

## **Bab 3**

# **Metode dan Perancangan Sistem**

Penelitian ini dimulai dari pengambilan data penjualan PT. Sinar Niaga Sejahtera *Point* Ambarawa yang kemudian diteruskan dengan permintaan ijin untuk melakukan replikasi sistem untuk membantu proses penelitian perjalanan data dan tabulasi data. Penelitian dilanjutkan dengan pemaparan tabulasi data dari proses *input* sampai dengan *output* sesuai dengan judul dan rumusan masalah yang telah ditentukan sebelumnya.

Untuk lebih menjelaskan mengenai sistem OLAP dan *Data Warehouse* yang telah dibangun di PT. Sinar Niaga Sejahtera *Point* akan digambarkan dengan beberapa diagram UML seperti *use-case diagram*, *activity diagram*, *class diagram*, serta *deployment diagram* untuk *user* yang berperan dalam proses tabulasi data dalam sistem ini.

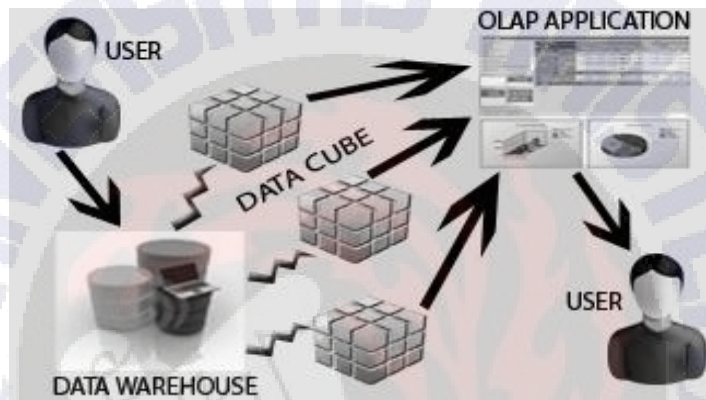
### **3.1 Lokasi dan Waktu Penelitian**

Penelitian yang dilakukan di PT. Sinar Niaga Sejahtera Ambarawa bertempat di Ruko I Ambarawa Jl. Brigjend Sudiarto Ambarawa yang dilakukan pada bulan April-Mei 2012.

Alasan pengambilan tempat penelitian di perusahaan ini adalah karena sistem OLAP dan *Data Warehouse* telah dipakai untuk membantu kerja karyawan yang ada dalam perusahaan ini dalam hal pelaporan dan analisa hasil penjualan perusahaan.

### 3.2 Metode Penelitian

Terdapat beberapa metode yang dilakukan dalam penelitian ini untuk menghasilkan proses tabulasi data di dalam sistem OLAP yang didukung dengan *Data Warehouse*. Tujuan penggunaan metode ini agar dapat dihasilkan tahapan perjalanan data di dalam sistem seperti yang terdapat di dalam Gambar 3.1.



**Gambar 3.1** Proses Perjalanan Data

Dalam Gambar 3.1 dijelaskan proses perjalanan data di dalam *Data Warehouse* dan teknologi OLAP dari proses *input* sampai dengan *output*. Dimulai dengan proses penyimpanan data di dalam *Data Warehouse* melalui proses ETL. Kemudian data-data yang terdapat di dalam *Data Warehouse* akan dipilih melalui beberapa OLAP *Cube* sehingga dapat menjadi suatu informasi yang berguna bagi pemilik sistem. Proses berikutnya adalah pengolahan data menjadi informasi dalam aplikasi OLAP yang berupa numerik, grafik, beserta hasil analisisnya sehingga pemilik sistem dapat dengan mudah menggunakan dan memanfaatkannya.

Beberapa metode digunakan untuk mendukung penelitian ini dalam hal pencarian data dan hasil analisa tabulasi data. Metode itu adalah sebagai berikut.

1. Wawancara

Wawancara merupakan proses tanya jawab yang dilakukan secara langsung dengan pihak-pihak yang terkait dalam hal ini adalah SNS *Point Admin* yang secara langsung menggunakan aplikasi ini.

2. Observasi

Observasi merupakan proses pengamatan yang dilakukan secara langsung yaitu penulis datang ke tempat penelitian untuk mendapatkan beberapa data dan informasi yang dibutuhkan dalam penelitian ini.

3. Studi Literatur

Studi literatur atau studi pustaka merupakan proses untuk mengumpulkan data atau informasi-informasi yang berhubungan dengan penelitian dengan cara membaca buku-buku, jurnal, referensi-referensi, prosedur-prosedur yang berhubungan dengan OLAP dan *Data Warehouse*.

4. Analisa Tabulasi Data

Setelah dilakukan beberapa proses tersebut yaitu wawancara, observasi, dan studi literatur. Maka dalam proses analisis dilakukan untuk mengetahui proses perjalanan data dengan penerapan OLAP dan *Data Warehouse*

Selain itu terdapat pula beberapa aturan dalam menjalankan sistem ini. Aturan-aturan tersebut dapat dijelaskan dalam beberapa kategori sebagai berikut.

## 1. Hak Akses Sistem

Hak akses dalam aplikasi OLAP dibagi menjadi tiga kategori akun yaitu *SuperAdmin*, *Admin SNS*, dan *SNS Point Head*. *SuperAdmin* hanya dimiliki oleh pembuat program, *Admin SNS* bekerja untuk mengolah data-data di dalam sistem OLAP yang dimiliki oleh PT. *SNS Point Ambarawa*, sedangkan *SNS Point Head* hanya sebagai penerima, pemantau, serta pemakai data hasil olahan untuk kepentingan memimpin perusahaan.

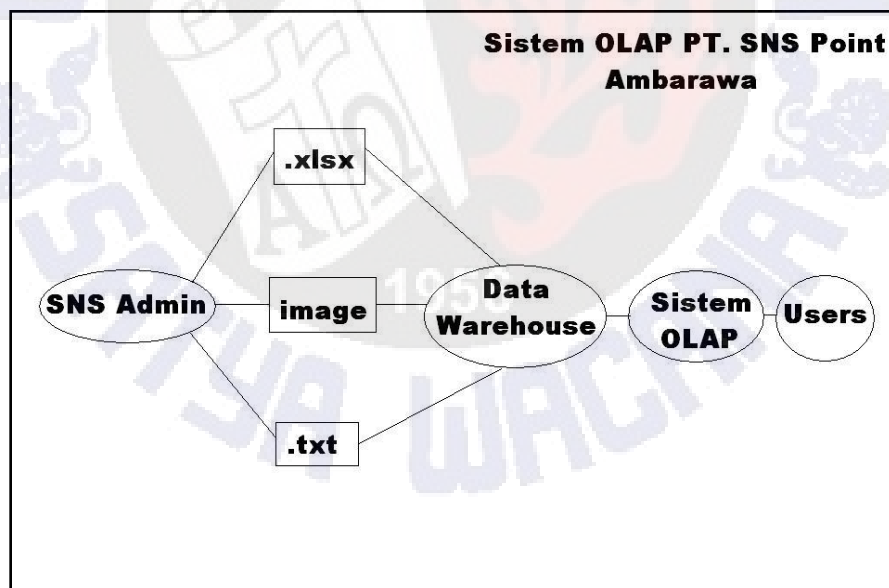
Alur pemakaian sistem dalam aplikasi ini adalah *user* memilih *web browser* sebagai media untuk akses sistem. Setelah mengisi alamat dari aplikasi OLAP, maka harus melewati proses verifikasi *username* dan *password* untuk memberikan hak akses kepada *user* sesuai dengan kategori hak akses dan untuk menjamin keamanan dalam sistem. Saat proses login telah dilalui benar dengan cara memasukkan verifikasi *username* dan *password*, *user* berhasil masuk ke dalam sistem untuk melakukan keperluannya dalam aplikasi OLAP. Alur sistem ini diakhiri dengan proses *logout* terlebih dahulu. Apabila *user* ingin mengakses kembali aplikasi ini maka alur sistem dimulai dari awal kembali.

## 2. Layanan Sistem Berdasarkan Hak Akses *User*

Layanan-layanan yang diberikan aplikasi ini didasarkan pada hak akses *user* yang dibaca oleh sistem melalui proses *login*. *User SuperAdmin* dapat mengakses seluruh kriteria menu dari program yang antara lain adalah *upload* laporan penjualan, lihat *edit* hapus laporan dan hasil analisis penjualannya, lihat *edit* hapus data pegawai, lihat edit hapus

forum SNS, tambah *user*, dan sebagainya. Sedangkan *Admin* SNS mendapatkan pelayanan untuk mengolah data-data yang dimiliki oleh PT. SNS Point Ambarawa seperti *upload* laporan penjualan, lihat *edit* hapus laporan dan hasil analisis penjualannya, lihat *edit* hapus data pegawai, lihat isi forum, lihat *edit* hapus data pegawai, dan sebagainya. Dan SNS Point Head mendapatkan *service* berupa aplikasi untuk melihat dan *download* data penjualan beserta analisisnya, lihat dan *download* data pegawai, dan isi forum.

Sistem OLAP yang telah dibangun dapat digambarkan arsitektur sistem OLAP PT. Sinar Niaga Sejahtera Point Ambarawa yang terdapat di dalam Gambar 3.2.



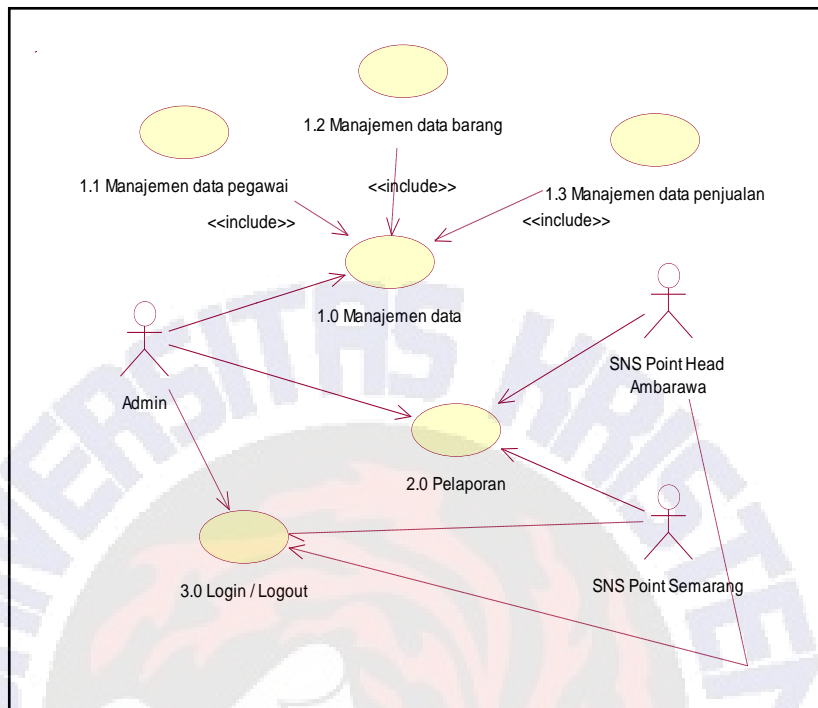
**Gambar 3.2** Arsitektur Sistem OLAP PT. SNS Point Ambarawa

Dalam Gambar 3.2 dijelaskan bahwa aplikasi bertumpu pada pihak *SNS Admin* yang bertugas memasukkan data baik bersifat *.xls*, *image .jpg* atau *.jpeg*, data *.text*, maupun data-data lainnya ke dalam *data warehouse*. Saat tersimpan ke dalam *data warehouse* maka saat *users* mengakses, sistem OLAP akan mengambil data-data tersebut dari dalam *Data Warehouse* untuk diolah sehingga menampilkan hasil-hasil analisa kepada *users* yang bersifat multidimensi dan mudah untuk dimengerti untuk mempermudah proses kerja dari perusahaan. *Users* disini yang dimaksud adalah semua pemakai aplikasi baik dalam kategori *SuperAdmin*, *SNS Admin*, serta *SNS Point Head*.

### **3.3 Unified Modelling Language (UML)**

#### **3.3.1 Use Case Diagram**

Menggunakan hasil dari metode penelitian wawancara dan observasi maka dapat dilakukan penggambaran sistem OLAP PT. *SNS Point Ambarawa* dan *Data Warehouse* menggunakan *use-case diagram* yang terdapat di dalam Gambar 3.3

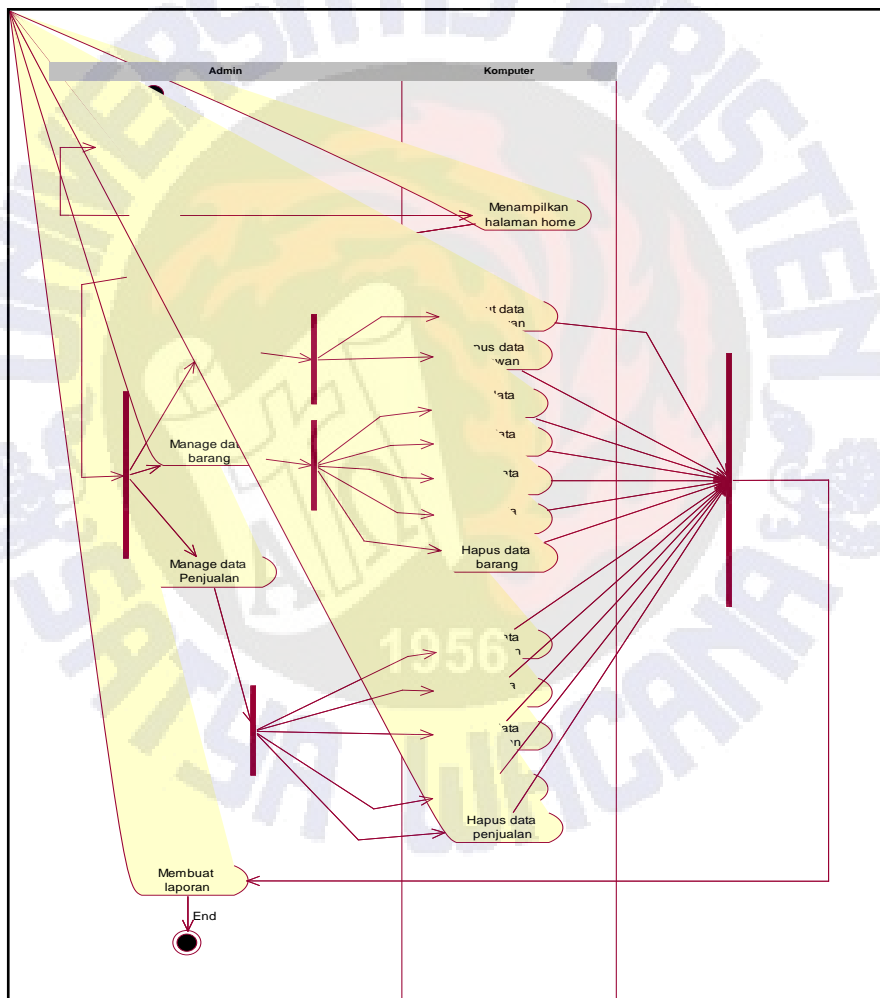


**Gambar 3.3** Use-Case Diagram Sistem

Gambar 3.3 merupakan *Use Case Diagram* dari sistem yang dikembangkan pada PT. SNS Point Ambarawa. Dijelaskan pada Gambar 3.3 inti dari sistem yang akan dibuat bertumpu pada pihak *Admin* yang bertugas untuk memasukkan dan mengolah data-data seperti data barang, data pegawai, dan data penjualan yang disatukan dalam bentuk laporan sistem yang dapat dilihat nantinya oleh pimpinan baik SNS Point Head Ambarawa maupun pihak SNS Point Semarang. Setiap *account* dalam sistem ini mempunyai *user*, *password login*, dan hak akses sistem yang berbeda-beda.

### 3.3.2 Activity Diagram

*Activity Diagram* yang akan digambarkan adalah beberapa aktivitas yang dilakukan oleh beberapa *user* di dalam sistem yang berhubungan dengan tabulasi data menggunakan *Data Warehouse* dan teknologi OLAP yaitu adalah *user Admin* yang terdapat di dalam Gambar 3.4.



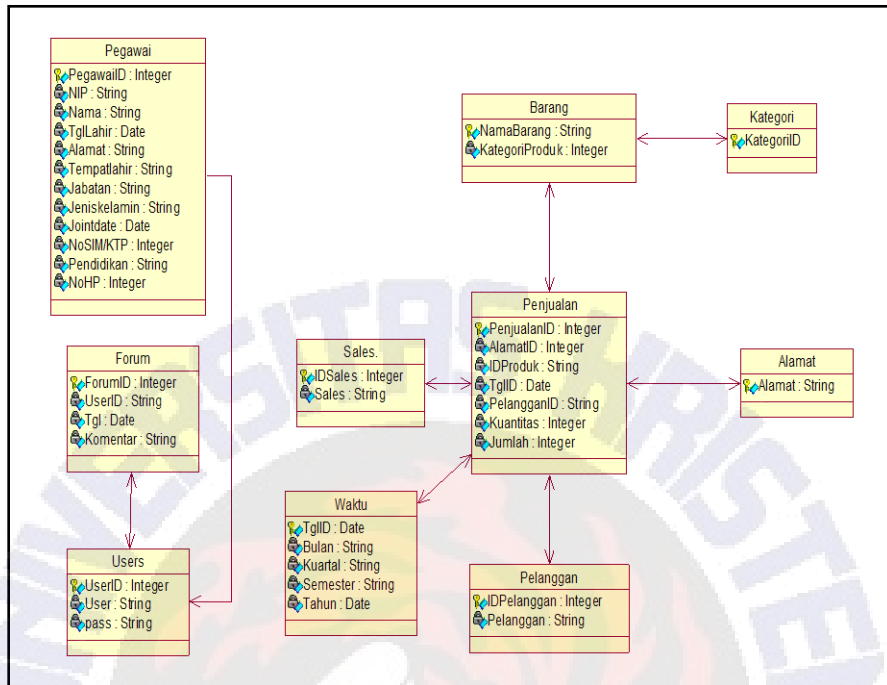
**Gambar 3.4** Activity Diagram Admin



Aktivitas yang dilakukan oleh *Admin* dalam penelitian ini adalah dalam hal proses *upload* (ETL) data penjualan perusahaan ke dalam sistem yang nantinya akan disimpan di dalam *Data Warehouse* dan diolah dengan teknologi OLAP untuk menghasilkan informasi yang relevan dan efektif.

### 3.3.3 *Class Diagram*

*Class* adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut atau properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda atau fungsi). *Class* diagram menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain. Dalam sistem yang telah dibangun dan berjalan digambarkan dengan beberapa class yang terdapat di dalam Gambar 3.5.



**Gambar 3.5** *Class Diagram* Sistem OLAP SNS Point Ambarawa

Gambar 3.5 adalah gambaran melalui *class diagram* SNS Point Ambarawa. Di dalam sistem ini *class* induk berisi enam tabel yaitu pegawai, sales, barang, penjualan, waktu, pelanggan serta informasi tentang pembuat aplikasi. Dari enam tabel induk inilah diturunkan lagi ke beberapa *class* dan tabel-tabel yang lebih kecil dengan informasi-informasi yang dibutuhkan oleh pihak pengguna dalam hal ini PT. SNS Point Ambarawa. Relasi antar *class* pada sistem ini adalah cukup kompleks dan banyak. *Class* Penjualan berelasi dengan *class* Barang, Alamat, Sales, Waktu, dan Pelanggan. Sedangkan untuk *class* lain seperti Pegawai, Forum, dan Users berdiri sendiri dan tidak bergantung pada berelasi dengan *class* Penjualan.

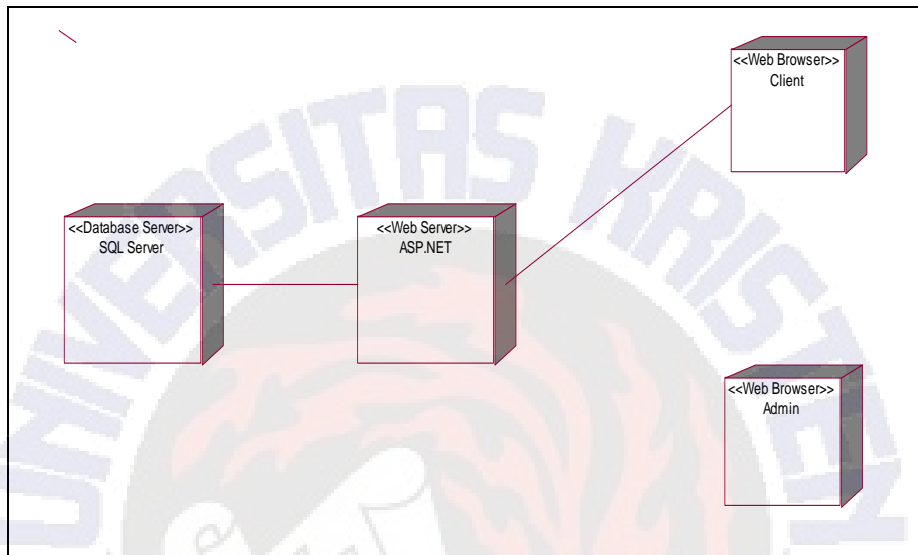
Relasi antara *class* Penjualan dengan *class* Barang, Alamat, Sales, Waktu, dan Pelanggan adalah *One to Many* yang berarti satu tabel Penjualan dapat memiliki lebih dari satu barang yang terdapat pada tabel Barang, memiliki lebih dari satu waktu untuk menjual barang yang terdapat pada tabel Waktu, memiliki lebih dari satu pelanggan yang terdapat pada tabel Pelanggan, memiliki lebih dari satu *sales* dalam menjual barang yang terdapat pada tabel *Sales*, dan memiliki lebih dari satu alamat pegawai dan pelanggan yang terdapat pada tabel Alamat. Sedangkan relasi antara *class* Barang, Waktu, Pelanggan, *Sales*, dan Alamat dengan *class* Penjualan adalah *Many to One*.

Pada relasi *class* Kategori dan *class* Barang adalah *One to Many* yang berarti satu kategori untuk barang dapat memiliki lebih dari satu jenis barang di dalamnya. Sedangkan relasi antara *class* Pegawai dan *class* Users adalah *One to One* yang artinya bahwa satu orang pegawai dalam sistem ini hanya dapat memiliki satu *user* saja. Relasi antara *class* Users dan *class* Forum adalah *One to Many* yang berarti satu buah *user* dapat menulis lebih dari satu forum dalam sistem.

#### 3.3.4 *Deployment Diagram*

*Deployment* atau *physical diagram* menggambarkan detail bagaimana komponen di-*deploy* dalam infrastruktur sistem, di mana komponen akan terletak (pada mesin, *server* atau piranti keras apa), bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi *server*, dan hal-hal lain yang bersifat fisik. Sebuah *node* adalah *server*, *workstation* atau piranti keras lain yang digunakan untuk melakukan *deploy* komponen dalam lingkungan sebenarnya.

Hubungan antar *node* (misalnya TCP atau IP) dan *requirement* dapat juga didefinisikan dalam diagram ini. *Deployment Diagram* system ini terdapat pada Gambar 3.6.



**Gambar 3.6** *Deployment Diagram*

Pada *Deployment Diagram* ini menggunakan *SQL Server* sebagai *Database Server* untuk sistem ini. Nantinya data-data yang berada pada *Database Server* ini akan diolah menjadi data yang lebih efisien sehingga *client* dapat membaca dan menggunakan sistem ini hanya dengan menggunakan bantuan *Web Browser*.

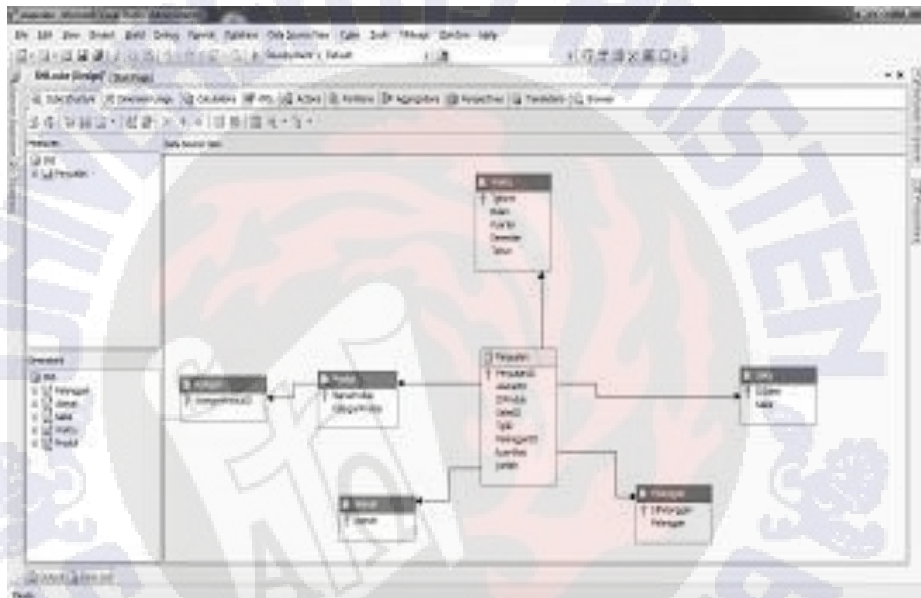
### 3.3.5 *Star Schema* dan *Snowflake Schema*

Dalam permodelan *Data Warehouse* merupakan suatu desain *logic* yang merepresentasikan data dalam bentuk standar, dan mendukung dilakukannya akses terhadap data dengan cepat. Terdapat beberapa istilah yang berkaitan dengan model ini, yaitu *fact table*, *dimension table*, dan hirarki. *Fact table* merupakan tabel

utama yang berisikan kumpulan dari *primary key* tabel lain. Setiap *fact table* dalam model konseptual *Data Warehouse* memiliki *composite key*, begitu juga sebaliknya, setiap tabel yang memiliki *composite key* adalah *fact table*. Dengan kata lain, setiap tabel yang menunjukkan hubungan banyak ke banyak (*many to many relationship*) pastilah merupakan *fact table*. *Dimension table* adalah tabel yang lebih sederhana dimana didalamnya terdapat *primary key* yang berhubungan dengan salah satu *composite key* yang ada pada *fact table*. Hirarki mendefinisikan urutan pemetaan dari konsep level bawah ke level yang lebih tinggi, konsep yang lebih umum. Dimensional model yang paling populer adalah *star schema*, *snowflakes schema*, dan *fact constellation schema*.

Di dalam implementasi *Data Warehouse* terdapat dua jenis skema *relational database* yaitu *Star Schema* dan *Snowflake Skema*. Kedua skema ini adalah hal yang umum dalam proses pembangunan *Data Warehouse*. *Star schema* adalah salah satu dimensional model dimana *fact table* terletak di pusat dan tabel lainnya, yaitu *dimension table* terletak disekelilingnya. Kebanyakan dari *fact table* pada *star schema* merupakan normalisasi bentuk ketiga dari *database*, sedangkan *dimensional table* adalah normalisasi bentuk kedua. *Star schema* merupakan bentuk dimensional model yang paling sederhana. *Snowflake schema* merupakan bentuk perluasan dari *star schema* dimana terjadi proses normalisasi dari beberapa atau seluruh *dimension table*. *Snowflake schema* biasanya digunakan pada tabel yang sangat besar dan jika *star schema* tidak mampu menggambarkan kompleksitas dari *database* tersebut (Sharma dan Gosain, 2010).

Dalam penelitian ini menggunakan *Star Schema* sebagai skema dalam pembangunan *Data Warehouse*. Pemilihan skema ini didasarkan pada keuntungan penggunaan *Star Schema* yaitu mudah dipahami karena struktur database yang sederhana dan *low maintenance*. Untuk Skema *Database* dalam sistem ini terdapat di dalam Gambar 3.7.



**Gambar 3.7** *Star Schema Data Warehouse*

Dalam Gambar 3.7 dijelaskan bahwa *Fact Table* dari sistem ini adalah Penjualan (*sale*) karena memang dalam sistem ini akan dianalisa laporan penjualan dari PT. Sinar Niaga Sejahtera *Point*. Untuk *Dimension Table* dari skema ini adalah tabel Produk (*Product*), Waktu (*Time*), Alamat (*Address*), Pelanggan (*Costumer*), dan Sales (*Salesman*). Untuk tabel Produk akan berelasi dengan Tabel Kategori (*Category*) secara *Many to One*. Sehingga dapat dikatakan skema dalam sistem ini adalah campuran dari *Star*

*Schema* dan *Snowflake Schema*, tetapi lebih dominan kepada *Star Schema*.

