

BAB II

Merancang Struktur Database

2.1 Penjelasan Singkat Keyword yang akan Digunakan

Berikut adalah penjelasan dari keyword syntax pada **dbdiagram.io** yang akan digunakan :

1. **Table**. Digunakan untuk mendefinisikan sebuah tabel dalam database.

Contoh :

```
Table user {  
    //kolom-kolom yang dibutuhkan  
}
```

2. **Ref**. Digunakan untuk mendefinisikan relasi antara dua tabel.

Contoh :

```
Ref {  
    produk.id_kategori > kategori.id_kategori  
}
```

3. **PK** atau singkatan dari **Primary Key**, digunakan untuk mendefinisikan kolom yang berfungsi sebagai identitas unik dalam sebuah tabel.

Contoh :

```
id int PK
```

4. **int** atau **Integer**, digunakan untuk mendefinisikan bahwa kolom tersebut memiliki tipe data *bilangan bulat*.
5. **varchar**, digunakan untuk mendefinisikan bahwa kolom tersebut memiliki tipe data string atau teks.
6. **tinyint**, digunakan untuk mendefinisikan bahwa kolom tersebut memiliki tipe data bilangan bulat yang sangat kecil, biasanya digunakan untuk menyimpan nilai numerik kecil, seperti **boolean** (0 atau 1).
7. **text**, digunakan untuk mendefinisikan bahwa kolom tersebut memiliki tipe data teks dalam jumlah besar.
8. **timestamp**, digunakan untuk mendefinisikan bahwa kolom tersebut memiliki tipe data informasi waktu (tanggal dan jam).

9. **unique**, digunakan untuk memastikan bahwa nilai dalam kolom tersebut harus unik, tidak boleh ada duplikasi.
10. **nullable**, menunjukkan bahwa kolom tersebut boleh kosong (null)
11. **default**, digunakan untuk menetapkan nilai bawaan untuk kolom jika tidak ada nilai yang diinput.

2.2 Langkah – Langkah membuat struktur database untuk aplikasi Point of Sales ini adalah :

2.2.1 Identifikasi Kebutuhan dan Fitur Aplikasi

Kebutuhan fitur yang akan ada dalam aplikasi POS antara lain :

- a) Manajemen pengguna (user)
- b) Manajemen kategori produk
- c) Manajemen produk
- d) Manajemen anggota (member)
- e) Manajemen pemasok (supplier)
- f) Manajemen pembelian
- g) Pencatatan detail pembelian
- h) Manajemen penjualan
- i) Pencatatan detail pembelian
- j) Pencatatan pengeluaran
- k) Pengelolaan informasi toko

2.2.2 Menggunakan dbdiagram.io untuk Mendesain Database

Untuk mengakses dbdiagram.io kamu bisa mengunjungi alamat berikut ini :

<https://dbdiagram.io/>

2.2.3 Membuat Tabel User

Tabel **user** ini berfungsi untuk menyimpan data pengguna aplikasi.

Berikut adalah penulisan code sql tabel user :

```
Table users{  
    id int PK
```

```

nama_user varchar
email_user varchar // unique
password_user varchar
foto_user varchar
level_user tinyint
remember_token varchar // nullable
created_at timestamp // nullable
deleted_at timestamp // nullable
}

```

❖ **Catatan :**

- ✓ ***remember_token** digunakan untuk menyimpan token keamanan dalam tabel autentikasi, sering kali digunakan dalam sistem login untuk mengingat pengguna yang sudah login.*
- ✓ ***created_at** adalah kolom untuk menyimpan waktu kapan data dibuat. Biasanya menggunakan tipe data **timestamp** atau **datetime**.*
- ✓ ***deleted_at** digunakan dalam konsep **soft delete**, yaitu data tidak benar-benar dihapus dari database, tapi hanya “disembunyikan” dengan memberikan nilai waktu pada kolom ini ketika data dihapus.*

2.2.4 Membuat Tabel Kategori

Tabel **Kategori** ini berfungsi untuk mengelompokkan produk.

Berikut adalah penulisan untuk tabel kategori :

```

Table kategori {
    id_kategori int PK
    nama_kategori varchar // unique
    created_at timestamp // nullable
    updated_at timestamp // nullable
}

```

```
}
```

❖ **Catatan :**

- ✓ ***updated_at** adalah kolom yang digunakan untuk menyimpan kapan data terakhir kali diperbarui.*

2.2.5 Membuat Tabel Produk

Tabel **Produk** ini berfungsi untuk menyimpan informasi produk yang dijual.

Berikut adalah penulisan code sql untuk tabel kategori :

```
Table produk{
    id_produk int PK
    id_kategori int
    nama_produk varchar // unique
    merk_produk varchar // nullable
    harga_beli_produk int
    diskon_produk int // nullable default 0
    harga_jual_produk int
    stok_produk int // default 0
    created_at timestamp // nullable
    updated_at timestamp // nullable
}
Ref {
    produk.id_kategori > kategori.id_kategori
}
```

2.2.6 Membuat Tabel Member

Tabel **Member** ini berfungsi untuk menyimpan informasi anggota atau pelanggan yang mendaftar.

Berikut adalah penulisan code sql untuk tabel member :

```
Table member {
```

```

    id_member int PK
    kode_member varchar // unique
    nama_member varchar
    alamat_member text // nullable
    telepon_member varchar
    created_at timestamp // nullable
    updated_at timestamp // nullable
}

```

2.2.7 Membuat Tabel Supplier

Tabel **Supplier** ini berfungsi untuk menyimpan data pemasok barang.

Berikut adalah penulisan untuk tabel supplier :

```

Table supplier {
    id_supplier int PK
    nama_supplier varchar
    alamat_supplier text
    telepon_supplier varchar
    created_at timestamp // nullable
    updated_at timestamp // nullable
}

```

2.2.8 Membuat Tabel Pembelian

Tabel **Pembelian** ini berfungsi untuk mencatat transaksi pembelian dari pemasok/supplier.

Berikut adalah penulisan code sql untuk tabel supplier :

```

Table pembelian {
    id_pembelian int PK
    id_supplier int
    total_item_pembelian int
}

```

```

total_harga_pembelian int
diskon_pembelian int // default 0
bayar_pembelian int
created_at timestamp // nullable
updated_at timestamp // nullable
}
Ref {
    pembelian.id_supplier > supplier.id_supplier
}

```

2.2.9 Membuat Tabel Pembelian Detail

Tabel **Pembelian Detail** ini berfungsi untuk mencatat rincian setiap barang yang dibeli dari pemasok/supplier.

Berikut adalah penulisan code sql untuk tabel detail_pembelian :

```

Table pembelian_detail {
    id_pembelian_detail int PK
    id_pembelian int
    id_produk int
    harga_pembelian_detail int
    jumlah_pembelian_detail int
    subtotal_pembelian_detail int
    created_at timestamp // nullable
    updated_at timestamp // nullable
}
Ref {
    pembelian_detail.id_pembelian > pembelian.id_pembelian
}
Ref {
    pembelian_detail.id_produk > produk.id_produk
}

```

```
}
```

2.2.10 Membuat Tabel Penjualan

Tabel **Penjualan** ini berfungsi untuk mencatat transaksi penjualan kepada pelanggan.

Berikut adalah penulisan code sql untuk tabel penjualan :

```
Table penjualan {  
    id_penjualan int PK  
    id_member int  
    total_item_penjualan int  
    total_harga_penjualan int  
    diskon_penjualan int // default 0  
    bayar_penjualan int  
    diterima_penjualan int // default 0  
    id_user int  
    created_at timestamp // nullable  
    updated_at timestamp // nullable  
}  
  
Ref {  
    penjualan.id_member > member.id_member  
}
```

2.2.11 Membuat Tabel Penjualan Detail

Tabel **Penjualan Detail** ini berfungsi untuk mencatat rincian barang yang dijual.

Berikut adalah penulisan code sql untuk tabel penjualan detail :

```
Table penjualan_detail {  
    id_penjualan_detail int PK  
    id_penjualan int  
    id_produk int
```

```

    harga_penjualan_detail int
    jumlah_penjualan_detail int
    diskon_penjualan_detail int // default 0
    subtotal_penjualan_detail int
    created_at timestamp // nullable
    updated_at timestamp // nullable
}
Ref {
    penjualan_detail.id_penjualan > penjualan.id_penjualan
}
Ref {
    penjualan_detail.id_produk > produk.id_produk
}

```

2.2.12 Membuat Tabel Pengeluaran

Tabel **Pengeluaran** ini berfungsi untuk mencatat pengeluaran lain dalam bisnis, misalnya ‘biaya operasional’.

Berikut adalah penulisan code sql untuk tabel pengeluaran:

```

Table pengeluaran {
    id_pengeluaran int PK
    deskripsi_pengeluaran text
    nominal_pengeluaran int
    created_at timestamp // nullable
    updated_at timestamp // nullable
}

```

2.2.13 Membuat Tabel Setting

Tabel **Setting** ini berfungsi untuk menyimpan pengaturan aplikasi seperti nama toko, logo, alamat, dan lainnya.

Berikut adalah penulisan code sql untuk tabel setting :


```
Table setting {  
    id_setting int PK  
    nama_perusahaan varchar  
    alamat text // nullable  
    telepon varchar  
    path_logo varchar // nullable  
    tipe_nota tinyint  
    path_kartu_member varchar  
    created_at timestamp // nullable  
    updated_at timestamp // nullable  
}
```