**STUDI KASUS RITEL**

PT. Segala Murah adalah sebuah perusahaan yang bergerak di bidang jual beli eceran(ritel) yang sudah tersebar di beberapa wilayah jawa barat, produk yang ditawarkan pun sangat beragam dimulai dari komoditi sehari-hari disediakan untuk memenuhi kebutuhan para konsumennya. Terkadang PT segala Murah memberikan potongan harga untuk memanjakan konsumennya dan memberikan harga yang bersaing agar tetap menjadi perusahaan ritel terdepan.

Setiap wilayah cabang akan dikelola oleh kepala cabang dan dibantu oleh kepala masing-masing bidang, supervisor dan para pegawai untuk membantu pengelolaan manajemen penjualan. Untuk pengelolaan pegawai perusahaan menetapkan 2 klasifikasi yaitu : 1. Tetap, dan 2. Kontrak dengan masa kerja 1 tahun sebagai pertimbangan ujicoba dan menjadi pegawai tetap apabila kinerjanya dianggap baik.

**Kebutuhan Bisnis**

1. PT. Segala murah memilik banyak gerai di wilayah jabar, dan total gerai yang dimiliki ialah 100 gerai.
2. Gerai adalah tempat terjadinya jual beli antara konsumen dan perusahaan. Setiap gerai akan dikelola oleh kepala cabang dan dibantu oleh kepala masing-masing bidang, supervisor dan para pegawai untuk membantu pengelolaan manajemen penjualan.
3. Pegawai ialah tenaga kerja yang membantu penangan perusahan dari level top management s/d lower management.
4. Pegawai lower management bisa dilakukan penerimaan dalam 2 bentuk : 1. Tetap, 2. Kontrak.
5. Konsumen yang dapat membeli produk tidak diwajibkan menjadi member, untuk member ada beberapa keuntungan seperti mendapatkan diskon setiap kali pembelian barang.
6. Membership memiliki batas kadaluwarsa dan harus selalu diperbaharui setiap tahun, dan membership dibagi menjadi 3, yakni, Silver, Gold, dan Platinum.
7. Untuk membership yang mencapai kadaluwarsa dan tidak diperpanjang akan dianggap hangus, untuk konsumen yang bersangkutan harus mengulangi pendaftaran membership dari awal.
8. Konsumen yang ingin menjadi membership harus memberikan informasi pribadi dan setiap aktivitas pembelian akan direkam, untuk konsumen yang rutin berbelanja dan pembelian pertahun mencapai 15 Juta Rupiah akan diberikan tawaran kenaikan kategori menjadi Gold dan untuk Platinum saat konsumen sudah mencapai pembelian di 25 Juta.
9. Produk yang dijual merupakan komoditi sehari-hari, untuk komoditi harian selain didatangkan dari pemasok(suppilier) terdapat produk yang di produksi sendiri oleh perusahaan seperti makan ringan dan air mineral.
10. Produk yang mendekati masa kadaluwarsa akan diberikan potongan yang besar untuk menghabiskan stok barang yang ada.
11. minProduk yang terjual harus terdata untuk produk yang terjual dibeli berikut konsumen selaku pembeli, dan rincian penjulannya lainnya seperti waktu event penjualan terjadi, tempat terjadinya penjualan, produk terjual dan cara pembayaran konsumen.
12. Setiap produk yang sudah dibeli dapat dikirimkan melalui jasa yang berikan oleh perusahaan untuk meningkatkan layanan. Setiap produk yang akan dikirimkan setiap konsumen akan diminta untuk mengisi alamat tujuan dan setiap pengiriman aakan dilakukan di hari terjadinya penjualan atau H+1.
13. Diskon diberikan pada semua produk yang diberikan dengan kententuan berlaku, yakni:
    1. Konsumen terdaftar sebagai member maka akan diberikan potongan sebesar 1% - 3% dari total biaya per item barang yang diberikan, dan besaran potongan ini bisa berubah sewaktu-waktu.
    2. Produk diberikan diskon dengan besaran yang berbeda berdasarkan kuantitas produk yang dibeli, dan tedapat batas minimal dan maksimal kuantitas yang dapat diberikan diskon.
    3. Diskon pada produk tidak diberlakukan setiap hari melainkan pada waktu-waktu tertentu saja.
14. Dalam proses pengiriman produk terjual, harus terekam waktu pengiriman dan penerimaan dan siapa yang mengirim dan menerima, hal ini digunakan untuk menjamin barang yang diantar dan memudahkan dalam penelusuran apabila ada suatu persoalan.

Diagram

Description automatically generated

Bualah Objek-objek berikut di Database anda berdasarkan model yang sudah diberikan:

1. **Constraint**
   1. **Computed Column Constraint**

Total\_Biaya\_per\_Produk = (harga\_satuan \* jumlah\_penjualan) - Nominal\_Diskon

* 1. **Check Constraint Constraint**
* Nilai di Kolom/ Atribut Jenis\_kelamin pada member harus Perempuan atau Pria
* Nilai di Kolom/ Atribut pernikahan pada pegawai harus Nikah atau Belum Nikah
* Nilai di Kolom/ Atribut Jumlah\_anak pada entitas pegawai harus berisi 0 Jika status pernikahan Belum Nikah
* Nilai di Kolom/ Atribut Persen\_Diskon pada entitas diskon harus berisikan nilai dari 0 - 100
* Nilai harus lebih dari 0 untuk Kolom/ Atribut berikut:
  + MinQty pada entitas diskon
  + harga\_satuan pada Rincian\_Penjualan
  + Nominal\_Diskon pada Rincian\_Penjualan
  + Total\_Biaya\_Per\_Produk pada Rincian\_Penjualan
  + harga\_satuan pada Produk
* Nilai Tanggal\_Akhir\_berlaku Harus lebih besar dari Tanggal\_Awal\_Berlaku
* Nilai di Kolom/ Atribut Satuan\_Berat pada Produk harus G atau Mg atau Kg atau L atau Ml
  1. **Unique Contraint**
* Entitas Member
  + Kolom/ Atribut No\_KTP
  + Kolom/ Atribut No\_Kontak
* Entitas Supplier
  + Kolom/ Atribut No\_Kontak
  1. **-- Tipe Data Bentukan / User-defined Data Type**
* Tipe data bernama Nama dan dapat menambung nilai alpanumerik dengan panjang data ialah 80
* Kolom yang menggunakan tipe data nama
  + Nama\_Penerima di entitas Pengirim
  + Nama di entitas Gerai
  + Nama di entitas Pegawai
  + Nama\_jabatan di entitas Jabatan
  + Nama di entitas Member
  + Nama\_\_Jenis\_Member di entitas Jenis\_member
  + Nama\_produk di entitas produk

1. **Daftar Fungsi**
   1. **Fungsi Bernama Register Member untuk menyimpan data member,**

**Parameter** yang digunakan No\_KTP, Nama, Jenis\_Kelamin, Alamat, Keluarahan, Kecamatan, Kabupaten\_Kota, Provinsi, No\_Kontak, MinTrans

**Nilai Yang dikembalikan** Berupa String

**Rules/Flow Algoritma**:

* Nilai member\_id diambil dengan format YYYYMMDDZZZZ, YY:tahun berjalan, MM:Bulan berjalan, DD: hari Berjalan, ZZZZ: Nomor terkhir pendaftaran pada tanggal berjalan Misal: 202101010001.
* Nilai pada atribut tanggal\_daftar\_member diambil dari tanggal saat ini di sistem komputer anda, dan tanggal\_kadaluarsa\_Member di tambah 1 tahun dari tanggal\_daftar\_member
* nilai atribut Jenis\_MemberJenis\_Member\_ID didapatkan pada jenis\_member\_ID yang didasarkan pada pencarian pada jenis member dengan menggunakan nilai di paramter mintrans dan dilakukan    pencarian data di jenis member yang sesuai.
* Jika nilai parameter mintrans tidak tersedia dalam entitas member maka akan dikembalikan pesan "Proses Regitrasi Ditolak karena minimal transaksi tidak memenuhi syarat". dan sebaliknya jika tersedia akan dikembalikan pesan "Selamat, Anda menjadi member kami dengan Jenis\_member dan keanggotaan anda berakhir Tanggal\_kadaluarsa", isi jenis\_member & Tanggal\_kadaluarsa dengan nilai jenis\_member yang sesuai hasil pencarian.

* 1. **Fungsi Bernama Mengambil Nama Pegawai**

**Parameter** yang digunakan Pegawai\_ID

**Nilai Yang dikembalikan Berupa String**

**Rules/Flow Algoritma:**

* Jika nilai pada parameter Pegawai\_ID maka anda akan mengembalikan nilai Nama pada entitas pegawai yang sesuai dan jika sebalikan maka akan dikembalikan Nilai kosong ('').
  1. **Fungsi Bernama Set bawahan**

**Parameter yang digunakan Pegawai\_ID, Atasan\_ID**

**Nilai Yang dikembalikan Berupa String**

**Rules/Flow Algoritma:**

* jika nilai pada parameter Pegawai\_ID dan/atau Atasan\_ID tidak tersedia di kumpulan data entitas pegawai maka akan dikembalikan pesan "Pegawai yang ada tetapkan tidak tersedia, silahkan cek kembali data pegawai yang akan di sunting". sebaliknya jika tersedia maka anda akan mengubah data atasan\_id dari bawahanya yang tentukan dan dikembalikan pesan sebagai berikut "Selamat nama\_pegawai\_bawahan menjadi bawahan dari nama\_pegawai\_atasan". isi Pegawai\_ID, nama\_pegawai\_bawahan, nama\_pegawai\_atasan dengan data yang sesuai.
* anda wajib menggunakan Fungsi Bernama Mengambil Nama Pegawai untuk kondisi jika nama pegawai\_ID dan atasan tersedia.
  1. **Fungsi Nominal Diskon**

**Parameter yang digunakan produk\_ID, kuantitas**

**Nilai Yang dikembalikan Berupa money**

**Rules/Flow Algoritma:**

* anda cari harga satuan produk berdasarkan produk\_ID jika tidak tersedia kembalikan nilai 0 dan fungsi langsung berakhir.
* anda cari persen diskon berdasarkan produk\_id dan tanggal saat ini yang disesuaikan dengan tanggal\_awal\_berlaku atau tanggal\_Akhir\_berlaku dan kuantitas harus berada diatas minimal\_Qty jika tidak tersedia isi dengan nilai 0 untuk persen dikson
* jika data produk id dan persen\_diskon tersedia maka dikembalikan nilai dengan formulasi (persen diskon/100) \* harga satuan

1. **Daftar Prosedur**
   1. **Prosedur Tambah Penjualan**

**Parameter** yang digunakan Pegawai\_ID, Member ID

**Rules/Flow Algoritma**:

* + Nilai parameter Member ID diizinkan untuk NULL sebagai konsumen non member
  + Jika nilai parameter pengawai\_ID tidak tersedia di pegawai dan berjabatan kasir munculkan error menggunakan raiserror dan menghentikan flow yang ada di prosedur, dan memberikan pesan "data pegawai tidak dikenali, cek kembali pegawai yang melakukan penjualan".
  + Tanggal\_transaksi di isi dengan tanggal sistem saat ini yang berjalan.
  + jenis pembayaran di isi dengan CASH
  + Nilai No\_transaksi diambil dengan format INV-YYYYMMDDZZZZ, YY:tahun berjalan, MM:Bulan berjalan, DD: hari Berjalan, ZZZZ: Nomor terkhir pendaftaran pada tanggal berjalan Misal: INV-202101010001.

* 1. **Prosedur Tambah Rincian Penjualan**

**Parameter** yang digunakan No\_transaksi, Produk ID, kuantitas

**Rules/Flow Algoritma**:

* + Nilai rincian\_ID diambil dengan format INV-YYYYMMDDZZZZ-AAAA, YY:tahun berjalan, MM:Bulan berjalan, DD: hari Berjalan, ZZZZ: Nomor terkhir pendaftaran pada tanggal berjalan, AAAA: nomor urut rincian Misal: INV-202101010001-0001.
  + nilai Diskon\_ProdukDiskon\_produk\_id ditetapkan berdasrkan parameter produk ID dan tanggal yang berjalan disistem saat ini.
  + harga satuan didapatkan dari produk berdasarkan produk ID
  + Nominal\_diskon di ambil dari fungsi Nominal diskon
  + total\_biaya\_per\_Produk = harga satuan \* kuantitas - Nomimal Diskon
  + munculkan pesan error jika no\_transaskia atau produk id tidak tersedia pada data penjualan atau produk
  + setelah anda simpan data dan sukses untuk tambahan rincian, anda perbaharui total\_pembayaran di penjualan dengan menggunakan nilai yang dikembalikan dari fungsi update total\_bayar.

1. **Trigger**
   1. **Update Total\_Bayar setiap kali proses penambahan data rincian berhasil dilakukan**
   * Anda cari dan lakukan kalkulasi total\_biaya\_per\_produk berdasarkan no\_trasaksi penjualan dan jika no\_trasaksi tidak ditemukan maka akan dikembalikan nilai 0