seperti pada tabel dibawah ini:

Tabel II.2 Simbol-Simbol Data Flow Diagram (DFD)

Simbol Model	Keterangan
<p>External Entity.</p> <p>Merupakan sebuah objek atau kesatuan yang berada di lingkungan luar sistem yang dapat berupa orang, department, organisasi atau sistem yang lain memberi (output) atau menerima (input).</p>
<p>Process</p> <p>Kegiatan atau kerja yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk kedalam proses untuk dihasilkan arus data yang akan keluar dari proses.</p>
<p>Data Store (Penyimpanan Data)</p> <p>Merupakan tempat penyimpanan data.</p>
<p>Data Flow (Arus Data)</p> <p>Menunjukkan data yang mengalir melalui sistem. Dimulai dengan sebagian input dan diolah atau diproses menjadi output.</p>

2.12 Flowchart Document/Flowmap

Flowchart Document/Flowmap adalah penggambaran secara fisik dari langkah-langkah dan urutan prosedur dari suatu program. *Flowmap* membantu analis dan programmer untuk memecahkan masalah kedalam segmen yang lebih kecil dan menolong dalam menganalisis alternative pengoperasian. Biasanya *Flowmap*

mempermudah penyelesaian suatu masalah khususnya masalah yang perlu dipelajari dan dievaluasi lebih lanjut.

Untuk membuat suatu analisis menggunakan *Flowmap* seorang analisis dan programmer memerlukan beberapa tahapan diantaranya:

1. *Flowmap* digambarkan dari halaman atas ke bawah dan dari kiri ke kanan.
2. Aktivitas yang digambarkan harus didefinisikan secara hati-hati dan definisi ini harus dapat dimengerti oleh pembacanya.
3. Kapan aktivitas dimulai dan berakhir harus ditentukan secara jelas.
4. Setiap langkah dari aktivitas harus diuraikan dengan menggunakan deskripsi kata kerja.
5. Setiap langkah dari aktivitas harus berada pada urutan yang benar.
6. Lingkup dari range dari aktivitas yang sedang digambarkan harus ditelusuri dengan hati-hati. Percabangan-percabangan yang memotong aktivitas yang sedang digambarkan tidak perlu digambarkan pada *Flowmap* yang sama. Simbol konektor harus digunakan dan percabangannya diletakkan pada halaman yang terpisah atau hilangkan seluruhnya bila percabangannya tidak berkaitan dengan sistem.
7. Gunakan simbol-simbol *Flowmap* yang standart.

Tabel II.3 Simbol-Simbol Dalam Flowchart Document/Flowmap

No	Simbol	Arti
1		Formulir, berfungsi sebagai media output dalam bentuk dokumen
2		Tampilan ke monitor
3		Proses ke computer
4		Proses manual
5		File
6		Disk/Databse
7		Arsip
8		Komunikasi jarak jauh
9		Pilihan
10		Konektor antar halaman

Menurut penelitian tentang *Flowchart Document/Flowmap* (Fairuzelsaid.wordpress.com/tag/), diperoleh bagan alir dokumen (*Document Flowchart*) atau disebut juga bagan alir formulir (*Form Flowchart*) atau *Paperwork Flowchart* merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan-tembusannya.

2.13 SQL (*Structured Query Language*) Server 2008

Menurut Andi (2010:2), menyatakan bahwa SQL (*Structured Query Language*) Server adalah sebuah DBMS (*Database management sistem*) yang dibuat oleh Microsoft untuk ikut berkecimpung dalam persaingan dunia pengolahan data menyusul pendahulunya seperti Oracle. SQL (*Structured Query Language*) Server dibuat pada saat kemajuan dalam bidang *Hardware* sedemikian pesat. Oleh karena itu sudah dapat dipastikan bahwa SQL (*Structured Query Language*) Server membawa beberapa terobosan dalam bidang pengolahan penyimpanan data.

2.14 ASP.NET (*Active Server Pages.Networking*)

Menurut Edison Siregar (2007:1), menyatakan bahwa ASP.NET (*Active Server Pages.Networking*) adalah sebuah sarana untuk membangun halaman-halaman web dengan menggunakan teknologi.NET (*Networking*). Dengan teknologi ini, pengembang akan dengan mudah membangun sebuah *web sites* yang kompleks. ASP.NET (*Active Server Pages.Networking*) ditujukan terutama untuk membangun halaman-halaman yang interaktif yang memungkinkan pengguna berinteraksi dengan cara yang sangat mudah dan menakjubkan.

Sebelum menggunakan ASP.NET (*Active Server Pages.Networking*) sebagai sarana untuk membangun *web sites*, akan lebih baik bila terlebih dahulu memahami konsep-konsep dasar teknologi internet.

2.15 Visual Studio 2010

Microsoft Visual Studio 2010 adalah alat penting untuk individu melakukan tugas-tugas pembangunan dasar. Ini menyederhanakan penciptaan, debugging, dan penyebaran aplikasi pada berbagai platform, termasuk SharePoint dan Cloud. Visual Studio 2010 dilengkapi dengan dukungan terpadu untuk pengembangan uji-didorong, serta alat debugging yang membantu memastikan solusi berkualitas tinggi.

1. Form

Form Designer merupakan suatu objek yang digunakan untuk merancang tampilan program. Form Designer juga dapat dikatakan sebagai objek utama pada pemrograman Visual Basic karena pada form inilah nantinya komponen dan kontrol Toolbox diletakan dan diatur sebgus mungkin. Berikut merupakan gambar Form Designer :

Gambar II.1 Form Designer

2. Toolbars

Toolbars pada aplikasi windows lainnya yang berisi tombol–tombol yang mewakili suatu perintah tertentu yang sering digunakan untuk keperluan dalam pemrograman dan lain-lain, toolbars dapat terlihat dalam bentuk icon. Berikut merupakan gambar dari Toolbars :

Gambar II.2 Toolbars

3. Menu Bar

Menu bar yang terdapat pada program– program aplikasi di Windows. Menu Bar digunakan untuk melakukan proses atau perintah- perintah tertentu. Menu bar dibagi menjadi beberapa pilihan sesuai dengan kegunaan nya, seperti menu bar File digunakan untuk memproses atau menjalankan perintah-perintah yang berhubungan dengan file, seperti membuka file baru, menyimpan file. selain itu juga terdapat Menu Bar lain seperti : Edit, View, Project, Build, Debug, Data, Format, Tools, Window, dan Help. Berikut merupakan gambar dari Menu Bar :

Gambar II.3 Menu Bar

4. Tittle Bar

Tittle Bar adalah tempat untuk menampilkan nama project yang sedang dibuat. Berikut merupakan gambar dari Tittle Bar :

Gambar II.4 Tittle Bar

5. Toolbox

ToolBox merupakan tempat penyimpanan kontrol-kontrol atau komponen standar yang nantinya akan diletakkan sebagai komponen program didalam Form saat merancang sebuah aplikasi. Toolbox adalah tempat dimana kontrol dan komponen yang dilambangkan dengan icon. Kontrol dan komponen sangat membantu pada saat proses merancang tampilan Form dalam pembuatan program. Berikut merupakan gambar dari Toolbox:

Gambar II.5 Toolbox

6. Solution Explorer

Solution Explorer adalah jendela yang menyimpan Informasi mengenai Solution, Project – project, beserta file-file, form – form ataupun resource yang digunakan pada program aplikasi. Berikut merupakan gambar dari Solution Explorer :

Gambar II.6 Solution Explorer

7. Properties

Jendela Properties berfungsi untuk memberikan informasi mengenai objek yang sedang aktif dan nama objek yang sedang aktif. Berikut merupakan gambar dari Properties:

Gambar II.7 Properties

BAB III

ANALISIS SISTEM BERJALAN

3.1 Tinjauan Perusahaan

3.1.1 Sejarah Singkat Matahari Department Store.Tbk Kota Binjai

Grup Matahari adalah peritel multi-format terkemuka modern Indonesia dengan bisnis ritel inti dalam *fashion* dan belanja rumah tangga yang ditargetkan untuk konsumen menengah keatas. Matahari didirikan pada tahun 1958 oleh pendirinya, seorang legenda hidup Indonesia ritel – Mr Hari Darmawan, yang reputasi dan pengalamannya sangat terkenal dipasar domestic dan internasional, dan merupakan pengusaha dihormati dalam industri ritel. Pada tahun 1996, kepemilikan mayoritas Matahari berubah tangan dari Bapak Darmawan untuk LIPPO GROUP – sebuah multi-bisnis terkemuka di Indonesia. Transisi kepemilikan membawa Matahari ke struktur baru dan strategi bisnis baru.

Saat ini dipimpin oleh Benjamin J. Mailool, mantan banker dan CEO dari bisnis property, sebagai Presiden Direktur & CEO, untuk menjadikan Matahari sebagai peritel modern terkemuka di Indonesia dalam semua aspek ritel. Dengan tujuan utama untuk menjadi pemimpin ritel di Indonesia, Matahari telah melakukan langkah-langkah inovatif untuk memperluas, mengintegrasikan dan menciptakan permintaan untuk berbagai kebutuhan ritel dan terutama berfokus pada bidang persediaan *fashion*, makanan dan minuman, kesehatan dan kecantikan dan pusat hiburan. Matahari terus memperluas bisnis inti dengan unit terkait lainnya seperti pusat distribusi dan saluran, serta kartu loyalitas pelanggan.

“Kualitas Produk dan Layanan Langsung dari Hati kami” adalah deklarasi yang tulus yang bekerja untuk menunjukkan cara matahari melakukan bisnis dan kehadiran seluruh rantai toko. Seperti matahari terus menggalang dukungan dan loyalitas dengan kehadiran pertumbuhan regional, kualitas yang semakin baik untuk berdiri menjadi yang terbaik dan berusaha menjadi toko yang konsisten. Matahari adalah jaringan toko yang tak tertandingi dan telah tersebar di lebih dari 50 kota di Indonesia, dengan pelayanan yang besar dan yang paling penting, pemahaman tentang apa yang dibutuhkan untuk menyenangkan setiap pelanggan.

Sebagai sebuah rantai ritel terkemuka yang membanggakan diri pada menawarkan produk terbaik dan melayani pelayanan terbaik, dalam hubungan pelanggan dan asosiasi pemasok, sehingga memberikan pesan sehat dari pilihan produk lezat dan belanja sempurna yang hanya dapat dialami pada setiap toko Matahari. Matahari bangga menyajikan pelayanan terbaik dalam menyediakan kebutuhan sehari-hari masyarakat Indonesia, terutama untuk kebutuhan *fashion* dan makanan, melalui toko nasional dalam tiga format bisnis utama; *Matahari Department Store*, *Hypermart* dan *Foodmart*. Matahari telah meningkat secara substansial toko di Indonesia dengan total 103 *department store*, 39 *hypermarket*, 29 *supermarket*, 46 pusat kesehatan dan kecantikan lebih dari 90 pusat hiburan keluarga aktif beroperasi secara nasional pada akhir kuartal pertama 2008.

Kantor pusat *Matahari Department Store* berbasis di Lippo Karawaci, Tangerang Indonesia. Namun, rantai toko kami meliputi kota-kota besar di seluruh Indonesia, dengan reputasi yang mengesankan di dalam negeri dan internasional termasuk pemberian “*Gold Award*” bergengsi di Pasifik Ritel Asia Top 500 Awards

2007, 2006, 2005 & 2004. Selain itu, Matahari juga menerima penghargaan paling bergengsi untuk pertama kalinya “*The most 2007 Retail Asia Pacific Top 500 awards*”, yang merupakan pengakuan tertinggi dalam industri ritel di wilayah Asia Pasifik untuk perusahaan berprestasi yang diakui secara internasional. Ini adalah pengakuan yang tegas menempatkan Matahari pada perusahaan ritel terkemuka dan paling dinamis di kawasan itu.

Matahari Binjai Supermall yang merupakan salah satu dari bagian PT. Matahari Department Store Tbk, didirikan pada tanggal 12 September 2007. Berdiri sekitar 5 tahun Matahari Binjai Supermall mampu menjadi *Department Store* yang terkemuka khususnya di Kota Binjai.

3.1.1.1 Visi PT. Matahari Department Store. Tbk

“*Consumers’ Most Preferred Retailer*” adalah Visi dari PT. Matahari Department Store. Tbk yang artinya adalah menjadi peritel pilihan utama konsumen.

3.1.1.2 Misi PT. Matahari Department Store. Tbk

“*Consistently bringing value and fashionright products and services to enhance the consumers’ quality of lifestyle*” yaitu konsisten menawarkan berbagai ragam produk bernilai dan tepat guna dengan pelayanan terbaik guna peningkatan kualitas dan gaya hidup konsumen.

3.1.1.3 Filosofi Matahari Departement Store.Tbk

Berikut adalah filosofi PT.Matahari Departement Store.Tbk :

1. PT.Matahari Departement Store.Tbk berusaha menciptakan tingkat hidup yang lebih baik bagi seluruh karyawan.
2. PT.Matahari Departement Store.Tbk berusaha menciptakan tempat kerja yang aman, nyaman, tentram, dan sejahtera sebagai pancaran cita-cita karyawan.
3. PT.Matahari Departement Store.Tbk berusaha menciptakan sistem organisasi terpadu demi masa depan perusahaan dan karyawan atas dasar efisiensi kerja yang maksimal.
4. PT.Matahari Departement Store.Tbk berusaha mendidik, melatih dan mengembangkan seluruh karyawan yang merata tanpa membedakan tradisi, agama, asal keturunan, sadar akan tugas dan kewajiban, menjunjung tinggi tujuan perusahaan sebagai penunjang perekonomian negara.
5. PT.Matahari Departement Store.Tbk berharap atas dasar sinkronisasi, saling percaya mempercayai, hormat menghormati, kerjasama yang baik dengan azas kekeluargaan untuk mencapai kemajuan yang kekal abadi.

3.1.2 Struktur Organisasi dan Fungsi

Gambar III.1 Struktur Organisasi PT.Matahari Department Store Kota Binjai

Sumber : PT.Matahari Department Store.Tbk Kota Binjai Supermall (2013)

Berikut merupakan tugas dan tanggung jawab pada masing-masing bidang di PT.Matahari Department Store.Tbk Kota Binjai yang terdiri dari :

1. Store Manajer

Adapun uraian dan tanggung jawab dari seorang store manager adalah :

- a. Mengawas dan memantau seluruh kegiatan karyawan Matahari mulai Assisten Manager, Supervisor, dan Koordinator.
- b. Memantau pencapaian sales pada tiap harinya harus sesuai dengan target yang telah ditentukan oleh kantor pusat.
- c. Bertanggung jawab atas seluruh asset yang ada di Matahari binjai yang diarahkan kepada department Loss Prevention.

2. Assisten Store Manager

Adapun uraian tugas dan tanggung jawab dari seorang Assisten Store Manager adalah :

- a. Bertanggung jawab atas pelaksanaan kegiatan oprasional pada setiap world (Ladies, Mens, Children, Shoes, Youth, Home, Intimate), Bazaar dan Suporting Unit (EDP, Ekspedisi, Kasir, Adm Personalia, Adm Toko, Scurity, Teknisi, Monitor).
- b. Bertanggung jawab atas kinerja dari seorang supervisor dan koordinator.
- c. Mampu mengetahui complain customer yang terjadi, dan mampu melakukan *action plain & follow up* untuk meminimalisir complain tersebut.
- d. Menjadi *Closing Duty*.

3. Supervisor HRD (*Human Resource Development*)

Adapun uraian tugas dan tanggung jawab dari seorang Supervisor HRD adalah :

- a. Melakukan pendataan karyawan, administrasi gaji dan lembur, jamsostek, penerimaan karyawan, training, mengatur budget operasional toko.
- b. Melakukan kegiatan yang berkaitan dengan pengontrolan kehadiran karyawan.
- c. Menjadi *Manager Of Duty* (MOD), *Opening Duty*, dan *Closing Duty*.

4. Supervisor Kasir

Adapun uraian tugas dan tanggung jawab dari seorang Supervisor Kasir adalah :

- a. Bertanggung jawab dalam kegiatan operasional di area kasir, CSO (*customer service officer*) yang dilaksanakan oleh supervisor kasir, serta memberikan pengarahan, bimbingan serta evaluasi kerja kepada seluruh team kasir pada tiap harinya.
- b. Mengkoordinir pengelolaan keuangan dalam mendukung operasional kasir di toko, mengelola uang dan voucher di brankas.
- c. Menjadi *Manager Of Duty* (MOD), *Opening Duty*, dan *Closing Duty*.

5. Supervisor Ekspedisi

Adapun uraian tugas dan tanggung jawab dari seorang Supervisor Ekspedisi adalah :

- a. Pemegang jabatan ini bertugas mengorganisir pekerjaan di ekspedisi yang meliputi dalam penerimaan barang dari supplier yang harus sesuai dengan apa yang diminta (po, faktur) seta memiliki dokumen yang sah.
- b. Mengelola sumber daya manusia di ekspedisi seperti melakukan *briefing* karyawan setiap harinya, memberiklan pengarahan bimbingan, teguran, evaluasi dan pelatihan kepada karyawan di ekspedisi.
- c. Menjadi *Manager Of Duty* (MOD), *Opening Duty*, dan *Closing Duty*.

6. Supervisor Area

Adapun uraian tugas dan tanggung jawab dari seorang Supervisor Area adalah :

- a. Pemegang jabatan ini bertugas mengorganisir pekerjaan di area yang meliputi dalam penyusunan dan stok barang.
- b. Mengelola sumber daya manusia di area seperti melakukan *briefing* karyawan setiap harinya, memberiklan pengarahan bimbingan, teguran, evaluasi dan pelatihan kepada karyawan di area.
- c. Menjadi *Manager Of Duty* (MOD), *Opening Duty*, dan *Closing Duty*.

7. Supervisor Visual

Adapun uraian tugas dan tanggung jawab dari seorang Supervisor Visual adalah :

- a. Memastikan seluruh promosi toko yang berkaitan dengan visual merchandising diseluruh area toko dalam keadaan sudah terpasang dan mendukung aktifitas kerja sepanjang hari.
- b. Bertanggung jawab terhadap performance dekorasi toko pada saat event promosi berlangsung.
- c. Menjadi *Manager Of Duty* (MOD), *Opening Duty*, dan *Closing Duty*.

8. Koordinator Kasir

Adapun uraian tugas dan tanggung jawab dari seorang Koordinator Kasir adalah :

- a. Melakukan pengawasan dalam kegiatan oprasional diarea kasir.
- b. Membuat laporan bulanan keuangan pada setiap transaksi pembelian *Customer* yang nantinya harus melalui persetujuan supervisor kasir.

9. Koordinator Area

Adapun uraian tugas dan tanggung jawab dari seorang Koordinator Area adalah :

- a. Melakukan pengawasan dalam kegiatan oprasional diarea.
- b. Membuat laporan persediaan barang diarea yang nantinya akan dilaporkan kepada supervisor area.

10. Adm HRD & Adm Personalia

Adapun uraian tugas dan tanggung jawab dari seorang Adm HRD & Adm Personalia adalah :

- a. Mengatur absensi karyawan.
- b. Mendata dan menghitung lembur karyawan.
- c. Membuat data laporan karyawan setiap harinya.
- d. Mencatat surat masuk dan surat keluar.
- e. Mencatat dan mendata biaya oprasional toko.

11. EDP (*electronic data processing*)

Adapun uraian tugas dan tanggung jawab dari seorang EDP adalah :

- a. Bertugas melakukan pengontrolan dan perawatan fisik,perlengkapan dan peralatan elektrik, mekanik, dan electrical data processing (edp).
- b. Memberikan laporan penjualan setiap harinya kepada masing-masing brand diarea.

12. Monitor (*Information*)

Adapun uraian tugas dan tanggung jawab dari seorang Monitor (*Information*) adalah :

- a. Menginformasikan acara-acara yang sedang berlangsung ditoko melalui audio promo.

- b. Membagikan catalogue kepada customer yang datang ke CSO(*customer service officer*) dan memberikan penjelasan singkat mengenai penawaran khusus yang ada di catalogue.
- c. Menawarkan produk-produk yang ada di CSO(*customer service officer*) seperti kartu matahari MCC(*master club card*) menjelaskan secara singkat dan melakukan transaksi administrasi kalau customer bersedia.

13. Adm Ekspedisi & Pelaksana Ekspedisi

Adapun uraian tugas dan tanggung jawab dari seorang Adm Ekspedisi & Pelaksana Ekspedisi adalah :

- a. Seorang adm ekspedisi memiliki tugas untuk melakukan penginputan RTV(*retur to vendor*), RTW(*retur to warehouse*), RMPB(*rekap masalah penerimaan barang*), LSPB(*laporan selisih penerimaan barang*), LPB(*laporan penjualan bulanan*).
- b. Seorang pelaksana ekspedisi mengecek barang masuk dari supplier yang harus sesuai dengan apa yang diminta (po,faktur).

14. Security

Adapun uraian tugas dan tanggung jawab dari seorang Security adalah :

- a. Mengkordinir keluar masuk karyawan .
- b. Menjaga kondisi toko agar tetap aman.
- c. Mengawasi toko dari pencurian baik itu pihak internal maupun eksternal.

15. Teknisi

Adapun uraian tugas dan tanggung jawab dari seorang Teknisi adalah :

- a. Bertugas dalam penanganan instansi toko yang terdiri dari listrik,air,dan lainnya.
- b. Penanganan terhadap mesin yang ada di toko harus dalam keadaan siap guna.

20. Pelaksana Visual

Adapun uraian tugas dan tanggung jawab dari seorang Pelaksana Visual adalah :

- a. Memajang promosi toko yang berkaitan dengan acara yang sedang berlangsung diseluruh area .
- b. Menyusun *performance* dekorasi toko pada saat *event* promosi berlangsung.

21. Kasir

Adapun uraian tugas dan tanggung jawab dari seorang kasir adalah :

- a. Menyapa customer, dan memberikan pelayanan saat transaksi pembayaran.
- b. Menyampaikan acara yang berlangsung ditoko baik dalam pemberitahuan diskon dan maupun MCC (*matahari club card*).

22. Pramuniaga,SPG/SPB,BA

Adapun uraian tugas dan tanggung jawab dari seorang Pramuniaga,SPG/SPB,BA adalah :

- a. Mengecek barang masuk dari supplier.

- b. Melakukan pemajangan di area dan merapikanya.
- c. Membuat laporan stok barang baik di area maupun di gudang.
- d. Melayani customer dengan pelayanan yang baik sesuai standart matahari department store.

3.2 Prosedur Sistem Berjalan

Prosedur sistem berjalan untuk mengetahui laporan penjualan bulanan di PT.Matahari Department Store.Tbk Binjai Supermall adalah sebagai berikut :

1. Bagian EDP(*electronic data processing*) mengecek ke AlphaPos untuk merekap data penjualan bulanan dan mencetak laporan, namun data penjualan yang bisa dilihat di AlphaPos hanya yang ada di toko tersebut dan tidak dapat melihat di toko lain.
2. Kemudian data diberikan kepada Assisten Manager dan Store Manajer.

3.3 Diagram Alir Data Sistem Berjalan

Untuk lebih memperjelas sistem laporan penjualan yang berjalan pada PT.Matahari Department Store.Tbk Binjai, dibawah ini Diagram Alir Data yang dapat ditunjukkan pada Tabel III.1.

Tabel III.1 Flow Map Sistem Berjalan Pada Matahari Department Store.Tbk

3.4 Kamus Data Sistem Berjalan

Adapun kamus data sistem berjalan pada PT.Matahari Department Store.Tbk

Binjai adalah sebagai berikut :

1. Data Toko

@Kd_Toko+Nm_Toko

2. Data Barang

@Sku+Nm_Barang+Department+Harga

3. Data Penjualan

@Tgl_Penjualan+Sku+Kd_Toko+Qty+Harga

3.5 Spesifikasi Sistem Berjalan

3.5.1 Spesifikasi Bentuk Dokumen Masukan

Adapun spesifikasi bentuk dokumen masukan adalah sebagai berikut :

Gambar III.2 Dokumen Masukan Penjualan

3.5.2 Spesifikasi Bentuk Dokumen Keluaran

Adapun spesifikasi bentuk dokumen keluaran adalah sebagai berikut :

Gambar III.3 Dokumen Keluaran Laporan Penjualan

3.5.3 Spesifikasi File

Adapun spesifikasi file yang ada dalam sistem laporan penjualan yang sedang berjalan pada PT.Matahari Department Store.Tbk Binjai adalah sebagai berikut :

Tabel III.2 Tabel Data Toko

No	Field Name	Type	Size
1	Kd_Toko	Text	3
2	Nm_Toko	Text	30

Tabel III.3 Tabel Data Barang

No	Field Name	Type	Size
1	Sku	Text	8
2	Nm_Barang	Text	30
3	Department	Number	4
4	Harga	Currency	Currency

Tabel III.4 Tabel Data penjualan

No	Field Name	Type	Size
1	Tgl_Penjualan	Date/Time	General
2	Sku	Text	8
3	Kd_Toko	Text	3
4	Qty	Number	4
5	Harga	Currency	Currency
6	Total	Currency	Currency

3.5.4 Struktur Kode

Dibawah ini adalah struktur kode sistem berjalan pada PT.Matahari
Department Store.Tbk.

1. Barang
 - 14607866 : Kemeja Formal
 - 14694736 : Boneka
 - 14651335 : Koper
 - 14777576 : Sepatu Sport
 - 14577023 : Baju Tidur Wanita
2. Toko
 - 309 : Binjai Supermall
 - 285 : Palladium
 - 261 : Medan Fair
 - 553 : Thamrin plaza
 - 645 : Medan Mall

3.5.5 Spesifikasi Program

Dibawah ini adalah spesifikasi program yang berjalan pada PT.Matahari
Department Store.Tbk Binjai.

1. Software
 - a. Paket Program : Microsoft Office Excel 2003
 - b. Sistem Operasi : Microsoft Windows XP pack 2

2. Hardware

- a. Processor : Intel Pentium IV 3.0 Gh
- b. Hardisk : 80 GB
- c. RAM : 1 GB
- d. Mouse : Optic
- e. Keyboard : Standart

3.6 Permasalahan

Permasalahan yang terjadi pada sistem laporan penjualan PT.Matahari Department Store.Tbk, hanya rekapan penjualan bulanan pada tiap-tiap toko tersebut dan tidak dapat melihat penjualan pada toko lain dimana hal itu bisa digunakan sebagai acuan untuk mengukur *performance* penjualan pada masing-masing toko, yang nantinya para Store Manager akan mengatur strategi untuk meningkatkan penjualan hingga melebihi target yang telah ditentukan oleh HO (*Head Office*).

3.7 Alternatif Pemecahan Masalah

Sistem yang berjalan pada PT.Matahari Department Store.Tbk, masih perlu dilakukan perubahan yaitu dengan membuat sistem pemetaan penjualan dengan memanfaatkan GOOGLE MAP. Untuk memetakan daerah pada toko-toko lain yang sudah tersedia informasi penjualan yang dapat diakses bebas oleh Assisten Manager dan Store Manajer masing-masing toko sebagai dasar pengambilan keputusan.

BAB IV

RANCANGAN SISTEM USULAN

4.1 Prosedur Sistem Usulan

Berikut ini merupakan prosedur yang diusulkan lebih ditekankan dari prosedur program yang akan dibangun yaitu prosedur laporan penjualan. Adapun prosedur sistem usulan yang diterapkan pada laporan penjualan di PT.Matahari Department Store.Tbk adalah sebagai berikut :

1. Prosedur Laporan Penjualan

Pada proses ini Store Manager masing-masing toko dapat melihat penjualan bulanan di alphapos, kemudian laporan penjualan tersebut dikirim via e-mail kepada administrator pusat yang saat ini bertempat di lippo karawachi,yang dimana pada e-mail tersebut disertakan pula data toko dan store managernya.

2. Prosedur Input Data Toko,Store Manager dan Penjualan.

Dalam hal ini administrator dapat mengambil laporan data toko, data penjualan dan data store manager masing-masing toko yang telah dikirim oleh para manager masing-masing cabang pada via e-mail tersebut, kemudian administrator menginput laporan tersebut ke sistem yang telah dibuat, yang dimana para manager masing-masing toko cabang nantinya dapat melihat penjualan toko lain yang tersebar di seluruh indonesia dengan memanfaatkan GOOGLE MAP.

4.2 Diagram Alir Data Sistem Berjalan

4.2.1 Diagram Konteks Sistem Usulan

Adapun diagram konteks sistem usulan dari Sistem Informasi Pemetaan Penjualan Retail pada PT.Matahari Department Store.Tbk adalah sebagai berikut :

Gambar IV.1 Diagram Konteks Sistem Usulan

4.2.2 Data Flow Diagram Level 0 Sistem Usulan

Gambar IV.2 Data Flow Diagram Level 0 Sistem Usulan

4.3 Kamus Data Sistem Usulan

Adapun kamus data sistem usulan pada PT.Matahari Department Store.Tbk adalah sebagai berikut :

1. Store Manager

@StoreManagerId + Nip + Nama + TglMasuk + Alamat + Email + Telp + Hp
+ Foto + Mime

2. Toko

@IdToko + KdToko + NamaToko + Email + Telp + Fax + LevelId +
@@StoreManagerId + Foto + Mime + Longitude + Latitude

3. Penjualan

IdTarget + @@IdToko + Bulan + Tahun + TargetBulanan +
PersentasePenjualan + Selisih + CapaiTarget

4.4 Spesifikasi Sistem Usulan

Sistem yang diusulkan adalah untuk proses pengolahan laporan penjualan pada masing-masing toko yang tersebar di seluruh Indonesia. Proses tersebut dibuat dalam rancangan form masukkan dan juga laporan serta menu utamanya. Penjelasan dari form masukkan dan laporannya akan diuraikan sedemikian rupa.

4.4.1 Spesifikasi Rancangan Dokumen Masukkan

Form input terdiri atas beberapa form yaitu Form Input Data Store Manager, Form Input Data Toko dan Form Input Data Penjualan. Bentuk seluruh form dapat dilihat pada gambar dibawah ini :

Form Input Data Store Manager	
Nip	FOTO
<input type="text"/>	
Tanggal Masuk	
<input type="text"/> <input type="button" value="Kalender"/>	
Nama	
<input type="text"/>	
Alamat	
<input type="text"/>	
Email	
<input type="text"/>	
Telepon/Hp	
<input type="text"/> / <input type="text"/>	
Foto	
<input type="button" value="Select"/>	
<input type="button" value="Simpan"/> <input type="button" value="Kembali"/>	

Gambar IV.3 Rancangan Form Input Data Store Manager

Form Input Data Toko	
Kode Toko	<div>LOGO</div>
<input type="text"/>	
Nama Toko	
<input type="text"/>	
Kelas	
<input type="text"/>	<input type="button" value="▼"/>
Store Manager	
<input type="text"/>	<input type="button" value="▼"/>
Alamat Toko	
<input type="text"/>	
Email	
<input type="text"/>	
Telefon/Fax	
<input type="text"/>	/ <input type="text"/>
Foto	
<input type="button" value="Select"/>	
Set Lokasi Toko	
<div>TAMPILAN PETA</div>	
<div>Simpan</div> <div>Kembali</div>	

Gambar IV.4 Rancangan Form Input Data Toko

Form Input Data Penjualan	
Bulan	
<input type="text"/>	<input type="button" value="▼"/>
Tahun	
<input type="text"/>	
Target Bulanan	
<input type="text"/>	<input type="button" value="▲"/> <input type="button" value="▼"/>
Penjualan	
<input type="text"/>	<input type="button" value="▲"/> <input type="button" value="▼"/>
<div>Simpan</div> <div>Kembali</div>	

Gambar IV.5 Rancangan Form Input Data Penjualan

4.4.2 Spesifikasi Rancangan Dokumen Keluaran

Adapun rancangan dokumen keluaran yang terdiri dari laporan data toko, laporan data store manager dan laporan data penjualan. Bentuk laporan dapat dilihat pada gambar dibawah ini :

<div> <div>Logo</div> <div>MDS Retail Mapping management</div> </div>		Log Off	
Home	Toko	Store Manager	Penjualan
<div>Grade</div> <div> ☐ A+ ☐ A ☐ A- ☐ B+ ☐ B ☐ B- ☐ C+ ☐ C ☐ C- ☐ D ☐ E ☐ F </div>	<div>Information Board Lingitude Latitude</div> <div>TAMPILAN PETA</div>		<div>Toko</div> <div>Tampilan Data Toko</div> <div>Manager</div> <div>Tampilan Data Manager</div> <div>Penjualan</div> <div>Tampilan Data Penjualan</div>
<div>Grafik</div> <div>TAMPILAN GRAFIK PENJUALAN</div>			

Gambar IV.6 Rancangan Laporan Data toko, Store Manager dan Penjualan

4.4.3 Normalisasi File

Normalisasi *file* dapat dijabarkan sebagai suatu proses pengelompokan elemen data ke dalam tabel yang menyatakan hubungan antar entitas sehingga terwujud suatu bentuk yang memudahkan adanya perubahan dengan dampak terkecil. Proses pada pengolahan data yang mengakibatkan efek samping yang tidak diharapkan sering disebut dengan istilah anomaly yang dapat terjadi akibat inserting,

updating atau deleting. Oleh karena itu diperlukan normalisasi *file*. Adapun teknik penyusunan normalisasi *file* adalah dengan menentukan hal-hal sebagai berikut :

1. Kunci Atribut

Setiap *file* selalu mempunyai kunci berupa satu *field* atau satu set *field* yang dapat mewakili *record*. Sebagai contoh pada table toko terdapat *field* kunci berupa 671 yang mewakili nama toko, alamat dan sebagainya.

2. Kunci Kandidat

Kunci Kandidat adalah satu atribut atau satu set minimal atribut yang mengidentifikasi secara unik suatu kejadian spesifik dari entiti. Jika kunci kandidat berisi lebih dari satu atribut, maka biasanya disebut sebagai kunci gabungan (*Composite Key*).

3. Kunci Primer

Kunci Primer adalah satu atribut atau satu set minimal atribut yang tidak hanya mengidentifikasi secara unik suatu kejadian spesifik tetapi juga dapat mewakili setiap kejadian dari sistem entiti. Setiap kunci kandidat mempunyai peluang untuk menjadi kunci primer, tetapi sebaiknya dipilih satu saja yang dapat mewakili secara menyeluruh terhadap entiti yang ada.

4. Kunci Alternatif

Kunci Alternatif adalah kunci kandidat yang tidak dipakai sebagai kunci primer. Biasanya kunci ini digunakan sebagai kunci pengurutan data saja, misalnya dalam pembuatan laporan.

5. Kunci Tamu

Kunci Tamu adalah satu atribut (atau satu *set atribut*) yang melengkapi satu hubungan yang menunjukkan ke tabel induknya. Kunci tamu biasanya ditempatkan pada entiti anak dan sama dengan kunci primer induk yang direlasikan. Hubungan antara entiti induk dengan anak adalah (biasanya) hubungan satu ke banyak (*one to many*).

4.4.4 Spesifikasi File

Adapun spesifikasi file usulan dalam sistem laporan penjualan pada PT.Matahari Department Store.Tbk adalah sebagai berikut :

Tabel IV.1 Tabel Data Store Manager

No	Field Name	Type	Size
1	StoreManagerId	begint	-
2	Nip	nvarchar	50
3	Nama	nvarchar	100
4	TglMasuk	date	-
5	Alamat	nvarchar	100
6	Email	nvarchar	50
7	Telp	nvarchar	20
8	Hp	nvarchar	20
9	Foto	image	-
10	Mime	nvarchar	50

Tabel IV.2 Tabel Data Toko

No	Field Name	Type	Size
1	IdToko	begint	-
2	KodeToko	nvarchar	20
3	NamaToko	nvarchar	50
4	AlamatToko	nvarchar	100
5	Email	nvarchar	100
6	Telp	nvarchar	20
7	Fax	nvarchar	20
8	LevelId	int	-
9	StoreManagerId	begint	-
10	Foto	image	-
11	Mime	nvarchar	100
12	Longitude	float	-
13	latitude	float	-

Tabel IV.3 Tabel Data Penjualan

No	Field Name	Type	Size
1	IdTarget	begint	-
2	IdToko	begint	-
3	Bulan	int	-
4	Tahun	int	-
5	TargetBulanan	float	-
6	Penjualan	float	-
7	PersentasePenjualan	float	-
8	Selisih	float	-
9	CapaiTarget	bit	-

Adapun bentuk rancangan relasi antar tabel adalah sebagai berikut :

Gambar IV.7 Rancangan Relasi Antar Tabel

Keterangan :

* : Kunci Primer

** : Kunci Tamu

4.4.5 Struktur Kode

Pada dasarnya struktur pengkodean digunakan untuk mengklasifikasikan data, memasukkan data kedalam komputer dan untuk mengambil bermacam-macam informasi yang berhubungan denganya. Dalam hal ini, kode dapat berupa kumpulan angka, huruf atau karakter khusus.

1. Petunjuk Pembuatan Kode

a. Harus Mudah Diingat

Agar kode mudah diingat, dalam hal ini penulis menggunakan kode yang sudah ada pada PT.Matahari Department Store.Tbk yaitu :

671,337,309,267,285 dst.

b. Harus Unik

Penggunaan kode harus unik untuk masing-masing item yang mewakilinya. Unik berarti tidak ada terdapat 2 (dua) buah kode yang sama.

c. Harus *Fleksibel*

Penggunaan kode harus *fleksibel* sehingga memungkinkan perubahan-perubahan atau penambahan item baru dapat tetap diwakili oleh kode.

d. Harus *Efisien*

Penggunaan kode harus sependek mungkin, selain mudah diingat juga akan *efisien* bila direkam disimpan luar komputer. Misalnya panjang dari kode cukup sepanjang 5 digit saja dan tidak akan *efisien* bila dipergunakan kode yang lebih dari 5 digit.

e. Harus *Konsisten*

Bilamana mungkin, kode harus *konsisten* dengan kode yang telah digunakan, misalnya perubahan hanya membeli sepeda motor dari seorang penjual saja, maka dapat dipergunakan kode-kode barang yang sudah dipergunakan oleh penjual.

f. Harus *distandardisasi*

Kode harus *standardisasi* untuk seluruh tingkatan dan department dalam organisasi. Kode yang tidak standar akan melibatkan kebingungan, salah pengertian dan dapat cenderung terjadi kesalahan pemakaian bagi yang menggunakan kode tersebut. Misalnya kode untuk tanggal harus standar , yaitu tanggal, bulan dan tahun atau bulan, tanggal dan tahun.

2. Tipe Dari Kode

a. Kode *Mnemonic*

Pengkodean *mnemonic* digunakan untuk tujuan supaya mudah diingat. Kode *mnemonic* dibuat dengan dasar singkatan atau mengambil sebagian karakter dari item yang akan diwakili dengan kode ini. Misalnya kode “A+” untuk menyatakan toko yang berukuran besar. Kode “C” untuk menyatakan toko yang dengan ukuran kecil dan tipe kode yang lainnya.

b. Kode Urut

Pengkodean urut disebut juga dengan kode seri merupakan kode yang nilainya urut antara satu kode dengan kode berikutnya.

4.4.6 Spesifikasi Program

Adapun spesifikasi program usulan yang dirancang dapat dilihat pada gambar IV.8 sebagai berikut :

Gambar IV.8 Rancangan Spesifikasi Program

4.5 Hasil Program

4.5.1 Form Login

Form login merupakan form untuk memasukkan user name dan password pengguna aplikasi. Bentuk tampilan form login sistem yang dibangun dapat dilihat pada gambar dibawah ini :

Gambar IV.9 Tampilan Form Login

4.5.2 Form Menu Utama

Menu utama merupakan tampilan awal pada saat sistem dijalankan. Bentuk tampilan menu utama dapat dilihat pada gambar dibawah ini :

Gambar IV.10 Tampilan Form Menu Utama

4.5.3 Form Input Data Toko

Berikut merupakan form input data toko, digunakan oleh administrator untuk menambah data toko.

Gambar IV.11 Tampilan Form Input Data Toko

4.5.4 Form Input Data Store Manager

Berikut ini adalah tampilan form input data store manager, digunakan oleh administrator untuk menambah data store manager.

Gambar IV.12 Tampilan Form Input Data Store Manager

4.5.5 Form Input Data Penjualan

Berikut ini adalah tampilan form input data penjualan, digunakan oleh administrator untuk menambah data penjualan.

Gambar IV.13 Tampilan Form Input Data Penjualan

4.5.6 Form User Management

Berikut ini merupakan form user management, digunakan administrator untuk menambah password store manager dan administrator.

Gambar IV.14 Tampilan Form User Management

4.5.7 Laporan Data Toko

MDS RETAIL MAPPING MANAGEMENT									
TOKO									
FOTO	KODE TOKO	KELAS	NAMA TOKO	STORE MANAGER	ALAMAT TOKO	EMAIL	TELEPON	FAX	COMMAND
	671	A	MDS Angso Duo Jambi	Suruso	Jl.Gatot subroto No.53	Mds671@matahari.co.id	07417550512	07417550513	
	337	A	MDS Basko Padang	Mentivia Susendra	Jl.Prof Dr.Hamka No.2A	mds337@matahari.co.id	07514488877	07514488878	
	309	C	MDS Binjai Supermall	Jhonson Hutabarat	Jl.Soekarno Hatta no.14	mds309@matahari.co.id	0618821750	06188228862	
	267	A	MDS Ciputra Seraya	Felly Chandra K	Jl.Riau No.58 Pekanbaru	mds267@matahari.co.id	0761888500	0761888501	
	285	C	MDS Grand Palladium	Cahya Hestawati	Jl.Kapt.Maulana Lubis	mds285@matahari.co.id	0614512411	0614511944	
	369	B	MdS Hermez palace	Muhammad Basri	Jl.Hasan Dek Banda Aceh	mds369@matahari.co.id	0651350519	0651350510	
	507	A+	MDS I'NTL Plaza	Mulyana	Jl.Jendral Sudirman	mds507@matahari.co.id	0711350519	0711350510	
	361	A	MDS Kepri Batam	Nurulina Tumanggor	Jl.Jendral Sudirman	mds361@matahari.co.id	07787490004	07787490008	
	261	A+	MDS Medan Fair	Erwin Ilham Syarif	Jl.gatot Subroto no.23	mds261@matahari.co.id	0619879203	0619893044	
	645	A	MDS Medan Mall	Aolomaziduhu Zebua	Jl.MT-Haryono No.8-9	mds645@matahari.co.id	0618741334	0619893997	

Gambar IV.15 Tampilan Laporan Data Toko

4.5.8 Laporan Data Store Manager

MDS RETAIL MAPPING MANAGEMENT							
STORE MANAGER							
FOTO	NIP	NAMA	ALAMAT	EMAIL	TELEPON	HP	COMMAND
00001	Johnson Hutabarat	Jl.Setia Budi No.45 Medan	jhonson@matahari.co.id	0618821750	08163121333

00002	Aolomadziduhu Zebua	Jl.Krakatau No.44 Medan	kono@matahari.co.id	0610009879	081260244324

00003	Erwin Ilham Syarif	Jl.Gatot Subroto No.120 Medan	Erwin@matahari.co.id	0610987678	081376091899

00004	Polman Mannurung	Jl.karya No.27 Medan	Polman@matahari.co.id	0610987678	0811765190

00005	Cahya Hestiawati	Jl.Jamin Ginting No.220 Medan	Cahya@matahari.co.id	0614533214	08128305448

00006	Felly Chandra K	Jl.Soekarno Hatta No.22 Pekanbaru	Felly@matahari.co.id	0614875667	082155558765

00007	Hamidi Chaidir	Jl. Teunku Amir Hamzah No.44 Pekanbaru	Hamidi@matahari.co.id	0616654778	087845546788

00008	Agus Budiyanto	Jl. Ketilang No.265 Batam	Agus@matahari.co.id	0615564656	081397917087

00009	Andi	Jl.Budi Karya No.132 Batam	Andi@matahari.co.id	0618876654	081365567887

00010	Mentivia Susendra	Jl. Patimurah No.21 Padang	Mentivia@matahari.co.id	0616654878	085377664554

Gambar IV.16 Tampilan Laporan Data Store Manager

4.5.9 Laporan Data Penjualan

MDS RETAIL MAPPING MANAGEMENT							
SALES							
KODE TOKO	KELAS	NAMA TOKO	STORE MANAGER	ALAMAT TOKO	EMAIL	TELEPON	FAX
309	C	MDS Binjai Superall	Jhonson Hutabarat	Jl.soekarno Hatta.o14 Binjai	Mds671@matahari.co.id	07417550512	07417550513
261	A+	MDS Medan Fair	Erwin Ilham Syarif	Jalan Gatot Soebroto No.23 Medan	m337@matahari.co.id	07514488877	07514488878
285	C	MDS Grand Palladium	Cahya Hestiawati	Jl.Kapt Maulana Lubir No.8 Medan	m309@matahari.co.id	0618821750	06188228862
553	A+	MDS Thamrin Plaza	Polman Mannurung	Jl. Thamrin No.75 Medan	m267@matahari.co.id	0761888500	0761888501
645	A	MDS Medan Mall	Aolomadziduhu Zebua	Jl.MT-Haryono No.8-9 Medan	m285@matahari.co.id	0614512411	0614511944
267	A	MDS Ciputra Seraya	Felly Chandra K	Jl. Riau No.58 Pekanbaru	m369@matahari.co.id	0651350519	0651350510
273	A+	MDS SKA Pekanbaru	Hamidi Chaidir	Jl.Soekarno Hatta Tuanku-tambusai, PKU	m507@matahari.co.id	0711350519	0711350510
275	A	MDS Mega Mall Batam	Agus Budianto	Jl.Engku Putri Batam Center	m361@matahari.co.id	07787490004	07787490008
305	A	MDS Nagoya Hill Batam	Andi	Jl.Teunku Umar Nagoya Batam	m261@matahari.co.id	0619879203	0619893044
337	A	MDS Basko Padang	Mentiva Susendra	Jl.Prof DR.Hamka NO.2 Padang	m645@matahari.co.id	0618741334	0619893997

Gambar IV.17 Tampilan Laporan Data Penjualan

4.5.10 Laporan Data User Management

MDS RETAIL MAPPING MANAGEMENT				
User Management				
NIP	FULLNAME	EMAIL	ROLES	COMMAND
00001	Johnson Hutabarat	Jhonson@matahari.co.id	Manager
00002	Aolomadziduhu Zebua	Kono@matahari.co.id	Manager
00003	Erwin Ilham Syarief	Erwin@matahari.co.id	Manager
00004	Polman Mannurung	Polman@matahari.co.id	Manager
00005	Cahya Hestiawati	Cahya@matahari.co.id	Manager
00006	Felly Chandra K	Felly@matahari.co.id	Manager
00007	Hamidi Chaidir	Hamidi@matahari.co.id	Manager
00008	Agus Budiyanto	Agus@matahari.co.id	Manager
00009	Andi	Andi@matahari.co.id	Manager
00010	Mentivia Susendra	Mentiva@matahari.co.id	Manager

Gambar IV.18 Tampilan Laporan Data User Management

4.6 Spesifikasi Sistem Komputer

Spesifikasi yang dibutuhkan sistem komputer untuk menerapkan rancangan sistem dibagi atas tiga bagian yaitu pemakai sistem (umum), perangkat keras dan perangkat lunaknya. Penjelasan dari spesifikasi ini akan lebih dijelaskan berikut ini.

4.6.1 Umum

Setelah sistem dirancang maka dibutuhkan seorang programmer yang akan menerapkan kedalam suatu aplikasi bahasa pemrograman. Dalam kasus ini dibutuhkan seseorang programmer yang memiliki spesifikasi bahasa pemrograman web. Sistem yang dibangun akan digunakan oleh seseorang operator yang memiliki pengetahuan mengenai komputer dan bahasa pemrograman khususnya web.

4.6.2 Spesifikasi Perangkat Keras

Spesifikasi perangkat keras yang digunakan dalam pembuatan sistem yang diusulkan adalah sebagai berikut :

- a. Prosesor : Minimal Pentium IV dengan kecepatan 2.66 GHZ
- b. Hardisk : Minimal 10 GB
- c. RAM : 512 MB
- d. VGA Card : 8 MB
- e. Monitor : SVGA GIC 15 “Digital
- f. Keyboard : Standar
- g. Mouse : Scroll / Optic

4.6.3 Spesifikasi Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan dalam pembuatan sistem yang diusulkan adalah :

- a. Sistem Operasi : Sistem Operasi Windows Seven
- b. Paket Program : Visual Studio 2010
- c. *Database* : SQL Server 2008

4.7 Jadwal Implementasi

Pada dasarnya, rencana implementasi adalah suatu prosedur atau uraian mengenai tahapan yang akan dilakukan, dan menjelaskan segala sesuatu yang dibutuhkan dalam proses implementasi berupa komputerisasi sistem. Oleh sebab itu

dipandang perlu beberapa tahapan yang akan mempermudah proses implementasi yaitu sebagai berikut :

1. Menyediakan perangkat keras (*Hardware*) dan perangkat lunak (*Software*).
2. Menjelaskan tentang sistem, hal ini bertujuan untuk memberikan pengertian dan pemahaman kepada user tentang sistem ini supaya user dapat bekerja dengan optimal.
3. Pemeliharaan terhadap sistem. Tujuannya adalah agar sistem dapat dioperasikan dengan baik.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan yang telah penulis paparkan pada bab-bab sebelumnya, maka penulis dapat menarik beberapa kesimpulan sebagai berikut :

1. Dengan adanya sistem yang baru ini diharapkan dapat mempercepat dan memudahkan proses penyampaian dan penerimaan laporan penjualan pada masing-masing toko yang tersebar diseluruh indonesia.
2. Memaksimalkan keakuratan laporan dan sebagai sarana akses informasi kepada semua pihak manager masing-masing cabang terhadap laporan penjualan retail.
3. Dengan adanya sistem yang baru ini diharapkan dapat meningkatkan kualitas pengendalian terhadap pelaksanaan laporan penjualan bulanan pada masing-masing toko PT.Matahari Department Store.Tbk dan dapat membantu dalam penyajian informasi oleh para manager sebagai bahan pertimbangan keputusan untuk meningkatkan penjualan pada masing-masing toko.

5.2 Saran

Berdasarkan kesimpulan diatas maka penulis mengemukakan beberapa saran yang diharapkan dapat menjadi bahan pertimbangan bagi kemajuan sistem dimasa yang akan datang. Dan beberapa saran dari penulis adalah sebagai berikut :

4. Pengembangan terhadap sistem laporan penjualan bulanan pada masing-masing toko PT.Matahari Department Store.Tbk sangat diperlukan, karena semakin meningkatnya kebutuhan akan informasi pada saat ini. Oleh karena itu untuk mendukung kelancaran kerja pada PT.Matahari Department Store.Tbk diperlukan seseorang yang mampu untuk mengoperasikan sistem ini. Dalam hal ini harus ada yang bertindak sebagai operator data atau IT operator, supaya penerapan sistem ini dapat dilakukan secara maksimal.
5. Dengan adanya sistem ini diharapkan dapat menjadi acuan untuk mengembangkan sistem komputerisasi ke arah yang lebih baik.
6. Penulis menyadari bahwa dalam setiap pengembangan sistem informasi yang baru pasti selalu terdapat ketidaksempurnaan sistem, begitu juga dengan pengembangan sistem laporan penjualan retail PT.Matahari Department Store.Tbk ini. Namun dengan berkembangnya teknologi yang semakin pesat saat ini diharapkan nantinya sistem ini bisa lebih baik dan berkembang sesuai dengan kebutuhan di PT.Matahari Department Store.Tbk.

LISTING PROGRAM

1. Listing Program Form Login

```

Imports
System.Diagnostics.CodeAnalysis
Imports System.Security.Principal
Imports System.Web.Routing
Public Class AccountController
    Inherits
System.Web.Mvc.Controller
    '
    ' GET: /Account/LogOn
    Public Function LogOn() As
ActionResult
        Return View()
    End Function
    '
    ' POST: /Account/LogOn
    <HttpPost()> _
    Public Function LogOn(ByVal
model As LogOnModel, ByVal
returnUrl As String) As ActionResult
        If ModelState.IsValid Then
            If
Membership.ValidateUser(model.UserName, model.Password) Then
FormsAuthentication.SetAuthCookie
(model.UserName,
model.RememberMe)
                If
Url.IsLocalUrl(returnUrl) AndAlso
returnUrl.Length > 1 AndAlso
returnUrl.StartsWith("/") _
                    AndAlso Not
returnUrl.StartsWith("//") AndAlso
Not returnUrl.StartsWith("\") Then
                    Return Redirect(returnUrl)
                Else
                    Return
RedirectToAction("Index", "Home")
                End If
            Else
                ModelState.AddModelError("", "The
user name or password provided is
incorrect.")
                End If
            End If
            ' If we got this far, something
failed, redisplay form
            Return View(model)
        End Function
    '
    ' GET: /Account/LogOff
    Public Function LogOff() As
ActionResult
        FormsAuthentication.SignOut()
        Return
RedirectToAction("Index", "Home")
    End Function
    '
    ' GET: /Account/Register
    Public Function Register() As
ActionResult
        Return View()
    End Function
    '
    ' POST: /Account/Register
    <HttpPost()> _
    Public Function Register(ByVal
model As RegisterModel) As
ActionResult
        If ModelState.IsValid Then
            ' Attempt to register the user
            Dim createStatus As
MembershipCreateStatus
            Membership.CreateUser(model.UserName, model.Password,
model.Email, Nothing, Nothing,
True, Nothing, createStatus)
            If createStatus =
MembershipCreateStatus.Success
                Then

```

```

FormsAuthentication.SetAuthCookie
(model.UserName, False)
    Return
RedirectToAction("Index", "Home")
    Else
ModelState.AddModelError("",
    ErrorCodeToString(createStatus))
    End If
    End If
    ' If we got this far, something
failed, redisplay form
    Return View(model)
    End Function
'
' GET: /Account/ChangePassword
<Authorize()> _
Public Function
ChangePassword() As ActionResult
    Return View()
    End Function
'
' POST:
/Account/ChangePassword
<Authorize()> _
<HttpPost()> _
Public Function
ChangePassword(ByVal model As
ChangePasswordModel) As
ActionResult
    If ModelState.IsValid Then
        ' ChangePassword will throw
an exception rather
        ' than return false in certain
failure scenarios.
        Dim
changePasswordSucceeded As
Boolean
        Try
            Dim currentUser As
MembershipUser =
Membership.GetUser(User.Identity.
Name, True)
            changePasswordSucceeded
=
currentUser.ChangePassword(model.
OldPassword, model.NewPassword)

```

```

Catch ex As Exception
    changePasswordSucceeded
= False
    End Try
    If changePasswordSucceeded
Then
        Return
RedirectToAction("ChangePassword
Success")
    Else
ModelState.AddModelError("", "The
current password is incorrect or the
new password is invalid.")
    End If
    End If
    ' If we got this far, something
failed, redisplay form
    Return View(model)
    End Function
'
' GET:
/Account/ChangePasswordSuccess
Public Function
ChangePasswordSuccess() As
ActionResult
    Return View()
    End Function
#Region "Status Code"
Public Function
ErrorCodeToString(ByVal
createStatus As
MembershipCreateStatus) As String
    ' See
http://go.microsoft.com/fwlink/?Link
ID=177550 for
    ' a full list of status codes.
    Select Case createStatus
        Case
MembershipCreateStatus.DuplicateU
serName
            Return "User name already
exists. Please enter a different user
name."
        Case
MembershipCreateStatus.DuplicateE
mail

```

Return "A user name for that e-mail address already exists. Please enter a different e-mail address."

Case
MembershipCreateStatus.InvalidPassword

Return "The password provided is invalid. Please enter a valid password value."

Case
MembershipCreateStatus.InvalidEmail

Return "The e-mail address provided is invalid. Please check the value and try again."

Case
MembershipCreateStatus.InvalidAnswer

Return "The password retrieval answer provided is invalid. Please check the value and try again."

Case
MembershipCreateStatus.InvalidQuestion

Return "The password retrieval question provided is invalid. Please check the value and try again."

Case
MembershipCreateStatus.InvalidUsername

Return "The user name provided is invalid. Please check the value and try again."

Case
MembershipCreateStatus.ProviderError

Return "The authentication provider returned an error. Please verify your entry and try again. If the problem persists, please contact your system administrator."

Case
MembershipCreateStatus.UserRejected

Return "The user creation request has been canceled. Please verify your entry and try again. If the problem persists, please contact your system administrator."

Case Else
Return "An unknown error occurred. Please verify your entry and try again. If the problem persists, please contact your system administrator."

End Select
End Function
#End Region
End Class

2. Listing Program Menu Utama

```
Imports Telerik.Web.Mvc
Imports System.IO
<Authorize(Roles:="administrator,manager")>
Public Class HomeController :
Inherits System.Web.Mvc.Controller
    Private Property MapRepo As
New MapRepository
    Private Property ManagerRepo As
New StoreManagerRepository
    Private Property TokoRepo As
New TokoRepository
    Private Property TargetRepo As
New TargetRepository
    Private Property SessionGrade As
New List(Of Integer)
#Region "IO"
    Sub BuildJson(ByVal model As
List(Of MapModel))
        For Each f As Object In
System.IO.Directory.GetFiles(Server
.MapPath("~/Scripts/Json"))
            System.IO.File.Delete(f)
```



```

Next
Dim i As Integer = 1
Dim FILE_NAME As String =
Server.MapPath("~/Scripts/Json/Data.js")
Dim objWriter As New
System.IO.StreamWriter(FILE_NAME, True)
objWriter.WriteLine("var long
= 120;")
objWriter.WriteLine("var lat =
3.5888;")
objWriter.WriteLine("var
arrIconSize = [26, 46];")
objWriter.WriteLine("var
arrIconAnchor = [12, 46];")
objWriter.WriteLine("var
arrInfoWindowAnchor = [12, 0];")
objWriter.WriteLine("var
ActualData=[ ")
If model IsNot Nothing Then
    If model.Count <> 0 Then
        For Each item In model
objWriter.WriteLine("{")
objWriter.WriteLine("latitude:" &
item.latitude.ToString.Replace(",",
"."))
objWriter.WriteLine(",longitude:" &
item.longitude.ToString.Replace(",",
"."))
objWriter.WriteLine(",title:" &
Chr(34) & item.title & Chr(34))
objWriter.WriteLine(",html:" &
Chr(34) & item.html & Chr(34))
            If item.icon IsNot
Nothing Then
objWriter.WriteLine(",icon:")
objWriter.WriteLine("{")
objWriter.WriteLine("image:" &
Chr(34) & item.icon.image &
Chr(34) & ", shadow:" & Chr(34) &
item.icon.shadow & Chr(34) & ",
tags: " & item.icon.tags.ToString &
", iconsize: arrIconSize,iconanchor:
arrIconAnchor,infowindowanchor:
arrInfoWindowAnchor")

```

```

objWriter.WriteLine("}")
        End If
objWriter.WriteLine("}")
        If i <> model.Count
Then
objWriter.WriteLine(",")
        End If
        i += 1
        Next
    End If
End If
objWriter.WriteLine("];")
objWriter.Close()
End Sub
#End Region
#Region "Ajax"
Function GetImageToko(ByVal id
As Integer) As ActionResult
    Dim model =
TokoRepo.GetById(id)
    Dim imageByte As Byte() =
model.Foto
    Return File(imageByte,
model.Mime)
End Function
Function
GetImageManager(ByVal id As
Integer) As ActionResult
    Dim model =
ManagerRepo.GetById(id)
    Dim imageByte As Byte() =
model.Foto
    Return File(imageByte,
model.Mime)
End Function
Function ShowInformation(ByVal
id As Integer) As JsonResult
    Dim mapModel =
MapRepo.Json(id)
    Dim tokoModel =
TokoRepo.GetById(id)
    Dim managerModel =
ManagerRepo.GetById(tokoModel.S
toreManagerId)

```

```

        Dim penjualanModel =
        TargetRepo.list(id).OrderByDescending(Function(a) a.IdTarget).ToList
        Return Json(New With { .map =
        mapModel, .toko = tokoModel,
        .manager = managerModel,
        .penjualan = penjualanModel },
        JsonRequestBehavior.AllowGet)
    End Function
    Function _SelectChart(ByVal id
    As Integer) As JsonResult
        Dim model =
        TargetRepo.list(id)
        Return Json(model,
        JsonRequestBehavior.AllowGet)
    End Function
    'Function SelectGrade(ByVal id
    As Integer, ByVal value As Integer)
    As JsonResult
        ' If value = 1 Then
        '     SessionGrade.Add(id)
        '     Session("SessionGrade") =
        SessionGrade
        ' Else
        '     SessionGrade =
        DirectCast(Session("SessionGrade"),
        List(Of Integer))
        '     If SessionGrade IsNot
        Nothing Then
        '         If SessionGrade.Count <>
        0 Then
        '             SessionGrade.Remove(id)
        '             End If
        '         End If
        '     End If
        '     Dim i As Integer = 1
        '     Dim ctx As New MapEntities
        '     Dim str As New StringBuilder
        '     str.Append("SELECT*FROM
        dbo.Toko ")
        '     If SessionGrade IsNot Nothing
        Then
        '         If SessionGrade.Count <> 0
        Then
        '             str.Append("WHERE ")

```

```

        '         For Each item In
        SessionGrade
        '             str.Append("LevelId="
        & item.ToString & " ")
        '             If i <>
        SessionGrade.Count Then
        '                 str.Append("OR ")
        '             End If
        '             i += 1
        '         Next
        '     End If
        ' End If
        ' Dim model As New List(Of
        MapModel)
        ' Dim Store =
        ctx.ExecuteStoreQuery(Of
        Toko)(str.ToString()).ToList
        ' For Each item In Store
        '     Dim Icon As New
        IconModel
        '     Dim LevelId = item.LevelId
        '     Dim imagePath = (From a In
        ImageDictionary() Where a.Key =
        LevelId).FirstOrDefault.Value
        '     Icon = New IconModel With
        { .image = imagePath, .iconsize =
        {26, 46}, .iconanchor = {12, 46},
        .infowindowanchor = {12, 0} }
        '     Dim MapItem As New
        MapModel With { .longitude =
        item.Longitude, .latitude =
        item.Latitude, .address =
        item.AlatamToko, .html =
        HTMLTemplate(item).ToString,
        .icon = Icon }
        '     model.Add(MapItem)
        ' Next
        ' BuildJson(model)
        ' Return Json(SessionGrade,
        JsonRequestBehavior.AllowGet)
    'End Function
#End Region
    Function Index() As ActionResult
        Dim model = MapRepo.Json
        BuildJson(model)
        Return View()

```

```

    End Function
    Function ChartPartial(ByVal id As
Integer) As ActionResult
        ViewBag.id = id
        Return PartialView()
    End Function
End Class

```

3. Listing Program Data Store Manager

```

Namespace Map
<Authorize(Roles:="administrator")
>
    Public Class ManagerController :
Inherits System.Web.Mvc.Controller
        Private Property ManagerRepo
As New StoreManagerRepository
#Region "ajax"
            Function GetImage(ByVal id
As Integer) As ActionResult
                Dim model =
ManagerRepo.GetById(id)
                Dim imageByte As Byte() =
model.Foto
                Return File(imageByte,
model.Mime)
            End Function
#End Region
            Function Index() As
ActionResult
                Return
View(ManagerRepo.list.OrderBy(Fu
nction(a) a.NIP).ToList)
            End Function
            Function Create() As
ActionResult
                Dim model As New
StoreManagerViewModel
                Return View("CreateOrEdit",
model)
            End Function
            Function Edit(ByVal id As
Integer) As ActionResult

```

```

                Dim model =
ManagerRepo.GetById(id)
                Return View("CreateOrEdit",
model)
            End Function
        <HttpPost>
        Function Post(ByVal model As
StoreManagerViewModel, ByVal
attachment As HttpPostedFileBase)
As ActionResult
            If attachment IsNot Nothing
Then
                If
attachment.ContentType.Contains("i
mage") = True Then
                    Dim
ImageByte(attachment.ContentLengt
h) As Byte
                    attachment.InputStream.Read(Image
Byte, 0, attachment.ContentLength)
                    model.Foto =
ImageByte
                    model.Mime =
attachment.ContentType
                Else
                    ModelState.AddModelError("attach
ment", "Hanya file gambar yang
boleh diupload")
                End If
            End If
            Try
                ManagerRepo.save(model)
            If
ManagerRepo.errorMessage <> ""
Then
                ModelState.AddModelError("",
ManagerRepo.errorMessage)
            Else
                Return
RedirectToAction("Index")
            End If
            Catch ex As Exception
                ModelState.AddModelError("",
ex.Message)

```

```

        End Try
        Return View("CreateOrEdit",
model)
    End Function
    Function Delete(ByVal id As
Integer) As ActionResult
        Dim stat As Integer = 0
        Dim err As String = ""
        Try
            ManagerRepo.Delete(id)
            If
ManagerRepo.errorMessage <> ""
Then
                err =
ManagerRepo.errorMessage
            Else
                stat = 1
            End If
            Catch ex As Exception
                err = ex.Message
            End Try
            Return Json(New With {.stat
= stat, .err = err},
JsonRequestBehavior.AllowGet)
        End Function
    End Class
End Namespace

```

4. Listing Program Data Penjualan

```

Imports Telerik.Web.Mvc
Namespace Map
<Authorize(Roles:="administrator")
>
    Public Class SalesController :
Inherits System.Web.Mvc.Controller
        Private Property TokoRepo As
New TokoRepository
        Private Property TargetRepo As
New TargetRepository
        #Region "Grid Action"
        <GridAction()>

```

```

        Function _SelectStore() As
ActionResult
            Dim TokoRepo As New
TokoRepository
            Return View(New
GridModel(Of TokoViewModel)
With {.Data = TokoRepo.list})
        End Function
        <GridAction()>
        Function _SelectItem(ByVal
IdToko As Integer) As ActionResult
            Dim TargetRepo As New
TargetRepository
            Return View(New
GridModel(Of TargetViewModel)
With {.Data =
TargetRepo.list(IdToko)})
        End Function
    #End Region
    Function Index() As
ActionResult
        Return
View(TokoRepo.list.OrderBy(Functi
on(a) a>NamaToko).ToList)
    End Function
    Function Delete(ByVal id As
Integer) As ActionResult
        Dim stat As Integer = 0
        Dim err As String = ""
        Try
            TargetRepo.Delete(id)
            If
TargetRepo.errorMessage <> ""
Then
                err =
TargetRepo.errorMessage
            Else
                stat = 1
            End If
            Catch ex As Exception
                err = ex.Message
            End Try
            Return Json(New With {.stat
= stat, .err = err},
JsonRequestBehavior.AllowGet)
        End Function

```

```

Function Create(ByVal IdToko
As Integer) As ActionResult
    ViewBag.Selectlist =
Selectlist()
    Return View("CreateOrEdit",
model:=New TargetViewModel
With {.IdToko = IdToko, .Tahun =
Now.Year})
End Function
Function Edit(ByVal id As
Integer, ByVal IdToko As Integer)
As ActionResult
    ViewBag.Selectlist =
Selectlist()
    Return View("CreateOrEdit",
TargetRepo.GetById(id))
End Function
<HttpPost(>)
Function CreateOrEdit(ByVal
model As TargetViewModel) As
ActionResult
    ViewBag.Selectlist =
Selectlist()
    Dim ctx As New MapEntities
    Dim valid As Boolean = True
    Dim check = (From a In
ctx.Targets
        Where a.Bulan =
model.Bulan And a.Tahun =
model.Tahun And a.IdToko =
model.IdToko
        Select
a).FirstOrDefault
    If check IsNot Nothing Then
        If check.IdTarget <>
model.IdTarget Then
            valid = False
ModelState.AddModelError("",
"Data penjualan bulan " &
MonthName(model.Bulan) & " dan
tahun " & model.Tahun.ToString & "
sudah ada.")
        End If
    End If
    If valid = True Then

```

```

        model.Selisih =
model.Penjualan -
model.TargetBulanan
        model.PersentasePenjualan
= (model.Penjualan /
model.TargetBulanan) * 100

model.PeriodeTargetBulanan =
MonthName(model.Bulan) & "-" &
model.Tahun.ToString
        TargetRepo.save(model)
    If
TargetRepo.errorMessage <> ""
Then
    ModelState.AddModelError("",
TargetRepo.errorMessage)
Else
    Return
RedirectToAction("index")
End If
End If
Return View(model)
End Function
Function Selectlist() As
SelectList
    Dim dict As New
Dictionary(Of Integer, String)
    For i = 1 To 12
        dict.Add(i, MonthName(i))
    Next
    Return New SelectList(dict,
"key", "value")
End Function
End Class
End Namespace

```

5. Listing Program Data Toko

```

Namespace Map
<Authorize(Roles:="administrator")
>
    Public Class StoreController :
Inherits System.Web.Mvc.Controller

```

```

        Private Property TokoRepo As
New TokoRepository
        Private Property ManagerRepo
As New StoreManagerRepository
        Private Property LevelRepo As
New LevelRepository
#Region "Ajax"

        Function GetImage(ByVal id
As Integer) As ActionResult
            Dim model =
TokoRepo.GetById(id)
            Dim imageByte As Byte() =
model.Foto
            Return File(imageByte,
model.Mime)
        End Function
#End Region

        Function Index() As
ActionResult
            Return
View(TokoRepo.list.OrderBy(Functi
on(a) a>NamaToko).ToList)
        End Function
        Function Create() As
ActionResult
            Dim model As New
TokoViewModel
            ViewBag.level =
LevelRepo.selectList
            ViewBag.manager =
ManagerRepo.selectList
            Return View("CreateOrEdit",
model)
        End Function
        Function Edit(ByVal id As
Integer) As ActionResult
            Dim model =
TokoRepo.GetById(id)
            ViewBag.level =
LevelRepo.selectList
            ViewBag.manager =
ManagerRepo.selectList
            Return View("CreateOrEdit",
model)
        End Function

```

```

        <HttpPost(>
        Function Post(ByVal model As
TokoViewModel, ByVal attachment
As HttpPostedFileBase) As
ActionResult
            ViewBag.level =
LevelRepo.selectList
            ViewBag.manager =
ManagerRepo.selectList
            If attachment IsNot Nothing
Then
                If
attachment.ContentType.Contains("i
mage") = True Then
                    Dim
ImageByte(attachment.ContentLengt
h) As Byte

attachment.InputStream.Read(Image
Byte, 0, attachment.ContentLength)
                    model.Foto =
ImageByte
                    model.Mime =
attachment.ContentType
                Else
ModelState.AddModelError("attach
ment", "Hanya file gambar yang
boleh diupload")
                End If
            End If
            Try
                TokoRepo.save(model)
                If TokoRepo.errorMessage
<> "" Then
ModelState.AddModelError("",
TokoRepo.errorMessage)
                Else
                    Return
RedirectToAction("Index")
                End If
            Catch ex As Exception
ModelState.AddModelError("",
ex.Message)
            End Try
            Return View("CreateOrEdit",
model)

```

```

        End Function
        Function Delete(ByVal id As
Integer) As ActionResult
            Dim stat As Integer = 0
            Dim err As String = ""
            Try
                TokoRepo.Delete(id)
                If TokoRepo.errorMessage
<> "" Then
                    err =
TokoRepo.errorMessage
                Else
                    stat = 1
                End If
            Catch ex As Exception
                err = ex.Message
            End Try
            Return Json(New With {.stat
= stat, .err = err},
JsonRequestBehavior.AllowGet)
        End Function
    End Class
End Namespace

```

6. Listing Program User Management

```

Imports Telerik.Web.Mvc
Namespace Workshop
<Authorize(Roles:="Administrator")
>
    Public Class
        UserManagementController :
            Inherits System.Web.Mvc.Controller
                Private userrepo As
IUserManagementRepository
                Private StoreManagerRepo As
New StoreManagerRepository
                Function Index() As
ActionResult
                    userrepo = New
UserManagementRepository
                    Dim model =
userrepo.GetUserList()

```

```

                Return View(model)
            End Function
            Function Create() As
ActionResult
                ViewBag.selectlist =
StoreManagerRepo.otherselectList
                ViewBag.Roles =
Roles.GetAllRoles()
                Return View()
            End Function
            <HttpPost>
            Function Create(ByVal model
As RegisterModel, ByVal Role() As
String) As ActionResult
                ViewBag.selectlist =
StoreManagerRepo.otherselectList
                ViewBag.Roles =
Roles.GetAllRoles()
                If Role Is Nothing Then
                    ModelState.AddModelError("",
"Role user belum dipilih")
                End If
                Dim StoreModel =
StoreManagerRepo.list.Where(Func
tion(a) a.NIP.Trim =
model.UserName.Trim).FirstOrDefault
                If StoreModel Is Nothing
Then
                    ModelState.AddModelError("",
"User tidak terdaftar")
                End If
                model.Email =
StoreModel.Email
                If ModelState.IsValid Then
                    ' Attempt to register the
user
                    Dim createStatus As
MembershipCreateStatus
                    Membership.CreateUser(model.User
Name, model.Password,
model.Email, Nothing, Nothing,
True, Nothing, createStatus)
                    If createStatus =
MembershipCreateStatus.Success
Then

```

```

        If Role IsNot Nothing
Then
Roles.AddUserToRoles(model.UserName, Role)
        End If
        Dim pf As ProfileBase =
ProfileBase.Create(model.UserName
)
pf.SetPropertyValue("FullName",
StoreModel>Nama)
pf.Save()
Return
RedirectToAction("Index")
Else
ModelState.AddModelError("",
ErrorCodeToString(createStatus))
End If
End If
' If we got this far, something
failed, redisplay form
Return View(model)
End Function
Function Edit(ByVal Id As
Guid) As ActionResult
userrepo = New
UserManagementRepository()
Dim model =
userrepo.GetUserById(Id)
With ViewBag
.selectlist =
StoreManagerRepo.otherselectList
.Roles =
Roles.GetAllRoles()
.UserRole =
Roles.GetRolesForUser(model.UserName)
End With
Return View(model)
End Function
<HttpPost()>
Function Edit(ByVal model As
EditRegisteredModel, ByVal Role()
As String) As ActionResult
With ViewBag
.selectlist =
StoreManagerRepo.selectList

```

```

.Roles =
Roles.GetAllRoles()
.UserRole =
Roles.GetRolesForUser(model.UserName)
End With
Dim StoreModel =
StoreManagerRepo.list.Where(Function(a) a.NIP.Trim =
model.UserName.Trim).FirstOrDefault
If StoreModel Is Nothing
Then
ModelState.AddModelError("",
"User tidak terdaftar")
End If
model.Email =
StoreModel.Email
If Role Is Nothing Then
ModelState.AddModelError("",
"Role user belum dipilih")
End If
'If ModelState.IsValid Then
Dim provider As
MembershipProvider =
Membership.Provider
Dim _user As
MembershipUser =
provider.GetUser(model.UserName,
False)
_user.Email = model.Email
provider.UpdateUser(_user)
For Each s In
Roles.GetAllRoles
If
Roles.IsUserInRole(model.UserName, s) Then
Roles.RemoveUserFromRole(model.
UserName, s)
End If
Next
If Role IsNot Nothing Then
Roles.AddUserToRoles(model.UserName, Role)
End If

```



```

        Dim pf As ProfileBase =
        ProfileBase.Create(model.UserName
        )
        pf.SetPropertyValue("FullName",
        StoreModel>Nama)
        pf.Save()
        If Not
        String.IsNullOrEmpty(model.Password
        rd) Then
            Dim newPass As String =
            provider.ResetPassword(model.UserName, "")
            provider.ChangePassword(model.UserName, newPass, model.Password)
            End If
            Return
            RedirectToAction("Index")
            'End If
            ' If we got this far, something
            failed, redisplay form
            Return View(model)
            End Function
            Function Delete(ByVal Id As
            Guid) As ActionResult
                If Not
                String.IsNullOrEmpty(Id.ToString)
                Then
                    'Membership.DeleteUser(Id, True)
                    Dim ctx As New
                    MapEntities
                    Dim aspnet_Users = (From
                    a In ctx.aspnet_Users
                    Where a.UserId
                    = Id Select a).FirstOrDefault
                    If aspnet_Users IsNot
                    Nothing Then
                        ctx.ExecuteStoreCommand("DELETE
                        FROM aspnet_UsersInRoles
                        WHERE UserId=@UserId",
                        New
                        SqlClient.SqlParameter("@UserId",
                        Id))
                    End If
                    If
                    aspnet_Users.aspnet_Membership
                    IsNot Nothing Then

```

```

                    If
                    aspnet_Users.aspnet_Profile IsNot
                    Nothing Then
                        ctx.aspnet_Profile.DeleteObject(aspnet_Users.aspnet_Profile)
                        ctx.SaveChanges()
                    End If
                    ctx.aspnet_Membership.DeleteObject(aspnet_Users.aspnet_Membership)
                    ctx.SaveChanges()
                    End If
                    ctx.aspnet_Users.DeleteObject(aspnet_Users)
                    ctx.SaveChanges()
                    End If
                    End If
                    Return
                    RedirectToAction("Index")
                    End Function
                    #Region "Status Code"
                    Public Function
                    ErrorCodeToString(ByVal
                    createStatus As
                    MembershipCreateStatus) As String
                        ' See
                        http://go.microsoft.com/fwlink/?Link
                        ID=177550 for
                        ' a full list of status codes.
                        Select Case createStatus
                            Case
                            MembershipCreateStatus.DuplicateU
                            serName
                                Return "User name ini
                                sudah ada. Silahkan gunakan user
                                name yang lain."
                            Case
                            MembershipCreateStatus.DuplicateE
                            mail
                                Return "User dengan
                                email ini sudah tersedia. Silahkan
                                gunakan email yang lain."
                            Case
                            MembershipCreateStatus.InvalidPass
                            word

```

Return "The password provided is invalid. Please enter a valid password value."

Case
MembershipCreateStatus.InvalidEmail

Return "Email yang digunakan bukan merupakan email yang valid. Silahkan periksa kembali"

Case
MembershipCreateStatus.InvalidAnswer

Return "The password retrieval answer provided is invalid. Please check the value and try again."

Case
MembershipCreateStatus.InvalidQuestion

Return "The password retrieval question provided is invalid. Please check the value and try again."

Case
MembershipCreateStatus.InvalidUsername

Return "User name yang digunakan tidak valid."

Case
MembershipCreateStatus.ProviderError

Return "The authentication provider returned an error. Please verify your entry and try again. If the problem persists, please contact your system administrator."

Case
MembershipCreateStatus.UserRejected

Return "The user creation request has been canceled. Please verify your entry and try again. If the problem persists, please contact your system administrator."

Case Else

Return "An unknown error occurred. Please verify your entry and try again. If the problem persists, please contact your system administrator."

End Select

End Function

#End Region

End Class

End Namespace