# **BAB IV**

# **ANALISA DAN PERANCANGAN**

1. **Analisa Sistem yang Berjalan**

Saat ini sistem pencatatan inventori dan penjualan barang masih menggunakan prosedur manual yang di lakukan oleh bagian EDP/*Entry data processing* yaitu mencatat di buku dan di *Microsoft Excel* dan menghitung kembali setiap ada pemasukan barang dan pengeluaran barang.

1. **Analisa Kebutuhan**

Dalam mengimplementasikan aplikasi ini di butuhkan empat buah komponen penting yang akan dianalisa kebutuhannya :

1. Perangkat yang disarankan untuk implementasi program

Spesifikasi perangkat keras yang dibutuhkan meliputi:

* 1. Komputer dengan *Processor* minimal Pentium IV
  2. *Monitor* VGA
  3. RAM *minimal* 512 Mb
  4. Hardisk berkapasitas *minimal* 40 Gb

1. Perangkat lunak yang disarankan untuk implementasi program
   1. Web Browser Google Chrome

Perangkat lunak ini di tujukan untuk memjalankan aplikasi *inventory* dengan berbasis web

* 1. XAMPP (Apache, MYSQL *Database*, PHP *Script* *Language*)

Perangkat lunak ini berfungsi sebagai server untuk mengelola pemprosesan dari aplikasi *inventory*.

1. Data dan Informasi

Dalam pembuatan aplikasi ini, daftar kebutuhan data dan informasi yang dikehendaki adalah :

* 1. Data Area
  2. Data Barang
  3. Laporan Penjualan

Dalam menyusun sistem baru, software dan pemprograman yang di gunakan adalah berbasis web meliputi perpaduan antara PHP, HTML, dan MYSQL.

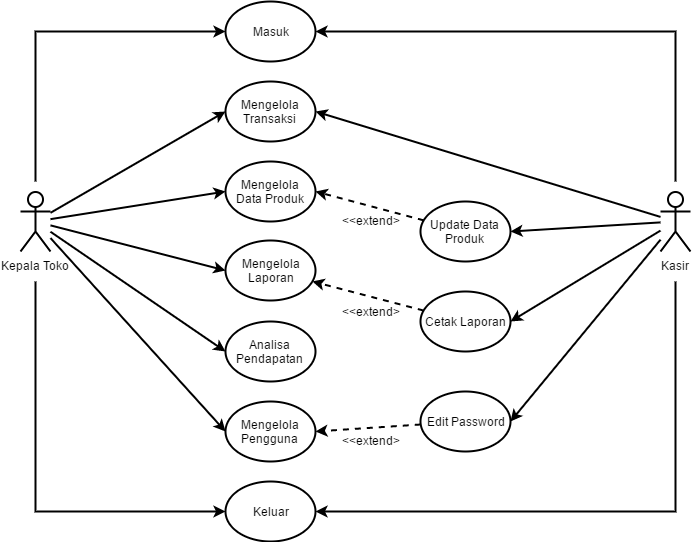
1. Pengguna (*Brainware*)

Brainware diperlukan Karena aplikasi ini tidak dapat beroperasi dengan sendirinya tanpa adanya intruksi dari pengguna. Dalam tahapan analisis ini pengguna dapat dikategorikan sebagai berikut:

* 1. Administrator yaitu seorang yang memiliki kewenangan dalam mengatur data. Mengatur disini antara lain mengelola data barang inventori, mulai dari penambah, mengedit, dan menghapus barang, serta menangani apabila terjadi error pada aplikasi.
  2. User yaitu seorang yang menggunakan website untuk melakukan proses penggajian, melakukan proses pengeluaran barang, cetak laporan penjualan.

1. **Desain Analisa**
2. *Unified Modeling Language* (UML)
   1. *Use Case Diagram*

*Use case diagram* merupakan diagram yang menampilkan aktor, *use case*, dan relasi dalam sebuah sistem, sebuah *use case* akan mempresentasikan sebuah interaksi antar pelaku dengan sistem.



Gambar 4.1 *Use Case Diagram*

Tabel 4.1 Keterangan aktor dalam perancangan aplikasi *inventory* dan penjualan

|  |  |  |
| --- | --- | --- |
| NO | Nama Aktor | Keterangan |
| 1 | Kepala Toko | * + - 1. Mengelola transaksi, dan mengontrol data produk,       2. Mengelola laporan untuk di analisa perkembangan perusahaan |
| 2 | Kasir | 1. Mencetak data transaksi untuk konsumen 2. Mengelola data dan cetak laporan untuk diberikan kepada kepala toko |

1. Sekenario *Use Case* Masuk

Nama *Use Case* : Masuk

Aktor : Kepala toko / Kasir

Tipe : *Primary*

Prekondisi : Masuk ke halaman web

Tujuan : Untuk masuk kehalaman Aplikasi web berdasarkan hak akses

Tabel 4.2 Sekenario *Use Case* Masuk

|  |  |
| --- | --- |
| Aktor | Sistem Aplikasi |
| 1. Kepala Toko atau Kasir memasukkan *username* dan *password* | 1. Sistem mengecek apakah *username* dan *password* sama dengan data yang di *database* |
|  | 1. Terdapat kondisi jika *username* dan *password* sama dengan data yang ada di *database,* maka masuk ke halaman utama aplikasi *web*, tetapi jika *username* dan *password* tidak terdapat atau tidak sama di *database* maka sistem akan kembali ke halaman masuk |

1. Sekenario *Use Case* Mengelola Transaksi

Nama *Use Case* : Mengelola Transaksi

Aktor : Kepala Toko / Kasir

Tipe : *Primary*

Prekondisi : Memasukkan *username* dan *password*

Tujuan : Mengelola Data Transaksi

Tabel 4.3 Sekenario *Use Case* Mengelola Transaksi

|  |  |
| --- | --- |
| Aktor | Sistem Aplikasi |
| 1. Kepala Toko / Kasir masuk ke halaman utama web | 1. Menampilkan Halaman Utama dengan daftar transaksi barang |
| 1. Kepala Toko mengakses menu Transaksi untuk melakukan aksi edit, dan hapus data transaksi | 1. Sistem melakukan perubahan data di-*database* dan menyimpannya |

1. Sekenario *Use Case* Mengelola Data Produk

Nama *Use Case* : Mengelola Data Produk

Aktor : Kepala Toko

Tipe : *Primary*

Prekondisi : Memasukkan *username* dan *password*

Tujuan : Mengelola Data Produk

Tabel 4.4 Sekenario *Use Case* Mengelola Data Produk

|  |  |
| --- | --- |
| Aktor | Sistem Aplikasi |
| 1. Kepala Toko masuk ke halaman utama web | 1. Menampilkan Halaman Utama dengan daftar data produk |
| 1. Kepala Toko mengakses menu data produk untuk melakukan aksi edit, dan hapus data produk | 1. Sistem melakukan perubahan data di-*database* dan menyimpannya |

1. Sekenario Use Case Mengelola Laporan

Nama Use Case : Mengelola Laporan

Aktor : Kepala Toko

Tipe : *Primary*

Prekondisi : Memasukkan *username* dan *password*

Tujuan : Mengelola Laporan

Tabel 4.5 Sekenario *Use Case* Mengelola Laporan

|  |  |
| --- | --- |
| Aktor | Sistem Aplikasi |
| 1. Kepala Toko masuk ke halaman utama web | 1. Menampilkan Halaman Utama dengan daftar menu laporan |
| 1. Kepala Toko mengakses menu laporan untuk melakukan aksi edit, dan hapus data laporan | 1. Sistem melakukan perubahan data di-*database* dan menyimpannya |

1. Sekenario *Use Case* Analisa Perkembangan

Nama Use Case : Melihat Analisa Perkembangan

Aktor : Kepala Toko

Tipe : *Primary*

Prekondisi : Memasukkan *username* dan *password*

Tujuan : Melihat Perkembangan Perusahaan

Tabel 4.6 Sekenario *Use Case* Analisa Perkembangan

|  |  |
| --- | --- |
| Aktor | Sistem Aplikasi |
| 1. Kepala Toko masuk ke halaman utama web | 1. Menampilkan Halaman Utama dengan daftar Analisa Perkembangan |
| 1. Kepala Toko mengakses menu untuk melihat perkembangan | 1. Sistem melakukan perhitungan perkembangan |

1. Sekenari *Use Case* Mengelola Pengguna

Nama Use Case : Mengelola Pengguna

Aktor : Kepala Toko

Tipe : *Primary*

Prekondisi : Memasukkan *username* dan *password*

Tujuan : Mengelola data pengguna aplikasi

Tabel 4.7 Sekenario *Use Case* Mengelola Pengguna

|  |  |
| --- | --- |
| Aktor | Sistem Aplikasi |
| 1. Kepala Toko masuk ke halaman utama web | 1. Menampilkan Halaman Utama dengan daftar Hak Akses |
| 1. Kepala Toko mengakses menu untuk mengelola edit, tambah, hapus hak akses pengguna | 1. Sistem melakukan perubahan data di-*database* dan disimpan |

1. Sekenario *Use Case* Keluar

Nama *Use Case* : Masuk

Aktor : Kepala toko / Kasir

Tipe : *Primary*

Prekondisi : Masuk ke halaman web

Tujuan : Keluar dari sistem aplikasi web

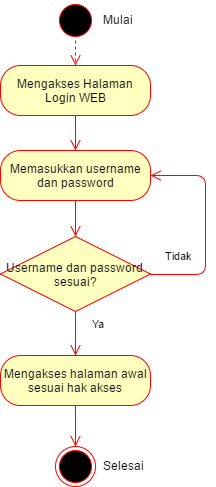
Tabel 4.8 Sekenario *Use Case* Masuk

|  |  |
| --- | --- |
| Aktor | Sistem Aplikasi |
| 1. Kepala Toko atau Kasir masuk dalam halaman web | 1. Menamplikan halaman menu |
| 1. Mengakses menu keluar | 1. Sistem menghapus semua sesi saat masuk dan keluar dari menu halaman utama web |

* 1. *Activity* *Diagram*

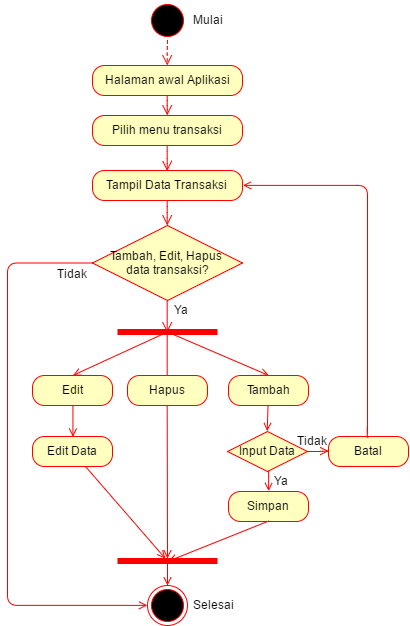
*Activity diagram* memberikan suatu gambaran ilustrasi alur dari setiap fungsi yang ada pada sistem.

1. *Activity diagram* masuk

**

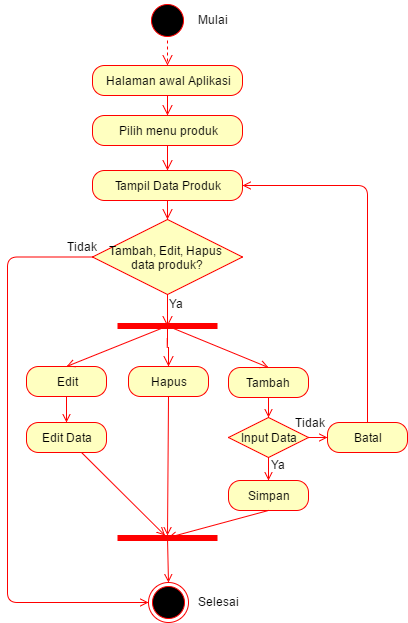
Gambar 4.2 *Activity diagram* masuk

1. *Activity diagram* mengelola transaksi



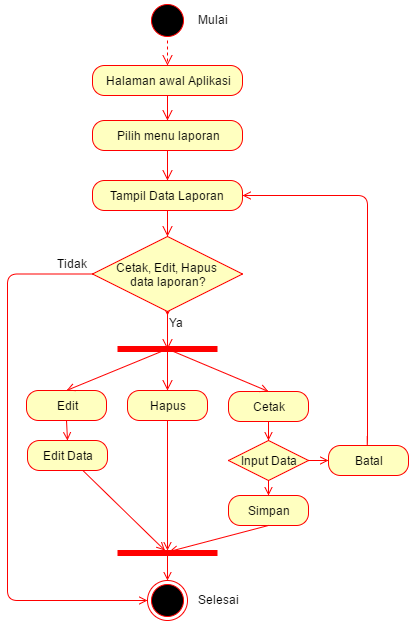
Gambar 4.3 *Activity diagram* mengelola transaksi

1. *Activity diagram* mengelola data produk



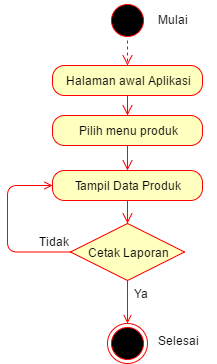
Gambar 4.4 *Activity diagram* mengelola data produk

1. *Activity diagram* mengelola laporan



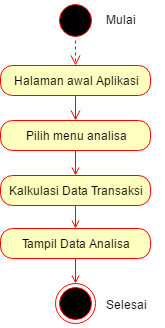
Gambar 4.5 *Activity diagram* mengelola laporan

1. *Activity diagram* cetak laporan



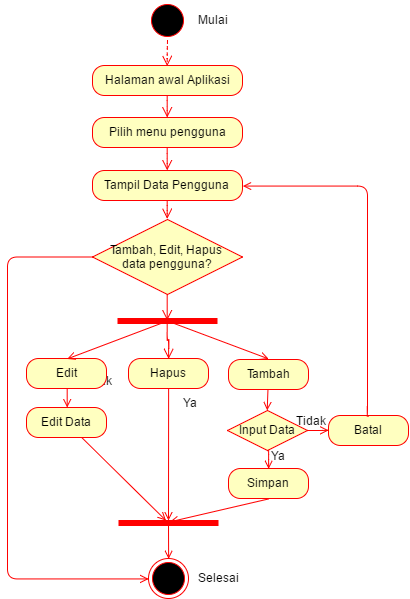
Gambar 4.6 *Activity diagram* cetak laporan

1. *Activity diagram* analisa pendapatan



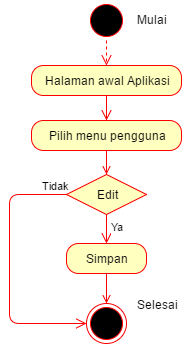
Gambar 4.7 *Activity diagram* analisa pendapatan

1. *Activity diagram* mengelola pengguna



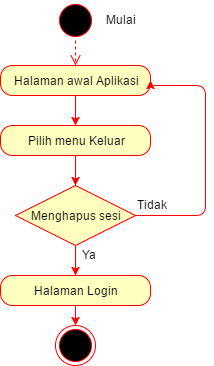
Gambar 4.8 *Activity diagram* mengelola pengguna

1. *Activity diagram* *edit password*



Gambar 4.9 *Activity diagram* *edit password*

1. *Activity diagram* keluar

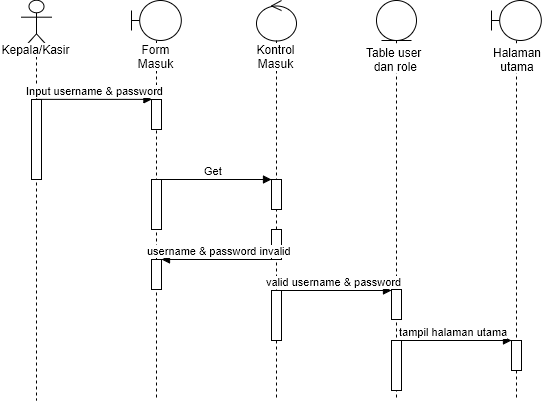


Gambar 4.10 *Activity diagram* Keluar

* 1. *Sequence Diagram*

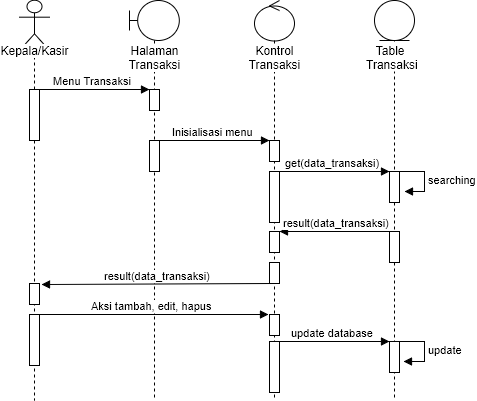
*Sequence diagram* menjelaskan interaksi antar objek-objek yang disusun dalam urutan waktu. Dalam sistem ini terdapat *sequence diagram* yang merupakan penjabaran dari masing-masing *usecase* yang ada pada *usecase diagram*.

1. *sequence diagram* masuk



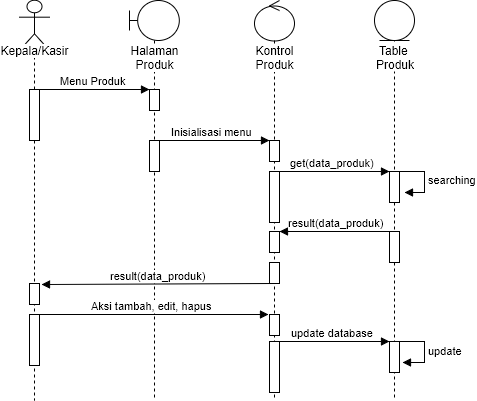
Gambar 4.11 *sequence diagram* masuk

1. *sequence diagram* transaksi



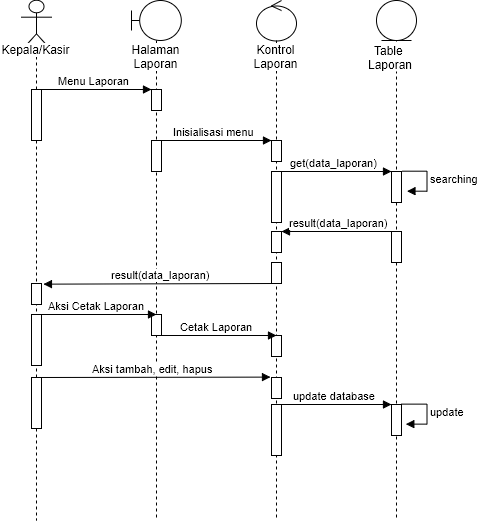
Gambar 4.12 *sequence diagram* transaksi

1. *sequence diagram* produk



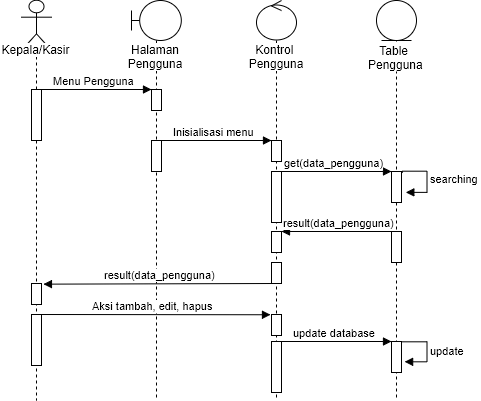
Gambar 4.13 *sequence diagram* produk

1. *sequence diagram* laporan



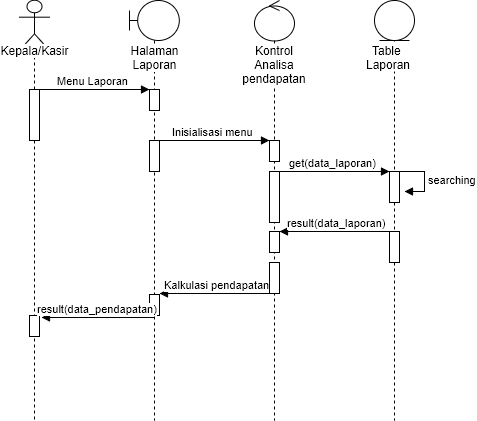
Gambar 4.14 *sequence diagram* laporan

1. *sequence diagram* pengguna



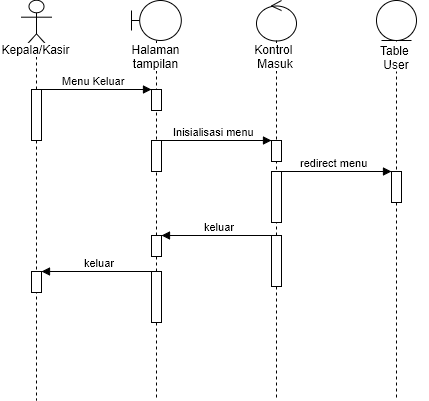
Gambar 4.15 *sequence diagram* pengguna

1. *sequence diagram* analisa pendapatan



Gambar 4.16 *sequence diagram* analisa pendapatan

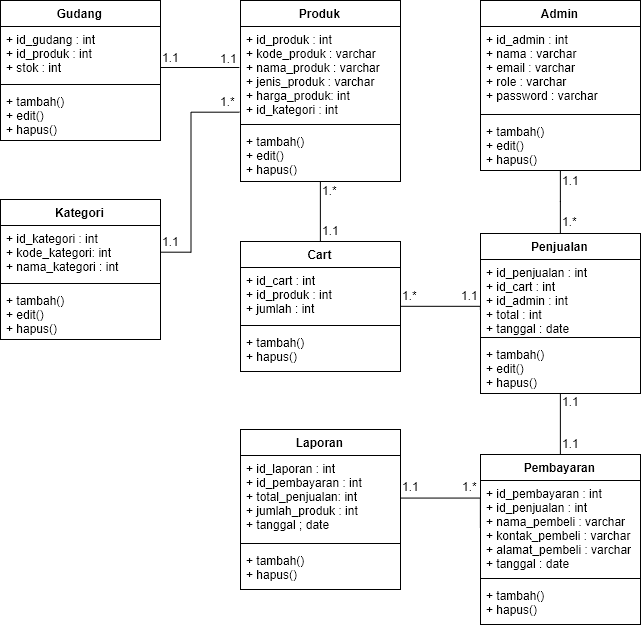
1. *sequence diagram* keluar



Gambar 4.17 *sequence diagram* keluar

* 1. *Class Diagram*

*Class Diagram* memperlihatkan hubungan antar kelas dan menggambarkan keadaan (atribut/properti) dari suatu sistem.



1. **Perancangan *Database***

Perancangan database dimaksud untuk membantu pembentukan database ketika mengimplikasikan system dalam bentuk aplikasi. Perancangan database ini juga menggambarkan tingkat relasi antar tabel satu dengan yang lainnya untuk membentuk table baru.

1. Rancangan Tabel Admin

Rancangan tabel admin akan memberikan standar pembentukan tabel untuk admin, baik itu *field* maupun atributnya.

Tabel 4.9 perancangan tabel admin

|  |  |  |
| --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** |
| id\_admin | Int | Primary key |
| email\_admin | Varchar |  |
| nama\_admin | Varchar |  |
| role\_admin | Varchar |  |
| password\_admin | Varchar |  |

1. Rancangan Tabel Produk

Rancangan tabel produk akan memberikan standar pembentukan tabel untuk produk, baik itu *field* maupun atributnya.

Tabel 4.10 perancangan tabel produk

|  |  |  |
| --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** |
| id\_produk | Int | Primary key |
| kode\_produk | Varchar |  |
| nama\_produk | Varchar |  |
| jenis\_produk | Varchar |  |
| harga\_produk | Int |  |
| id\_kategory | int | Foreign key |

1. Rancangan Tabel Kategori

Rancangan tabel kategori akan memberikan standar pembentukan tabel untuk kategori, baik itu *field* maupun atributnya.

Tabel 4.11 perancangan tabel kategori

|  |  |  |
| --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** |
| id\_kategori | Int | Primary key |
| kode\_kategori | Varchar |  |
| nama\_kategori | Varchar |  |

1. Rancangan Tabel Gudang

Rancangan tabel gudang akan memberikan standar pembentukan tabel untuk gudang, baik itu *field* maupun atributnya.

Tabel 4.12 perancangan tabel gudang

|  |  |  |
| --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** |
| id\_gudang | Int | Primary key |
| id\_produk | Int | Foreign key |
| stok | int |  |

1. Rancangan Tabel *Cart*

Rancangan tabel *cart* akan memberikan standar pembentukan tabel untuk *cart*, baik itu *field* maupun atributnya.

Tabel 4.13 perancangan tabel cart

|  |  |  |
| --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** |
| id\_cart | Int | Primary key |
| id\_produk | Varchar | Foreign key |
| jumlah | Varchar |  |

1. Rancangan Tabel Penjualan

Rancangan tabel kategori akan memberikan standar pembentukan tabel untuk kategori, baik itu *field* maupun atributnya.

Tabel 4.14 perancangan tabel penjualan

|  |  |  |
| --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** |
| id\_penjualan | Int | Primary key |
| id\_cart | Int | Foreign key |
| id\_admin | Int | Foreign key |
| total | Int |  |
| tanggal | Date |  |

1. Rancangan Tabel Pembayaran

Rancangan tabel penjualan akan memberikan standar pembentukan tabel untuk penjualan, baik itu *field* maupun atributnya.

Tabel 4.15 perancangan tabel pembayaran

|  |  |  |
| --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** |
| id\_pembayaran | Int | Primary key |
| id\_penjualan | Int | Foreign key |
| nama\_pembeli | Varchar |  |
| kontak\_pembeli | Varchar |  |
| alamat\_pembeli | Varchar |  |
| tanggal | Date |  |

1. Rancangan Tabel Laporan

Rancangan tabel laporan akan memberikan standar pembentukan tabel untuk laporan, baik itu *field* maupun atributnya.

Tabel 4.16 perancangan tabel laporan

|  |  |  |
| --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** |
| id\_laporan | Int | Primary key |
| id\_pembayaran | Int | Foreign key |
| total\_penjualan | Int |  |
| jumlah\_produk | Int |  |
| tanggal | Date |  |

1. **Perancangan Antar Muka**

Berdasarkan perancangan aplikasi yang dibuat, maka akan di design perancangan aplikasi. Perancangan ini akan menjadi acuan bagi pihak yang akan mengembangkan dan menggunakan aplikasi.