**APLIKASI PENJUALAN BARANG BAHAN BANGUNAN**

**(*POINT OF SALE*)BERBASIS WEB DI TOKO**

**BAHAN BANGUNAN SATRIA JAYA**

**KERJA PRAKTEK**

Diajukan Sebagai Salah Satu Syarat Kelulusan Mata Kuliah Kerja Praktek

Program Studi Teknik Informatika Fakultas Teknik Universitas Surya Kancana



**OLEH**

NAMA : HARIS ANWAR

NPM : 5520118042

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS SURYA KANCANA**

**CIANJUR**

**2021-2022**

# LEMBAR PERSETUJUAN

# LEMBAR KERJA PERSETUJUAN

Nama : HARIS ANWAR

NPM : 5520118042

Program Studi : Fakultas Teknik Informatika

Judul : APLIKASI PENJUALAN BARANG BAHAN BANGUNAN *(POINT OF SALE)* BERBASIS WEB DI TOKO BAHAN BANGUNAN SATRIA JAYA

Telah disetujui untuk disidangkan pada Sidang Kerja Praktek Pada Program Sarjana Strata-1 (S1), Program Studi Teknik Informatika Fakultas Teknik Universitas Suryakancana.

Cianjur, 27 Februari 2021

Pembimbing Kerja Praktek Pembimbing

Lapangan

**Hasbu Naim.S.Skom,MKom Hasbu Naim.S.SKom,MKom**

**NIDN.90290904394093 NIDN.90290904394093**

# LEMBAR PENGESAHAN

# LAPORAN KERJA PRAKTEK

Nama : HARIS ANWAR

NPM : 5520118042

Program Studi : Fakultas Teknik Informatika

Judul : APLIKASI PENJUALAN BARANG BAHAN BANGUNAN *(POINT OF SALE)* BERBASIS WEB DI TOKO BAHAN BANGUNAN SATRIA JAYA

Telah disidangkan dan dinyatakan Lulus Sidang Praktek pada Program Sarjana Strata-1 (S1), Program Studi Teknik Informatika Fakultas Teknik Universitas Suryakancana pada tanggal …..

Nama Penguji Tanda Tangan

1. Nama
2. Nama

Mengetahui:

Ketua Prodi Studi

Teknik Informatika

**Tarmin Abdulghani ST, MT.**

**NIDN. 04018750**

# LEMBAR PERNYATAAN KEASLIAN

# LAPORAN KERJA PRAKTEK

Nama : HARIS ANWAR

NPM : 5520118042

Program Studi : Fakultas Teknik Informatika

Judul : APLIKASI PENJUALAN BARANG BAHAN BANGUNAN *(POINT OF SALE)* BERBASIS WEB DI TOKO BAHAN BANGUNAN SATRIA JAYA

Dengan ini penulis menyatakan dalam laporan penelitian ini tidak terdapat karya yang pernah diajukan untuk memperoleh kelulusan **Kerja Praktek ataupun gelar kesarjanaan** di suatu Perguruan Tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh oranglain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka

Cianjur, 27 Februari 2021

Materai

**Haris Anwar**

**NPM. 5520118042**

# ABSTRAK

Pencatatan barang masuk, barang keluar, serta transaksi merupakan bagian yang dilakukan dalam operasional suatu toko, terutama toko yang bergerak di bidang perdagangan. TOKO BARANG BAHAN BANGUNAN SATRIA JAYA merupa suatu toko yang bergerak di bidang perdagangan, pencatatan data barang masuk, barang keluar, serta transaksi dalam operasional toko.

Pada saat ini pencatatan barang masuk dari *supplier* ataupun barang keluar berupa *return* masih dicatat secara manual, begitu pun setiap transaksi penjualan dicatat secara manual, hal tersebut menimbulkan beberapa masalah diantaranya yaitu hilangnya data atau dokumen bukti penjualan ataupun data barang berupa kertas atau fisik, serta pada saat transaksi penjualan barang terjadi kesalahan penghitungan harga, klasifikasi atau kategori barang yang belum tersusun, membutuhkan waktu lama saat pencatatan barang,. Mengacu pada pentingnya proses bisnis yang berjalan, maka dibutuhkan suatu aplikasi penjualan barang (*point of sale*) yang dapat menangani permasalahan tersebut, sehingga proses bisnis dapat berjalan menjadi lebih efektif.

Dalam pembuatan perangkat lunak ini, digunakan metode pengembangan sistem berdasarkan paradigma *prototype* yang terdiri kominikasi, perencanaan, pemodelan. Bahasa pemrograman yang digunakan untuk membangun aplikasi penjualan barang (*point of sale*) yaitu menggunakan bahasa pemrograman *php, jquery, javascript,* dengan menggunakan framework *codeignaiter* dan *bootsrap* dan DBMS MYSQL dengan perangkat lunak XAMPP,. Untuk metoda perancangan sistem menggunakan konsep perancangan beriorentasi object dimana menggunakan *tools* perancangan yaitu UML.

Hasil dari perancangan pada penelitian ini, di implementasikan menjadi perangkat lunak ataupun aplikasi penjualan barang (*point of sale*) berbasis web, yang berfungsi untuk pengelolaan barang masuk, ataupun barang keluar, kategori barang, serta transaksi yang terkomputerisasi. Dengan pembuatan perangkat lunak ini diharapkan proses bisnis dapat berjalan dengan efektif.

Kata Kunci : aplikasi, barang masuk, barang keluar, transaksi, *php, MyQL, Prototype*

# KATA PENGANTAR

Puji syukur penulis panjatkan kehadiran Allah SWT, karena hanya dengan limpahan rahmat dan ridha-Nya penulis dapat menyelesaikan laporan kerja praktek yang berjudul “APLIKASI PENJUALAN BARANG (*POINT OF SALE*) BAHAN BANGUNAN BERBASIS WEB DI TOKO BAHAN BANGUNAN SATRIA JAYA”.

Laporan ini merupakan salah satu persyaratan kelulusan pada matakuliah kerja praktek di Program Studi Teknik Informatika di Fakultas Teknik Universitas Suryakancana Cianjur.

Penelitian dan penyusunan laporan kerja praktek ini adalah suatu proses yang tidak lepas dari bantuan, bimbingan, serta dukungan berbagai pihak. Untuk itu penulis menguncapkan rasa terima kasih dan penghargaan yang sebesar-besanya kepada:

1. Bapak. Tarmin Abdulghani ST.MT selaku ketua jurusan Fakultas Teknik Informatika
2. Bapak.Hasbu Naim S.S.Kom.,MKom selaku pembimbing kerja praktek
3. Ibu Siti Nazillah ST.M.Kom selaku koordinator kerja praktek
4. Bapak. H. Adhe S.E selaku pemilik Toko Bahan Bangunan Satria Jaya
5. Kedua Orangtua yang selalu memberi dukungan tanpa henti dan semangat kepada penulis

Penulis menyadari bahwa laporan ini masih jauh dari sempurna, oleh karena itu penulis mengharapkan masukan berupa saran dari semua pihak bagi perbaikan dan pengembangan penelitian ini. Akhir kata penulis berharap agar laporan kerja praktek ini dapat memberikan manfaat bagi perkembangan teknologi informasi saat ini.

Cianjur, 27 April 2021

Penulis

# DAFTAR ISI

[LEMBAR PERSETUJUAN i](#_Toc72484151)

[LEMBAR KERJA PERSETUJUAN i](#_Toc72484152)

[LEMBAR PENGESAHAN ii](#_Toc72484153)

[LAPORAN KERJA PRAKTEK ii](#_Toc72484154)

[LEMBAR PERNYATAAN KEASLIAN iii](#_Toc72484155)

[LAPORAN KERJA PRAKTEK iii](#_Toc72484156)

[ABSTRAK iv](#_Toc72484157)

[KATA PENGANTAR v](#_Toc72484158)

[DAFTAR ISI vi](#_Toc72484159)

[DAFTAR TABEL x](#_Toc72484160)

[DAFTAR GAMBAR xi](#_Toc72484161)

[DAFTAR SIMBOL xv](#_Toc72484162)

[A. Simbol *Use Case Diagram* xv](#_Toc72484163)

[B. Simbol *Swimlane Diagram* xvi](#_Toc72484164)

[C. Simbol *Class Diagram* xvii](#_Toc72484165)

[D. Simbol *State Diagram* xviii](#_Toc72484166)

[E. Simbol *Sequence Diagram* xix](#_Toc72484167)

[F. Simbol *Entity Relationship Diagram* xxi](#_Toc72484168)

[*G.* Simbol *Deployment Diagram* xxii](#_Toc72484169)

[BAB I 1](#_Toc72484170)

[PENDAHULUAN 1](#_Toc72484171)

[1.1 Latar Belakang 1](#_Toc72484172)

[1.2 Rumusan Masalah 2](#_Toc72484173)

[1.3 Maksud Dan Tujuan 2](#_Toc72484174)

[1.4 Batasan Masalah 3](#_Toc72484175)

[1.5 Batasan Masalah 3](#_Toc72484176)

[1.6 Tempat dan Waktu 6](#_Toc72484177)

[1.7 Sistematika Penulisan 6](#_Toc72484178)

[BAB II 8](#_Toc72484179)

[TINJAUAN PUSTAKA 8](#_Toc72484180)

[2.1 Tinjauan Pustaka 8](#_Toc72484181)

[2.1.1 Sejarah Singkat Toko Satria Jaya 8](#_Toc72484182)

[2.1.2 Struktur Organisasi Satria Jaya 9](#_Toc72484183)

[2.1.3 *Job* Deskripsi 9](#_Toc72484184)

[2.2 Landasan Teori 10](#_Toc72484185)

[2.2.1 Definisi Aplikasi 10](#_Toc72484186)

[2.2.2 Definisi Penjualan 10](#_Toc72484187)

[2.2.3 Definisi Barang 10](#_Toc72484188)

[2.2.4 Definisi Bahan 11](#_Toc72484189)

[2.2.5 Definisi Bangunan 11](#_Toc72484190)

[2.2.6 Definisi POS *(Point Of Sale)* 11](#_Toc72484191)

[2.2.7 Definisi Toko 12](#_Toc72484192)

[2.2.8 Definisi PHP (*Hypertext Preprocessor*) 12](#_Toc72484193)

[2.2.9 Definisi CSS (*Cascading Style Sheet*) 13](#_Toc72484194)

[2.2.10 Definisi HTML (*Hyper Text Markup Language*) 13](#_Toc72484195)

[2.2.11 Definisi Basis Data (*Database*) 14](#_Toc72484196)

[2.2.12 Definisi DBMS (*Database Management System*) 14](#_Toc72484197)

[2.2.13 Definisi UML (*Unified Modeling Language*) 14](#_Toc72484198)

[2.2.14 Definisi Motode Pengujian (*Blackbox*) 16](#_Toc72484199)

[2.3 *Tools* yang digunakan 17](#_Toc72484200)

[2.3.1 *Bootstrap* 17](#_Toc72484201)

[2.3.2 Balsamiq *Mockup* 17](#_Toc72484202)

[2.3.3 *CodeIgniter* 18](#_Toc72484203)

[2.3.4 *Google Chrome* 18](#_Toc72484204)

[2.3.5 *Visual Studio Code* 18](#_Toc72484205)

[2.3.6 *MySQL* 19](#_Toc72484206)

[2.3.7 *Xampp* 19](#_Toc72484207)

[2.3.8 *Visual Paradigm* 20](#_Toc72484208)

[BAB III 21](#_Toc72484209)

[ANALISIS DAN PERANCANGAN SISTEM 21](#_Toc72484210)

[3.1 Analisis Sistem 21](#_Toc72484211)

[3.2 Analisis Masalah 22](#_Toc72484212)

[3.3 Analisis sitem yang sedang berjalan 22](#_Toc72484213)

[3.4 Analisis kebutuhan non fungsional 25](#_Toc72484214)

[3.4.1 Analisa Pengguna 25](#_Toc72484215)

[3.4.2 Analisa kebutuhan perangkat keras 25](#_Toc72484216)

[3.4.3 Analisa kebutuhan perangkat lunak 26](#_Toc72484217)

[3.5 Analisis kebutuhan fungsional 27](#_Toc72484218)

[3.5.1 *Actor List* 27](#_Toc72484219)

[3.5.2 *Usecase Scenario* (Naratif) 28](#_Toc72484220)

[3.5.3 *Usecase List* 39](#_Toc72484221)

[3.5.4 *Usecase Diagram* 41](#_Toc72484222)

[3.5.5 *Swimline Diagram* 42](#_Toc72484223)

[3.5.6 *Class Diagram* 59](#_Toc72484224)

[3.5.7 *State Diagram* 61](#_Toc72484225)

[3.5.8 *Sequence Diagram* 69](#_Toc72484226)

[3.5.9 *Perancangan ERD (Entity Relationship Diagram)* 84](#_Toc72484227)

[3.5.10 *Struktur Tabel* 86](#_Toc72484228)

[3.5.11 *Perancangan Struktur Menu* 90](#_Toc72484229)

[3.5.12 *Perancangan Antarmuka* 91](#_Toc72484230)

[3.5.13 *Deployment Diagram* 131](#_Toc72484231)

[BAB IV 133](#_Toc72484232)

[IMPLEMENTASI PERANCANGAN 133](#_Toc72484233)

[DAN PENGUJIAN SISTEM 133](#_Toc72484234)

[4.1 Implementasi 133](#_Toc72484235)

[4.1.1 Kebutuhan Perangkat Keras 133](#_Toc72484236)

[4.1.2 Kebutuhan Perangkat Lunak 134](#_Toc72484237)

[4.1.3 Kebutuhan Pengguna 134](#_Toc72484238)

[4.2 Tampilan antarmuka 135](#_Toc72484239)

[4.3 Pengujian 175](#_Toc72484240)

[BAB V 202](#_Toc72484241)

[KESIMPULAN DAN SARAN 202](#_Toc72484242)

[1.1 Kesimpulan 202](#_Toc72484243)

[1.2 Saran 203](#_Toc72484244)

[DAFTAR PUSTAKA 205](#_Toc72484245)

[LAMPIRAN 206](#_Toc72484246)

# DAFTAR TABEL

[Tabel 1. 1 Jadwal Kegiatan 6](#_Toc72484472)

[Tabel 3. 1 Tabel Kebutuhan Perangkat Keras 26](#_Toc72484487)

[Tabel 3. 2 Actor List 27](#_Toc72484488)

[Tabel 3. 3 *Usecase Scenario* Kelola *Login* 28](#_Toc72484489)

[Tabel 3. 4 *Usecase Scenario* Kelola *Supplier* 29](#_Toc72484490)

[Tabel 3. 5 *Usecase Scenario* Kelola *Customer* 30](#_Toc72484491)

[Tabel 3. 6 *Usecase Scenario* Kelola Produk 31](#_Toc72484492)

[Tabel 3. 7 *Usecase Scenario* Kelola *Stock* Barang 33](#_Toc72484493)

[Tabel 3. 8 *Usecase Scenario* Kelola *User* 35](#_Toc72484494)

[Tabel 3. 9 *Usecase Scenario* Kelola Laporan 36](#_Toc72484495)

[Tabel 3. 10 Usecase Scenario Kelola Penjualan 38](#_Toc72484496)

[Tabel 3. 11 *Usecase List* 39](#_Toc72484497)

[Tabel 3. 12 Tabel *User* 86](#_Toc72484498)

[Tabel 3. 13 Tabel *Stock* 86](#_Toc72484499)

[Tabel 3. 14 Tabel *Supplier* 87](#_Toc72484500)

[Tabel 3. 15 Tabel Detail Penjualan 87](#_Toc72484501)

[Tabel 3. 16 Tabel Item 87](#_Toc72484502)

[Tabel 3. 17 Tabel Satuan 88](#_Toc72484503)

[Tabel 3. 18 Tabel Kategori 88](#_Toc72484504)

[Tabel 3. 19 Tabel Detail Penjualan 88](#_Toc72484505)

[Tabel 3. 20 Tabel *Customer* 89](#_Toc72484506)

[Tabel 3. 21 Tabel Transaksi 89](#_Toc72484507)

[Tabel 4. 1 Pengujian Black *Box* 176](#_Toc72484508)

# DAFTAR GAMBAR

[Gambar 1. 1 Prototype Pressman (Pressman, 2015:42) 4](#_Toc72484811)

[Gambar 3. 1 Analisis Proses Bisnis Pengelolaan *Stock* Barang 23](#_Toc72484820)

[Gambar 3. 2 Analisis Proses Bisnis dari Pelanggan ke Toko Satria Jaya 24](#_Toc72484821)

[Gambar 3. 3 *Usecase Diagram* 41](#_Toc72484822)

[Gambar 3. 4 *Swimeline Diagram* *Login* 42](#_Toc72484823)

[Gambar 3. 5 *Swimeline Diagram Supplier* 44](#_Toc72484824)

[Gambar 3. 6 *Swimeline Diagram Produk* 46](#_Toc72484825)

[Gambar 3. 7 *Swimeline Diagram Stock* Barang 48](#_Toc72484826)

[Gambar 3. 8 *Swimeline Diagram User* 50](#_Toc72484827)

[Gambar 3. 9 *Swimeline Diagram* Laporan 52](#_Toc72484828)

[Gambar 3. 10 *Swimeline Diagram Customer* 55](#_Toc72484829)

[Gambar 3. 11 *Swimeline Diagram* Penjualan 57](#_Toc72484830)

[Gambar 3. 12 *Class Diagram* Toko Satria Jaya 59](#_Toc72484831)

[Gambar 3. 13 *State Diagram Login* 61](#_Toc72484832)

[Gambar 3. 14 *State Diagram User* 62](#_Toc72484833)

[Gambar 3. 15 *State Diagram Supplier* 63](#_Toc72484834)

[Gambar 3. 16 *State Diagram Customer* 64](#_Toc72484835)

[Gambar 3. 17 *State Diagram* Penjualan 65](#_Toc72484836)

[Gambar 3. 18 *State Diagram Stock* Barang 66](#_Toc72484837)

[Gambar 3. 19 *State Diagram* Produk 67](#_Toc72484838)

[Gambar 3. 20 *Sequence Diagram Login* 69](#_Toc72484839)

[Gambar 3. 21 *Sequence Diagram Supplier* 70](#_Toc72484840)

[Gambar 3. 22 *Sequence Diagram Supplier* 72](#_Toc72484841)

[Gambar 3. 23 *Sequence Diagram* Produk 74](#_Toc72484842)

[Gambar 3. 24 *Sequence Diagram Stock* Barang 76](#_Toc72484843)

[Gambar 3. 25 *Sequence Diagram User* 78](#_Toc72484844)

[Gambar 3. 26 *Sequence Diagram* Laporan 80](#_Toc72484845)

[Gambar 3. 27 *Sequence Diagram* Penjualan 82](#_Toc72484846)

[Gambar 3. 28 *ERD* Toko Satria Jaya 84](#_Toc72484847)

[Gambar 3. 29 Struktur Menu Toko Satria Jaya 90](#_Toc72484848)

[Gambar 3. 30 Antarmuka Halaman *Login* 91](#_Toc72484849)

[Gambar 3. 31 Antarmuka Halaman Admin 92](#_Toc72484850)

[Gambar 3. 32 Antarmuka Halaman Pengguna 93](#_Toc72484851)

[Gambar 3. 33 Antarmuka Halaman Tambah Pengguna 94](#_Toc72484852)

[Gambar 3. 34 Antarmuka Halaman Ubah Pengguna 95](#_Toc72484853)

[Gambar 3. 35 Antarmuka Halaman Admin Menu Hapus Pengguna 96](#_Toc72484854)

[Gambar 3. 36 Antarmuka Halaman Admin Laporan Penjualan 97](#_Toc72484855)

[Gambar 3. 37 Antarmuka Halaman Admin Laporan *Stock* Barang 98](#_Toc72484856)

[Gambar 3. 38 Antarmuka Halaman Kasir 99](#_Toc72484857)

[Gambar 3. 39 Antarmuka Halaman Kasir Menu Pelanggan 100](#_Toc72484858)

[Gambar 3. 40 Antarmuka Halaman Kasir Menu Tambah Pelanggan 101](#_Toc72484859)

[Gambar 3. 41 Antarmuka Halaman Kasir Menu Ubah Pelanggan 102](#_Toc72484860)

[Gambar 3. 42 Antarmuka Halaman Kasir Menu Hapus Pelanggan 103](#_Toc72484861)

[Gambar 3. 43 Antarmuka Halaman Kasir Menu Penjualan 104](#_Toc72484862)

[Gambar 3. 44 Antarmuka Halaman Kasir Menu Tambah Penjualan 105](#_Toc72484863)

[Gambar 3. 45 Antarmuka Halaman Kasir Menu Ubah Penjualan 106](#_Toc72484864)

[Gambar 3. 46 Antarmuka Halaman Kasir Menu Hapus Penjualan 107](#_Toc72484865)

[Gambar 3. 47 Antarmuka Halaman Kasir Menu Proses Penjualan 108](#_Toc72484866)

[Gambar 3. 48 Antarmuka Halaman Admin Menu Barang Masuk 109](#_Toc72484867)

[Gambar 3. 49 Antarmuka Halaman Admin Menu Tambah Barang Masuk 110](#_Toc72484868)

[Gambar 3. 50 Antarmuka Halaman Admin Menu Hapus Barang Masuk 111](#_Toc72484869)

[Gambar 3. 51 Antarmuka Halaman Kasir Menu Barang Keluar 112](#_Toc72484870)

[Gambar 3. 52 Antarmuka Halaman Kasir Menu Tambah Barang Keluar 113](#_Toc72484871)

[Gambar 3. 53 Antarmuka Halaman Admin Menu Hapus Barang Keluar 114](#_Toc72484872)

[Gambar 3. 54 Antarmuka Halaman Admin Menu *Supplier* 115](#_Toc72484873)

[Gambar 3. 55 Antarmuka Halaman Admin Menu Tambah *Supplier* 116](#_Toc72484874)

[Gambar 3. 56 Antarmuka Halaman Admin Menu Ubah *Supplier* 117](#_Toc72484875)

[Gambar 3. 57 Antarmuka Halaman Admin Menu Hapus Supplier 118](#_Toc72484876)

[Gambar 3. 58 Antarmuka Halaman Admin Menu Kategori Produk 119](#_Toc72484877)

[Gambar 3. 59 Antarmuka Halaman Admin Menu Tambah Kategori Produk 120](#_Toc72484878)

[Gambar 3. 60 Antarmuka Halaman Admin Menu Ubah Kategori Produk 121](#_Toc72484879)

[Gambar 3. 61 Antarmuka Halaman Admin Menu Hapus Kategori Produk 122](#_Toc72484880)

[Gambar 3. 62 Antarmuka Halaman Admin Menu Satuan Produk 123](#_Toc72484881)

[Gambar 3. 63 Antarmuka Halaman Admin Menu Tambah Satuan Produk 124](#_Toc72484882)

[Gambar 3. 64 Antarmuka Halaman Admin Menu Ubah Satuan Produk 125](#_Toc72484883)

[Gambar 3. 65 Antarmuka Halaman Admin Menu Hapus Satuan Produk 126](#_Toc72484884)

[Gambar 3. 66 Antarmuka Halaman Admin Menu Item Produk 127](#_Toc72484885)

[Gambar 3. 67 Antarmuka Halaman Admin Menu Tambah Item Produk 128](#_Toc72484886)

[Gambar 3. 68 Antarmuka Halaman Admin Menu Ubah Item Produk 129](#_Toc72484887)

[Gambar 3. 69 Antarmuka Halaman Admin Menu Hapus Item Produk 130](#_Toc72484888)

[Gambar 3. 70 *Deployement Diagram* 131](#_Toc72484889)

[Gambar 4. 1 Antarmuka Halam *Login* 135](#_Toc72484890)

[Gambar 4. 2 Antarmuka Halaman Admin 136](#_Toc72484891)

[Gambar 4. 3 Antarmuka Halaman Pengguna 137](#_Toc72484892)

[Gambar 4. 4 Antarmuka Halaman Tambah Pengguna 138](#_Toc72484893)

[Gambar 4. 5 Antarmuka Halaman Ubah Pengguna 139](#_Toc72484894)

[Gambar 4. 6 Antarmuka Halaman Admin Menu Hapus Pengguna 140](#_Toc72484895)

[Gambar 4. 7 Antarmuka Halaman Admin Laporan Penjualan 141](#_Toc72484896)

[Gambar 4. 8 Antarmuka Halaman Admin Laporan Stock Barang 142](#_Toc72484897)

[Gambar 4. 9 Antarmuka Halaman Kasir 143](#_Toc72484898)

[Gambar 4. 10 Antarmuka Halaman Kasir Menu Pelanggan 144](#_Toc72484899)

[Gambar 4. 11 Antarmuka Halaman Kasir Menu Tambah Pelanggan 145](#_Toc72484900)

[Gambar 4. 12 Antarmuka Halaman Kasir Menu Ubah Pelanggan 146](#_Toc72484901)

[Gambar 4. 13 Antarmuka Halaman Kasir Menu Hapus Pelanggan 147](#_Toc72484902)

[Gambar 4. 14 Antarmuka Halaman Kasir Menu Penjualan 148](#_Toc72484903)

[Gambar 4. 15 Antarmuka Halaman Kasir Menu Tambah Penjualan 149](#_Toc72484904)

[Gambar 4. 16 Antarmuka Halaman Kasir Menu Ubah Penjualan 150](#_Toc72484905)

[Gambar 4. 17 Antarmuka Halaman Kasir Menu Hapus Penjualan 151](#_Toc72484906)

[Gambar 4. 18 Antarmuka Halaman Kasir Menu Proses Penjualan 152](#_Toc72484907)

[Gambar 4. 19 Antarmuka Halaman Admin Menu Barang Masuk 153](#_Toc72484908)

[Gambar 4. 20 Antarmuka Halaman Admin Menu Tambah Barang Masuk 154](#_Toc72484909)

[Gambar 4. 21 Antarmuka Halaman Admin Menu Hapus Barang Masuk 155](#_Toc72484910)

[Gambar 4. 22 Antarmuka Halaman Kasir Menu Barang Keluar 156](#_Toc72484911)

[Gambar 4. 23 Antarmuka Halaman Admin Menu Tambah Barang Keluar 157](#_Toc72484912)

[Gambar 4. 24 Antarmuka Halaman Admin Menu Hapus Barang Keluar 158](#_Toc72484913)

[Gambar 4. 25 Antarmuka Halaman Admin Menu *Supplier* 159](#_Toc72484914)

[Gambar 4. 26 Antarmuka Halaman Admin Menu Tambah *Supplier* 160](#_Toc72484915)

[Gambar 4. 27 Antarmuka Halaman Admin Menu Ubah *Supplier* 161](#_Toc72484916)

[Gambar 4. 28 Antarmuka Halaman Admin Menu Hapus *Supplier* 162](#_Toc72484917)

[Gambar 4. 29 Antarmuka Halaman Admin Menu Kategori Produk 163](#_Toc72484918)

[Gambar 4. 30 Antarmuka Halaman Admin Menu Tambah Kategori Produk 164](#_Toc72484919)

[Gambar 4. 31 Antarmuka Halaman Admin Menu Ubah Kategori Produk 165](#_Toc72484920)

[Gambar 4. 32 Antarmuka Halaman Admin Menu Hapus Kategori Produk 166](#_Toc72484921)

[Gambar 4. 33 Antarmuka Halaman Admin Menu Satuan Produk 167](#_Toc72484922)

[Gambar 4. 34 Antarmuka Halaman Admin Menu Tambah Satuan Produk 168](#_Toc72484923)

[Gambar 4. 35 Antarmuka Halaman Admin Menu Ubah Satuan Produk 169](#_Toc72484924)

[Gambar 4. 36 Antarmuka Halaman Admin Menu Hapus Satuan Produk 170](#_Toc72484925)

[Gambar 4. 37 Antarmuka Halaman Admin Menu Item Produk 171](#_Toc72484926)

[Gambar 4. 38 Antarmuka Halaman Admin Menu Tambah Item Produk 172](#_Toc72484927)

[Gambar 4. 39 Antarmuka Halaman Admin Menu Ubah Item Produk 173](#_Toc72484928)

[Gambar 4. 40 Antarmuka Halaman Admin Menu Hapus Item Produk 174](#_Toc72484929)

# DAFTAR SIMBOL

# Simbol *Use Case Diagram*

|  |  |
| --- | --- |
| **Gambar** | **Penjelasan** |
|  | ***Usecase*** : adalah menggambarkan apa yang dilakukan aktor melalui transaksi atau fungsi dari awal sampai akhir. |
|  | ***Actor*** : adalah menggambarkan peran aktor baik itu manusia, media ataupun sistem lain. Disebut juga alat untuk komunikasi dengan bagian *usecase* dalam sistem. |
|  | ***Dependency*** : adalah mengambarkan relasi antar dua model elemen jika satu berubah akan menyebabkan perubahan pada elemen lain. |
|  | ***Include*** :adalah menggambarkan suatu *use case* yang tidak dapat dilakukan apabila *use case* lain belum dilakukan. |
|  | ***Extends*** : adalah menggambarkan suatu *use case*  dapat dilakukan tanpa ada persyaratan tertentu. |
|  | ***Generalization*** : adalah menggambarkan turunan dari suatu aktor, contohnya pelanggan ada pelanggan *member* dan *non member*. |
|  | ***System*** : adalah menggambarkan Lingkup suatu system |

# Simbol *Swimlane Diagram*

|  |  |
| --- | --- |
| **Gambar** | **Penjelasan** |
|  | ***Activity*** : adalah koordinasi eksekusi dari subordinat menggunakan kontrol atau *data flow model.* |
|  | ***Activity parameter node*** : adalah aksi yang diterima. |
|  | ***Action*** : adalah merepresentasikan tahapan tunggal. |
|  | ***Decision node*** : menggambarkan sebuah pengkondisian , dari satu aliran token yang datang menjadi beberapa aliran yang keluar |
|  | ***Marge node* :** Mengabungkan dari beberap aliran yang telah di pecah oleh decision node |
|  | ***Fork node :*** menerima satu aliran untuk di bagi ke beberapa aliran activity |
|  | ***Join node* :** Mengabungkan bebearap action yang telah di pecah fork node |
|  | ***Initial node*** : mengawali aktivitas. |
|  | ***Activity final node*** : mengakhiri semua aktivitas. |
|  | ***Expansion node*** : yang melewati batas. |
|  | ***Horizontal swimlane*** : digunakan untuk partisi *children.* |
|  | ***Control flow*** : aktivitas *node* setelah *node*  yang lain selesai. |

# Simbol *Class Diagram*

|  |  |
| --- | --- |
| **Gambar** | **Penjelasan** |
|  | ***Class*** : merepresentasikan *entity* dalam *system.* |
|  | ***Generalization*** : ada diantara dua kelas dimana satu kelas terspesifikasi. |
|  | ***Aggregation* :** Hubungan yang menggambarkan suatu merupakan bagian dari kelas lain tapi bisa berdidri sendiri |
|  | ***Composition :***bentuk khusu dari agregasi jadi kelas yang menjadi part baru dapat di ciptakan setalah kelas lain yang menjadi *whole* dibuat. jika kelas yang menjadi *whole* dimusnakan kelas part akan hilang. |
|  | ***Usage*** : relasi antara dua elemen untuk implementasi atau operasi. |
|  | ***Association*** : relasi antar kelas. |
|  | ***N-ary association*** : model relasi beberapa kelas. |
|  | ***Association class*** : relasi antar kelas yang mempunyai atribut. |
|  | ***Dependency*** : relasi antar dua kelas dimana jika ada perubahan dalam satu kelas, maka akan mempengaruhi kelas lain. |
|  | ***Abstraction*** : konsep sama pada *level* berbeda. |

|  |  |
| --- | --- |
| **Gambar** | **Penjelasan** |
|  | ***Collaboration*** : grup elemen yang saling memiliki fungsi sama. |
|  | ***Model*** : kontainer untuk model elemen yang disimpan. |
|  | ***Note*** : area teks untuk menandai. |
|  | ***Anchor*** : konektor yang mengkoneksikan tambahan *note.* |
|  | ***Constraint*** : kondisi dibawah perilaku yang dieksekusi. |
|  | ***Containment*** : relasi yang menandakan ‘*part of relationship’.* |

# Simbol *State Diagram*

|  |  |
| --- | --- |
| **Gambar** | **Penjelasan** |
|  | ***State*** : merupakan indikasikan aksi yang terjadi dikarenakan adanya *event* |
|  | ***Initial pseudo State***: merupakan suatu awal *State* |
|  | ***Final State***: merupakan suatu akhir dari *State* |
|  | ***Transition***: merupakan relasi antara satu *State* ke *State* lain. |

# Simbol *Sequence Diagram*

|  |  |
| --- | --- |
| **Gambar** | **Penjelasan** |
|  | ***Lifeline*** : merupkana suatu reprentasi *role* atau objek yang dalam model interaksi. |
|  | ***Message*** : merupakan definisi dari komunikasi antar *lifeline*. |
|  | ***Duration message*** : dapat menampilkan jarak antara dua *instance* waktu dalam pesan. |
|  | ***Create message*** : merupakan pesan yang merepresentasikan instansiasi *lifeline*. |
|  | ***Self message*** : merupakan pesan yang menampilkan operasi dalam objek *lifeline* yang sama. |
|  | ***Recursive message*** : merupakan tipe pesan yang dieksekusi secara rekursif. |
|  | ***Found message*** : merupakan pesan yang menerima *event* yang diketahui tapi tidak tahu pengirimannya. |
|  | ***Lost message*** : merupakan pesan yang diketahui *event*- nya tapi tidak tahu pengirimnya. |
|  | ***Reentrant message*** : merupakan *point* untuk aktivasi di atasnya. |

|  |  |
| --- | --- |
| **Gambar** | **Penjelasan** |
|  | ***Interaction use*** : menyalin konten dari referensi interaksi. |
|  | ***Frame*** : mendeskripsikan unit lingkungan yang berfokus dalam perubahan informasi. |
|  | ***Actor*** : merepresentasikan peran manusia, atau sistem lain atau alat untuk komunikasi dengan bagian *use case* dalam sistem. |
|  | ***Continuation*** : cara sintetik untuk mendefinisikan *branch* yang berbeda |
|  | ***Gate*** : cara sintatik untuk. |
|  | ***Duration constraint*** : *constraint* mengacu pada *interval* durasi. |
|  | ***Note*** : area teks untuk menandai. |
|  | ***Anchor*** : konektor yang mengkoneksikan tambahan *note*. |
|  | ***Constraint*** : kondisi dibawah perilaku yang dieksekusi. |

|  |  |
| --- | --- |
| **Gambar** | **Keterangan** |
|  | ***Relationship line*** : Garis Penghubung yang digunakan pada diagram |
|  | ***Entitas*** : Digunakan untuk menggambarkan obyek yang dapat di identifikasikan dalam lingkungan pemakai |
|  | ***Atribut*** : Menggambarkan elemen-elemen dari suatu entitas yang menggambarkan entitas. |
|  | **Satu ke satu** : Satu entitas di dalam himpunan entitas A dihubungkan dengan paling banyak satu entitas di dalam himpunan entitas B, dan entitas di dalam himpunan entitas B dihubungkan dengan paling banyak satu entitas dalam himpunan entitas A. |
|  | **Satu ke banyak** : Satu entitas di dalam himpunan entitas A dihubungkan dengan lebih dari satu entitas di dalam himpunan entitas B, dan entitas di dalam himpunan entitas B hanya dapat dihubungkan dezxngan paling banyak satu entitas dalam himpunan entitas A. |
|  | **Banyak ke satu** : Satu entitas didalam himpunan entitas A dihubungkan dengan paling banyak satu entitas di dalam himpunan entitas B di dalam himpunan entitas B dapat dihubungkan dengan lebih dari satu entitas dalam himpunan entitas A. |

# Simbol *Entity Relationship Diagram*

|  |  |
| --- | --- |
| **Gambar** | **Keterangan** |
|  | **Banyak ke banyak** : Satu entitas di dalam himpunan entitas A dapat dihubungkan dengan lebih dari satu entitas di dalam himpunan entitas B, dan entitas di dalam himpunan entitas B dapat dihubungkan dengan lebih dari satu entitas dalam himpunan entitas A. |

# Simbol *Deployment Diagram*

|  |  |
| --- | --- |
| **Gambar** | **Penjelasan** |
|  | ***Node*** : merepresentasikan sumber daya komputasi tempat artefak dapat digunakan untuk eksekusi. |
|  | ***Artifact*** : Spesifikasi dari informasi fisik yang digunakan atau diproduksi oleh proses pengembangan perangkat lunak. |
|  | ***Deployment Specification*** : Sekumpulan property yang menentukan parameter |
|  | ***Component*** : Mewakili bagian modular dari sistem yang merangkum kontenya dan yang manifestasinya dapat diterima Kembali di dalam lingkunganya. |
|  | ***Interfaces*** : mewakili deklarasi sekumpulan fitur dan kewajiban public yang koheren. |

|  |  |
| --- | --- |
| **Gambar** | **Penjelasan** |
|  | ***Port*** : Menentukan titik interaksi yang berbeda antara pengklasifikasi itu dan lingkunganya. |
|  | ***Found message*** : Diperpanjang dengan kemampuan menjadi target penerapan dalam hubungan penerapan. |
|  | ***Association*** : Mebentukan hubungan semantik yang dapat terjadi di antara *instance* type. |
|  | ***Port*** : Menentukan titik interaksi yang berbeda antara pengklasifikasi itu dan lingkunganya. |
|  | ***Link*** : Sebuah asosiasi menyatakan bahwa mungin ada tautan antara *instance* dari tipe terkait. |
|  | ***Dependency*** : relasi antar dua node dimana jika ada perubahan dalam satu node, maka akan mempengaruhi node lain. |
|  | ***Interaction use*** : ada diantara dua node dimana satu node terspesifikasi. |
|  | ***Note*** : area teks untuk menandai.. |

# BAB I

# PENDAHULUAN

# Latar Belakang

Kebutuhan akan suatu aplikasi merupakan hal yang sangat penting dan sangat berpengaruh dalam kehidupan. Peranan aplikasi sangat penting bagi kemajuan di segala bidang, terutama bidang perdagangan, teknologi diperuntukan bagi kemudahan manusia dalam mengerjakan segala kegiatan. Aplikasi juga merupakan salah satu aspek yang penting bagi toko, serta bagi semua aspek bidang, karena dapat membantu toko untuk meningkatakan produktifitas, baik dalam memperoleh, mengolah serta menggunakan aplikasi secara akurat.

Pada toko, pengelolaan data barang secara manual menimbulkan ketidak efektifan dalam pengecekan kembali barang, begitu pun transaksi manual yang seringkali menimbulkan ketidakpastian dalam jual beli, kepuasan konsumen selalu menjadi nomor satu bagi toko, terutama toko yang bergerak di bidang perdagangan. Perkembangan dan pemanfaatan teknologi, aplikasi saat ini sangat diperlukan terutama dalam pemanfaatan teknologi komputer, untuk pengolahan data juga menjadi perhatian di toko besar. Berbagai kegiatan toko bisa dikembangkan menjadi sistem yang kinerjanya menggunakan perangkat komputer atau kinerja di toko bisa ditingkatkan, seperti pengolahan data penjualan barang, pengelolaan *stock* barang di gudang.

Mengenai hal tersebut tentunya pemanfaatan teknologi komputer sangat dibutuhkan di berbagai bidang, terutama bidang perdagangan dimana setiap barang masuk dan barang keluar, bisa terpantau secara *realtime*, selama ini bidang usaha, toko Bangunan Satria Jaya masih menggunakan tulis tangan, baik dalam pencatatan transaksi kepada konsumen ataupun pendataan barang bahan bangunan di gudang.

Untuk membantu dalam pengelolaan barang dan transaksi terhadap konsumen, maka penulis menemukan sebuah celah untuk membuat APLIKASI PENJUALAN BARANG (*POINT OF SALE*) BAHAN BANGUNAN BERBASIS

WEB DI TOKO BAHAN BANGUNAN SATRIA JAYA, dengan adanya aplikasi penjualan barang (*point of sale*), pengelolaan akan terkoordinasi dengan baik, dan mengefensiensikan waktu, serta kemudahan dalam pengecekan barang dan transaksi terhadap konsumen,. Selain itu aplikasi ini dibuat sebagai sarana penelitian dan pengabdian penulis sebagai mahasiswa.

# Rumusan Masalah

Berdasarkan latar belakang yang penulis sampaikan di atas, maka dapat disimpulkan yaitu:

1. Bagaimana membuat sistem yang dapat mengelola barang masuk dan barang keluar ?
2. Bagaimana membuat sistem perhitungan penjualan barang atau transaksi penjualan?
3. Bagaimana membuat laporan transaksi penjualan dari tulis tangan, ke *digital* atau komputerisasi?

Dengan demikian dapat disimpulkan bahwa **TOKO BAHAN BANGUNAN SATRIA JAYA,** membutuhkan aplikasi yang dapat membantu dalam pengelolaan barang bahan bangunan serta transaksi yang terjadi,.

# Maksud Dan Tujuan

1. **Maksud**

Pembuatan aplikasi penjualan barang bahan bangunan (*point of sale*), ini dimaksudkan untuk membantu mengelola data barang, serta transaksi di Toko Bangunan Satria Jaya yang saat ini masih menggunakan metode tulis tangan.

Aplikasi penjualan barang bahan bangunan (*point of sale*), ini diperlukan untuk mengelola data barang masuk, barang keluar, atau *stock* barang, serta transaksi dengan harapan mempermudah operasional kerja.

1. **Tujuan**

Tujuan dari pembuatan aplikasi penjualan (*point of sale*) barang bahan bangunan ini, adalah sebagai berikut :

1. Membuat sistem yang dapat mengelola barang masuk dan barang keluar, agar stock barang yang tersedia akan lebih mudah dipantau, dan dikendalikan.
2. Membuat sistem perhitungan penjualan barang atau transaksi penjualan, agar mempermudah proses transaksi dengan konsumen
3. Membuat laporan transaksi penjualan dari tulis tangan, ke *digital* atau komputerisasi

# Batasan Masalah

Agar pembuatan aplikasi penjualan barang bahan bangunan (*point of sale*) ini terarah dengan baik, maka dalam penulisan ini hanya akan membahas :

1. Pengelolaan data barang masuk
2. Pengelolaan data barang keluar
3. Transaksi penjualan
4. Laporan transaksi penjualan

# Batasan Masalah

* 1. Metode pengumpulan data yang penulis gunakan sebagai berikut:

1. Wawancara, yaitu suatu kegiatan tanya jawab yang dilakukan oleh seseorang atau lebih, antara pembuat sistem dengan pihak toko. Ini dilakukan agar mendapatkan data yang akurat dan benar serta menghindari kesalahan informasi.
2. Observasi, pengumpulan data dengan mengamati langsung, melihat dan mengambil data yang dibutuhkan oleh pembuat aplikasi.
3. Tinjauan Pustaka, yaitu mencari suatu data yang bersumber dari para ahli, dengan mempelajari buku-buku serta *up to date* terhadap aplikasi.
   1. Metode Rekayasa Perangkat Lunak

Metode pada aplikasi penjualan barang bahan bangunan (*point of sale*) ini adalah metode *prototype*, karena merupakan model sederhana cocok digunakan untuk pengembangan perangkat lunak, dalam skala project kecil ke sedang, dengan melakukan pendekatan dengan *client*, mulai dari komunikasi hingga *prototype* dari aplikasi yang akan dibuat, setiap perubahan di deskripsikan

ke *client*, kemudian *client* akan merevisi jika rancangan aplikasi tidak sesuai, jika sesuai maka akan dilakukan ke tahap berikutnya.

Metode *Prototype* adalah metode pengembangan perangkat lunak yang memungkinkan adanya interaksi antara pengembang sistem dengan pengguna sistem, sehingga dapat mengatasi ketidakserasian antara pengembang dan pengguna (Pressman, 2012: 50). Adapun model pengembangan *Prototype* digambarkan pada gambar 1.1.

|  |
| --- |
|  |
| Gambar 1. 1 Prototype Pressman (Pressman, 2015:42) |

Umumnya pelanggan mendefinisikan sejumlah sasaran perangkat lunak secara umum, tetapi tidak bisa mengidentifikasi spesifikasi kebutuhan yang rinci, untuk fungsi-fungsi dan fitur-fitur yang nantinya akan dimiliki perangkat lunak yang akan dikembangkan. Dalam suatu kasus dan dalam situasi yang lain, paradigma pembuatan *prototype* (*prototyping*) mungkin menawarkan pendekatan yang paling baik (Pressman, 2012:50), dimulai dengan dilakukannya komunikasi antara tim pengembang perangkat lunak dengan pelanggan.

Tim perangkat lunak akan melakukan petemuan pertemuan dengan *stakeholder* untuk mendefinisikan sasaran keseluruhan untuk perangkat lunak yang akan dikembangkan, mengidentifikasi spesifikasi kebutuhan apa pun yang saat ini diketahui, dan menggambarkan area-area dimana didefinisi lebih jauh pada iterasi selanjutnya merupakan keharusan.

Berikut tahap-tahap pada metode pengembangan sistem yang digunakan pada penelitian ini:

1. Komunikasi, merupakan tahapan awal dari model *prototype* guna mengidentifikasi permasalahan-permasalahan yang ada, serta informasi-informasi lain untuk di analisis dan diperlukan untuk membangun sistem.

2. Perencanaan merupakan tahapan yang dikerjakan dengan kegiatan penentuan sumber daya ataupun tenaga ahli yang dibutuhkan dalam membangun sistem, spesifikasi untuk pengembangan berdasarkan kebutuhan sistem, dan tujuan berdasarkan pada hasil komunikasi yang dilakukan agar pengembangan dapat sesuai dengan yang diharapkan.

3. Pemodelan merupakan, tahapan selanjutnya dimana representasi atau menggambarkan model sistem yang akan dikembangkan seperti proses dengan perancangan menggunakan *Unified Modeling Language (UML)* . Dalam tahap ini, *Prototype* yang dibangun dengan sistem rancangan sementara kemudian di *evaluasi* terhadap *customer* apakah sudah sesuai dengan yang diinginkan, atau masih perlu untuk di *evaluasi* kembali. Setelah sistem dianggap sesuai dengan apa yang diharapkan *customer*, langkah berikutnya yaitu pembuatan aplikasi (pengkodingan) dari rancangan sistem yang dibuat diterjemahkan ke dalam bahasa pemrograman hingga menjadi sebuah sistem atau aplikasi.

4. Konstruksi merupakan tahapan yang digunakan untuk membangun *prototype* dan menguji-coba sistem yang dibangun. Proses instalasi dan penyediaan *user support* juga dilakukan agar sistem dapat berjalan dengan sesuai.

5. Penyerahan merupakan tahapan yang dibutuhkan untuk mendapatkan *feedback* dari pengguna, sebagai hasil *evaluasi* dari tahapan sebelumnya dan implementasi dari sistem yang dikembangkan.

# Tempat dan Waktu

Tempat Kerja Praktek dilaksanakan di :

Nama Instansi : TOKO BAHAN BANGUNAN SATRIA JAYA

Alamat : Jl.Aria Wiratanudatar KM.5 Sindanglaka, WA:0878 2030 2023,

Kabupaten Cianjur, Jawa Barat.

Waktu : Februari 2021 – Mei 2021

Tabel 1. 1 Jadwal Kegiatan

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Uraian Langkah Kegiatan | Bulan | | | |
| Februari | Maret | April | Mei |
|
| 1 | Rumusan Masalah |  |  |  |  |
|
| 2 | Pengumpulan Data |  |  |  |  |
|
| 3 | Analisis |  |  |  |  |
|
| 4 | Desain |  |  |  |  |
|
| 5 | Implementasi |  |  |  |  |
|
| 6 | Pengujian |  |  |  |  |
|
| 7 | Penyusunan Laporan |  |  |  |  |
|
| 8 | Seminar |  |  |  |  |
|

# Sistematika Penulisan

Sistematika penulisan laporan kerja praktek ini dibagi menjadi 5 (lima) bab. Berikut penjelasan mengenai masing-masing bab :

**BAB I PENDAHULUAN**

Bab ini menjelaskan mengenai latar belakang masalah, rumusan masalah, maksud dan tujuan penelitian, batasan masalah yang berguna untuk memfokuskan pada pengelolaan data barang masuk, barang keluar dan transaksi sehingga dapat mencapai hasil yang optimal.

**BAB II TINJAUAN PUSTAKA**

Bab ini menjelaskan tentang teori-teori yang mendukung laporan kerja praktek. Pembahasan landasan teori ini meliputi penjelasan tentang definisi-definisi, penjelasan mengenai perangkat lunak yang digunakan dalam perancangan sistem informasi dan istilah istilah yang terkait didalamnya.

**BAB III ANALISIS DAN PERANCANGAN**

Bab ini menjelaskan uraian tentang analisa sistem yang akan dibuat dan bagaimana merancangnya sehingga menjadi sebuah sistem. Selain itu mencari informasi-informasi yang berkaitan dengan sistem yang akan dibangun.

**BAB IV IMPLEMENTASI & PENGUJIAN**

Bab ini menjelaskan mengenai pembuatan sistem, yang merupakan pengimplementasian dari hasil analisa dan perancangan yang berisi gambaran struktur menu serta tampilan layar dan langkah pengujian dari sistem yang akan dibangun.

**BAB V PENUTUP**

Bab ini berisi kesimpulan tentang seluruh pembahasan dan pemecahan masalah yang telah dilakukan serta mengenai hal-hal yang terdapat dalam bab-bab sebelumnya dan saran-saran untuk mengembangkan aplikasi berikutnya.

# BAB II

# TINJAUAN PUSTAKA

# Tinjauan Pustaka

Tinjauan pustaka ini membahas tentang gambaran toko, tempat survey yang didalamnya mencakup sejarah singkat toko, struktur organisasi toko dan deskripsi kerja.

## Sejarah Singkat Toko Satria Jaya

Pada tahun 1960 berdiri toko bangunan, yang bahan bernama toko Bahan Bangunan Bapak Fakih, Bapak Fakih adalah kakek dari Bapak Haji Deden (Pemilik Toko Satria Jaya), Bapak Fakih mempunyai anak dua orang, pada tahun 1972 Bapak Fakih mewariskan toko bangunan tersebut kepada anak nya yaitu Bapak Haji Cece dan toko bangunan itu bernama Toko Bangunan Susan Jaya yang beralamat di Jalan Aria Wiratanudatar Cimenteng Cianjur.

Alhamdulillah toko bahan bangunan tersebut sekarang sudah mempunyai 4 cabang yang di kelola oleh anak-anak nya,. Toko Bahan Bangunan Satria Jaya adalah cabang ke 3 (tiga) dari Toko Susan Jaya. Toko Bahan Bangunan Satria Jaya Cabang ke-3 (tiga) berdiri pada tahun 2007 yang beralamat di Jalan Aria Wiratanudatar Sindanglaka Cianjur, yang menjual bahan bangunan seperti kayu, bata, besi, semen, cat, paku, pasir, kramik, dan lain-lain.

Alhamdulillah Toko Satria Jaya sudah memiliki banyak pelanggan, mulai dari pelanggan biasa, sampai pelanggan menengah ke atas, harga jual bahan bangunan di Toko Satria Jaya bersaing dengan toko lain. Pemelik Toko Bahan Bangunan Satria Jaya berharap, semoga Toko Satria Jaya bisa berkembang dan tentunya bisa diwariskan ke generasi selanjutnya

|  |
| --- |
| **Pemilik**  Bapak H.Deden S.E  **Kepala Bagian**  Ibu Hj. Devi ST  **Bagian Umum**  Bapak Acep  **Kasir**  Ibu Evi  **Bagian Kepegawaian**  Bapak Asep |
| Gambar 2. 1 Struktur Organisasi Toko Bahan Bangunan Satri Jaya |

## Struktur Organisasi Satria Jaya

## *Job* Deskripsi

1. Pemilik
2. Mengotrol semua aktifitas penjualan
3. Mengotrol semua aktifitas *stock* barang
4. Melakukan pengecekan laporan penjualan
5. Melakukan pengecekan laporan *stock* barang
6. Kepala Bagian
7. Memastikan pelaksanaan sebuah program
8. Menangani kinerja karyawan
9. Menjadi pusat koordinasi
10. Bagian Kepegawaian
11. Menjalankan prosedur penjadwalan pegawai
12. Mengelola penggajian pegawai
13. Menjalankan prosedur budaya kerja
14. Kasir
15. Menjalankan proses penjualan
16. Melakukan proses pengelolaan *stock* barang
17. Melakukan pencatatan atas semua transaksi
18. Membantu pelanggan dalam memberikan informasi mengenai suatu produk
19. Bagian Umum
20. Mengelola sarana dan prasaran toko
21. Menerima keluhan dari pelanggan atau *customer*
22. Bertanggung jawab terhadap alat yang digunakan di toko

# Landasan Teori

## Definisi Aplikasi

Arti dari Aplikasi Menurut Supriyanto (2005:2) adalah program yang memiliki aktifitas pemrosesan perintah yang diperlukan untuk melaksanakan permintaan pengguna dengan tujuan tertentu.

Dari definisi di atas penulis dapat menyimpulkan bahwa aplikasi merupakan penerapan dari rancangan sistem untuk mengolah data atau mengolah dokumen yang dilakukan oleh pengguna sistem, yang di dalamnya terdapat intruksi atau pernyataan sehingga sehingga komputer bisa memproses input menjadi output yang di inginkan.

## Definisi Penjualan

Menurut Basu Swastha (2001:8) penjualan merupakan ilmu dan seni mempengaruhi pribadi yang dilakukan oleh penjual untuk mengajak orang lain agar bersedia membeli barang yang ditawarkan. Adapun faktor-faktor yang mempengaruhi penjulalan adalah kondisi dan kemampuan penjual, kondisi pasar, modal, dan kondisi organisasi toko.

Dari definisi diatas penulis dapat menyimpulkan bahwa penjualan merupakan suatu kegiatan menjual *product* barang atau jasa dengan tujuan utama yaitu mendatangkan keuntungan.

## Definisi Barang

Definisi Barang menurut Fandy Tjiptono merupakan “produk yang berwujud fisik sehingga dapat bisa dilihat, disentuh, dirasa, dipegang, disimpan,dan perlakuan fisik lainnya” (1999:98).

Dari definisi diatas penulis dapat menyimpulkan barang merupakan *object* yang dapat dilihat dengan kasat mata dan memiliki suatu nilai ataupun bobot.

## Definisi Bahan

Bahan merupakan persediaan yang dibeli oleh perusahaan untuk diproses menjadi barang setengah jadi dan akhirnya barang jadi atau produk akhir dari perusahaan (Syamsuddin, 2001:281)

Dari definisi diatas penulis dapat menyimpulkan bahan merupakan material atau barang yang akan dibuat oleh seseorang atau sekelompok orang, melalui proses-proses tertentu hingga menjadi suatu karya yang dapat digunakan.

## Definisi Bangunan

Bangunan merupakan wujud fisik hasil pekerjaan konstruksi yang menyatu dengan tempat kedudukan baik yang diatas ataupun yang di bawah tanah dan menyatu dengan tempat kedudukan di air (Ariestadi, 2008:1)

Dari uraian definisi diatas, penulis dapat menyimpulkan Bangunan merupakan wujud fisik hasil pekerjaan konstruksi yang menyatu dengan tempat.

## Definisi POS *(Point Of Sale)*

Menurut Sugihartono, Satoto dan Widianto (2015:445), *Point of Sale* (POS) merupakan aplikasi yang digunakan dalam membantu pengolahan data-data seperti data pembelian, penjualan eceran, transaksi hutang, transaksi *retur* pembelian, dan pelaporan transaksi yang dibutuhkan dalam pengambilan keputusan oleh para pebisnis.

POS terdiri dari perangkat keras *(PC, receipt printer, cash drawer, terminal pembayaran, barcode scanner*) dan perangkat lunak (manajemen *stock*, pelaporan, pembelian, manajemen pelanggan, standar keamanan transaksi, dan proses retur) dimana kedua komponen tersebut digunakan dalam setiap proses transaksi yang terjadi (Yuarita dan Marisa, 2017:168).

Perangkat lunak POS adalah komponen utama dari POS yang akan menentukan proses berjalannya sistem tersebut, seperti apa yang harus dilakukan dan bagaimana cara melakukannya. Akan tetapi, perangkat keras POS dibutuhkan dalam membantu menjalankan fungsi POS, membantu proses transaksi pembayaran dan membuat tanda terima untuk pelanggan.

Perangkat keras dapat disesuaikan dengan kebutuhan organisasi agar dapat membantu mempercepat proses pemasukkan barang hingga proses pelayanan penjualan (Novita dan Djatikusuma, 2010). Manfaat penggunaan *Point of Sale* disebuah organisasi adalah untuk mengolah transaksi-transaksi, mengurangi biaya serta menghasilkan pendapatan yang menguntungkan bagi organisasi tersebut sebagai salah satu produk maupun pelayanan yang diberikan organisasi kepada pelanggannya. Penggunaan *Point of Sale* membantu mempertahankan persediaan stok barang di tingkat paling rendah agar konsisten dengan jenis barang yang tersedia sehingga organisasi dapat menghindari terjadinya stok habis dan pelanggan yang kecewa akan produk habis (Sugihartono, Satoto dan Widianto, 2015:445).

Dari definisi diatas penulis dapat menyimpulkan bahwa POS (*Poin Of Sale)* merupakan sistem atau aplikasi yang penerapannya di dampingi oleh *hardware* yang diperlukan guna untuk membantu dalam pengolahan data yang digunakan pada proses transaksi.

## Definisi Toko

Setiap lembaga bisnis dan penjualnya terutama berasal dari usaha retailing disebut retailer atau retail store bila lembaga tersebut berupa sebuah toko. Pengecer atau toko eceran adalah usaha bisnis yang volume penjualannya terutama berasal dari penjualan eceran (Kotler; 2008; 140)

Dari definisi diatas penulis dapat menyimpulkan bahwa, toko merupakan suatu tempat dimana terjadinya suatu kegiatan jual beli atau transaksi berupa barang antara penjual dan pembeli, dimana mempunyai tujuan suatu tujuan tertentu untuk mecapai kesepakatan bersama.

## Definisi PHP (*Hypertext Preprocessor*)

PHP adalah bahasa pemrograman *interpreter* yaitu proses penerjemahan baris kode sumber menjadi kode mesin yang dimengerti komputer secara langsung pada saat baris kode dijalankan. (Sibero, 2011) “PHP” (*Hypertext Preprocessor*)

PHP adalah sebuah bahasa pemrograman yang berbentuk *scripting*, sistem kerja dari program ini adalah sebagai *interpreter* bukan sebagai *compiler*”. (Nugroho, 2013)

Dari definisi diatas penulis dapat menyimpulkan bahwa, bahasa pemrograman php merupakan bahasa pemrograman yang digunakan untuk mengembangkan situs *web* statis atau situs *web* dinamis atau Aplikasi *Web*. PHP singkatan dari *Hypertext Pre-processor*, yang sebelumnya disebut Personal Home Pages.

## Definisi CSS (*Cascading Style Sheet*)

(Bekti, 2015) Berdasarkan pendapat yang telah dikemukakan dapat disimpulkan bahwa CSS (*Cascading Style Sheets*) adalah salah satu jenis bahasa pemrograman untuk mengatur proses pengolahan pada komponen tampilan *web* menjadi bentuk *web* yang lebih indah dan menarik.

CSS merupakan singkatan dari *Cascading Style Sheet*. Kegunaannya adalah untuk mengatur tampilan dokumen HTML, contohnya seperti pengaturan jarak antar baris, teks, warna dan format border bahkan penampilan file gambar. (Jayan, 2010)

Dari definisi diatas penulis dapat menyimpulkan CSS, merupakan *script code* yang berfungsi untuk mengatur *layout* web, tulisan, warna, seluruh *style* tampilan halaman web.

## Definisi HTML (*Hyper Text Markup Language*)

*Hyper Text Markup Language* atau HTML adalah bahasa yang digunakan pada dokumen *web* sebagai bahasa untuk pertukaran dokumen web. (Sibero, 2013)

HTML (*Hyper Text Markup Language*) adalah simbol-simbol atau tag-tag yang dituliskan dalam sebuah file yang dimaksudkan untuk menampilkan halaman pada *web browser*. Tag-tag HTML selalu diawali dengan dan diakhiri dengan dimana x tag HTML seperti b, i, u dan sebagainya. Namun ada juga tag yang tidak diakhiri dengan tanda, Astamal (2006:8)

Dari definisi diatas penulis dapat menyimpulkan CSS, merupakan *script code* yang berfungsi untuk mengatur *layout* *web*, tulisan, warna, seluruh *style* tampilan halaman *web*.

## Definisi Basis Data (*Database*)

*Database* adalah koleksi terpadu dari data-data yang saling berkaitan dari suatu *enterprise* (toko, instansi pemerintahan atau swasta). (Ladjamudin, 2013), *Database* (Basis Data) merupakan sekumpulan data yang sangat kompleks, kemudian data tersebut memiliki hubungan antara data yang satu dengan data yang lainya. (Nugroho, 2013)

Dari definisi diatas penulis dapat menyimpulkan *Database*, merupakan sekumpulan data yang tersimpan secara *digital* yang mempunyai hubungan antara satu dengan data yang lain

## Definisi DBMS (*Database Management System*)

DBMS adalah “*Database Management System* (DBMS) merupakan kumpulan program aplikasi yang digunakan untuk mengelola basis data. Taufik, 2013). DBMS adalah kumpulan program yang digunakan untuk mendefinisikan, mengatur, dan memproses *database*, sedangkan *database* itu sendiri esensinya adalah sebuah struktur yang dibangun untuk keperluan penyimpanan data. (Budi Raharjo, 2011).

Dari definisi diatas penulis dapat menyimpulkan DBMS , merupakan suatu tempat atau sekumpulan berkas data yang sudah terkomputerisasi yang bertujuan untuk memelihara data atau informasi.

## Definisi UML (*Unified Modeling Language*)

Menurut Hend (2006:5) “*Unified Modeling Language* (UML) adalah bahasa yang telah telah menjadi standard untuk *visualisasi*, menetapkan, membangun dan mendokumentasikan artifak suatu sistem perangkat lunak”. UML menyediakan sembilan jenis diagram, yang lain menyebutkan delapan karena ada beberapa diagram yang digabung, misanya diagram komunikasi, diagram urutan dan diagram pewaktuan digabung menjadi diagram interaksi. (*He UML* adalah metodologi kolaborasi antara metoda-metoda *Booch, OMT (Object Modeling Technique),* serta *OOSE* (*Object Oriented Software Enggineering*) dan beberapa metoda lainnya, merupakan metodologi yang paling sering digunakan saat ini untuk analisa dan perancangan sistem dengan metodologi berorientasi objek mengadaptasi maraknya penggunaan bahasa ‘pemrograman berorientasi objek’ (OOP). (Nugroho, 2009) rlawati, 2011).

Diagram berbentuk grafik yang menunjukan simbol elemen model yang

disusun untuk mengilustrasikan bagian atau aspek tertentu dari sistem. Sebuah diagram merupakan bagian dari suatu *view* tertentu dan ketika digambarkan biasanya dialokasikan untuk *view* tertentu. Adapun jenis-jenis diagram antara lain:

* 1. *Use Case Diagram*

*Use Case* Diagram merepresentasikan sejumlah external *actors* dan hubungannya ke use case yang diberikan oleh sistem. *Use case* adalah deskripsi fungsi yang disediakan oleh sistem dalam bentuk teks sebagai dokumentasi dari *use case symbol* namun dapat juga dilakukan dalam *activity* diagram.

* 1. *Activity Diagram*

Diagram ini menggambarkan rangkaian aliran dari aktivitas didalam sebuah sistem yang bersifat dinamis, diagram ini digunakan untuk mendeskripsikan aktifitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya seperti *use case* atau interaksi.

* 1. *Class Diagram*

Class Diagram adalah pandangan aplikasi yang bersifat statis. Class Diagram tidak hanya menggambarkan visualisasi, tetapi juga menggambarkan dan mendokumentasikan aspek yang berbeda dalam sistem, tetapi juga untuk kontruksi eksekusi kode dalam software atau aplikasi.

Class Diagram digunakan untuk mengelompokan hal-hal inti dari setiap proses yang ingin dilakukan. Semua proses dimasukkan ke dalam tiap-tiap class dan saling dihubungkan pada class-class lainnya yang saling berhubungan.

* 1. *State Diagram*

Menggambarkan semua *state* (kondisi) yang dimiliki oleh suatu *object* dari suatu *class* dan keadaan yang menyebabkan *state* berubah. Kejadian dapat berupa *object* lain yang mengirim pesan.

* 1. *Sequence Diagram*

Diagram ini menggambarkan kolaborasi dinamis antara sejumlah *object* yang bersifat dinamis. Kegunaanya untuk menunjukkan rangkaian pesan yang dikirim terhadap *object* juga interaksi antara *object*, sesuatu yang terjadi pada titik

tertentu dalam eksekusi sistem.

* 1. *Deployment Diagram*

Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan (*run-time*) yang bersifat *statis*. Memuat simpul-simpul berserta komponen-komponen yang ada di dalamnya. Diagram *deployment* berhubungan erat dengan diagram komponen dimana diagram ini memuat satu atau lebih komponen-komponen. Diagram ini sangat berguna saat aplikasi berlaku sebagai aplikasi yang dijalankan pada banyak mesin (*distributed computing*).

Dari definisi diatas penulis dapat menyimpulkan UML (*Unified Modeling Language),* Definisi UML merupakan model atau bahasa yang sudah menjadi standar pada *visualisasi*, perancangan, dan juga pendokumentasian sistem aplikasi yang akan dibangun.

## Definisi Motode Pengujian (*Blackbox*)

“*Blackbox* *testing* merupakan tipe *testing* yang memperlakukan perangkat lunak yang tidak diketahui kinerja *internal*nya”. Sedangkan menurut Mustaqbal, dkk (2015:34) “*BlackBox Testing* befokus pada spesifikasi fungsional dari perangkat lunak, kumpulan kondisi input dan melakukan pengetesan pada fungsional program”. Menurut Rizky (2011:264) “*Blackbox testing* merupakan menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program”. Menurut Rosa dan Salahuddin (2015:275).

Menurut Simarmata Janner (2010 : 316) Klasifikasi *blackbox testing* mencakup beberapa pengujian yaitu :

1. Pengujian fungsional (*functional testing*)
2. Pengujian tegangan (*stress testing*)
3. Pengujian beban (*load testing*)
4. Pengujian khusus (*ad-hoc testing*)

Dari definisi diatas penulis dapat menyimpulkan metode pengujian *Blackbox*, merupakan metode untuk menguji kelancaran program yang telah dibuat sebelumnya. Pengujian dilakukan agar program tidak terjadi kesalahan ketika dijalankan.

# *Tools* yang digunakan

Untuk memudahkan dalam pembuatan sistem, maka diperlukan beberapa alat atau *tools* yang digunakan dalam pembuatan aplikasi penjualan barang bahan bangunan, serta untuk proses pelaksanaan pembuatan sistem yang akan dibangun.

## *Bootstrap*

Menurut para ahli *Bootstrap* merupakan kerangka kerja *front-end* atau *design* untuk pengembangan *web* yang lebih cepat dan mudah tentunya gratis. *Bootstrap* mencakup *template* desain berbasis HTML dan CSS untuk *tipografi*, bentuk, tombol, tabel, *navigasi, modals*, korsel gambar dan banyak lainnya, serta *plugin JavaScript* opsional. *Bootstrap* juga memberikan sobat kemampuan untuk membuat desain *responsif* dengan mudah.

*Responsive* *Layout* maka aplikasi *web* yang didesain dengan menggunakan *Bootstrap* akan langsung menyesuaikan dengan lebar dari media perambahannya. Sehingga *Framework* ini support untuk semua jenis device baik dari *smartphone, tablet*, laptop ataupun *PC* *Desktop*. Di samping itu, *Boostrap* juga sudah support untuk HTML 5 dan C.

## Balsamiq *Mockup*

Balsamiq Mockup merupakan suatu program aplikasi yang digunakan, dalam pembuatan tampilan *user interface* sebuah aplikasi. *Software* ini sudah menyediakan *tools* yang dapat memudahkan dalam membuat desain *prototyping* aplikasi yang akan dibuat. Berfokus pada konten yang ingin digambar dan fungsionalitas yang dibutuhkan oleh pengguna.

Menggambar sketsa (*wireframe*) atau *prototype* rancangan desain *website* di atas kertas balsamiq *mockups* membantu seorang *web designer* dapat membuat tampilan *web* dalam bentuk gambar di komputer. Tujuannya agar membuat tampilan (*design*) *website* menarik juga dapat menyesuaikan dengan kebutuhan *customer* (pelanggan). Dengan alat pembuat *mockup* maka seorang *web designer* dapat menganalisa tata letak, desain dan fungsi.

## *CodeIgniter*

CodeIgniter adalah salahsatu *framework* PHP yang dibuat berdasarkan *model view Controlleer* (MVC). CI memiliki *library* lengkap untuk mengerjakan operasi-operasi yang dibutuhkan oleh aplikasi berbasis web misalnya mengakses *database*, *memvalidasi form* sistem yang akan dikembangkan mudah.

Codelgniter (*CI*) merupakan suatu *Framework* pengembangan aplikasi (*Application Development Framework*) dengan menggunakan bahasa PHP, pengembangan dapat langsung menghasilkan program dengan cepat, dengan mengikuti kerangka kerja membuat yang telah disiapkan oleh *Framework* CI. Dengan menggunkan *Framework* tidak perlu membuat program dari awal, tetapi sudah diberikan *library* fungsi-fungsi yang sudah diorganisasi untuk dapat membuat suatu program dengan cepat. Hanya perlu memasukkan data yang akan diproses dan bagaimana menampilkannya.

## *Google Chrome*

*Google Chrome* merupakan suatu aplikasi peramban yang digunakan untuk menjelajah dunia maya seperti halnya Firefox, Opera ataupun *Microsoft Edge*. Jika *Firefox* dikembangkan oleh *Mozilla, Google Chrome* dibuat dan dirancang oleh Google, toko internet terbesar di dunia yang juga mempunyai Android.

Proyek yang dimiliki yakni *open source* digunakan oleh Google disebut Chromium, menggunakan mesin rendering Webkit sampai dengan versi 27 dan dirancang untuk bekerja dengan kecepatan di atas rata-rata namun tetap ringan dijalankan di perangkat desktop dan mobile.

## *Visual Studio Code*

*Visual Studio* *Code* (*VS Code*) merupakan suatu teks editor ringan dan handal yang dibuat oleh *Microsoft* untuk sistem operasi *multiplatform*, artinya tersedia juga untuk versi *Linux*, *Mac,* dan *Windows*. Teks editor ini secara langsung mendukung bahasa pemrograman *JavaScript, Typescript,* dan *Node.js,* serta bahasa pemrograman lainnya dengan bantuan plugin yang dapat dipasang *via marketplace* Visual Studio Code (seperti *C++, C#, Python, Go, Java*, dst).

Banyaknya fitur yang disediakan oleh *Visual Studio Code*, diantaranya *Intellisense*, *Git Integration, Debugging*, dan fitur ekstensi yang menambah kemampuan teks editor. Fitur tersebut akan terus bertambah seiring dengan bertambahnya versi *Visual Studio Code*. Pembaruan versi *Visual Studio Code* ini juga dilakukan berkala setiap bulan, dan inilah yang membedakan *VS Code* dengan teks editor-teks editor yang lain.

## *MySQL*

MySQL merupakan suatu DBMS singkatan dari (*Database Management System*) menggunakan perintah SQL (*Structured Query Language*) yang banyak digunakan saat ini dalam pembuatan aplikasi berbasis website. MySQL dibagi menjadi dua lisensi, pertama adalah *Free Software* dimana perangkat lunak dapat diakses oleh siapa saja. Dan kedua adalah *Shareware* dimana perangkat lunak berpemilik memiliki batasan dalam penggunaannya.

MySQL termasuk ke dalam RDBMS singkatan daari (*Relational Database Management System*). Sehingga, menggunakan tabel, kolom, baris, di dalam struktur database -nya. Jadi, dalam proses pengambilan data menggunakan metode relational *database*. Dan juga menjadi penghubung antara perangkat lunak dan *database server*.

## *Xampp*

XAMPP merupakan suatu paket perangkat lunak (*software*) komputer yang sistem penamaannya diambil dari akronim kata Apache, MySQL (dulu) / Maria DB (sekarang), PHP, dan Perl. Sementara imbuhan huruf “X” yang terdapat pada awal kata berasal dari istilah *cross platform* sebagai simbol bahwa aplikasi ini bisa dijalankan di empat sistem operasi berbeda, seperti *OS Linux, OS Windows, Mac OS*, dan juga *Solaris.*

XAMPP merupakan sebuah *web server multi-platform* yang bebas lisensi

atau gratis, yang terdiri dari Apache HTTP Server, database MySQL dan penerjemah untuk script yang dituliskan dalam PHP dan bahasa pemrograman. XAMPP dipakai untuk membuat web server lokal di komputer. Hal ini akan memudahkan Anda dalam mengembangkan, mendesain, dan *keperluan testing website*.

## *Visual Paradigm*

Visual Paradigm suatu *software* model sistem visualisasi yang memungkinkan model yang telah dibuat dapat digunakan sebagai representasi proyek-proyek lain dilengkapi dengan beberapa fitur lengkap.

Visual Paradigm sebuah software model dengan sistem *visualisasi* memungkinkan model yang telah dibuat dapat digunakan sebagai representasi proyek-proyek lain dilengkapi dengan beberapa fitur yang ada didalamnya sampai pada menganalisa sebuah proyek yang akan dikerjakan. Diagram dapat disusun sedemikian rupa sehingga dapat dipustakakan menjadi proyek per proyek yang saling berkaitan.

# BAB III

# ANALISIS DAN PERANCANGAN SISTEM

# Analisis Sistem

Aplikasi penjualan bahan bangunan *Point Of Sale* (POS) yang dibangun merupakan suatu sistem yang dapat melakukan proses transaksi, pengelolaan barang yakni persediaan barang masuk (*stock in)*, persediaan barang keluar (*stock out)*, aplikasi *Point Of Sale* (POS) umumnya terdapat perangkat keras dan perangkat lunak yang digunakan seperti *scanner* *barcode*  ketika melakukan *checkout* barang pada aplikasi yang digunakan,.

Aplikasi panjualan bahan bangunan *Point Of Sale* (POS) ber *evolusi* setiap tahunnya, sesuai dengan perkembangan ilmu pengetahuan dan teknologi, dan banyak digunakan di berbagai toko, terutama toko yang bergerak di bidang perdagangan untuk memperoleh kemudahan proses perhitungan barang pesanan oleh pelanggan atau *customer*, pencarian barang, pembuatan laporan secara cepat. Proses transaksi dapat dikategorikan sebagai berikut

1. Pengelolaan *stock in* barang

Pada tahap pengelolaan *stock in* barang, *suppplier* mengirimkan barang ke Toko Satria Jaya, pada proses ini terjadi perubahan pada *stock* *in* barang atau barang bartambah, dan akan mempengaruhi perubahan *stock* di gudang

1. Pengelolaan *stock out* barang

Pada tahap pengelolaan *stock out*  barang, *suppplier* mengirimkan barang ke Toko Satria Jaya, dikarenakan barang tidak sesuai, maka barang dicatat pada *stock out* barang, pada proses ini terjadi perubahan pada *stock in* barang atau barang berkurang, sedangkan pencatatan pada *stock out* barang bertambah, dan akan mempengaruhi perubahan *stock* di gudang.

1. Penjualan barang

Pada tahap ini penjualan barang terjadi antara penjual dan pembeli (*customer*), proses penjualan ini akan mempengaruhi perubahan terhadap *stock* barang di gudang,.

# Analisis Masalah

Masalah yang timbul terjadi pada operasional toko, umumnya setiap masalah yang timbul bermacam - macam tergantung kondisi dalam toko, dengan harapan setiap masalah yang terjadi dapat ditemukannya solusi atau pemecahan masalah.

Masalah yang timbul di Toko Bangunan Satria Jaya yakni terjadi perhitungan barang secara *manual*, transaksi terhadap *customer* atau pelanggan masih menggunkan kalkulator, pencarian barang secara *manual*, pencatatan laporan menggunakan media tulis tangan, kehilangan data atau catatan, potongan harga yang keliru atau tidak sesuai perhitungan*,*.

Terjadinya proses pengelolaan *stock* barang yang masih tidak sesuai dengan *stock* barang yang tersedia di gudang, serta proses penjualan barang terhadap pelanggan atau *customer*, umumnya mempengaruhi *stock* barang di gudang, namun yang terjadi kadang kala tidak dilakukan proses perubahan terhadap *stock* di gudang, karena petugas toko lupa untuk melakukan perubahan *stock* di gudang.

# Analisis sitem yang sedang berjalan

Berikut ini adalah proses bisnis berdasarkan, analisa sistem yang sedang berjalan, sebagai berikut:

1. Pengelolaan *stock* barang
2. *Supplier* melakukan pengiriman barang untuk Toko Satria Jaya
3. Toko Satria Jaya melakukan pemeriksaan atau pengecekan barang yang dikirim oleh *supplier*
4. Jika barang yang diterima sesuai (tidak terdapat cacat atau rusak), maka barang dicatat pada *stock in*  sebagai penerimaan barang di gudang, jika barang yang diterima tidak sesuai, cacat atau rusak maka barang dicatat sebagai *stock out*  dan dilakukan pengiriman kepada *supplier*

|  |
| --- |
|  |
| Gambar 3. 1 Analisis Proses Bisnis Pengelolaan *Stock* Barang |

Berikut ini merupakan proses pengelolaan *stock* barang, berdasarkan analis sistem berjalan dari uraian diatas

1. Penjualan
2. *Customer*  atau pelanggan melakukan pesanan barang
3. Toko Satria Jaya menerima pesanan dan menghitung total pemesanan.
4. *Customer* atau pelanggan, mengecek barang yang di pesan, jika sesuai maka dilakukan pembayaran, jika tidak sesuai maka melakukan pesanan kembali barang.
5. Toko Satria Jaya menerima pembayaran

|  |
| --- |
|  |
| Gambar 3. 2 Analisis Proses Bisnis dari Pelanggan ke Toko Satria Jaya |

# Analisis kebutuhan non fungsional

Kebutuhan *non* Fungsional adalah kebutuhan yang menitikberatkan pada *property* perilaku yang dimiliki oleh sistem, atau suatu fungsi yang menggambarkan kebutuhan yang diperlukan untuk menjalankan aplikasi yang dibangun. Untuk menjalankan aplikasi penjualan bahan bangunan POS *(Point Of Sale)* ada beberapa kebutuhan yang harus dipersiapkan diantaranya yaitu kebutuhan perangkat keras, kebutuhan perangkat lunak, dan pengguna yang menggunakan aplikasi. Analisa kebutuhan *non* fungsional dibangun dengan tujuan aplikasi dapat di digunakan oleh pemilik atau karyawan Satria Jaya.

## Analisa Pengguna

Analisa pengguna merupakan suatu aktivitas yang menunjukan siapa saja yang menggunakan sistem dan apa saja yang bisa dilakukan oleh pengguna di dalam sistem. Berikut ini merupakan pengguna dari aplikasi penjualan bahan bangunan *Poin Of Sale* Satria Jaya:

1. Admin

Admin memiliki tugas untuk mengelola dan menggunakan sistem secara keseluruhan, adapun syarat yang harus dimiliki yaitu sebagai berikut:

1. Paham dan dapat mengoperasikan komputer dengan baik dan benar
2. Teliti dan jujur
3. Memahami alur secara keseluruhan dari aplikasi penjualan bahan bangunan *Point Of Sale*
4. Kasir

Kasir memiliki tugas untuk menggunakan sistem, adapun syarat yang harus dimiliki yaitu sebagai berikut:

1. Paham dan dapat mengoperasikan komputer dengan baik dan benar
2. Teliti dan jujur
3. Bisa menggunakan aplikasi penjualan bahan bangunan *Point Of Sale*

## Analisa kebutuhan perangkat keras

Analisa kebutuhan perangkat keras atau *hardware* bertujuan agar aplikasi penjualn bahan bangunan *point of sale* dapat berjalan dengan baik, spesifikasi yang dibutuhkan untuk menjalankan aplikasi penjualan bahan bangunan *point of sale* yaitu sebagai berikut:

Tabel 3. 1 Tabel Kebutuhan Perangkat Keras

|  |  |  |
| --- | --- | --- |
| No | Perangkat | Spesifikasi |
| 1 | *Processor* | *Dual core i3 up to* |
| 2 | *Memory Ram* | *3 GB up to* |
| 3 | *Hardisk* | 500 GB *up to* |
| 4 | *Monitor* | *Standart* |
| 5 | *Keyboard* | *Standart* |
| 6 | *Mouse* | *Standart* |

## Analisa kebutuhan perangkat lunak

Analisa perangkat lunak merupakan suatu perangkat lunak yang berkaitan dengan operasional kerja komputer, dan dapat digunakan oleh komputer untuk menjadi pendukung agar sistem dapat berjalan dengan baik,diantaranya dalah sebagai berikut

1. Sistem Operasi

Sistem operasi digunakan sebagai perantara antara perangkat lunak dan perangkat keras komputer terdapat berbagai macam sistem operasi baik itu *Windows* 7/8/10 atau *linux* dalam aplikasi ini sistem operasi yang di rekomendasikan yaitu *windows* 10 karena mudah dalam operasional maupun sistem yang *up date*. Sistem operasi menjadi pengatur atau bertanggung jawab dalam penggunaan perangkat keras dan bertugas untuk menjalankan perangkat lunak atau aplikasi yang digunakan

1. Database

Dengan menggunakan *database*, maka pengguna diberikan fasilitas untuk melakukan penyimpanan, pengolahan, dan transformasi data dalam basis data, dalam aplikasi ini *database* yang di rekomendasikan yakni *MYSQL* karena mudah dalam penggunaan dan lebih familiar.

1. Server lokal

Dengan menggunakan *server* lokal maka pengguna diberikan fasilitas untuk menggunakan *database mysql*, untuk *server* lokal yang direkomendasikan yakni *xampp*, karena lebih *portable*, familiar dan mudah untuk digunakan.

# Analisis kebutuhan fungsional

Analisis aplikasi penjualan barang bahan bangunan *point of sale* yang akan di bangun pada Toko Bahan Bangunan Satria Jaya dirancang menggunakan UML (*Unifed Modeling Leanguage*), diantaranya *Secenario-Based Modeling* yang meliputi *Use Case Secenario* (Naratif). *Actor List*, *Use Case List, Use Case Diagram,* dan *Swimlane Diagram*. *Data modeling* yang meliputi *Class Diagram,* dan *Behavioral Modeling* yaitu *Sequence Diagram*.

## *Actor List*

Aktor merupakan orang yang menggunakan sistem atau perangkat baik *software/hardware*, serta memiliki sifat dan fungsi yang dijelaskan dalam sebuah konteks yang memiliki peran terhadap sistem, berikut adalah pendefinisian aktor pada aplikasi penjualan barang bahan bangunan (*pont of sale*) di toko bangunan satria jaya

Tabel 3. 2 Actor List

|  |  |  |
| --- | --- | --- |
| No | Aktor | Deskripsi |
| 1 | Admin | Orang yang mengelola sistem dan akun untuk pengguna atau *user*, serta menerima laporan dari setiap transaksi pada sistem yang berjalan. |
| 2 | Kasir | Orang yang mempunyai hak akses untuk masuk ke sistem dan melakukan proses transaksi. |

## *Usecase Scenario* (Naratif)

*Usecase scenario* merupakan urutan langkah-langkah dalam proses yang dilakukan aktor terhadap sistem maupun yang dilakukan oleh sistem terhadap aktor. Berikut merupakan *use case scenario* aplikasi penjualan bahan bangunan *point of sale* Satria Jaya sebagai berikut:

1. *Usecase Scenario Login*

Tabel 3. 3 *Usecase Scenario* Kelola *Login*

|  |  |
| --- | --- |
| **Identifkasi** | |
| **Nama** | Kelola *Login* |
| **Aktor** | Admin, Kasir |
| **Deskripsi** | *Usecase* ini menggambarkan sistem *login*, dimana setiap aktor melakukan *login* sesuai dengan hak aksesnya, *username* dan *password* di input untuk dapat mengakses halaman utama. |
| **Skenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Membuka Aplikasi | 1. Menampilkan Halaman *Login* |
| 1. Mengisi form *login*, lalu klik “*Login*” atau “*Cancel*” jika batal | 1. *Validasi Username* dan *Password* |
|  | 1. Menampilkan pesan “*Login* Berhasil Selamat Datang” jika *username* dan *password* cocok |
| 1. Klik oke | 1. Menampilkan Halaman Utama |
| **Skenario Alternatif (Proses Gagal)** | |
|  | 1. Menampikan pesan “*username password* salah” |
| 1. Klik oke | 1. Menampilkan kembali *form* *login* |
| **Kondisi Akhir** | Setiap aktor atau pengguna masuk kedalam sistem sesuai hak aksesnya. |

Tabel 3. 4 *Usecase Scenario* Kelola *Supplier*

|  |  |
| --- | --- |
| **Identifkasi** | |
| **Nama** | Kelola *Supplier* |
| **Aktor** | Admin |
| **Deskripsi** | *Usecase* ini menggambarkan aktivitas admin dalam mengelola *supplier* barang seperti tambah, ubah,hapus data *supplier.* |
| **Skenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
|  | 1. Memeriksa status *login* |
| 1. Memasukan data *supplier* sesuai dengan kolom yang ada | 1. Memeriksa validasi sesuai kolom yang telah di isi |
| 1. Klik Simpan Data | 1. Menyimpan data *supplier* ke basis data |
|  | 1. Menampilkan pesan “Data *Supplier* Berhasil Ditambahkan” |
| **Skenario Alternatif (Proses Gagal)** | |
|  | 1. Menampilkan pesan “Data yang anda masukan tidak *valid*” |
| 1. Memperbaiki data masukan yang tidak *valid* | 1. Menampilkan pesan “Data *Supplier* Berhasil Ditambahkan” |
| **Kondisi Akhir** | admin mempunyai akses menambah, ubah, hapus data *supplier* . |

Tabel 3. 5 *Usecase Scenario* Kelola *Customer*

|  |  |
| --- | --- |
| **Identifkasi** | |
| **Nama** | Kelola *Customer* |
| **Aktor** | Admin dan Kasir |
| **Deskripsi** | *Usecase* ini menggambarkan aktivitas admin dan kasir dalam mengelola *customer* atau pelanggan seperti tambah, ubah, hapus *customer.* |
| **Skenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
|  | 1. Memeriksa status *login* |
| 1. Memasukan data *customer* sesuai dengan kolom yang ada | 1. Memeriksa validasi sesuai kolom yang telah di isi |
| 1. Klik Simpan Data | 1. Menyimpan data *customer* ke basis data |
|  | 1. Menampilkan pesan “Data *Customer* Berhasil Ditambahkan” |
| **Skenario Alternatif (Proses Gagal)** | |
|  | 1. Menampilkan pesan “Data yang anda masukan tidak *valid*” |
| 1. Memperbaiki data masukan yang tidak *valid* | 1. Menampilkan pesan “Data *Customer* Berhasil Ditambahkan” |
| **Kondisi Akhir** | Admin dan kasir mempunyai akses menambah, ubah, hapus data *customer* atau pelanggan. |

Tabel 3. 6 *Usecase Scenario* Kelola Produk

|  |  |
| --- | --- |
| **Identifkasi** | |
| **Nama** | Kelola Produk |
| **Aktor** | Admin |
| **Deskripsi** | *Usecase* ini menggambarkan aktivitas admin dalam mengelola *product* seperti tambah, ubah, hapus *product* dimana setiap *product* mempunyai nama barang, kategori, satuan barang. |
| **Skenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
|  | 1. Memeriksa status *login* |
| 1. Klik menu *product* | 1. Menampikan menu barang, kategori, satuan barang |
| 1. Klik menu kategori | 1. Menampilkan kolom kategori |
| 1. Mengisi kolom kategori | 1. Memeriksa *validasi* sesuai kolom yang telah di isi |
| 1. Klik Simpan Data | 1. Menyimpan data kategorike basis data |
|  | 1. Menampilkan pesan “Data Kategori BarangBerhasil Ditambahkan” |
| 1. Klik menu satuan | 1. Menampilkan kolom satuan |
| 1. Mengisi kolom satuan | 1. Memeriksa *validasi* sesuai kolom yang telah di isi |
| 1. Klik Simpan Data | 1. Menyimpan data satuanke basis data |
|  | 1. Menampilkan pesan “Data Satuan BarangBerhasil Ditambahkan” |
| 1. Klik menu item | 1. Menampilkan kolom item |
| 1. Mengisi kolom item | 1. Memeriksa *validasi* sesuai kolom yang telah di isi |
| 1. Klik Simpan Data | 1. Menyimpan data barangke basis data |
|  | 1. Menampilkan pesan “Data BarangBerhasil Ditambahkan” |
| **Skenario Alternatif (Proses Gagal)** | |
| 1. Mengisi kolom kategori | 1. Menampilkan pesan “Data yang anda masukan tidak *valid*” |
| 1. Memperbaiki data masukan yang tidak *valid* | 1. Menampilkan pesan “Data KategoriBerhasil Ditambahkan” |
| 1. Mengisi kolom Satuan | 1. Menampilkan pesan “Data yang anda masukan tidak *valid*” |
| 1. Memperbaiki data masukan yang tidak *valid* | 1. Menampilkan pesan “Data Satuan BarangBerhasil Ditambahkan” |
| 1. Mengisi kolom Item | 1. Menampilkan pesan “Data yang anda masukan tidak *valid*” |
| 1. Memperbaiki data masukan yang tidak *valid* | 1. Menampilkan pesan “Data BarangBerhasil Ditambahkan” |
| **Kondisi Akhir** | Admin mempunyai akses menambah, ubah, hapus data *product* yaitu meliputi satuan, kategori dan item barang. |

Tabel 3. 7 *Usecase Scenario* Kelola *Stock* Barang

|  |  |
| --- | --- |
| **Identifkasi** | |
| **Nama** | Kelola *Stock* Barang |
| **Aktor** | Admin |
| **Deskripsi** | *Usecase* ini menggambarkan aktivitas admin dalam mengelola *stock* barang, barang masuk (*stock in)* terhadap supplier, ataupun berupa retur barang (*stock out)*, admin mempunyai akses cari, tambah , hapusdata *stock*  barang masuk (*stock in),* retur barang (*stock out)*,. |
| **Skenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
|  | 1. Memeriksa status *login* |
| 1. Klik menu *Stock* Barang | 1. Menampikan menu admin, pengelolaan barang masuk (*stock in)*, pengelolaan barang keluar (*stock out).* |
| 1. Klik menu pengelolaan barang masuk (*stock in)* | 1. Sistem menampilkan halaman antarmuka kelola barang masuk (*stock in)*, admin mempunyai akses cari, hapus, simpan barang. |
| 1. Mengisi kolom yang tersedia pada menu *stock in* barang | 1. Memeriksa validasi sesuai kolom yang telah di isi |
| 1. Klik Simpan Data | 1. Menyimpan data *stock in* barangke basis data |
|  | 1. Menampilkan pesan “Data Barang MasukBerhasil Ditambahkan” |
| 1. Klik menu pengelolaan barang keluar (*stock out)* | 1. Sistem menampilkan halaman antarmuka kelola barang keluar (*stock out)*, kasir mempunyai akses cari, tambah, hapus, simpan barang. |
| 1. Mengisi kolom yang tersedia pada menu *stock out*  barang | 1. Memeriksa validasi sesuai kolom yang telah di isi |
| 1. Klik Simpan Data | 1. Menyimpan data *stock out* barangke basis data |
|  | 1. Menampilkan pesan “Data Barang KeluarBerhasil Ditambahkan” |
| **Skenario Alternatif (Proses Gagal)** | |
| 1. Mengisi kolom *stock in* barang | 1. Menampilkan pesan “Data yang anda masukan tidak *valid*” |
| 1. Memperbaiki data masukan yang tidak *valid* | 1. Menampilkan pesan “Data Barang MasukBerhasil Ditambahkan” |
| 1. Mengisi kolom *stock out* barang | 1. Menampilkan pesan “Data yang anda masukan tidak *valid*” |
| 1. Memperbaiki data masukan yang tidak *valid* | 1. Menampilkan pesan “Data Barang KeluarBerhasil Ditambahkan” |
| **Kondisi Akhir** | Admin mempunyai akses malakukan *stock* barang masuk dan keluar, meliputi tambah, cari, dan hapus data *stock* barang |

Tabel 3. 8 *Usecase Scenario* Kelola *User*

|  |  |
| --- | --- |
| **Identifkasi** | |
| **Nama** | Kelola *User* |
| **Aktor** | Admin |
| **Deskripsi** | *Usecase* ini menggambarkan aktivitas admin dalam mengelola *user* atau pengguna sistem, admin mempunyai akses tambah, ubah, hapus *user* atau pengguna sistem,*.* |
| **Skenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
|  | 1. Memeriksa status *login* |
| 1. Memasukan data *user* atau penggunasesuai dengan kolom yang ada | 1. Memeriksa *validasi* sesuai kolom yang telah di isi |
| 1. Klik Simpan Data | 1. Menyimpan data *user* ke basis data |
|  | 1. Menampilkan pesan “Data *user* Berhasil Ditambahkan” |
| **Skenario Alternatif (Proses Gagal)** | |
|  | 1. Menampilkan pesan “Data yang anda masukan tidak *valid*” |
| 1. Memperbaiki data masukan yang tidak *valid* | 1. Menampilkan pesan “Data *user* atau PenggunaBerhasil Ditambahkan” |
| **Kondisi Akhir** | Admin mempunyai akses menambah, ubah, hapus data *user* atau pengguna sistem |

Tabel 3. 9 *Usecase Scenario* Kelola Laporan

|  |  |
| --- | --- |
| **Identifkasi** | |
| **Nama** | Kelola Laporan |
| **Aktor** | Admin |
| **Deskripsi** | *Usecase* ini menggambarkan aktivitas admin dalam mengelolaLaporan meliputi Laporan barang atau *stock* barangdan penjualan, admin mempunyai akses cetak dan hapus laporan |
| **Skenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
|  | 1. Memeriksa status *login* |
| 1. Klik menu laporan | 1. Menampikan menu laporan penjualan dan laporan barang atau *stock* |
| 1. Klik menu laporan penjualan | 1. Sistem menampilkan halaman atau antarmuka penjualan barang, admin mempunyai akses cari, cetak, dan hapus laporan berdasarkan tanggal penjualan. |
| 1. Klik tombol cetak | 1. Sistem menampilkan halaman penjualan yang akan di cetak, kemudian sistem mengeluarkan data transaksi penjualan |
| 1. Klik tombol laporan barang atau *stock* | 1. Sistem menampilkan halaman atau antarmuka *stock* barang, admin mempunyai akses cari, cetak laporan secara keseluruhan |
| 1. Klik tombol cetak | 1. Sistem menampilkan halaman *stock* barang yang akan di cetak, kemudian sistem mengeluarkan data *stock* barang |
| **Skenario Alternatif (Proses Gagal)** | |
|  | 1. Menampilkan “Data yang anda cari kosong” |
| 1. Memperbaiki data masukan yang sesuai | 1. Table menampilkan data yang di cari di urutkan |
| **Kondisi Akhir** | Admin mempunyai akses cari, cetak, dan hapus pada menu laporan. |

Tabel 3. 10 Usecase Scenario Kelola Penjualan

|  |  |
| --- | --- |
| **Identifkasi** | |
| **Nama** | Kelola Laporan |
| **Aktor** | Admin |
| **Deskripsi** | *Usecase* ini menggambarkan aktivitas admin dalam mengelolapenjualan terhadap proses transaksi kepada *customer* |
| **Skenario Utama** | |
| **Aksi Aktor** | **Reaksi Sistem** |
|  | 1. Memeriksa status *login* |
| 1. Klik menu penjualan | 1. Sistem menampilkan halaman antarmuka transaksi penjualan barang, kasir mempunyai akses cari, tambah, ubah, hapus barang dan cetak transaksi. |
| 1. Memasukan data yang tersedia pada menu kasir, sesuai dengan kolom yang ada | 1. Memeriksa validasi sesuai kolom yang telah di isi |
| 1. Klik tombol proses | 1. Menyimpan data transaksi, dan menampilkan data transaksi penjualan atau *struct* |
| 1. Klik tombol cetak | 1. Sistem mengeluarkan *output* data transaksi penjualan |
| **Skenario Alternatif (Proses Gagal)** | |
| 1. Mengisi kolom penjualan | 1. Menampilkan pesan “Data yang anda masukan tidak *valid*” |
| 1. Memperbaiki data masukan yang tidak *valid* | 1. Menyimpan data penjualan, dan menampilkan data transaksi penjualan atau *struct* |
| **Kondisi Akhir** | Kasir mempunyai akses cari, cetak, dan hapus pada menu laporan. |

## *Usecase List*

Dafar *usecase* menjelaskan tentang *usecase* yang ada dalam sistem, daftar *usecase* berupa kode *usecase,* nama *usecase* dan keterangan dari tiap *usecase.* Adapun *usecase list* yang terdapat pada aplikasi penjualan bahan bangunan Toko Satria Jaya, sebagai berikut:

Tabel 3. 11 *Usecase List*

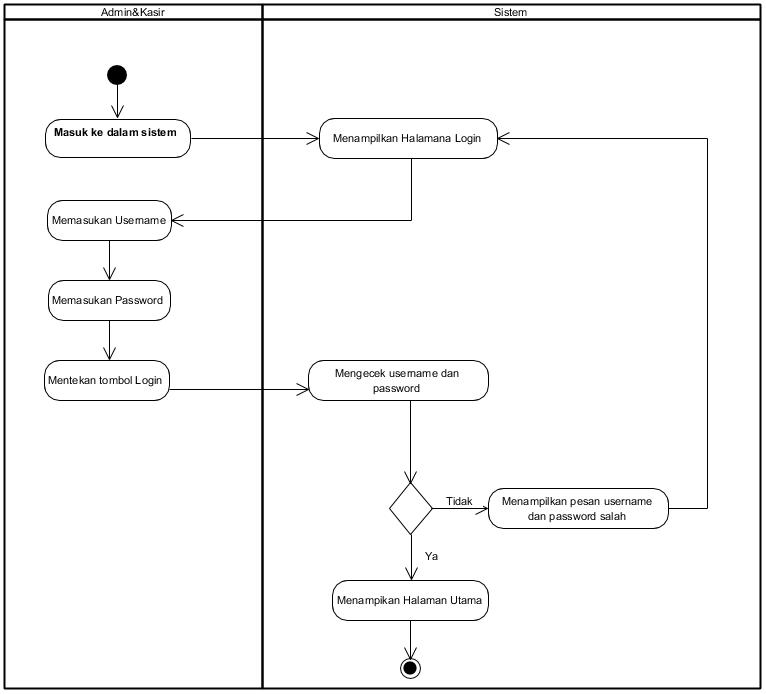
|  |  |  |
| --- | --- | --- |
| Kode | Sistem *Usecase* | Keterangan |
| U1 | *Login* | Digunakan untuk memastikan setiap *user* masuk kedalam sistem, sesuai hak aksesnya berdasarkan *username* dan *password* yang disertakan. |
| U2 | Kelola *supplier* | Digunakan oleh aktor untuk mengelola data *supplier* |
| U3 | Kelola *customer* | Digunakan oleh aktor untuk mengelola data *customer* atau pelanggan. |
| U4 | Kelola Produk | Digunakan oleh aktor untuk mengelola data produk meliputi *item*, kategori, satuan barang. |
| U5 | Kelola Penjualan | Digunakan oleh aktor untuk mengelola data penjualan dengan *customer* |
| U6 | Kelola *User* | Digunakan oleh aktor untuk mengelola data *user* atau pengguna sistem |
| U7 | Kelola Laporan | Digunakan oleh aktor untuk menerima laporan meliputi laporan penjualan dan laporan barang atau *stock* barang |
| U8 | Kelola *Stock* Barag | Digunakan oleh aktor untuk mengelola data *stock* barang masuk dan barang keluar |

## *Usecase Diagram*

|  |
| --- |
|  |
| Gambar 3. 3 *Usecase Diagram* |

Gambar 3.3 *Usecase* Diagram ini terdapat 8 fungsi dan 2 aktor yaitu admin dan kasir, dimana *usecase login* dapat di akses oleh aktor tersebut, sedangakan untuk *use case* kelola *customer, supplier,* laporan yang merupakan *generalisasi* dari laporan penjualan dan *stock* barang, *user,* *stock* barang yang merupakan *generalisasi* dari *stock* barang masukdan keluar, produk yang merupakan *generalisasi* dari kategori, satuan, item, hanya dapat diakses oleh aktor admin, sedangkan untuk *usecase* kelola *customer* dan penjualan hanya dapat diakses oleh aktor kasir.

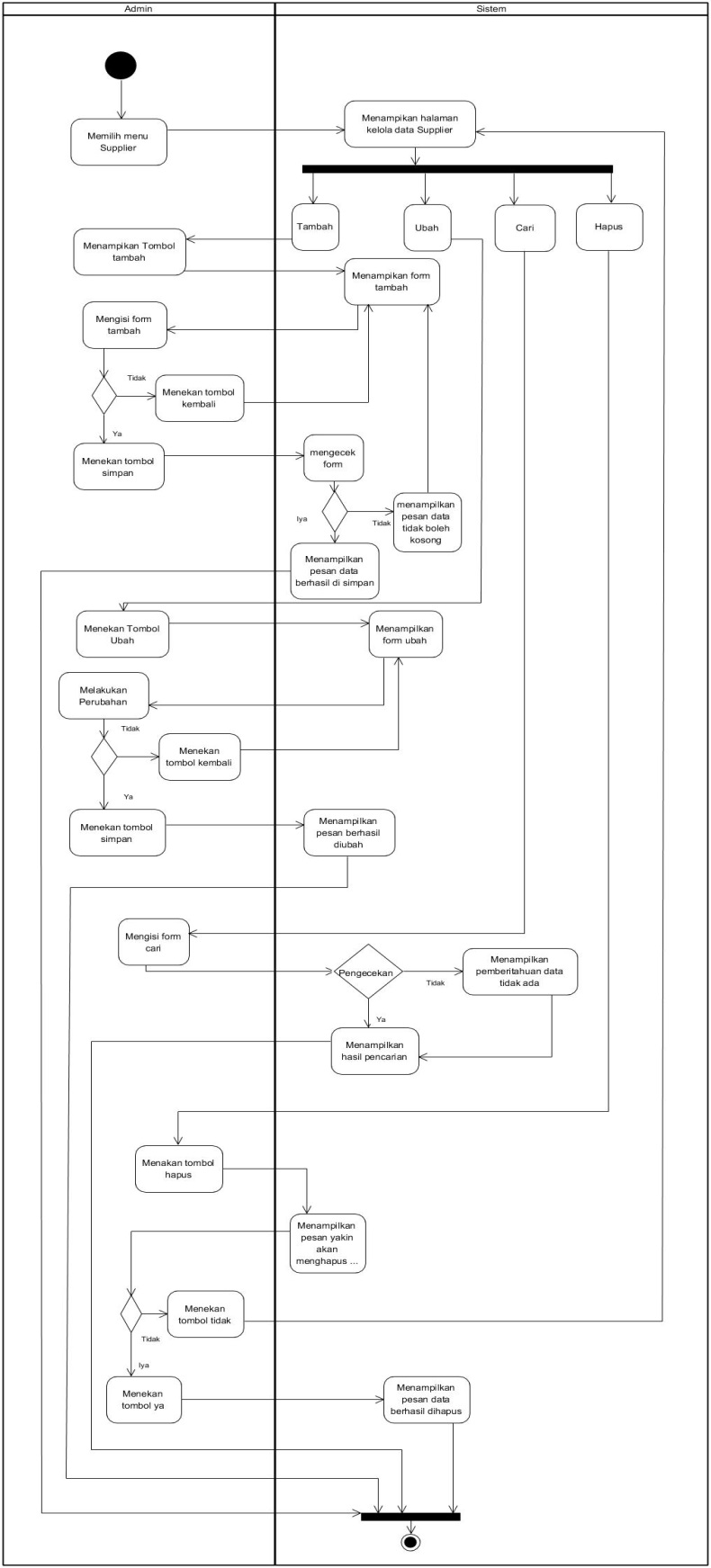
## *Swimline Diagram*

1. *Swimlne Diagram Login*

|  |
| --- |
|  |
| Gambar 3. 4 *Swimeline Diagram* *Login* |

Keterangan:

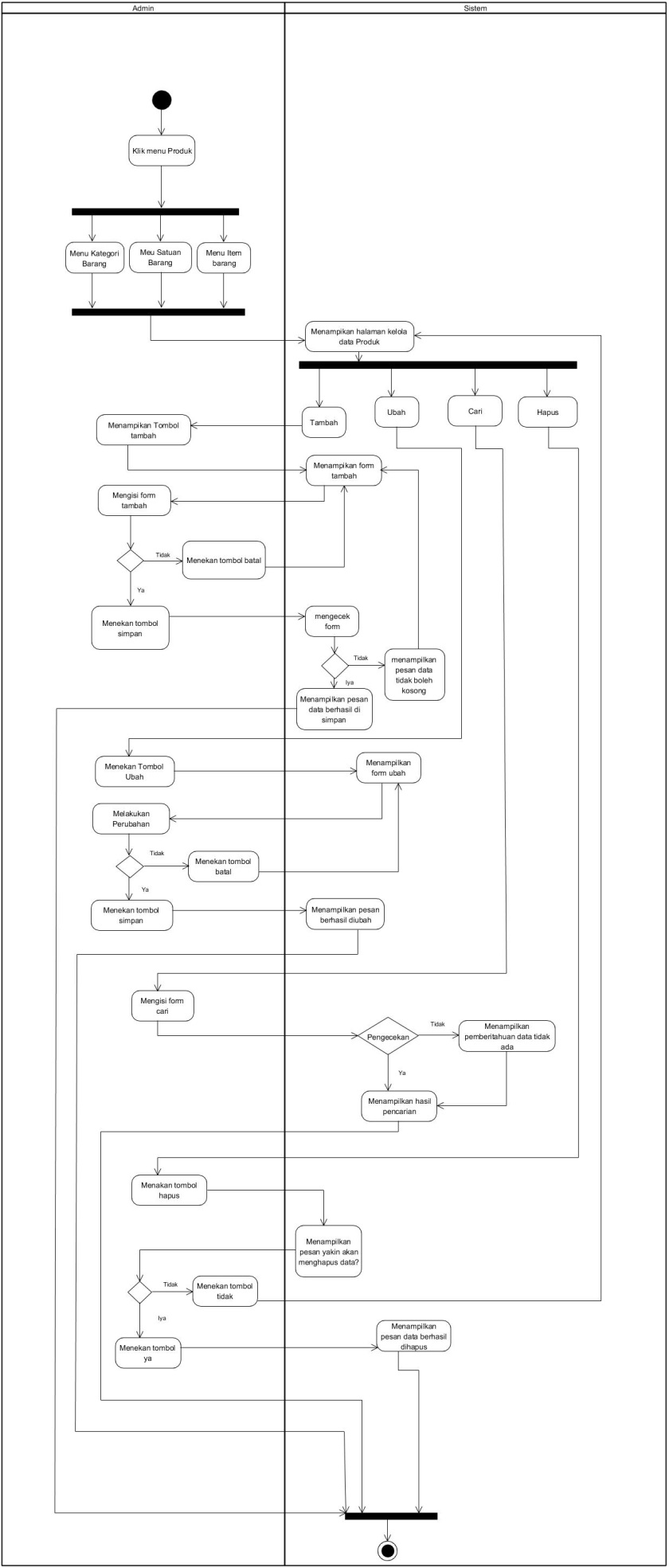
Gambar 3.4 *Swimeline Diagram Login* menunjukan aktifitas aktor yaitu *user* yang terdiri dari admin dan kasir, *user* masuk ke dalam sistem menampilkan halaman *login*, lalu *user* memasukan *username* dan *password* dan menekan tombol *login*, sistem akan memvalidasi data yang dimasukan, jika data tidak sesuai makan akan menampilkan pesan kesalahan, dan jika sesuai maka user akan masuk ke dalam sistem sesuai hak aksesnya.

1. *Swimlne* Diagram Kelola *Supplier*

|  |
| --- |
|  |
| Gambar 3. 5 *Swimeline Diagram Supplier* |

Keterangan:

Gambar 3.5 *Swimeline Diagram* Kelola *Supplier* menunjukan aktifitas *user* yaitu admin saat masuk dalam sistem, meliputi proses cari, tambah, ubah, dan hapus data kelola *supplier*, proses pengisian *form supplier*, sistem akan mem-*validasi* data yang dimasukan oleh *user*, jika sesuai maka data akan tersimpan dalam basis data, jika tidak sesuai sistem akan menampilkan pesan kesalahan inputan data.

1. *Swimlne* Diagram Kelola Produk

|  |
| --- |
|  |
| Gambar 3. 6 *Swimeline Diagram Produk* |

Keterangan:

Gambar 3.6 *Swimeline Diagram* Kelola Produk meliputi kategori, item, dan satuan barang, menunjukan aktifitas *user* yaitu admin saat masuk dalam sistem, meliputi proses cari, tambah, ubah, dan hapus data kelola produk, proses pengisian pada tiap-tiap menu pada *form* produk, sistem akan mem-*validasi* data yang dimasukan oleh *user*, jika sesuai maka data akan tersimpan dalam basis data, jika tidak sesuai sistem akan menampilkan pesan kesalahan inputan data.

1. *Swimlne* Diagram Kelola *Stock* Barang

|  |
| --- |
|  |
| Gambar 3. 7 *Swimeline Diagram Stock* Barang |

Keterangan:

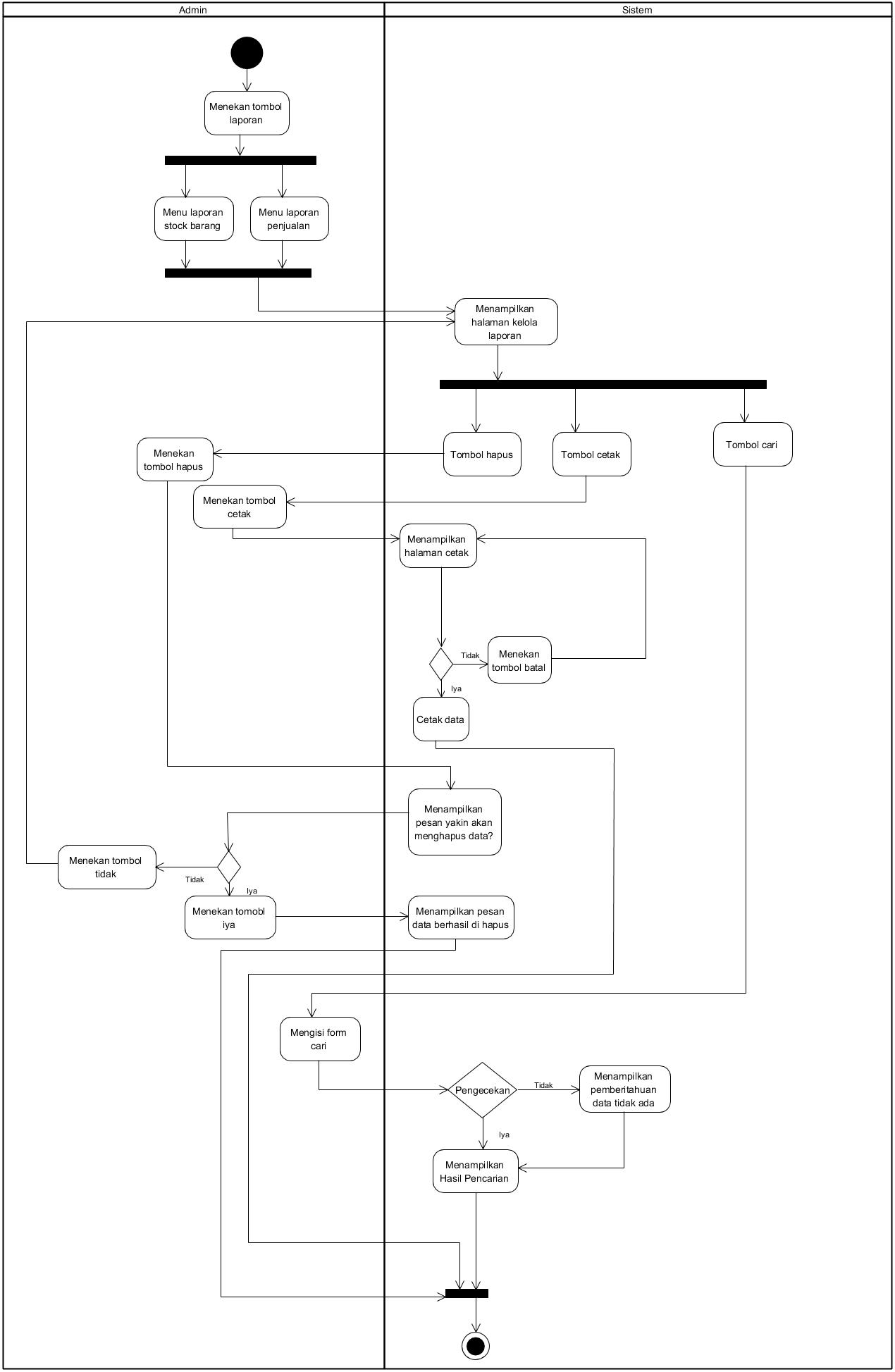
Gambar 3.7 *Swimeline Diagram* Kelola *Stock* Barang meliputi barang masuk dan barang keluar, menunjukan aktifitas *user* yaitu admin saat masuk dalam sistem, meliputi proses cari, tambah, dan hapus data kelola *stock*barang, proses pengisian pada tiap-tiap menu pada *form stock* barang, sistem akan mem-*validasi* data yang dimasukan oleh *user*, jika sesuai maka data akan tersimpan dalam basis data, jika tidak sesuai sistem akan menampilkan pesan kesalahan inputan data.

1. *Swimlne* Diagram Kelola *User*

|  |
| --- |
|  |
| Gambar 3. 8 *Swimeline Diagram User* |

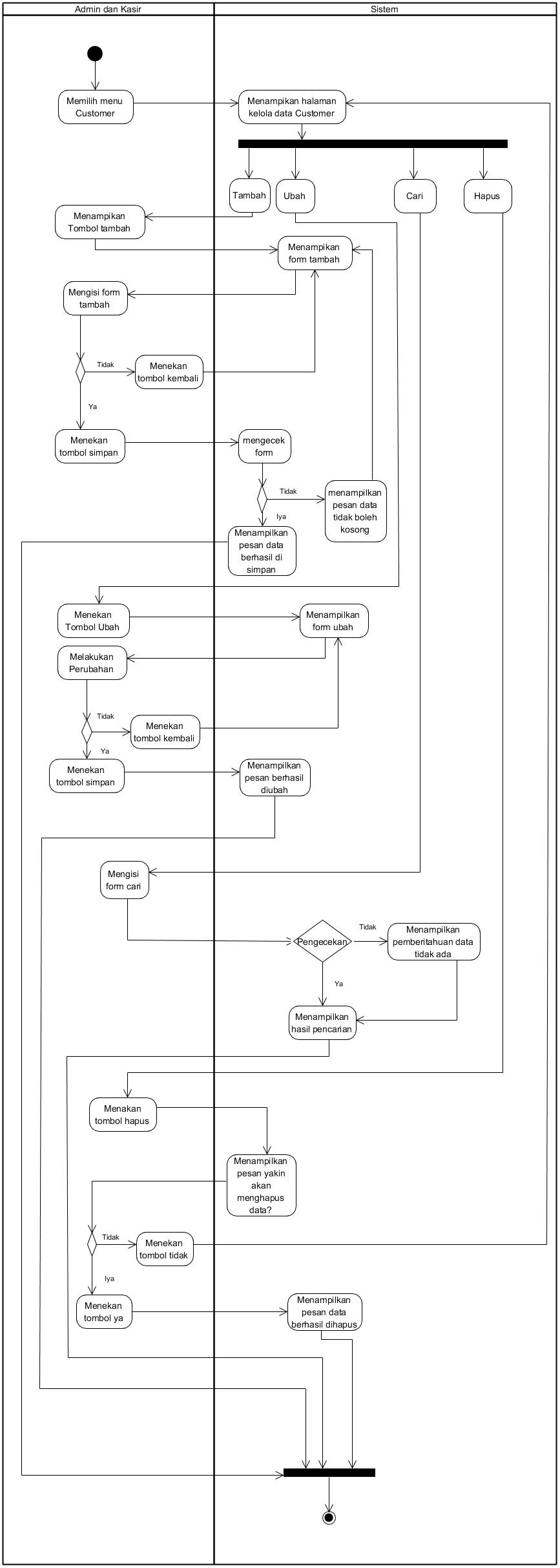
Keterangan:

Gambar 3.8 *Swimeline Diagram* Kelola *User* menunjukan aktifitas *user* yaitu admin saat masuk dalam sistem, meliputi proses cari, tambah, ubah, dan hapus data kelola *user*, proses pengisian *form user*, sistem akan mem-*validasi* data yang dimasukan oleh *user*, jika sesuai maka data akan tersimpan dalam basis data, jika tidak sesuai sistem akan menampilkan pesan kesalahan inputan data.

1. *Swimlne* Diagram Kelola Laporan

|  |
| --- |
|  |
| Gambar 3. 9 *Swimeline Diagram* Laporan |

Gambar 3.9 *Swimeline Diagram* Kelola Laporan meliputi laporan penjualan, dan laporan *stock* barang, menunjukan aktifitas *user* yaitu admin saat masuk dalam sistem, meliputi proses cari, cetak, dan hapus data, proses pencarian data berdasarkan pengisian data yang dimasukan oleh user, sistem akan mem-*validasi* data yang dimasukan oleh *user*, jika sesuai maka data akan ditemukan, jika tidak sesuai sistem akan menampilkan pesan bahwa data tidak ada atau tidak tersedia.

1. *Swimlne Diagram Kelola Customer*

|  |
| --- |
|  |
| Gambar 3. 10 *Swimeline Diagram Customer* |

Keterangan:

Gambar 3.10 *Swimeline Diagram* Kelola *Customer* menunjukan aktifitas *user* yaitu admin dan kasir saat masuk dalam sistem, meliputi proses cari, tambah, ubah, dan hapus data kelola *user*, proses pengisian *form user*, sistem akan mem-*validasi* data yang dimasukan oleh *user*, jika sesuai maka data akan tersimpan dalam basis data, jika tidak sesuai sistem akan menampilkan pesan kesalahan inputan data.

1. *Swimlne Diagram* KelolaPenjualan

|  |
| --- |
|  |
| **Gambar 3. 11 *Swimeline Diagram* Penjualan** |

Keterangan:

Gambar 3.11 *Swimeline Diagram* Kelola Penjualan menunjukan aktifitas *user* yaitu kasir saat masuk dalam sistem, meliputi proses cari, cetak tambah, ubah, dan hapus data kelola penjualan, proses pengisian *form* penjualan, sistem akan mem-*validasi* data yang dimasukan oleh *user*, jika sesuai maka data akan tersimpan dalam basis data dan di lanjutkan dengan proses cetak data penjualan, jika tidak sesuai sistem akan menampilkan pesan kesalahan inputan data.

## *Class Diagram*

|  |
| --- |
|  |
| Gambar 3. 12 *Class Diagram* Toko Satria Jaya |

Keterangan

Berdasarkan Gambar 3.12 *Class Diagram* diatas , terdapat 10 *class* yaitu *login,* kelola *supplier,* kelola *customer,* kelola *user*, kelola transaksi, kelola detail penjualan, kelola *stock* barang, kelola produk yang merupakan *generalisasi* dari kelola kategori, kelola satuan, kelola *item,* kelola penjualan, kelola laporan yang merupakan *generalisasi* dari kelola laporan penjualan dan kelola laporan *stock* barang. *class* kelola *customer* yang memiliki hubungan *composition* dengan *login* dan hubungan *aggregation* dengan penjualan, kelola *supplier* yang memiliki hubungan *composition* dengan *login* dan hubungan *aggregation* dengan kelola *stock* barang , kelola *user* yang memiliki hubungan *composition* dengan *login* dan hubungan *aggregation* dengan transaksi,.

Kelola transaksi yang memiliki hubungan *composition* dengan *login* dan hubungan *aggregation* dengan kelola *user,* kelola detail penjualan memiliki hubungan *composition* dengan *login* dan hubungan *aggregation* dengan kelola penjualan, keloal penjualan memiliki hubungan *compositon* dengan *login* dan hubungan *aggregation* dengan *customer,* kelola *customer* memiliki hubungan *compositon* dengan *login* dan hubungan *aggregation* dengan kelola penjualan, kelola produkmemiliki hubungan *compositon* dengan *login* dan hubungan *aggregation* dengan kelola detail penjualan dan kelola laporan, kelola *stock* barangmemiliki hubungan *compositon* dengan *login* dan kelola produk,sedangkan hubungan *aggregation* dengan kelola *supplier*, kelola laporan memiliki hubungan *compositon* dengan *login* dan kelola penjualan, sedangkan hubungan *aggregation* dengan kelola produk,.

## *State Diagram*

1. *State Diagram Login*

|  |
| --- |
|  |
| **Gambar 3. 13 *State Diagram Login*** |

Keterangan :

Pada Gambar 3.13 *State Diagram Login* terdapat 5 fungsi yaitu, *view\_login*()*,* verifikasi*login*(), *login*(), *logout*(), dan *view*(). Fungsi *view\_login*() akan menampilkan halaman *login*, verifikasi*login*() akan memverifikasi *username* dan *password*, fungsi *login*() akan menampilkan halaman menu,*view*() akan menampilkan menu aplikasi yakni *view\_user*()*, view\_supplier*()*, view\_customer*()*, view\_*penjualan(), *view\_stock*()*, view\_*produk(), *view\_*laporan(), dan fungsi *logout*() akan keluar dari halaman menu utama.

1. *State Diagram User*

|  |
| --- |
|  |
| **Gambar 3. 14 *State Diagram User*** |

Keterangan :

Pada Gambar 3.14 *State Diagram User* terdapat 5 fungsi yaitu, *view\_user*()*, add\_user*(), *update\_user*(), *delete\_user*(), dan *search\_user*(). Fungsi *view\_user*()*,* akan menampilkan halaman *user,* *add\_user*() akan menampilkan pesan “data berhasil ditambahkan”, *update\_user*() akan menampilkan pesan “data berhasil diubah”, *delete\_user*() akan menampilkan pesan “data berhasil dihapus”,*search\_user*() akan menampilkan *list* data.

1. *State Diagram Supplier*

|  |
| --- |
|  |
| **Gambar 3. 15 *State Diagram Supplier*** |

Keterangan :

Pada Gambar 3.15 Dalam *State Diagram Supplier* terdapat 5 fungsi yaitu, *view\_supplier(), add\_supplier*(), *update\_supplier*(), *delete\_supplier*(), dan *search\_supplier()*.

Fungsi *view\_supplier*()*,* akan menampilkan halaman *supplier,* *add\_supplier*() akan menampilkan pesan “data berhasil ditambahkan”, *update\_supplier*() akan menampilkan pesan “data berhasil diubah”, *delete\_supplier*() akan menampilkan pesan “data berhasil dihapus”, *search\_supplier*() akan menampilkan *list* data.

1. *State Diagram Customer*

|  |
| --- |
|  |
| **Gambar 3. 16 *State Diagram Customer*** |

Keterangan :

Pada Gambar 3.16 *State Diagram Customer* terdapat 5 fungsi yaitu, *view\_customer*()*, add\_customer*(), *update\_customerr*(), *delete\_customer*(), dan *search\_customer*().

Fungsi *view\_customer*()*,* akan menampilkan halaman *customer,* *add\_customer*() akan menampilkan pesan “data berhasil ditambahkan”, *update\_customer*() akan menampilkan pesan “data berhasil diubah”, *delete\_customer*() akan menampilkan pesan “data berhasil dihapus”, *search\_customer*() akan menampilkan *list* data.

1. *State Diagram* Penjualan

|  |
| --- |
|  |
| **Gambar 3. 17 *State Diagram* Penjualan** |

Keterangan :

Pada Gambar 3.17 *State Diagram* Penjualanterdapat 7 fungsi yaitu, *view\_*penjualan()*, search\_*penjualan(), *add\_*penjualan(), *update\_*penjualan(), *delete\_*penjualan()*, process\_*penjualan(), dan *print­­\_*penjualan().

Fungsi *view\_*penjualan()*,* akan menampilkan halamanpenjualan *,* *search\_*penjualan() akan menampilkan data barang , *add\_*penjualan() akan menampilkan data dalam tabel, *update\_*penjualan() akan menampilkan pesan “data berhasil diubah”, *delete \_*penjualan() akan menampilkan pesan “data berhasil dihapus”, *process\_*penjualan() akan menampilkan data barang yang dipesan dan *print\_*penjualan() akan menampilkan halaman cetak data barang.

1. *State Diagram Stock* Barang

|  |
| --- |
|  |
| Gambar 3. 18 *State Diagram Stock* Barang |

Keterangan :

Pada Gambar 3.18 *State Diagram* Penjualanterdapat 9 fungsi yaitu, *view\_stock*()*, view\_stock-*in(), *add\_stock-in*(), *delete\_stock-in*(), *search\_stock-in*(), *view\_stock-out*(), *add\_stock-out*(), *delete\_stock-out*(), dan *search\_stock-out*()*,*.

Fungsi *view\_stock*()*,* akan menampilkan halaman *stock* barang*,* *view\_stock-in*() akan menampilkan data barang masuk , *add\_stock-in*() akan menampilkan pesan “data berhasil ditambahkan”, *delete\_stock-in* () akan menampilkan pesan “data berhasil dihapus”, *search\_stock-in*() akan menampilkan data dalam tabel, *view\_stock-out*() akan menampilkan data barang keluar , *add\_stock-out*() akan menampilkan pesan “data berhasil ditambahkan”, *delete\_stock-out*() akan menampilkan pesan “data berhasil dihapus”, dan *search\_stock-out*() akan menampilkan data dalam tabel,.

1. *State Diagram* Produk

|  |
| --- |
|  |
| Gambar 3. 19 *State Diagram* Produk |

Keterangan :

Pada Gambar 3.19 *State Diagram* Produkterdapat 16 fungsi adalah berikut ini: *view\_*produk()*, view\_*kategori(), *add\_*kategori(), *update\_*kategori(), *delete\_*kategori(), *search\_*kategori(), *view\_*satuan(), *add\_*satuan(), *update\_*satuan(), *delete\_*satuan (), dan *search\_*satuan()*, view\_*item(), *add\_*item(), *update\_*item(), *delete\_*item(), dan *search\_*item()*.*

Fungsi *view\_*produk()*,* akan menampilkan halamanproduk*,* *view\_*kategori() akan menampilkan data kategori barang, *add\_*kategori() akan menampilkan pesan “data berhasil ditambahkan”, *update\_*kategori() akan menampilkan pesan “data berhasil diubah”, *delete\_*kategori() akan menampilkan pesan “data berhasil dihapus”, *search\_*kategori() akan menampilkan *list* data dalam tabel, *view\_*satuan() akan menampilkan data satuan barang, *add\_*satuan() akan menampilkan pesan “data berhasil ditambahkan”, *update\_*satuan() akan menampilkan pesan “data berhasil diubah”, *delete\_*satuan() akan menampilkan pesan “data berhasil dihapus”, *search\_*satuan() akan menampilkan *list* data dalam tabel, *view\_item*() akan menampilkan data *item* barang, *add\_item*() akan menampilkan pesan “data berhasil ditambahkan”, *update\_item*() akan menampilkan pesan “data berhasil diubah”, *delete\_item*() akan menampilkan pesan “data berhasil dihapus”, dab *search\_item*() akan menampilkan *list* data dalam tabel,.

## *Sequence Diagram*

1. *Sequence Diagram Login*

|  |
| --- |
|  |
| Gambar 3. 20 *Sequence Diagram Login* |

Keterangan:

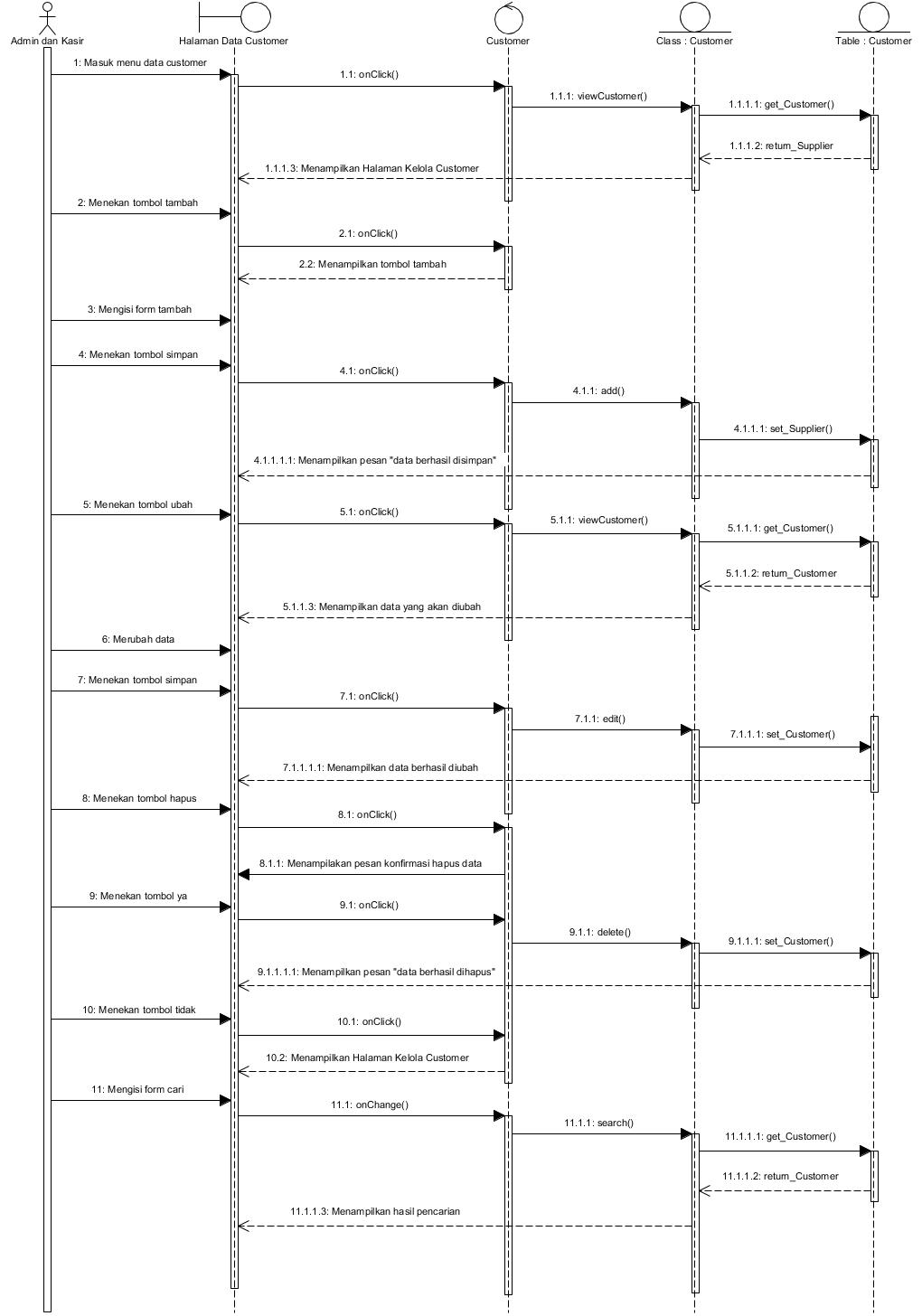
Gambar 3.20 *Sequence Diagram Login* menggambarkan *user* yang terdiri dari admin dan kasir, *user* memasukan *username* dan *password* untuk masuk ke dalam sistem.

1. *Sequence Diagram Supplier*

|  |
| --- |
|  |
| Gambar 3. 21 *Sequence Diagram Supplier* |

Keterangan:

Gambar 3.21 *Sequence Diagram Supplier* menggambarkan admin memiliki hak akses untuk melakukan kelola *supplier* meliputi tambah, ubah, hapus data *supplier*

1. *Sequence Diagram Customer*

|  |
| --- |
|  |
| Gambar 3. 22 *Sequence Diagram Supplier* |

Keterangan:

Gambar 3.22 *Sequence Diagram Customer* menggambarkan admin dan kasir memiliki hak akses untuk melakukan kelola *customer* meliputi tambah, ubah, hapus data *customer*

1. *Sequence Diagram* Produk

|  |
| --- |
|  |
| Gambar 3. 23 *Sequence Diagram* Produk |

Keterangan:

Gambar 3.23 *Sequence Diagram* Produk menggambarkan admin memiliki hak akses untuk kelola produk, kelola produk memiliki menu kategori, satuan, dan item produk, setiap menu bisa melakukan tambah, ubah, dan hapus, yang disertakan bentuk *loop* yang berarti menunjukan proses yang sama terjadi pada setiap menu tersebut.

1. *Sequence Diagram Stock* Barang

|  |
| --- |
|  |
| Gambar 3. 24 *Sequence Diagram Stock* Barang |

Keterangan:

Gambar 3.24 *Sequence Diagram Stock* Barang menggambarkan admin memiliki hak akses untuk kelola *stock* barang, kelola *stock* barang memiliki menu barang masuk, dan barang keluar, setiap menu bisa melakukan cari, tambah, hapus data *stock* barang

1. *Sequence Diagram User*

|  |
| --- |
|  |
| Gambar 3. 25 *Sequence Diagram User* |

Keterangan:

Gambar 3.25 *Sequence Diagram User* menggambarkan admin memiliki hak akses untuk melakukan kelola *user* sistem meliputi tambah, ubah, hapus data *user.*

1. *Sequence Diagram* Laporan

|  |
| --- |
|  |
| Gambar 3. 26 *Sequence Diagram* Laporan |

Keterangan:

Gambar 3.26 *Sequence Diagram* Laporanmenggambarkan admin memiliki hak akses untuk kelola laporan, kelola laporan memiliki menu laporan penjualan dan laporan *stock*  barang, setiap menu bisa melakukan cari, cetak, dan hapus yang disertakan bentuk *loop* yang berarti menunjukan proses yang sama terjadi pada setiap menu tersebut.

1. *Sequence Diagram* penjualan

|  |
| --- |
|  |
| Gambar 3. 27 *Sequence Diagram* Penjualan |

Keterangan:

Gambar 3.27 *Sequence Diagram* Kelola Penjualan menunjukan aktifitas *user* yaitu kasir saat masuk dalam sistem, meliputi proses cari, cetak tambah, ubah, dan hapus data kelola penjualan, proses pengisian *form* penjualan, sistem akan mem-*validasi* data yang dimasukan oleh *user*, jika sesuai maka data akan tersimpan dalam basis data dan di lanjutkan dengan proses cetak data penjualan, jika tidak sesuai sistem akan menampilkan pesan kesalahan inputan data.

## *Perancangan ERD (Entity Relationship Diagram)*

|  |
| --- |
|  |
| Gambar 3. 28 *ERD* Toko Satria Jaya |

Keterangan

Berdasarkan Gambar 3.28 Terdapat 10 tabel yakni penjualan, *customer*, *user, stock, supplier*, transaksi, item, satuan, kategori, detail penjualan, setiap tabel saling terhubung satu sama lain dengan *primary key* ataupun *foreign key* yang terdapat dalam setiap tabel, dalam satu tabel memungkinkan memilki *foreign key* lebih dari satu, dalam kardinalitas terdapat *one to one, one to many*, dan *many to many,* yang implementasinya terdapat pada gambar *ERD* (*Entity Relationship Diagram)* di atas.

## *Struktur Tabel*

Struktur tabel merupakan gambaran detail dari tabel yang berisi *field,* tipe data, *range data,* *primary key, foreign key* dari field dan lain sebagainya. Berikut merupakan *table* yang terdapat pada sistem Toko Satria Jaya

1. Table *user*

Tabel 3. 12 Tabel *User*

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Deskripsi** |
| User\_id | integer | 10 | *Primary key* |
| Username | varchar | 10 |  |
| Nama | varchar | 10 |  |
| Password | varchar | 10 |  |
| Alamat | varchar | 50 | *Nullabel* |
| Level | integer | 10 |  |

1. Table *stock*

Tabel 3. 13 Tabel *Stock*

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Deskripsi** |
| Stock\_id | integer | 10 | *Primary key* |
| Item\_id | integer | 10 | *Foreign key* |
| Type | enum | 10 |  |
| Detail | varchar | 25 |  |
| Supplier\_id | integer | 10 | *Foreign key* |
| Jumlah | integer | 10 |  |
| date | date |  |  |
| User\_id | integer | 10 | *Foreigm key* |

1. Table *Supplier*

Tabel 3. 14 Tabel *Supplier*

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Deskripsi** |
| Supplier\_id | integer | 10 | *Primary key* |
| Nama | varchar | 25 |  |
| No\_telepon | varchar | 15 |  |
| Alamat | varchar | 25 |  |
| Deskripsi | text | 25 | *Nullable* |

1. Table Detail Penjualan

Tabel 3. 15 Tabel Detail Penjualan

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Deskripsi** |
| Detail\_id | integer | 10 | *Primary key* |
| Penjualan\_id | integer | 10 | *Foreign key* |
| Item\_id | integer | 10 | *Foreign key* |
| Harga | integer | 10 |  |
| Jumlah | integer | 10 |  |
| total | integer | 10 |  |

1. Table *Item*

Tabel 3. 16 Tabel Item

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Deskripsi** |
| Item\_id | integer | 10 | *Primary key* |
| Nama\_item | varchar | 25 |  |
| barcode | varchar | 100 | *Nullable* |
| Kategori\_id | integer | 10 | *Foreign key* |
| Satuan\_id | integer | 10 | *Foreign key* |
| Harga | integer | 10 |  |
| stock | integer | 10 | *Nullable* |

1. Table Satuan

Tabel 3. 17 Tabel Satuan

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Deskripsi** |
| Satuan\_id | integer | 10 | *Primary key* |
| Nama\_satuan | varchar | 25 |  |

1. Table Kategori

Tabel 3. 18 Tabel Kategori

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Deskripsi** |
| Kategori\_id | integer | 10 | *Primary key* |
| Nama\_Kategori | varchar | 25 |  |

1. Table Penjualan

Tabel 3. 19 Tabel Detail Penjualan

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Deskripsi** |
| Detail\_id | integer | 10 | *Primary key* |
| Nota | varchar | 25 |  |
| Customer\_id | integer | 10 | *Foreign key* |
| Total\_harga | integer | 10 |  |
| diskon | integer | 10 |  |
| Harga\_akhir | integer | 10 |  |
| date | date | 10 |  |
| Cash | integer | 10 |  |
| Change | integer | 10 |  |
| User\_id | integer | 10 | *Foreign key* |
| Note | text | 10 |  |

1. Table *Customer*

Tabel 3. 20 Tabel *Customer*

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Deskripsi** |
| Customer\_id | integer | 10 | *Primary key* |
| Nama | varchar | 25 |  |
| Jenis\_kelamin | varchar | 10 | *Nullable* |
| No\_telepon | varchar | 15 | *Nullable* |
| Alamat | varchar | 30 | *Nullable* |

1. Table Transaksi

Tabel 3. 21 Tabel Transaksi

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Deskripsi** |
| Transaksi\_id | integer | 10 | *Primary key* |
| Item\_id | varchar | 25 | *Forign key* |
| Harga | varchar | 100 |  |
| Jumlah | integer | 10 |  |
| Total | integer | 10 |  |
| User\_id | integer | 10 | *Forign key* |

## *Perancangan Struktur Menu*

Perancangan struktur menu adalah gambaran jalur pemakaian yang akan dibangun. Berikut merupakan struktur menu aplikasi penjualan bahan barang bangunan di Toko Satria Jaya.

|  |
| --- |
|  |
| Gambar 3. 29 Struktur Menu Toko Satria Jaya |

Keterangan

Berdasarkan Gambar 3.29 Terdapat 7 Menu yakni menu Pelanggan, Penjual, *Stock* Barang yang memiliki *sub* menu Barang Masuk dan Barang Keluar, Pemasok, Produk yang memiliki *sub* menu Kategori, Satuan, dan Item, User, Laporan yang memiliki *sub* menu Laporan Penjualan dan Laporan *Stock* Barang.

## *Perancangan Antarmuka*

1. Antarmuka Halaman *Login*

|  |
| --- |
|  |
| Gambar 3. 30 Antarmuka Halaman *Login* |

Keterangan

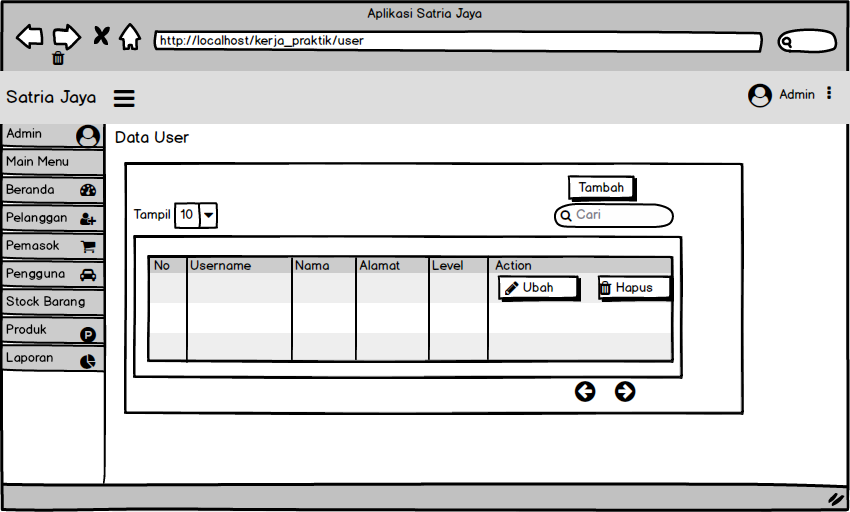
Antarmuka Halaman *Login* menggambarkan suatu aktifitas yang dapat dilakukan oleh user yaitu kasir dan admin mengisi *form* yang telah ada, untuk masuk ke dalam sistem sesuai hak aksesnya masing-masing, jika data yang dimasukan sesuai maka akan masuk ke dalam sistem, jika tidak akan menampilkan pesan kesalahan.

1. Antarmuka Halaman Admin

|  |
| --- |
|  |
| Gambar 3. 31 Antarmuka Halaman Admin |

Keterangan

Antarmuka Halaman Admin menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut, terdapat menu yang tersedia yakni beranda, pengguna, pelanggan, *supplier*, *stock* barang, produk, dan laporan

1. Antarmuka Halaman Admin Menu Pengguna

|  |
| --- |
|  |
| Gambar 3. 32 Antarmuka Halaman Pengguna |

Keterangan

Antarmuka Halaman Admin Menu Pengguna menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni tambah, ubah, hapus, dan cari data pengguna,.

1. Antarmuka Halaman Admin Menu Tambah Pengguna

|  |
| --- |
|  |
| Gambar 3. 33 Antarmuka Halaman Tambah Pengguna |

Keterangan

Antarmuka Halaman Admin Menu Tambah Pengguna menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menambah data pengguna ,. Klik tombol tambah dan isi sesuai *form* yang telah di sediakan lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Admin Menu Ubah Pengguna

|  |
| --- |
|  |
| Gambar 3. 34 Antarmuka Halaman Ubah Pengguna |

Keterangan

Antarmuka Halaman Admin Menu Ubah Pengguna menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni mengubah data pengguna,. Klik tombol ubah dan isi sesuai *form* yang akan diubah lalu tekan tombol simpan maka data akan tersimpan

1. Antarmuka Halaman Admin Menu Hapus Pengguna

|  |
| --- |
|  |
| Gambar 3. 35 Antarmuka Halaman Admin Menu Hapus Pengguna |

Keterangan

Antarmuka Halaman Admin Menu Hapus Pengguna menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menghapus data pengguna,. Pilih data yang akan di hapus, dan klik tombol hapus, maka akan menampilkan pesan konfirmasi “apakah yakin menghpus data” ?

Jika menekan tombol ya maka, data akan terhapus, jika menekan tombol tidak maka data tidak akan terhapus dan kembali ke halaman awal.

1. Antarmuka Halaman Admin Laporan Penjualan

|  |
| --- |
|  |
| Gambar 3. 36 Antarmuka Halaman Admin Laporan Penjualan |

Keterangan

Antarmuka Halaman Admin Menu Laporan Penjualan menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menghapus dan mencetak data laporan penjualan, pilih tombol *print* untuk mencetak laporan dan pilih data yang akan di hapus, dan klik tombol hapus, maka akan menampilkan pesan konfirmasi “apakah yakin menghpus data” ?

Jika menekan tombol ya maka, data akan terhapus, jika menekan tombol tidak maka data tidak akan terhapus dan kembali ke halaman awal

1. Antarmuka Halaman Admin Laporan *Stock* Barang

|  |
| --- |
|  |
| Gambar 3. 37 Antarmuka Halaman Admin Laporan *Stock* Barang |

Keterangan

Antarmuka Halaman Admin Menu Lapora *Stock* Barang menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni mencetak data laporan *stock* barang, pilih tombol *print* untuk mencetak laporan *stock* barang.

1. Antarmuka Halaman Kasir

|  |
| --- |
|  |
| Gambar 3. 38 Antarmuka Halaman Kasir |

Keterangan

Antarmuka Halaman Kasir menggambarkan suatu aktifitas yang dapat dilakukan oleh kasir pada halaman tersebut, terdapat menu yang tersedia yakni beranda, pelanggan, dan penjualan dimanan setiap menu memiliki hak akses penuh baik tambah, ubah, hapus, dan cari.

1. Antarmuka Halaman Kasir Menu Pelanggan

|  |
| --- |
|  |
| Gambar 3. 39 Antarmuka Halaman Kasir Menu Pelanggan |

Keterangan

Antarmuka Halaman Kasir Menu Pelanggan menggambarkan suatu aktifitas yang dapat dilakukan oleh kasir pada halaman tersebut yakni tambah, ubah, hapus, dan cari data pelanggan,.

1. Antarmuka Halaman Kasir Menu Tambah Pelanggan

|  |
| --- |
|  |
| Gambar 3. 40 Antarmuka Halaman Kasir Menu Tambah Pelanggan |

Keterangan

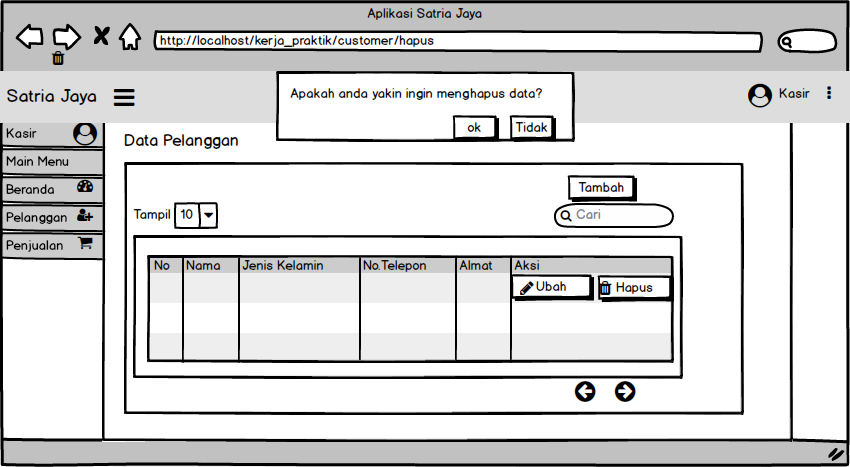
Antarmuka Halaman Kasir Menu Tambah Pelanggan menggambarkan suatu aktifitas yang dapat dilakukan oleh kasir pada halaman tersebut yakni menambah data pelanggan,. Klik tombol tambah dan isi sesuai *form* yang telah di sediakan lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Kasir Menu Ubah Pelanggan

|  |
| --- |
|  |
| Gambar 3. 41 Antarmuka Halaman Kasir Menu Ubah Pelanggan |

Keterangan

Antarmuka Halaman Admin Menu Ubah Pelanggan menggambarkan suatu aktifitas yang dapat dilakukan oleh kasir pada halaman tersebut yakni mengubah data pelanggan,. Klik tombol ubah dan isi sesuai *form* yang akan diubah lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Kasir Menu Hapus Pelanggan

|  |
| --- |
|  |
| Gambar 3. 42 Antarmuka Halaman Kasir Menu Hapus Pelanggan |

Keterangan

Antarmuka Halaman Kasir Menu Hapus Pelanggan menggambarkan suatu aktifitas yang dapat dilakukan oleh kasir pada halaman tersebut yakni menghapus data pelanggan,. Pilih data yang akan di hapus, dan klik tombol hapus, maka akan menampilkan pesan konfirmasi “apakah yakin menghpus data” ?

Jika menekan tombol ya maka, data akan terhapus, jika menekan tombol tidak maka data tidak akan terhapus dan kembali ke halaman awal.

1. Antarmuka Halaman Kasir Menu Penjualan

|  |
| --- |
|  |
| Gambar 3. 43 Antarmuka Halaman Kasir Menu Penjualan |

Keterangan

Antarmuka Halaman Kasir Menu Penjualan menggambarkan suatu aktifitas yang dapat dilakukan oleh kasir pada halaman tersebut yakni tambah, ubah, hapus, cari, dan cetak data barang,.

1. Antarmuka Halaman Kasir Menu Tambah Penjualan

|  |
| --- |
|  |
| Gambar 3. 44 Antarmuka Halaman Kasir Menu Tambah Penjualan |

Keterangan

Antarmuka Halaman Kasir Menu Tambah Penjualan menggambarkan suatu aktifitas yang dapat dilakukan oleh kasir pada halaman tersebut yakni menambah data barang ke daftar *list* penjualan,. Klik tombol tambah data akan masuk dalam tabel penjualan untuk di proses.

1. Antarmuka Halaman Kasir Menu Ubah Penjualan

|  |
| --- |
|  |
| Gambar 3. 45 Antarmuka Halaman Kasir Menu Ubah Penjualan |

Keterangan

Antarmuka Halama Kasir Menu Ubah Penjualan menggambarkan suatu aktifitas yang dapat dilakukan oleh kasir pada halaman tersebut yakni mengubah data penjualan,. Klik tombol ubah dan isi sesuai *form* yang akan diubah lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Kasir Menu Hapus Penjualan

|  |
| --- |
|  |
| Gambar 3. 46 Antarmuka Halaman Kasir Menu Hapus Penjualan |

Keterangan

Antarmuka Halaman Kasir Menu Hapus Penjualan menggambarkan suatu aktifitas yang dapat dilakukan oleh kasir pada halaman tersebut yakni menghapus data penjualan,. Pilih data yang akan di hapus, dan klik tombol hapus, maka akan menampilkan pesan konfirmasi “apakah yakin menghpus data” ?

Jika menekan tombol ya maka, data akan terhapus, jika menekan tombol tidak maka data tidak akan terhapus dan kembali ke halaman awal.

1. Antarmuka Halaman Kasir Menu Proses Penjualan

|  |
| --- |
|  |
| Gambar 3. 47 Antarmuka Halaman Kasir Menu Proses Penjualan |

Keterangan

Antarmuka Halaman Kasir Menu Proses Penjualan menggambarkan suatu aktifitas yang dapat dilakukan oleh kasir pada halaman tersebut yakni memproses data penjualan,. Kemudian data akan diproses dan akan ditampilkan halaman cetak penjualan.

1. Antarmuka Halaman Admin Menu Barang Masuk

|  |
| --- |
|  |
| Gambar 3. 48 Antarmuka Halaman Admin Menu Barang Masuk |

Keterangan

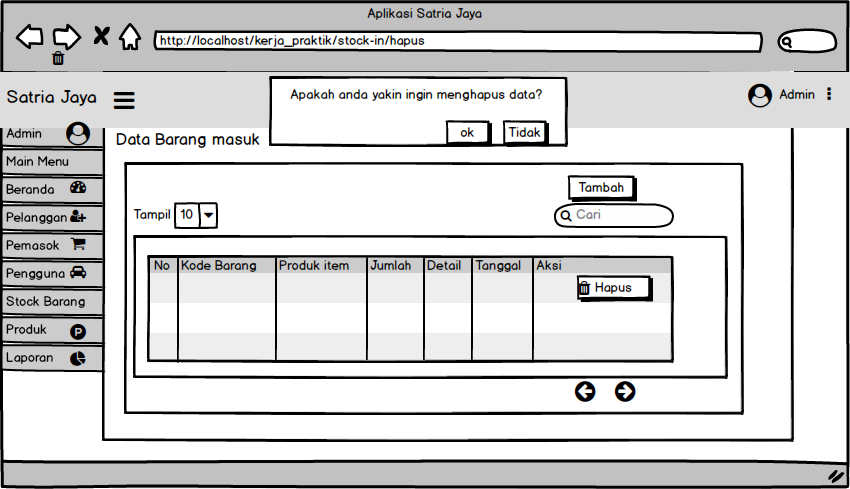
Antarmuka Halaman Admin Menu Barang Masuk menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni tambah, hapus, dan cari data barang masuk.

1. Antarmuka Halaman Admin Menu Tambah Barang Masuk

|  |
| --- |
|  |
| Gambar 3. 49 Antarmuka Halaman Admin Menu Tambah Barang Masuk |

Keterangan

Gambar 3.42 Antarmuka Halaman Admin Menu Tambah Barang Masuk menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menambah data barang masuk,. Klik tombol tambah dan isi sesuai *form* yang telah di sediakan lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Admin Menu Hapus Barang Masuk

|  |
| --- |
|  |
| Gambar 3. 50 Antarmuka Halaman Admin Menu Hapus Barang Masuk |

Keterangan

Antarmuka Halaman Admin Menu Hapus Baraang Masuk menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menghapus data barang masuk,. Pilih data yang akan di hapus, dan klik tombol hapus, maka akan menampilkan pesan konfirmasi “apakah yakin menghpus data” ?

Jika menekan tombol ya maka, data akan terhapus, jika menekan tombol tidak maka data tidak akan terhapus dan kembali ke halaman awal.

1. Antarmuka Halaman Admin Menu Barang Keluar

|  |
| --- |
|  |
| Gambar 3. 51 Antarmuka Halaman Kasir Menu Barang Keluar |

Keterangan

Antarmuka Halaman Admin Menu Barang Keluar menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni tambah, hapus, dan cari data barang keluar.

1. Antarmuka Halaman Admin Menu Tambah Barang Keluar

|  |
| --- |
|  |
| Gambar 3. 52 Antarmuka Halaman Kasir Menu Tambah Barang Keluar |

Keterangan

Antarmuka Halaman Admin Menu Tambah Barang Keluar menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menambah data barang keluar,. Klik tombol tambah dan isi sesuai *form* yang telah di sediakan lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Admin Menu Hapus Barang Keluar

|  |
| --- |
|  |
| Gambar 3. 53 Antarmuka Halaman Admin Menu Hapus Barang Keluar |

Keterangan

Antarmuka Halaman Admin Menu Hapus Barang Keluar menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menghapus data barang keluar,. Pilih data yang akan di hapus, dan klik tombol hapus, maka akan menampilkan pesan konfirmasi “apakah yakin menghpus data” ?

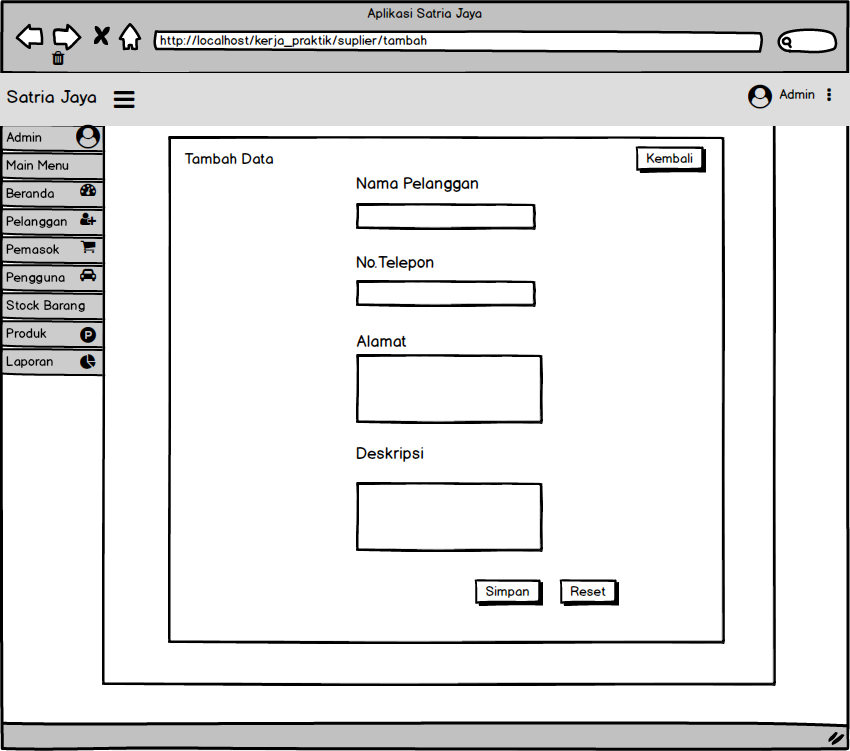
Jika menekan tombol ya maka, data akan terhapus, jika menekan tombol tidak maka data tidak akan terhapus dan kembali ke halaman awal.

1. Antarmuka Halaman Admin Menu *Supplier*

|  |
| --- |
|  |
| Gambar 3. 54 Antarmuka Halaman Admin Menu *Supplier* |

Keterangan

Antarmuka Halaman Admin Menu Supplier menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni tambah, ubah, hapus, cari, dan cetak data *supplier*,.

1. Antarmuka Halaman Admin Menu Tambah *Supplier*

|  |
| --- |
|  |
| Gambar 3. 55 Antarmuka Halaman Admin Menu Tambah *Supplier* |

Keterangan

Antarmuka Halaman Admin Menu Tambah Supplier menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menambah data *supplier*,. Klik tombol tambah dan isi sesuai f*orm* yang telah di sediakan lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Admin Menu Ubah *Supplier*

|  |
| --- |
|  |
| Gambar 3. 56 Antarmuka Halaman Admin Menu Ubah *Supplier* |

Keterangan

Antarmuka Halaman Admin Menu Ubah Pemasok menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni mengubah data *supplier* ,. Klik tombol ubah dan isi sesuai *form* yang akan diubah lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Admin Menu Hapus *Supplier*

|  |
| --- |
|  |
| Gambar 3. 57 Antarmuka Halaman Admin Menu Hapus Supplier |

Keterangan

Antarmuka Halaman Admin Menu Hapus Supplier menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menghapus data *supplier*,. Pilih data yang akan di hapus, dan klik tombol hapus, maka akan menampilkan pesan konfirmasi “apakah yakin menghpus data” ?

Jika menekan tombol ya maka, data akan terhapus, jika menekan tombol tidak maka data tidak akan terhapus dan kembali ke halaman awal.

1. Antarmuka Halaman Admin Menu Kategori Produk

|  |
| --- |
|  |
| Gambar 3. 58 Antarmuka Halaman Admin Menu Kategori Produk |

Keterangan

Antarmuka Halaman Admin Menu Kategori menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni tambah, ubah, hapus, cari, dan cetak data kategori produk,.

1. Antarmuka Halaman Admin Menu Tambah Kategori Produk

|  |
| --- |
|  |
| Gambar 3. 59 Antarmuka Halaman Admin Menu Tambah Kategori Produk |

Keterangan

Antarmuka Halaman Admin Menu Tambah Kategori Produk menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menambah data kategori produk,. Klik tombol tambah dan isi sesuai *form* yang telah di sediakan lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Admin Menu Ubah Kategori Produk

|  |
| --- |
|  |
| Gambar 3. 60 Antarmuka Halaman Admin Menu Ubah Kategori Produk |

Keterangan

Antarmuka Halaman Admin Menu Ubah Kategori Produk menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni mengubah data kategori produk ,. Klik tombol ubah dan isi sesuai *form* yang akan diubah lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Admin Menu Hapus Kategori Produk

|  |
| --- |
|  |
| Gambar 3. 61 Antarmuka Halaman Admin Menu Hapus Kategori Produk |

Keterangan

Antarmuka Halaman Admin Menu Hapus Kategori Produk menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menghapus data kategori produk,. Pilih data yang akan di hapus, dan klik tombol hapus, maka akan menampilkan pesan konfirmasi “apakah yakin menghpus data” ?

Jika menekan tombol ya maka, data akan terhapus, jika menekan tombol tidak maka data tidak akan terhapus dan kembali ke halaman awal.

1. Antarmuka Halaman Admin Menu Satuan Produk

|  |
| --- |
|  |
| Gambar 3. 62 Antarmuka Halaman Admin Menu Satuan Produk |

Keterangan

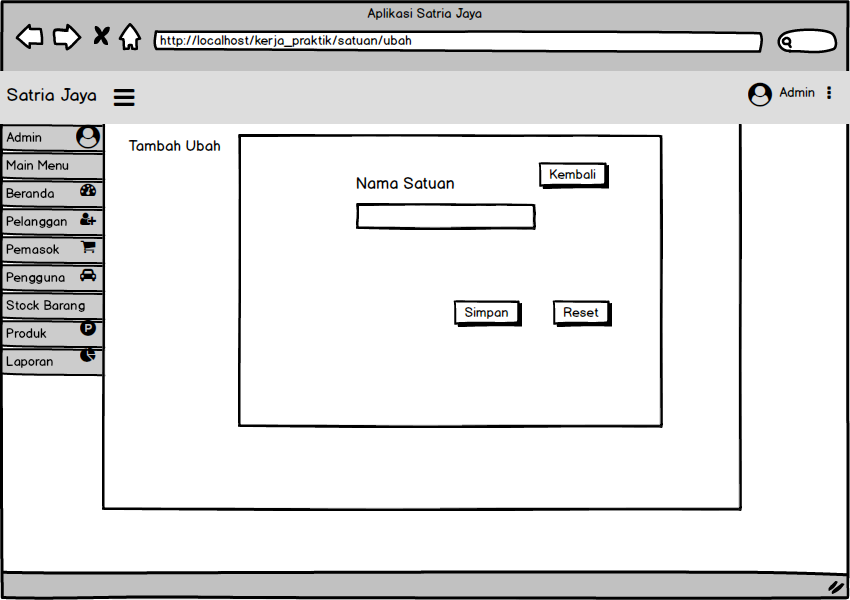
Antarmuka Halaman Admin Menu Kategori menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni tambah, ubah, hapus, cari, dan cetak data satuan produk,.

1. Antarmuka Halaman Admin Menu Tambah Satuan Produk

|  |
| --- |
|  |
| Gambar 3. 63 Antarmuka Halaman Admin Menu Tambah Satuan Produk |

Keterangan

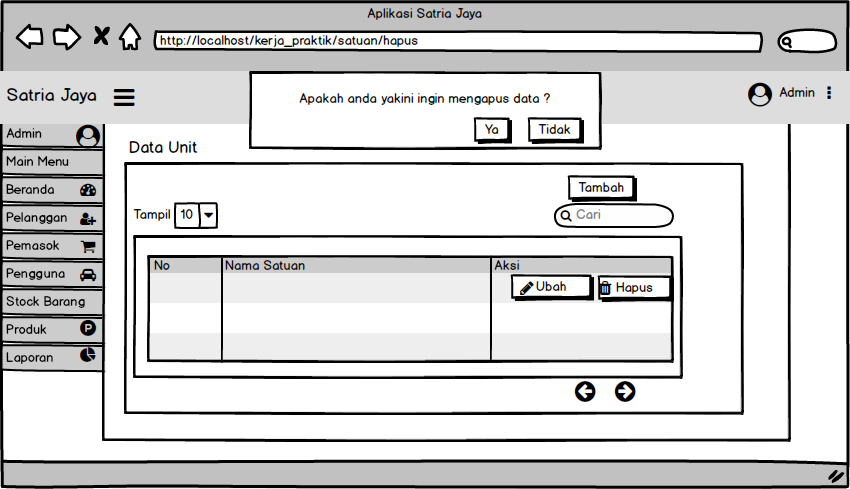
Antarmuka Halaman Admin Menu Tambah Satuan Produk menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menambah data satuan produk,. Klik tombol tambah dan isi sesuai *form* yang telah di sediakan lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Admin Menu Ubah Satuan Produk

|  |
| --- |
|  |
| Gambar 3. 64 Antarmuka Halaman Admin Menu Ubah Satuan Produk |

Keterangan

Antarmuka Halaman Admin Menu Ubah Satuan Produk menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni mengubah data satuan produk ,. Klik tombol ubah dan isi sesuai *form* yang akan diubah lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Admin Menu Hapus Satuan Produk

|  |
| --- |
|  |
| Gambar 3. 65 Antarmuka Halaman Admin Menu Hapus Satuan Produk |

Keterangan

Antarmuka Halaman Admin Menu Hapus Satuan Produk menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menghapus data satuan produk,. Pilih data yang akan di hapus, dan klik tombol hapus, maka akan menampilkan pesan konfirmasi “apakah yakin menghpus data” ?

Jika menekan tombol ya maka, data akan terhapus, jika menekan tombol tidak maka data tidak akan terhapus dan kembali ke halaman awal.

1. Antarmuka Halaman Admin Menu Item Produk

|  |
| --- |
|  |
| Gambar 3. 66 Antarmuka Halaman Admin Menu Item Produk |

Keterangan

Antarmuka Halaman Admin Menu Item menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni tambah, ubah, hapus, cari, dan cetak data item produk,.

1. Antarmuka Halaman Admin Menu Tambah Item Produk

|  |
| --- |
|  |
| Gambar 3. 67 Antarmuka Halaman Admin Menu Tambah Item Produk |

Keterangan

Antarmuka Halaman Admin Menu Tambah Item Produk menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menambah data item produk,. Klik tombol tambah dan isi sesuai *form* yang telah di sediakan lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Admin Menu Ubah Item Produk

|  |
| --- |
|  |
| Gambar 3. 68 Antarmuka Halaman Admin Menu Ubah Item Produk |

Keterangan

Antarmuka Halaman Admin Menu Ubah Item Produk menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni mengubah data item produk ,. Klik tombol ubah dan isi sesuai *form* yang akan diubah lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Admin Menu Hapus Item Produk

|  |
| --- |
|  |
| Gambar 3. 69 Antarmuka Halaman Admin Menu Hapus Item Produk |

Keterangan

Antarmuka Halaman Admin Menu Hapus Item Produk menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menghapus data item produk,. Pilih data yang akan di hapus, dan klik tombol hapus, maka akan menampilkan pesan konfirmasi “apakah yakin menghpus data”?

Jika menekan tombol ya maka, data akan terhapus, jika menekan tombol tidak maka data tidak akan terhapus dan kembali ke halaman awal.

## *Deployment Diagram*

*Deployment Diagram* merupakan bagaimana aplikasi dalam sistem atau pun luar sistem berfungsi dan subsistem dapat dialokasikan kedalam lingkungan terkomputerisasi dan akan mendukung sistem. Berikut merupakan *deployment diagram* dari Aplikasi Penjualan Bahan Barang Bangunan (*point of sale)* Toko Satria Jaya

|  |
| --- |
|  |
| Gambar 3. 70 *Deployement Diagram* |

Keterangan

Diagram tersebut menunjukan *device* dalam sistem maupun luar sistem yang digunakan, yaitu dengan menggunakan Sistem *Operasi Windows 10*, aplikasi *Visual Studio Code, Web Browser Google Chrome* 99.0.4430.093, serta *Web Server* yang digunakan dalam membuat Aplikasi Penjualan Bahan Barang Bangunan (*point of sale)* Toko Satria Jaya yaitu Apache 3.4.43. *Artifact* di dalam sistem tersebut yaitu *Login.php* yang merepresentasikan ke fungsi halaman- login, *User*.php yang merepresentasikan ke fungsi v-*user*, *Supplier*.php yang merepresntasikan ke fungsi v-*supplier*, *Customer*.php yang merepresentasikan ke fungsi v-*customer*, Penjualan.php yang merepresentasikan ke fungsi v-penjualan, Produk.php yang merepresentasikan ke fungsi v-produk, *Stock*\_Barang.php yang merepresentasikan ke fungsi v-*stock\_*barang dan Laporan.php yang merepresentasikan ke fungsi v-laporan. Semua *artifact* membutuhkan suatu *library* yaitu *config.php* dan deployment *spec index.php* yang dihubungkan ke *database system MySQL* 7.4.8 dengan *server MariaDB*-11.3.14 melalui *protocol TCP/IP*.

# BAB IV

# IMPLEMENTASI PERANCANGAN

# DAN PENGUJIAN SISTEM

# Implementasi

Implementasi merupakan penerapan bagaiman suatu sistem bekerja berdasarkan hasil analisa dan juga perancangan yang telah dibuat sebelumnya ke dalam suatu bahasa pemrograman tertentu.

Tahapan implementasi adalah tahap penciptaan perangkat lunak, tahap kelanjutan dari kegiatan perancangan sistem. Tahap ini merupakan tahap dimana sistem siap untuk dioperasikan, yang terdiri dari penjelasan mengenai lingkungan implementasi sistem, dan implementasi program.

# Kebutuhan Perangkat Keras

Spesifikasi perangkat keras atau *hardware* yang dibutuhkan untuk membangun Aplikasi Penjualan Barang Bahan Bangunan di Toko Satria Jaya adalah sebagai berikut:

1. Monitor LCD 1366x768
2. *Processor* Intel(R) *Core™i3-27350 cpu @2.4GHz (4 CPUs)*
3. *VGA Intel(R) HD Graphics 3000*
4. *Memory DDR3 3.00 GB RAM*
5. *Mouse and Keyboard*
6. Koneksi Internet

Sementara itu spesifikasi minimal dari perangkat keras yang digunakan untuk menjalankan aplikasi penjualan barang bahan bangunan di Toko Satria Jaya adalah sebagai berikut:

1. *Processor :Intel Pentium* [*4 @ 1.5*](mailto:4@1.5) *GHz*
2. *Hardisk 500 GB*
3. *RAM 512*
4. *Monitor :Resolusi 1024x76*
5. *Mouse and Keyboard*

# Kebutuhan Perangkat Lunak

Perangkat lunak yang digunakan untuk membangun aplikasi penjualan barang bahan bangunan adalah sebagai berikut:

1. *Microsoft Windwos 10*, perangkat lunak sistem yang mengatur sumber daya dari perangkat keras (*Hardware*) dan perangkat lunak (*Software*),. Tanpa sistem operasi, pengguna tidak dapat menjalankan program aplikasi pada komputer mereka, kecuali program booting.
2. *Xampp*, merupakan suatu aplikasi yang dipergunakan untuk *database server*. Berikut adalah paket aplikasi yang terdapat pada aplikasi xampp yaitu sebagai berikut:
3. Apache versi 5.4.4
4. MySQL versi 7.5.32
5. *Visual paradigm for UML 1.,0,* merupakan suatu aplikasi yang dipergunakan untuk membuat diagram yang digunakan dalam proses perancangan.
6. *Visual studio code,* merupakan sebuah aplikasi yang digunakan sebagai *code editor* dalam membuat sebuah program.

Kebutuhan perangkat lunak minimum yang digunakan oleh *user* untuk menjalankan Aplikasi Penjualan Barang Bahan Bangunan Di Toko Satria Jaya sebagai berikut:

1. Sistem Operasi : *Windows* 10
2. *Packet Driver* sesuai dengan spesifikasi komputer.

# Kebutuhan Pengguna

Pengguna dalam aplikasi penjualan barang bahan bangunan Toko Satria Jaya merupakan aktor yang mengoperasikan atau menggunakan aplikasi. Maka dari itu seorang pengguna aplikasi harus memahami cara mengoperasikan komputer dan sistemnya. Berikut adalah syarat untuk seorang pengguna aplikasi:

1. Paham dan dapat mengoperasikan komputer dengan baik dan benar
2. Teliti dan jujur
3. Bisa menggunakan aplikasi penjualan bahan bangunan *Point Of Sale*

# Tampilan antarmuka

Antarmuka merupakan tampilan halaman aplikasi yang sudah dibuat. Berikut ini akan diuraikan mengenai antarmuka Aplikasi Penjualan Bahan Barang Bangunan Di Toko Satria Jaya.

1. Antarmuka Halaman *Login*

|  |
| --- |
|  |
| Gambar 4. 1 Antarmuka Halam *Login* |

Keterangan

Antarmuka Halaman Login menggambarkan suatu aktifitas yang dapat dilakukan oleh user yaitu kasir dan admin mengisi *form* yang telah ada, untuk masuk ke dalam sistem sesuai hak aksesnya masing-masing, jika data yang dimasukan sesuai maka akan masuk ke dalam sistem, jika tidak akan menampilkan pesan kesalahan.

1. Antarmuka Halaman Admin

|  |
| --- |
|  |
| Gambar 4. 2 Antarmuka Halaman Admin |

Keterangan

Antarmuka Halaman Admin menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut, terdapat menu yang tersedia yakni beranda, pengguna, pelanggan, supplier, *stock* barang, produk, dan laporan

1. Antarmuka Halaman Admin Menu Pengguna

|  |
| --- |
|  |
| Gambar 4. 3 Antarmuka Halaman Pengguna |

Keterangan

Antarmuka Halaman Admin Menu Pengguna menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni tambah, ubah, hapus, dan cari data pengguna,.

1. Antarmuka Halaman Admin Menu Tambah Pengguna

|  |
| --- |
|  |
| Gambar 4. 4 Antarmuka Halaman Tambah Pengguna |

Keterangan

Antarmuka Halaman Admin Menu Tambah Pengguna menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menambah data pengguna ,. Klik tombol tambah dan isi sesuai *form* yang telah di sediakan lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Admin Menu Ubah Pengguna

|  |
| --- |
|  |
| Gambar 4. 5 Antarmuka Halaman Ubah Pengguna |

Keterangan

Antarmuka Halaman Admin Menu Ubah Pengguna menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni mengubah data pengguna,. Klik tombol ubah dan isi sesuai *form* yang akan diubah lalu tekan tombol simpan maka data akan tersimpan

1. Antarmuka Halaman Admin Menu Hapus Pengguna

|  |
| --- |
|  |
| Gambar 4. 6 Antarmuka Halaman Admin Menu Hapus Pengguna |

Keterangan

Antarmuka Halaman Admin Menu Hapus Pengguna menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menghapus data pengguna,. Pilih data yang akan di hapus, dan klik tombol hapus, maka akan menampilkan pesan konfirmasi “apakah yakin menghpus data” ?

Jika menekan tombol ya maka, data akan terhapus, jika menekan tombol tidak maka data tidak akan terhapus dan kembali ke halaman awal

1. Antarmuka Halaman Admin Laporan Penjualan

|  |
| --- |
|  |
| Gambar 4. 7 Antarmuka Halaman Admin Laporan Penjualan |

Keterangan

Antarmuka Halaman Admin Menu Laporan Penjualan menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menghapus dan mencetak data laporan penjualan, pilih tombol *print* untuk mencetak laporan dan pilih data yang akan di hapus, dan klik tombol hapus, maka akan menampilkan pesan konfirmasi “apakah yakin menghpus data” ?

Jika menekan tombol ya maka, data akan terhapus, jika menekan tombol tidak maka data tidak akan terhapus dan kembali ke halaman awal

1. Antarmuka Halaman Admin Lapora *Stock* Barang

|  |
| --- |
|  |
| Gambar 4. 8 Antarmuka Halaman Admin Laporan Stock Barang |

Keterangan

Antarmuka Halaman Admin Menu Lapora *Stock* Barang menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni mencetak data laporan *stock* barang, pilih tombol *print* untuk mencetak laporan *stock* barang.

1. Antarmuka Halaman Kasir

|  |
| --- |
|  |
| Gambar 4. 9 Antarmuka Halaman Kasir |

Keterangan

Antarmuka Halaman Kasir menggambarkan suatu aktifitas yang dapat dilakukan oleh kasir pada halaman tersebut, terdapat menu yang tersedia yakni beranda, pelanggan, dan penjualan dimanan setiap menu memiliki hak akses penuh baik tambah, ubah, hapus, dan cari.

1. Antarmuka Halaman Kasir Menu Pelanggan

|  |
| --- |
|  |
| Gambar 4. 10 Antarmuka Halaman Kasir Menu Pelanggan |

Keterangan

Antarmuka Halaman Kasir Menu Pelanggan menggambarkan suatu aktifitas yang dapat dilakukan oleh kasir pada halaman tersebut yakni tambah, ubah, hapus, dan cari data pelanggan,.

1. Antarmuka Halaman Kasir Menu Tambah Pelanggan

|  |
| --- |
|  |
| Gambar 4. 11 Antarmuka Halaman Kasir Menu Tambah Pelanggan |

Keterangan

Antarmuka Halaman Kasir Menu Tambah Pelanggan menggambarkan suatu aktifitas yang dapat dilakukan oleh kasir pada halaman tersebut yakni menambah data pelanggan,. Klik tombol tambah dan isi sesuai *form* yang telah di sediakan lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Kasir Menu Ubah Pelanggan

|  |
| --- |
|  |
| Gambar 4. 12 Antarmuka Halaman Kasir Menu Ubah Pelanggan |

Keterangan

Antarmuka Halaman Admin Menu Ubah Pelanggan menggambarkan suatu aktifitas yang dapat dilakukan oleh kasir pada halaman tersebut yakni mengubah data pelanggan,. Klik tombol ubah dan isi sesuai *form* yang akan diubah lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Kasir Menu Hapus Pelanggan

|  |
| --- |
|  |
| Gambar 4. 13 Antarmuka Halaman Kasir Menu Hapus Pelanggan |

Keterangan

Antarmuka Halaman Kasir Menu Hapus Pelanggan menggambarkan suatu aktifitas yang dapat dilakukan oleh kasir pada halaman tersebut yakni menghapus data pelanggan,. Pilih data yang akan di hapus, dan klik tombol hapus, maka akan menampilkan pesan konfirmasi “apakah yakin menghpus data” ?

Jika menekan tombol ya maka, data akan terhapus, jika menekan tombol tidak maka data tidak akan terhapus dan kembali ke halaman awal.

1. Antarmuka Halaman Kasir Menu Penjualan

|  |
| --- |
|  |
| Gambar 4. 14 Antarmuka Halaman Kasir Menu Penjualan |

Keterangan

Antarmuka Halaman Kasir Menu Penjualan menggambarkan suatu aktifitas yang dapat dilakukan oleh kasir pada halaman tersebut yakni tambah, ubah, hapus, cari, dan cetak data barang,.

1. Antarmuka Halaman Kasir Menu Tambah Penjualan

|  |
| --- |
|  |
| Gambar 4. 15 Antarmuka Halaman Kasir Menu Tambah Penjualan |

Keterangan

Antarmuka Halaman Kasir Menu Tambah Penjualan menggambarkan suatu aktifitas yang dapat dilakukan oleh kasir pada halaman tersebut yakni menambah data barang ke daftar *list* penjualan,. Klik tombol tambah data akan masuk dalam tabel penjualan untuk di proses.

1. Antarmuka Halaman Kasir Menu Ubah Penjualan

|  |
| --- |
|  |
| Gambar 4. 16 Antarmuka Halaman Kasir Menu Ubah Penjualan |

Keterangan

Antarmuka Halama Kasir Menu Ubah Penjualan menggambarkan suatu aktifitas yang dapat dilakukan oleh kasir pada halaman tersebut yakni mengubah data penjualan,. Klik tombol ubah dan isi sesuai *form* yang akan diubah lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Kasir Menu Hapus Penjualan

|  |
| --- |
|  |
| Gambar 4. 17 Antarmuka Halaman Kasir Menu Hapus Penjualan |

Keterangan

Antarmuka Halaman Kasir Menu Hapus Penjualan menggambarkan suatu aktifitas yang dapat dilakukan oleh kasir pada halaman tersebut yakni menghapus data penjualan,. Pilih data yang akan di hapus, dan klik tombol hapus, maka akan menampilkan pesan konfirmasi “apakah yakin menghpus data” ?

Jika menekan tombol ya maka, data akan terhapus, jika menekan tombol tidak maka data tidak akan terhapus dan kembali ke halaman awal.

1. Antarmuka Halaman Kasir Menu Proses Penjualan

|  |
| --- |
|  |
| Gambar 4. 18 Antarmuka Halaman Kasir Menu Proses Penjualan |

Keterangan

Antarmuka Halaman Kasir Menu Proses Penjualan menggambarkan suatu aktifitas yang dapat dilakukan oleh kasir pada halaman tersebut yakni memproses data penjualan,. Kemudian data akan diproses dan akan ditampilkan halaman cetak penjualan.

1. Antarmuka Halaman Admin Menu Barang Masuk

|  |
| --- |
|  |
| Gambar 4. 19 Antarmuka Halaman Admin Menu Barang Masuk |

Keterangan

Antarmuka Halaman Admin Menu Barang Masuk menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni tambah, hapus, dan cari data barang masuk.

1. Antarmuka Halaman Admin Menu Tambah Barang Masuk

|  |
| --- |
|  |
| Gambar 4. 20 Antarmuka Halaman Admin Menu Tambah Barang Masuk |

Keterangan

Gambar 4.20 Antarmuka Halaman Kasir Menu Tambah Barang Masuk menggambarkan suatu aktifitas yang dapat dilakukan oleh kasir pada halaman tersebut yakni menambah data barang masuk,. Klik tombol tambah dan isi sesuai *form* yang telah di sediakan lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Admin Menu Hapus Barang Masuk

|  |
| --- |
|  |
| Gambar 4. 21 Antarmuka Halaman Admin Menu Hapus Barang Masuk |

Keterangan

Antarmuka Halaman Admin Menu Hapus Baraang Masuk menggambarkan suatu aktifitas yang dapat dilakukan oleh kasir pada halaman tersebut yakni menghapus data barang masuk,. Pilih data yang akan di hapus, dan klik tombol hapus, maka akan menampilkan pesan konfirmasi “apakah yakin menghpus data” ?

Jika menekan tombol ya maka, data akan terhapus, jika menekan tombol tidak maka data tidak akan terhapus dan kembali ke halaman awal.

1. Antarmuka Halaman Admin Menu Barang Keluar

|  |
| --- |
|  |
| Gambar 4. 22 Antarmuka Halaman Kasir Menu Barang Keluar |

Keterangan

Antarmuka Halaman Admin Menu Barang Keluar menggambarkan suatu aktifitas yang dapat dilakukan oleh kasir pada halaman tersebut yakni tambah, hapus, dan cari data barang keluar.

1. Antarmuka Halaman Admin Menu Tambah Barang Keluar

|  |
| --- |
|  |
| Gambar 4. 23 Antarmuka Halaman Admin Menu Tambah Barang Keluar |

Keterangan

Antarmuka Halaman Admin Menu Tambah Barang Keluar menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menambah data barang keluar,. Klik tombol tambah dan isi sesuai *form* yang telah di sediakan lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Admin Menu Hapus Barang Keluar

|  |
| --- |
|  |
| Gambar 4. 24 Antarmuka Halaman Admin Menu Hapus Barang Keluar |

Keterangan

Antarmuka Halaman Admin Menu Hapus Barang Keluar menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menghapus data barang keluar,. Pilih data yang akan di hapus, dan klik tombol hapus, maka akan menampilkan pesan konfirmasi “apakah yakin menghpus data” ?

Jika menekan tombol ya maka, data akan terhapus, jika menekan tombol tidak maka data tidak akan terhapus dan kembali ke halaman awal.

1. Antarmuka Halaman Admin Menu *Supplier*

|  |
| --- |
|  |
| Gambar 4. 25 Antarmuka Halaman Admin Menu *Supplier* |

Keterangan

Antarmuka Halaman Admin Menu Supplier menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni tambah, ubah, hapus, cari, dan cetak data *supplier*,.

1. Antarmuka Halaman Admin Menu Tambah *Supplier*

|  |
| --- |
|  |
| Gambar 4. 26 Antarmuka Halaman Admin Menu Tambah *Supplier* |

Keterangan

Antarmuka Halaman Admin Menu Tambah Supplier menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menambah data *supplier*,. Klik tombol tambah dan isi sesuai f*orm* yang telah di sediakan lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Admin Menu Ubah *Supplier*

|  |
| --- |
|  |
| Gambar 4. 27 Antarmuka Halaman Admin Menu Ubah *Supplier* |

Keterangan

Antarmuka Halaman Admin Menu Ubah *Supplier* menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni mengubah data supplier ,. Klik tombol ubah dan isi sesuai *form* yang akan diubah lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Admin Menu Hapus *Supplier*

|  |
| --- |
|  |
| Gambar 4. 28 Antarmuka Halaman Admin Menu Hapus *Supplier* |

Keterangan

Antarmuka Halaman Kasir Menu Hapus *Supplier* menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menghapus data supplier,. Pilih data yang akan di hapus, dan klik tombol hapus, maka akan menampilkan pesan konfirmasi “apakah yakin menghpus data” ?

Jika menekan tombol ya maka, data akan terhapus, jika menekan tombol tidak maka data tidak akan terhapus dan kembali ke halaman awal.

1. Antarmuka Halaman Admin Menu Kategori Produk

|  |
| --- |
|  |
| Gambar 4. 29 Antarmuka Halaman Admin Menu Kategori Produk |

Keterangan

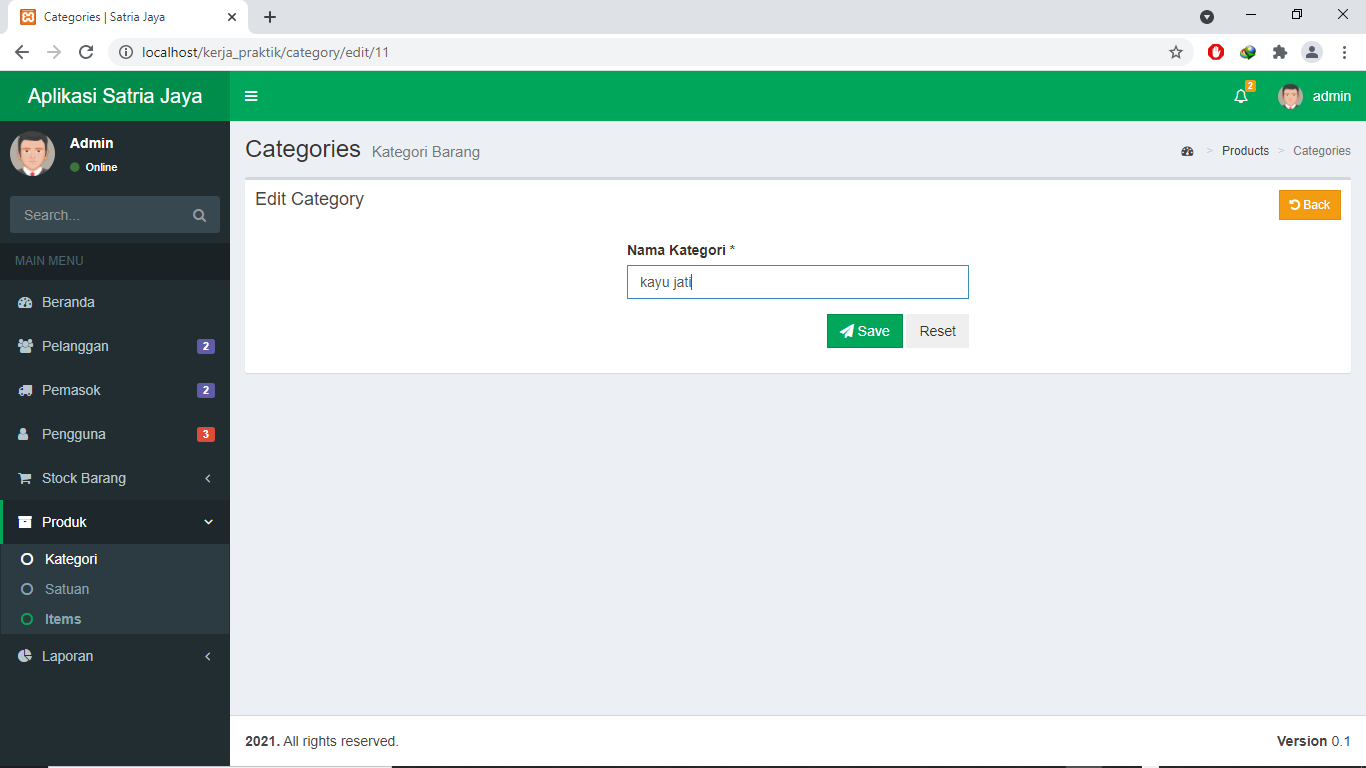
Antarmuka Halaman Admin Menu Kategori menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni tambah, ubah, hapus, cari, dan cetak data kategori produk,.

1. Antarmuka Halaman Admin Menu Tambah Kategori Produk

|  |
| --- |
|  |
| Gambar 4. 30 Antarmuka Halaman Admin Menu Tambah Kategori Produk |

Keterangan

Antarmuka Halaman Admin Menu Tambah Kategori Produk menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menambah data kategori produk,. Klik tombol tambah dan isi sesuai *form* yang telah di sediakan lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Admin Menu Ubah Kategori Produk

|  |
| --- |
|  |
| Gambar 4. 31 Antarmuka Halaman Admin Menu Ubah Kategori Produk |

Keterangan

Antarmuka Halaman Admin Menu Ubah Kategori Produk menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni mengubah data kategori produk ,. Klik tombol ubah dan isi sesuai *form* yang akan diubah lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Admin Menu Hapus Kategori Produk

|  |
| --- |
|  |
| Gambar 4. 32 Antarmuka Halaman Admin Menu Hapus Kategori Produk |

Keterangan

Antarmuka Halaman Admin Menu Hapus Kategori Produk menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menghapus data kategori produk,. Pilih data yang akan di hapus, dan klik tombol hapus, maka akan menampilkan pesan konfirmasi “apakah yakin menghpus data” ?

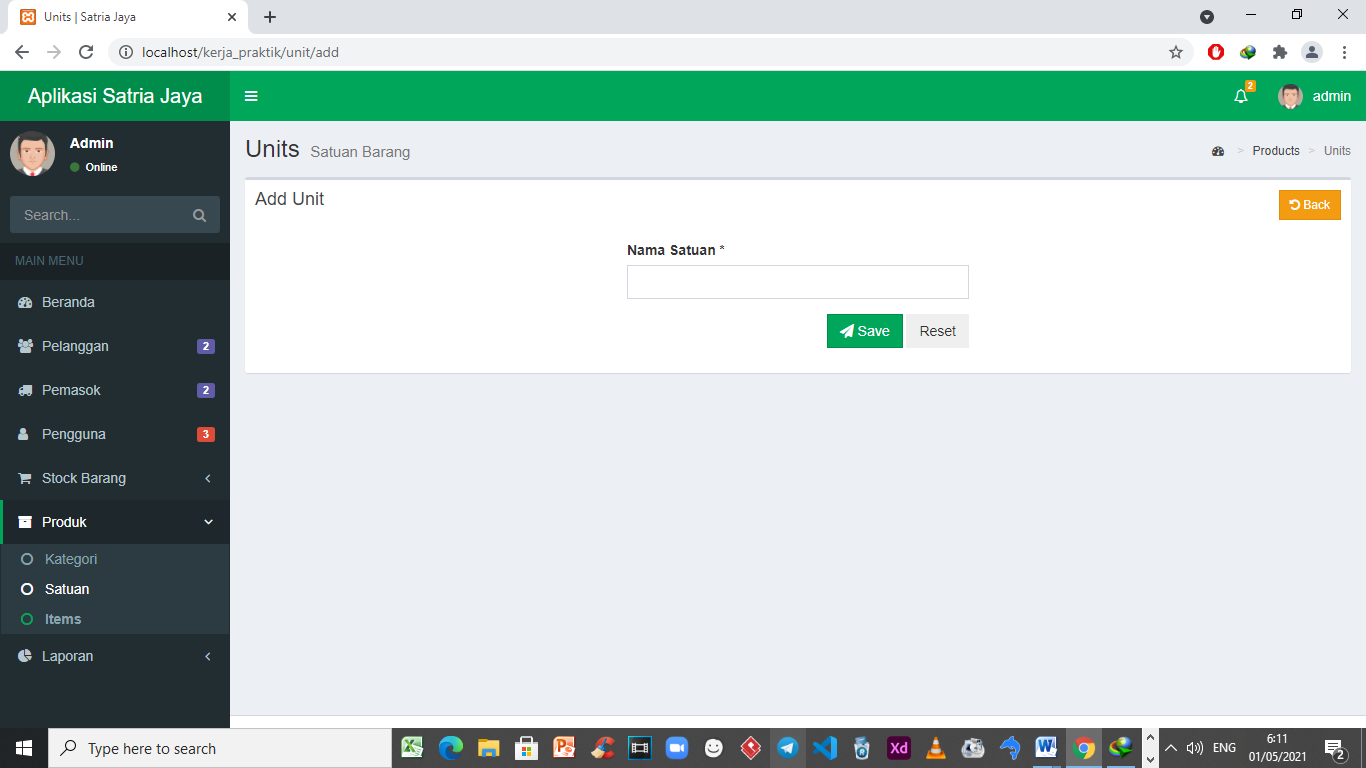
Jika menekan tombol ya maka, data akan terhapus, jika menekan tombol tidak maka data tidak akan terhapus dan kembali ke halaman awal.

1. Antarmuka Halaman Admin Menu Satuan Produk

|  |
| --- |
|  |
| Gambar 4. 33 Antarmuka Halaman Admin Menu Satuan Produk |

Keterangan

Antarmuka Halaman Admin Menu Satuan menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni tambah, ubah, hapus, cari, dan cetak data satuan produk,.

1. Antarmuka Halaman Admin Menu Tambah Satuan Produk

|  |
| --- |
|  |
| Gambar 4. 34 Antarmuka Halaman Admin Menu Tambah Satuan Produk |

Keterangan

Antarmuka Halaman Admin Menu Tambah Satuan Produk menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menambah data satuan produk,. Klik tombol tambah dan isi sesuai *form* yang telah di sediakan lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Admin Menu Ubah Satuan Produk

|  |
| --- |
|  |
| Gambar 4. 35 Antarmuka Halaman Admin Menu Ubah Satuan Produk |

Keterangan

Antarmuka Halaman Admin Menu Ubah Satuan Produk menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni mengubah data satuan produk ,. Klik tombol ubah dan isi sesuai *form* yang akan diubah lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Admin Menu Hapus Satuan Produk

|  |
| --- |
|  |
| Gambar 4. 36 Antarmuka Halaman Admin Menu Hapus Satuan Produk |

Keterangan

Antarmuka Halaman Admin Menu Hapus Satuan Produk menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menghapus data satuan produk,. Pilih data yang akan di hapus, dan klik tombol hapus, maka akan menampilkan pesan konfirmasi “apakah yakin menghpus data” ?

Jika menekan tombol ya maka, data akan terhapus, jika menekan tombol tidak maka data tidak akan terhapus dan kembali ke halaman awal.

1. Antarmuka Halaman Admin Menu Item Produk

|  |
| --- |
|  |
| Gambar 4. 37 Antarmuka Halaman Admin Menu Item Produk |

Keterangan

Antarmuka Halaman Admin Menu Item menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni tambah, ubah, hapus, cari, dan cetak data item produk,.

1. Antarmuka Halaman Admin Menu Tambah Item Produk

|  |
| --- |
|  |
| Gambar 4. 38 Antarmuka Halaman Admin Menu Tambah Item Produk |

Keterangan

Antarmuka Halaman Admin Menu Tambah Item Produk menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menambah data item produk,. Klik tombol tambah dan isi sesuai *form* yang telah di sediakan lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Admin Menu Ubah Item Produk

|  |
| --- |
|  |
| Gambar 4. 39 Antarmuka Halaman Admin Menu Ubah Item Produk |

Keterangan

Antarmuka Halaman Admin Menu Ubah Item Produk menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni mengubah data item produk ,. Klik tombol ubah dan isi sesuai *form* yang akan diubah lalu tekan tombol simpan maka data akan tersimpan.

1. Antarmuka Halaman Admin Menu Hapus Item Produk

|  |
| --- |
|  |
| Gambar 4. 40 Antarmuka Halaman Admin Menu Hapus Item Produk |

Keterangan

Antarmuka Halaman Admin Menu Hapus Item Produk menggambarkan suatu aktifitas yang dapat dilakukan oleh admin pada halaman tersebut yakni menghapus data item produk,. Pilih data yang akan di hapus, dan klik tombol hapus, maka akan menampilkan pesan konfirmasi “apakah yakin menghapus data”?

Jika menekan tombol ya maka, data akan terhapus, jika menekan tombol tidak maka data tidak akan terhapus dan kembali ke halaman awal.

# Pengujian

Pengujian merupakan bagian yang penting dalam siklus pengembangan perangkat lunak. Pengujian dilakukan untuk menjamin kualitas dan juga mengetahui kelamahan dari perangkat lunak. Tujuan dari pengujian ini adalah untuk menjamin bahwa perangkat lunak yang dibangun memiliki kualitas yang handal. Pengujian perangkat lunak ini menggunakan metode pengujian *blackbox*.

Metode uji coba *blackbox* memfokuskan pada keperluan fungsional dari software. Karena itu ujicoba *blackbox* memungkinkan pengembangan *software* untuk membuat himpunan kondisi input yang akan melatih seluruh syarat-syarat fungsional suatu program.

Tabel 4. 1 Pengujian Black *Box*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 1. | Mengisi *username* dan mengosongkan *password* kemudian menekan tombol *Login* |  | Menampilkan pesan “Please fill out this field” |  | Sesuai |
| 2. | Mengisi *password* dan mengosongkan*username* kemudian menekan tombol *Login* |  | Menampilkan pesan“Please fill out this field” |  | Sesuai |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 3. | Mengisi *username* sesuai dan *password* tidak sesuai |  | Menampilkan pesan peringatan “*Username* atau *password* anda salah” |  | Sesuai |
| 4. | Mengisi *username* tidak sesuai dan *password* sesuai |  | Menampilkan pesan peringatan “*Username* atau *password* anda salah” |  | Sesuai |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 5. | Mengisi *username* dan *password* tidak sesuai |  | Menampilkan pesan peringatan “*Username* atau *password* anda salah” |  | Sesuai |
| 6. | Mengisi *username* dan *password* sesuai |  | Menampilkan pesan *Succes login* berhasil |  | Sesuai |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 7. | Memilih menu Pelanggan |  | Menampilkan halaman Pelanggan |  | Sesuai |
| 8. | Menekan tombol tambah Pelanggan |  | Menampilkan *form* Tambah Pelanggan |  | Sesuai |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 9. | Menekan tombol ubah Pelanggan |  | Menampilkan halaman ubah Pelanggan |  | Sesuai |
| 10. | Menekan tombol hapus Pelanggan |  | Menampilkan pesan konfirmasi hapus data |  | Sesuai |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 11. | Memilih tombol ya saat konfirmasi hapus data Pelanggan |  | Menampilkan pesan bahwa data berhasil dihapus |  | Sesuai |
| 12. | Melih menu Penjualan |  | Menampilkan *form* Penjualan |  | Sesuai |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 13. | Menekan tombol cari barang |  | Menampilkan *pop up* *list* databarang |  | Sesuai |
| 14. | Memilih tambah barang |  | Data tampil dalam bentuk *table* Penjualan |  | Sesuai |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 15. | Memilih ubah barang |  | Menampilkan *form* ubah barang |  | Sesuai |
| 16. | Menekan tombol hapus barang |  | Menampilkan *pop up* konfirmasi hapus data |  | Sesuai |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 17. | Menekan tombol proses pembayaran |  | Menampilkan notifikasi pembayaran |  | Sesuai |
| 18. | Menekan *cancel* proses transaksi |  | Menampilkan konfirmasi *cancel* data transaksi |  | Sesuai |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 19. | Menekan tombol cetak transaksi |  | Menampilkan halaman cetak data transaksi |  | Sesuai |
| 20. | Memilih menu Pengguna |  | Menampilkan halaman Pengguna |  | Sesuai |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 21. | Menekan tombol tambah Pengguna |  | Menampilkan *form* Tambah Pengguna |  | Sesuai |
| 22. | Menekan tombol ubah Pengguna |  | Menampilkan halaman ubah Pengguna |  | Sesuai |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 23. | Menekan tombol hapus Pengguna |  | Menampilkan pesan konfirmasi hapus data |  | Sesuai |
| 24. | Memilih tombol ya saat konfirmasi hapus data Pengguna |  | Menampilkan pesan bahwa data berhasil dihapus |  | Sesuai |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 25. | Memilih menu Pemasok |  | Menampilkan halaman Pemasok |  | Sesuai |
| 26. | Menekan tombol tambah Pemasok |  | Menampilkan *form* tambah Pemasok |  | Sesuai |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 27. | Memilih tombol ubah Pemasok |  | Menampilkan halaman ubah Pemasok |  | Sesuai |
| 28. | Menekan tombol hapus Pemasok |  | Menampilkan pesan konfirmasi hapus data |  | Sesuai |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 29. | Memilih tombol ya, saat konfirmasi hapus data Pemasok |  | Menampilkan pesan bahwa data berhasil dihapus |  | Sesuai |
| 30. | Menekan *Stock* Barang Masuk |  | Menampilkan Halaman *Stock* Barang Masuk |  | Sesuai |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 31. | Menekan tombol Tambah Barang Masuk |  | Menampilkan *form* halaman *Stock* Barang Masuk |  | Sesuai |
| 32. | Menekan tombol hapus *Stock* Barang Masuk |  | Menampilkan pesan konfirmasi hapus data |  | Sesuai |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 33. | Memilih tombol ya, saat konfirmasi hapus data Barang Masuk |  | Menampilkan pesan bahwa data berhasil dihapus |  | Sesuai |
| 33. | Menekan *Stock* Barang Keluar |  | Menampilkan Halaman *Stock* Barang Keluar |  | Sesuai |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 34. | Menekan tombol Tambah Barang Keluar |  | Menampilkan *form* Barang Keluar |  | Sesuai |
| 35. | Menekan tombol hapus *Stock* Barang Keluar |  | Menampilkan pesan konfirmasi hapus data |  | Sesuai |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 36. | Memilih tombol ya, saat konfirmasi hapus *Stock* Barang Keluar |  | Menampilkan pesan bahwa data berhasil dihapus |  | Sesuai |
| 37 | Menekan tombol Kategori |  | Menampilkan Halaman Kategori |  | Sesuai |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 38. | Menekan tombol Tambah Kategori |  | Menampilkan Halaman Tambah Kategori |  | Sesuai |
| 39 | Menekan tombol Ubah Kategori |  | Menampilkan Halaman Ubah Kategori |  | Sesuai |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 40. | Menekan tombol Hapus Kategori |  | Menampilkan pesan konfirmasi hapus data |  | Sesuai |
| 41 | Menekan tombol ya saat konfirmasi pesan data |  | Menampilkanpesan bahwa data berhasil dihapus |  | Sesuai |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 42. | Menekan tombol Satuan |  | Menampilkan Halaman Satuan |  | Sesuai |
| 43 | Menekan tombol Tambah Satuan |  | MenampilkanHalaman Satuan |  | Sesuai |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 44. | Menekan tombol Ubah Satuan |  | Menampilkan Halaman Ubah Satuan |  | Sesuai |
| 45 | Menekan tombol Hapus Satuan |  | Menampilkan konfirmasi hapus data |  | Sesuai |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 46. | Menekan tombol ya ketika pesan konfirmasi hapus data |  | Menampilkan pesan bahwa data berhasil dihapus |  | Sesuai |
| 47 | Menekan tombol Item |  | Menampilkan Halaman Item |  | Sesuai |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 48. | Menekan tombol Tambah Item |  | Menampilkan Halaman Tambah Item |  | Sesuai |
| 49 | Menekan tombol Ubah Item |  | Menampilkan Halaman Halaman Ubah Item |  | Sesuai |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil yang**  **Diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| 50. | Menekan tombol Hapus Item |  | Menampilkan pesan konfirmasi hapus data |  | Sesuai |
| 51 | Menekan tombol ya ketika pesan konfirmasi data muncul |  | Menampilkan pesan bahwa data dihapus |  | Sesuai |

# BAB V

# KESIMPULAN DAN SARAN

# Kesimpulan

TOKO SATRIA JAYA merupa suatu toko yang bergerak di bidang perdagangan,. Pada saat ini pencatatan barang masuk dari supplier ataupun barang keluar berupa *return* masih dicatat secara manual, begitu pun setiap transaksi penjualan dicatat secara manual,

Dari hasil kerja praktek yang telah dilakukan, penulis dapat menyimpulkan beberapa hal sebagai berikut :

1. Aplikasi ini dapat meningkatan efektifitas dalam opersional transaksi dengan *customer*.
2. Aplikasi ini dapat mengelola data barang masuk maupun keluar agar informasi yang dihasilkan akurat dan aktual.
3. Aplikasi ini dapat mengelola *stock* barang.
4. Aplikasi ini dapat membantu dalam membuat laporan secara cepat dan tepat.

Pembuatan aplikasi penjualan barang bahan bangunan (*point of sale*), ini dimaksudkan untuk membantu mengelola data barang, serta transaksi di TOKO BANGUNAN SATRIA JAYA yang saat ini masih menggunakan metode tulis tangan.

# Saran

Berdasarkan hasil analisis dan perancangan sistem yang telah dibuat maka penulis mengajukan beberapa saran kepada pihak yang akan mengembangkan sebagai pengembang aplikasi ini selanjutnya antara lain:

* 1. Aplikasi *Point Of Sale* ini dapat dikembangkan sesuai dengan kebutuhan fungsional Toko Satria Jaya
  2. Perlu adanya pemeliharaan dan pengembangan dari *developer* lain untuk mengembangkan aplikasi agar lebih baik dan bermanfaat untuk menunjang operasional kerja.

Penulis menyadari bahwa laporan ini banyak terdapat kesalahan baik dari teori maupun pembahasan, dan sangat jauh dari kesempurnaan. Tentunya, penulis akan terus memperbaiki laporan ini dengan mengacu pada sumber yang dapat dipertanggungjawabkan nantinya. Oleh karena itu, penulis sangat mengharapkan kritik dan saran yang sifatnya membangun dari pembaca, agar laporan ini dapat lebih baik

DAFTAR RIWAYAT HIDUP



NPM : 5520118042

Nama : Haris Anwar

Tempat/Tgl. Lahir : Cianjur, 24 Februari 1998

Jenis Kelamin : Laki-laki

Agama : Islam

Program Studi : Teknik Informatika

Alamat Rumah : Kp. Babakan Rt/Rw. 03/03 Desa Nyalindung Kec. Cugenang

Kab. Cianjur

Telp : 088808676966

Emai/Gmail : [harisanwar129@gmail.com](mailto:harisanwar129@gmail.com)

|  |  |  |  |
| --- | --- | --- | --- |
| **Pendidikan** | **Nama Lembaga** | **Program Studi** | **Tahun Lulus** |
| MI | MI ALKHOERIYYAH | - | 2011 |
| SMP | MTSN CIHERANG | - | 2014 |
| SMA/SMK | SMKN 1 CIPANAS | ADMINISTRASI PERAKANTORAN | 2017 |
| PT | UNIV. SURYAKANCANA | TEKNIK INFORMATIKA | 2022 |

Demikian daftar riwayat hidup ini dibuat dengan sebenarnya.

Cianjur, 21 April 2021

Mahasiswa Ybs.

Haris Anwar

# DAFTAR PUSTAKA

Use the "Insert Citation" button to add citations to this document.

Diana, A. (2011). Sistem Informasi Akuntansi. Yogyakarta: Andi Yogyakarta.

F K, Alexander. Sibero. (2011). Kitab Suci Web Programing. Yogyakarta: Mediakom.

F K, Alexander. Sibero. (2013). Web Programing Power Pack. Yogyakarta: Mediakom.

Grady, B. (2005). Object Oriented Analysis and Design with Application 2nd

Edition. Unitid State of Amerika.

H, E. (2002). Pedoman Penggunaan Laboratorium Sekolah. Bandung: PT Remaja

Roesda Karya.

SISTEM INFORMASI KEPEGAWAIAN DISKOMINFO(rev1). (n.d.).

García Reyes, L. E. (2013). 済無No Title No Title. Journal of Chemical Information and Modeling, 53(9), 1689–1699.

F K, Alexander. Sibero. (2011). Kitab Suci Web Programing. Yogyakarta: Mediakom.

F K, Alexander. Sibero. (2013). Web Programing Power Pack. Yogyakarta: Mediakom.

Oliver, J. (2019). Bab Ii Tinjauan Pustaka Aplikasi. Hilos Tensados, 1, 1–476.

Chandra, K., Ciputra, U., & Town, U. C. (2017). Rancang Bangun Sistem Informasi Point of Sales Berbasis Website Pada Distributor Kain Hoggy Djaya. Sistem Informasi, 1(3), 192–200.

Praktek, K. (2020). Perancangan aplikasi sistem pengelolaan barang di labtif berbasis web.

[http://repository.potensi-utama.ac.id/jspui/bitstream/123456789/2990/6/BAB II.pdf](http://repository.potensi-utama.ac.id/jspui/bitstream/123456789/2990/6/BAB%20II.pdf) ww.yukoding.com

<http://weekly.cnbnews.com/news/article.html?no=124000>

Belakang, L. (2005). Bab I. Sasmitodia, 1–5.

# LAMPIRAN

**SOURCE CODE**

|  |
| --- |
| Model *Category* |
| <?php  class Category\_m extends CI\_Model  {  var $table = "kategori";  public function get($id = null)  {  $this->db->select('\*');  $this->db->from($this->table);  if ($id != null) {  $this->db->where('category\_id', $id);  }  $this->db->order\_by('cname', 'asc');  $query = $this->db->get();  return $query;  }  public function add($data)  {  $params = array(  'cname' => $data['cname']  );  $this->db->insert($this->table, $params);  }  public function edit($data)  {  $params = array(  'cname' => $data['cname'],  'updated' => date('Y-m-d H:i:s')  );  $this->db->where('category\_id', $data['id']);  $this->db->update($this->table, $params);  }  public function del($id)  {  $this->db->where('category\_id', $id);  $this->db->delete($this->table);  }  } |

|  |
| --- |
| Model *Customer* |
| <?php  class Category\_m extends CI\_Model  {  var $table = "kategori";  public function get($id = null)  {  $this->db->select('\*');  $this->db->from($this->table);  if ($id != null) {  $this->db->where('category\_id', $id);  }  $this->db->order\_by('cname', 'asc');  $query = $this->db->get();  return $query;  }  public function add($data)  {  $params = array(  'cname' => $data['cname']  );  $this->db->insert($this->table, $params);  }  public function edit($data)  {  $params = array(  'cname' => $data['cname'],  'updated' => date('Y-m-d H:i:s')  );  $this->db->where('category\_id', $data['id']);  $this->db->update($this->table, $params);  }  public function del($id)  {  $this->db->where('category\_id', $id);  $this->db->delete($this->table);  }  } |

|  |
| --- |
| Model *Item* |
| <?php  class Item\_m extends CI\_Model  {  var $table = "p\_item";  public function get($id = null)  {  $this->db->select('p\_item.name,  p\_item.barcode,  kategori.cname as category\_name,  satuan.uname as unit\_uname,  p\_item.price,  p\_item.stock,  p\_item.item\_id,  p\_item.category\_id,  p\_item.unit\_id');  $this->db->from($this->table);  $this->db->join('kategori', 'p\_item.category\_id = kategori.category\_id');  $this->db->join('satuan', 'p\_item.unit\_id = satuan.unit\_id');  if ($id != null) {  $this->db->where('item\_id', $id);  }  $this->db->order\_by('barcode', 'asc');  $query = $this->db->get();  return $query;  }  function check\_barcode($code, $id = null)  {  $this->db->select('\*');  $this->db->from($this->table);  $this->db->where('barcode', $code);  if ($id != null) {  $this->db->where('item\_id !=', $id);  }  $query = $this->db->get();  return $query;  }  public function add($data)  {  $params = array(  'name' => $data['name'],  'barcode' => $data['barcode'],  'category\_id' => $data['category'],  'unit\_id' => $data['unit'],  'price' => $data['price']  );  $this->db->insert($this->table, $params);  }  public function edit($data)  {  $params = array(  'name' => $data['name'],  'barcode' => $data['barcode'],  'category\_id' => $data['category'],  'unit\_id' => $data['unit'],  'price' => $data['price'],  'updated' => date('Y-m-d H:i:s')  );  $this->db->where('item\_id', $data['id']);  $this->db->update($this->table, $params);  }  public function del($id)  {  $this->db->where('item\_id', $id);  $this->db->delete($this->table);  }  function update\_stock\_in($data)  {  $qty = $data['qty'];  $id = $data['item\_id'];  $sql = "UPDATE p\_item SET stock = stock + '$qty' WHERE item\_id = '$id'";  $this->db->query($sql);  }  function update\_stock\_out($data)  {  $qty = $data['qty'];  $id = $data['item\_id'];  $sql = "UPDATE p\_item SET stock = stock - '$qty' WHERE item\_id = '$id'";  $this->db->query($sql);  }  } |

|  |
| --- |
| Model Penjualan |
| <?php  class Sale\_m extends CI\_Model  {  function invoice\_no()  {  $query = $this->db->query("SELECT MAX(MID(invoice,9,4)) AS invoice\_no  FROM penjualan WHERE MID(invoice,3,6) = DATE\_FORMAT(CURDATE(),'%y%m%d')");  if ($query->num\_rows() > 0) {  $r = $query->row();  $n = ((int)$r->invoice\_no) + 1;  $no = sprintf("%'.04d", $n);  } else {  $no = "0001";  }  $invoice = "SJ" . date('ymd') . $no;  return $invoice;  }  public function get\_cart($cart\_id = null, $item\_id = null)  {  $this->db->select('\*, p\_item.name as item\_name, transaksi.price as cart\_price');  $this->db->from('transaksi');  $this->db->join('p\_item', 'transaksi.item\_id = p\_item.item\_id');  $this->db->join('satuan', 'p\_item.unit\_id = satuan.unit\_id');  if ($cart\_id != null) {  $this->db->where('card\_id', $cart\_id);  }  if ($item\_id != null) {  $this->db->where('transaksi.item\_id', $item\_id);  }  $this->db->where('user\_id', $this->session->userdata('userid'));  $query = $this->db->get();  return $query;  }  public function add\_cart($data)  {  $query = $this->db->query("SELECT MAX(cart\_id) AS cart\_no FROM transaksi");  if ($query->num\_rows() > 0) {  $r = $query->row();  $cart\_no = ((int)$r->cart\_no) + 1;  } else {  $cart\_no = "1";  }  $params = array(  'cart\_id' => $cart\_no,  'item\_id' => $data['item\_id'],  'price' => $data['price'],  'discount\_item' => 0,  'qty' => $data['qty'],  'total' => $data['price'] \* $data['qty'],  'user\_id' => $this->session->userdata('userid')  );  $this->db->insert('transaksi', $params);  }  public function update\_cart\_qty($data)  {  $sql = "UPDATE transaksi  SET price = '$data[price]',  qty = qty + '$data[qty]',  total = '$data[price]' \* qty  WHERE item\_id = '$data[item\_id]'";  $this->db->query($sql);  }  public function edit\_cart($data)  {  $params = array(  'price' => $data['price'],  'qty' => $data['qty'],  'discount\_item' => $data['discount'],  'total' => $data['total'],  'updated' => date('Y-m-d H:i:s')  );  $this->db->where('cart\_id', $data['cart\_id']);  $this->db->update('transaksi', $params);  }  public function del\_cart($cart\_id = null, $user\_id = null)  {  if ($cart\_id != null) {  $this->db->where('cart\_id', $cart\_id);  }  if ($user\_id != null) {  $this->db->where('user\_id', $user\_id);  }  $this->db->delete('transaksi');  }  public function add\_sale($data)  {  $params = array(  'invoice' => $this->invoice\_no(),  'customer\_id' => $data['customer\_id'] == "" ? null : $data['customer\_id'],  'total\_price' => $data['subtotal'],  'discount' => $data['discount'],  'final\_price' => $data['grandtotal'],  'cash' => $data['cash'],  'change' => $data['change'],  'note' => $data['note'],  'date' => $data['date'],  'user\_id' => $this->session->userdata('userid')  );  $this->db->insert('penjualan', $params);  return $this->db->insert\_id();  }  public function add\_sale\_detail($params)  {  $this->db->insert\_batch('t\_sale\_detail', $params);  }  public function get\_sale($id = null)  {  $this->db->select('\*, customer.name as customer\_name, user.username as user\_name, penjualan.created as sale\_created');  $this->db->from('penjualan');  $this->db->join('customer', 'penjualan.customer\_id = customer.customer\_id', 'left');  $this->db->join('user', 'penjualan.user\_id = user.user\_id');  if ($id != null) {  $this->db->where('sale\_id', $id);  }  $query = $this->db->get();  return $query;  }  public function get\_sale\_detail($sale\_id = null)  {  $this->db->from('t\_sale\_detail');  $this->db->join('p\_item', 't\_sale\_detail.item\_id = p\_item.item\_id');  if ($sale\_id != null) {  $this->db->where('t\_sale\_detail.sale\_id', $sale\_id);  }  $query = $this->db->get();  return $query;  }  function get\_sale\_pagination($limit = null, $start = null)  {  $post = $this->session->userdata('search');  $this->db->select('\*, customer.name as customer\_name, user.username as user\_name, penjualan.created as sale\_created');  $this->db->from('penjualan');  $this->db->join('customer', 'penjualan.customer\_id = customer.customer\_id', 'left');  $this->db->join('user', 'penjualan.user\_id = user.user\_id');  if (!empty($post['date1']) && !empty($post['date2'])) {  $this->db->where("penjualan.date BETWEEN '" . db\_date($post['date1']) . "' AND '" . db\_date($post['date2']) . "'");  }  if (!empty($post['customer'])) {  if ($post['customer'] == 'null') {  $this->db->where("penjualan.customer\_id IS NULL");  } else {  $this->db->where("penjualan.customer\_id", $post['customer']);  }  }  if (!empty($post['invoice'])) {  $this->db->like("invoice", $post['invoice']);  }  $this->db->limit($limit, $start);  $this->db->order\_by('penjualan.created', 'desc');  $query = $this->db->get();  return $query;  }  public function del\_sale($id)  {  $this->db->where('sale\_id', $id);  $this->db->delete('penjualan');  }  } |

|  |
| --- |
| Model *Stock* |
| <?php  class stock\_m extends CI\_Model  {  var $table = "t\_stock";  public function get($id = null)  {  $this->db->select('\*');  $this->db->from($this->table);  if ($id != null) {  $this->db->where('stock\_id', $id);  }  // $this->db->order\_by('stock\_id', 'desc');  $query = $this->db->get();  return $query;  }  public function del($id)  {  $this->db->where('stock\_id', $id);  $this->db->delete($this->table);  }  public function get\_stock\_in()  {  $this->db->select('t\_stock.stock\_id,  p\_item.barcode,  p\_item.name as item\_name,  qty, date, detail,  supplier.name as supplier\_name,  p\_item.item\_id');  $this->db->from($this->table);  $this->db->join('p\_item', 't\_stock.item\_id = p\_item.item\_id');  $this->db->join('supplier', 't\_stock.supplier\_id = supplier.supplier\_id', 'left');  $this->db->where('type', 'in');  $this->db->order\_by('stock\_id', 'desc');  $query = $this->db->get();  return $query;  }  public function add\_stock\_in($data)  {  $params = array(  'item\_id' => $data['item\_id'],  'type' => "in",  'detail' => $data['detail'],  'supplier\_id' => $data['supplier'] == '' ? null : $data['supplier'],  'qty' => $data['qty'],  'date' => $data['date'],  'user\_id' => $this->session->userdata('userid')  );  $this->db->insert($this->table, $params);  }  public function get\_stock($id = null)  {  $post = $this->session->userdata('search');  $this->db->select('\*, satuan.uname as satuan\_uname,kategori.cname as category\_cname, p\_item.name as p\_item\_name');  $this->db->from('p\_item');  $this->db->join('satuan', 'p\_item.unit\_id = satuan.unit\_id');  $this->db->join('kategori', 'p\_item.category\_id = kategori.category\_id');  $query = $this->db->get();  return $query;  }  public function get\_stock\_detail($stock\_id = null)  {  $this->db->from('stock\_detail');  $this->db->join('p\_item', 'stock\_detail.item\_id = p\_item.item\_id');  if ($stock\_id != null) {  $this->db->where('stock\_detail.stock\_id', $stock\_id);  }  $query = $this->db->get();  return $query;  }  function get\_stock\_pagination($limit = null, $start = null)  {  $post = $this->session->userdata('search');  $this->db->select('\*, satuan.uname as satuan\_uname,kategori.cname as category\_cname, p\_item.name as p\_item\_name');  $this->db->from('p\_item');  $this->db->join('satuan', 'p\_item.unit\_id = satuan.unit\_id');  $this->db->join('kategori', 'p\_item.category\_id = kategori.category\_id');  if (!empty($post['date1']) && !empty($post['date2'])) {  $this->db->where("p\_items.created BETWEEN '" . db\_date($post['date1']) . "' AND '" . db\_date($post['date2']) . "'");  }  if (!empty($post['kategori'])) {  if ($post['kategori'] == 'null') {  $this->db->where("p\_item.category\_id IS NULL");  } else {  $this->db->where("p\_item.category\_id", $post['kategori']);  }  }  if (!empty($post['name'])) {  $this->db->like("name", $post['name']);  }  $this->db->limit($limit, $start);  $this->db->order\_by('p\_item.created', 'desc');  $query = $this->db->get();  return $query;  }  public function get\_stock\_out()  {  $this->db->select('\*');  $this->db->from($this->table);  $this->db->join('p\_item', 't\_stock.item\_id = p\_item.item\_id');  $this->db->where('type', 'out');  $this->db->order\_by('stock\_id', 'desc');  $query = $this->db->get();  return $query;  }  public function add\_stock\_out($data)  {  $params = array(  'item\_id' => $data['item\_id'],  'type' => "out",  'detail' => $data['detail'],  'qty' => $data['qty'],  'date' => $data['date'],  'user\_id' => $this->session->userdata('userid')  );  $this->db->insert($this->table, $params);  }  } |

|  |
| --- |
| Model *Supplier* |
| <?php  class Supplier\_m extends CI\_Model {  var $table = "supplier";  public function get($id = null)  {  $this->db->select('\*');  $this->db->from($this->table);  if($id != null) {  $this->db->where('supplier\_id', $id);  }  $this->db->order\_by('name', 'asc');  $query = $this->db->get();  return $query;  }  public function add($data)  {  $params = array(  'name' => $data['name'],  'phone' => $data['phone'],  'address' => $data['addr'],  'description' => $data['desc'] == '' ? null : $data['desc']  );  $this->db->insert($this->table, $params);  }  public function edit($data)  {  $params = array(  'name' => $data['name'],  'phone' => $data['phone'],  'address' => $data['addr'],  'description' => empty($data['desc']) ? null : $data['desc'],  'updated' => date('Y-m-d H:i:s')  );  $this->db->where('supplier\_id', $data['id']);  $this->db->update($this->table, $params);  }  public function del($id)  {  $this->db->where('supplier\_id', $id);  $this->db->delete($this->table);  }  } |

|  |
| --- |
| Model Satuan |
| <?php  class Unit\_m extends CI\_Model  {  var $table = "satuan";  public function get($id = null)  {  $this->db->select('\*');  $this->db->from($this->table);  if ($id != null) {  $this->db->where('unit\_id', $id);  }  $this->db->order\_by('uname', 'asc');  $query = $this->db->get();  return $query;  }  public function add($data)  {  $params = array(  'uname' => $data['uname']  );  $this->db->insert($this->table, $params);  }  public function edit($data)  {  $params = array(  'uname' => $data['uname'],  'updated' => date('Y-m-d H:i:s')  );  $this->db->where('unit\_id', $data['id']);  $this->db->update($this->table, $params);  }  public function del($id)  {  $this->db->where('unit\_id', $id);  $this->db->delete($this->table);  }  } |

|  |
| --- |
| Model *User* |
| <?php  class User\_m extends CI\_Model {  var $table = "user";  public function login($data)  {  $this->db->select('\*');  $this->db->from($this->table);  $this->db->where('username', $data['username']);  $this->db->where('password', sha1($data['password']));  $query = $this->db->get();  return $query;  }  public function get($id = null)  {  $this->db->select('\*');  $this->db->from($this->table);  if($id != null) {  $this->db->where('user\_id', $id);  }  $query = $this->db->get();  return $query;  }  public function add($data)  {  $params['username'] = $data['username'];  $params['password'] = sha1($data['password']);  $params['name'] = $data['name'];  $params['address'] = empty($data['address']) ? null : $data['address'];  $params['level'] = $data['level'];  $this->db->insert($this->table, $params);  }  public function edit()  {  $data = $this->input->post(null, TRUE);  $params['username'] = str\_replace(' ', '', $data['username']);  if(!empty($data['password'])) {  $params['password'] = sha1($data['password']);  }  $params['name'] = $data['name'];  $params['address'] = empty($data['address']) ? null : $data['address'];  $params['level'] = $data['level'];  $params['updated'] = date('Y-m-d H:i:s');  $this->db->where('user\_id', $data['user\_id']);  $this->db->update($this->table, $params);  }  public function del($id)  {  $this->db->where('user\_id', $id);  $this->db->delete($this->table);  }  } |

|  |
| --- |
| Model *Category* |
| <?php  defined('BASEPATH') or exit('No direct script access allowed');  class Category extends CI\_Controller  {  var $title = "categories";  function \_\_construct()  {  parent::\_\_construct();  check\_login\_session();  $this->load->model('category\_m', 'category');  }  public function index()  {  $data['title'] = $this->title;  $data['row'] = $this->category->get();  $this->template->load('\_template', 'product/category/category\_data', $data);  }  public function add()  {  $category = new stdClass();  $category->category\_id = null;  $category->cname = null;  $data = array(  'title' => $this->title,  'page' => 'add',  'row' => $category  );  $this->template->load('\_template', 'product/category/category\_form', $data);  }  public function edit($id = null)  {  if ($id != null) {  $query = $this->category->get($id);  if ($query->num\_rows() > 0) {  $category = $query->row();  $data = array(  'title' => $this->title,  'page' => 'edit',  'row' => $category  );  $this->template->load('\_template', 'product/category/category\_form', $data);  } else {  echo "<script>alert('Data tidak ditemukan');  window.location='" . site\_url('category') . "';</script>";  }  } else {  echo "<script>window.location='" . site\_url('category') . "';</script>";  }  }  public function process()  {  if (isset($\_POST['add'])) {  $data = $this->input->post(null, TRUE);  $this->category->add($data);  } else if (isset($\_POST['edit'])) {  $data = $this->input->post(null, TRUE);  $this->category->edit($data);  }  if ($this->db->affected\_rows() == 1) {  echo "<script>alert('Data berhasil disimpan');  window.location='" . site\_url('category') . "';</script>";  } else {  redirect('category');  }  }  public function del($id = null)  {  $this->category->del($id);  if ($this->db->affected\_rows() == 1) {  echo "<script>alert('Data berhasil dihapus');  window.location='" . site\_url('category') . "';</script>";  } else {  echo "<script>window.location='" . site\_url('category') . "';</script>";  }  }  } |

|  |
| --- |
| Controller *Customer* |
| <?php  defined('BASEPATH') OR exit('No direct script access allowed');  class Customer extends CI\_Controller {  var $title = "customers";  function \_\_construct()  {  parent::\_\_construct();  check\_login\_session();  $this->load->model('customer\_m', 'customer');  }    public function index()  {  $data['title'] = $this->title;  $data['row'] = $this->customer->get()->result();  $this->template->load('\_template', 'customer/customer\_data', $data);  }    public function add()  {  $customer = new stdClass();  $customer->customer\_id = null;  $customer->name = null;  $customer->gender = null;  $customer->phone = null;  $customer->address = null;  $data = array(  'title' => $this->title,  'page' => 'add',  'row' => $customer  );  $this->template->load('\_template', 'customer/customer\_form', $data);  }  public function edit($id = null)  {  if($id != null) {  $query = $this->customer->get($id);  if($query->num\_rows() > 0) {  $customer = $query->row();  $data = array(  'title' => $this->title,  'page' => 'edit',  'row' => $customer  );  $this->template->load('\_template', 'customer/customer\_form', $data);  } else {  echo "<script>alert('Data tidak ditemukan');  window.location='".site\_url('customer')."';</script>";  }  } else {  echo "<script>window.location='".site\_url('customer')."';</script>";  }  }  public function process()  {  if(isset($\_POST['add'])) {  $data = $this->input->post(null, TRUE);  $this->customer->add($data);  } else if(isset($\_POST['edit'])) {  $data = $this->input->post(null, TRUE);  $this->customer->edit($data);  }  if($this->db->affected\_rows() == 1) {  echo "<script>alert('Data berhasil disimpan');  window.location='".site\_url('customer')."';</script>";  } else {  redirect('customer');  }  }  public function del($id = null)  {  $this->customer->del($id);    if($this->db->affected\_rows() == 1) {  echo "<script>alert('Data berhasil dihapus');  window.location='".site\_url('customer')."';</script>";  } else {  echo "<script>window.location='".site\_url('customer')."';</script>";  }  }  } |

|  |
| --- |
| Controller *Dashboard* |
| <?php  defined('BASEPATH') or exit('No direct script access allowed');  class Dashboard extends CI\_Controller  {  function \_\_construct()  {  parent::\_\_construct();  check\_login\_session();  }  public function index()  {  $query = $this->db->query("SELECT t\_sale\_detail.item\_id, p\_item.name, (SELECT SUM(t\_sale\_detail.qty)) AS sold  FROM t\_sale\_detail  INNER JOIN penjualan ON t\_sale\_detail.sale\_id = penjualan.sale\_id  INNER JOIN p\_item ON t\_sale\_detail.item\_id = p\_item.item\_id  WHERE MID(penjualan.date,6,2) = DATE\_FORMAT(CURDATE(),'%m')  GROUP BY t\_sale\_detail.item\_id  ORDER BY sold desc  LIMIT 10");  $data['title'] = "Dashboard";  $data['row'] = $query->result();  $this->template->load('\_template', 'dashboard', $data);  }  } |

|  |
| --- |
| Controller *Item* |
| <?php  defined('BASEPATH') or exit('No direct script access allowed');  class Item extends CI\_Controller  {  var $title = "items";  function \_\_construct()  {  parent::\_\_construct();  check\_login\_session();  $this->load->model('item\_m', 'item');  }  public function index()  {  $data['title'] = $this->title;  $data['row'] = $this->item->get()->result();  $this->template->load('\_template', 'product/item/item\_data', $data);  }  public function add()  {  $item = new stdClass();  $item->item\_id = null;  $item->name = null;  $item->barcode = null;  $item->price = null;  $this->load->model('category\_m', 'category');  $query\_ctg = $this->category->get();  $category[null] = '- Pilih -';  foreach ($query\_ctg->result() as $ctg) {  $category[$ctg->category\_id] = $ctg->cname;  }  $this->load->model('unit\_m', 'unit');  $query\_unt = $this->unit->get();  $unit[null] = '- Pilih -';  foreach ($query\_unt->result() as $unt) {  $unit[$unt->unit\_id] = $unt->uname;  }  $data = array(  'title' => $this->title,  'page' => 'add',  'row' => $item,  'category' => $category, 'selectedcategory' => null,  'unit' => $unit, 'selectedunit' => null  );  $this->template->load('\_template', 'product/item/item\_form', $data);  }  public function edit($id = null)  {  if ($id != null) {  $query = $this->item->get($id);  if ($query->num\_rows() > 0) {  $item = $query->row();  $this->load->model('category\_m', 'category');  $query\_ctg = $this->category->get();  $category[null] = '- Pilih -';  foreach ($query\_ctg->result() as $ctg) {  $category[$ctg->category\_id] = $ctg->cname;  }  $this->load->model('unit\_m', 'unit');  $query\_unt = $this->unit->get();  $unit[null] = '- Pilih -';  foreach ($query\_unt->result() as $unt) {  $unit[$unt->unit\_id] = $unt->uname;  }  $data = array(  'title' => $this->title,  'page' => 'edit',  'row' => $item,  'category' => $category, 'selectedcategory' => $item->category\_id,  'unit' => $unit, 'selectedunit' => $item->unit\_id  );  $this->template->load('\_template', 'product/item/item\_form', $data);  } else {  echo "<script>alert('Data tidak ditemukan');  window.location='" . site\_url('item') . "';</script>";  }  } else {  echo "<script>window.location='" . site\_url('item') . "';</script>";  }  }  public function process()  {  $data = $this->input->post(null, TRUE);  if (isset($\_POST['add'])) {  if ($this->item->check\_barcode($data['barcode'])->num\_rows() > 0) {  echo "<script>alert('Barcode ini sudah dipakai barang lain');  window.location='" . site\_url('item/add') . "';</script>";  } else {  $this->item->add($data);  echo "<script>alert('Data berhasil disimpan');  window.location='" . site\_url('item') . "';</script>";  }  } else if (isset($\_POST['edit'])) {  if ($this->item->check\_barcode($data['barcode'], $data['id'])->num\_rows() > 0) {  echo "<script>alert('Barcode ini sudah dipakai barang lain');  window.location='" . site\_url('item/edit/' . $data['id']) . "';</script>";  } else {  $this->item->edit($data);  echo "<script>alert('Data berhasil disimpan');  window.location='" . site\_url('item') . "';</script>";  }  } else {  redirect('item');  }  }  public function del($id = null)  {  $this->item->del($id);  if ($this->db->affected\_rows() == 1) {  echo "<script>alert('Data berhasil dihapus');  window.location='" . site\_url('item') . "';</script>";  } else {  echo "<script>window.location='" . site\_url('item') . "';</script>";  }  }  } |

|  |
| --- |
| Controller Laporan |
| <?php  defined('BASEPATH') or exit('No direct script access allowed');  class Report extends CI\_Controller  {  function \_\_construct()  {  parent::\_\_construct();  check\_login\_session();  $this->load->model('sale\_m', 'sale');  //baru  $this->load->model('stock\_m', 'stock');  }  public function sale()  {  $this->load->model('customer\_m', 'customer');  $this->load->library('pagination');  if (isset($\_POST['reset'])) {  $this->session->unset\_userdata('search');  }  if (isset($\_POST['filter'])) {  $post = $this->input->post(null, TRUE);  $this->session->set\_userdata('search', $post);  } else {  $post = $this->session->userdata('search');  }  $config['base\_url'] = site\_url('report/sale');  $config['uri\_segment'] = 3;  $config['total\_rows'] = $this->sale->get\_sale\_pagination()->num\_rows();  $config['per\_page'] = 5;  $config['first\_link'] = 'First';  $config['last\_link'] = 'Last';  $config['next\_link'] = 'Next';  $config['prev\_link'] = 'Prev';  $config['num\_tag\_open'] = '<li>';  $config['num\_tag\_close'] = '</li>';  $config['cur\_tag\_open'] = '<li class="active"><a>';  $config['cur\_tag\_close'] = '</a></li>';  $config['next\_tag\_open'] = '<li>';  $config['next\_tag\_close'] = '</li>';  $config['prev\_tag\_open'] = '<li>';  $config['prev\_tag\_close'] = '</li>';  $config['first\_tag\_open'] = '<li>';  $config['first\_tag\_close'] = '</li>';  $config['last\_tag\_open'] = '<li>';  $config['last\_tag\_close'] = '</li>';  $this->pagination->initialize($config);  $data['title'] = 'Sales Report';  $data['pagination'] = $this->pagination->create\_links();  $data['customer'] = $this->customer->get()->result();  $data['row'] = $this->sale->get\_sale\_pagination($config['per\_page'], $this->uri->segment(3));  $data['post'] = $post;  $this->template->load('\_template', 'report/sale\_report', $data);  }  public function sale\_product($sale\_id = null)  {  $detail = $this->sale->get\_sale\_detail($sale\_id)->result();  echo json\_encode($detail);  }  //baru  public function stock()  {  $this->load->model('category\_m', 'category');  $this->load->library('pagination');  if (isset($\_POST['reset'])) {  $this->session->unset\_userdata('search');  }  if (isset($\_POST['filter'])) {  $post = $this->input->post(null, TRUE);  $this->session->set\_userdata('search', $post);  } else {  $post = $this->session->userdata('search');  }  $config['base\_url'] = site\_url('report/stock');  $config['uri\_segment'] = 3;  $config['total\_rows'] = $this->stock->get\_stock\_pagination()->num\_rows();  $config['per\_page'] = 5;  $config['first\_link'] = 'First';  $config['last\_link'] = 'Last';  $config['next\_link'] = 'Next';  $config['prev\_link'] = 'Prev';  $config['num\_tag\_open'] = '<li>';  $config['num\_tag\_close'] = '</li>';  $config['cur\_tag\_open'] = '<li class="active"><a>';  $config['cur\_tag\_close'] = '</a></li>';  $config['next\_tag\_open'] = '<li>';  $config['next\_tag\_close'] = '</li>';  $config['prev\_tag\_open'] = '<li>';  $config['prev\_tag\_close'] = '</li>';  $config['first\_tag\_open'] = '<li>';  $config['first\_tag\_close'] = '</li>';  $config['last\_tag\_open'] = '<li>';  $config['last\_tag\_close'] = '</li>';  $this->pagination->initialize($config);  $data['title'] = 'Sales Report';  $data['pagination'] = $this->pagination->create\_links();  $data['category'] = $this->category->get()->result();  $data['row'] = $this->stock->get\_stock\_pagination($config['per\_page'], $this->uri->segment(3));  $data['post'] = $post;  $this->template->load('\_template', 'report/stock\_report', $data);  }  public function stock\_product($stock\_id = null)  {  $detail = $this->stock->get\_stock\_detail($stock\_id)->result();  echo json\_encode($detail);  }  } |

|  |
| --- |
| Controller *Stock* |
| <?php  defined('BASEPATH') or exit('No direct script access allowed');  class stock extends CI\_Controller  {  var $title = "stock in";  var $title2 = "stock out";  function \_\_construct()  {  parent::\_\_construct();  check\_login\_session();  $this->load->model('stock\_m', 'stock');  $this->load->model('item\_m', 'item');  }  public function cetak($id)  {  $data = array(  'stock' => $this->stock->get\_stock($id)->result(),  'stock\_detail' => $this->stock->get\_stock($id)->result(),  );  $this->load->view('transaction/stock/receipt\_print', $data);  }  public function stock\_in\_data()  {  $data['title'] = $this->title;  $data['row'] = $this->stock->get\_stock\_in()->result();  $this->template->load('\_template', 'transaction/stock\_in/stock\_in\_data', $data);  }  public function stock\_in\_add()  {  $this->load->model('supplier\_m', 'supplier');  $supplier = $this->supplier->get()->result();  $item = $this->item->get()->result();  $data = array(  'title' => $this->title,  'page' => 'in\_add',  'supplier' => $supplier,  'item' => $item  );  $this->template->load('\_template', 'transaction/stock\_in/stock\_in\_form', $data);  }  public function stock\_in\_del()  {  $item\_id = $this->uri->segment(4);  $stock\_id = $this->uri->segment(5);  $qty = $this->stock->get($stock\_id)->row()->qty;  $data = ['qty' => $qty, 'item\_id' => $item\_id];  $this->item->update\_stock\_out($data);  $this->stock->del($stock\_id);  if ($this->db->affected\_rows() > 0) {  echo "<script>alert('Data berhasil dihapus');  window.location='" . site\_url('stock/in') . "';</script>";  }  }  public function stock\_out\_data()  {  $data['title'] = $this->title2;  $data['row'] = $this->stock->get\_stock\_out()->result();  $this->template->load('\_template', 'transaction/stock\_out/stock\_out\_data', $data);  }  public function stock\_out\_add()  {  $item = $this->item->get()->result();  $data = array(  'title' => $this->title2,  'page' => 'out\_add',  'item' => $item  );  $this->template->load('\_template', 'transaction/stock\_out/stock\_out\_form', $data);  }  public function stock\_out\_del()  {  $item\_id = $this->uri->segment(4);  $stock\_id = $this->uri->segment(5);  $qty = $this->stock->get($stock\_id)->row()->qty;  $data = ['qty' => $qty, 'item\_id' => $item\_id];  $this->item->update\_stock\_in($data);  $this->stock->del($stock\_id);  if ($this->db->affected\_rows() > 0) {  echo "<script>alert('Data berhasil dihapus');  window.location='" . site\_url('stock/out') . "';</script>";  }  }  public function process()  {  if (isset($\_POST['in\_add'])) {  $data = $this->input->post(null, TRUE);  $this->stock->add\_stock\_in($data);  $this->item->update\_stock\_in($data);  if ($this->db->affected\_rows() > 0) {  echo "<script>alert('Data berhasil disimpan');  window.location='" . site\_url('stock/in') . "';</script>";  }  } else if (isset($\_POST['out\_add'])) {  $data = $this->input->post(null, TRUE);  $row\_item = $this->item->get($this->input->post('item\_id'))->row();  if ($row\_item->stock < $this->input->post('qty')) {  echo "<script>alert('Stock tidak mencukupi');  window.location='" . site\_url('stock/out') . "';</script>";  } else {  $this->stock->add\_stock\_out($data);  $this->item->update\_stock\_out($data);  if ($this->db->affected\_rows() > 0) {  echo "<script>alert('Data berhasil disimpan');  window.location='" . site\_url('stock/out') . "';</script>";  }  }  }  }  } |

|  |
| --- |
| Controller *Supplier* |
| <?php  defined('BASEPATH') or exit('No direct script access allowed');  class stock extends CI\_Controller  {  var $title = "stock in";  var $title2 = "stock out";  function \_\_construct()  {  parent::\_\_construct();  check\_login\_session();  $this->load->model('stock\_m', 'stock');  $this->load->model('item\_m', 'item');  }  public function cetak($id)  {  $data = array(  'stock' => $this->stock->get\_stock($id)->result(),  'stock\_detail' => $this->stock->get\_stock($id)->result(),  );  $this->load->view('transaction/stock/receipt\_print', $data);  }  public function stock\_in\_data()  {  $data['title'] = $this->title;  $data['row'] = $this->stock->get\_stock\_in()->result();  $this->template->load('\_template', 'transaction/stock\_in/stock\_in\_data', $data);  }  public function stock\_in\_add()  {  $this->load->model('supplier\_m', 'supplier');  $supplier = $this->supplier->get()->result();  $item = $this->item->get()->result();  $data = array(  'title' => $this->title,  'page' => 'in\_add',  'supplier' => $supplier,  'item' => $item  <?php  defined('BASEPATH') OR exit('No direct script access allowed');  class Supplier extends CI\_Controller {  var $title = "suppliers";  function \_\_construct()  {  parent::\_\_construct();  check\_login\_session();  $this->load->model('supplier\_m', 'supplier');  }    public function index()  {  $data['title'] = $this->title;  $data['row'] = $this->supplier->get()->result();  $this->template->load('\_template', 'supplier/supplier\_data', $data);  }    public function add()  {  $supplier = new stdClass();  $supplier->supplier\_id = null;  $supplier->name = null;  $supplier->phone = null;  $supplier->address = null;  $supplier->description = null;  $data = array(  'title' => $this->title,  'page' => 'add',  'row' => $supplier  );  $this->template->load('\_template', 'supplier/supplier\_form', $data);  }  public function edit($id = null)  {  if($id != null) {  $query = $this->supplier->get($id);  if($query->num\_rows() > 0) {  $supplier = $query->row();  $data = array(  'title' => $this->title,  'page' => 'edit',  'row' => $supplier  );  $this->template->load('\_template', 'supplier/supplier\_form', $data);  } else {  echo "<script>alert('Data tidak ditemukan');  window.location='".site\_url('supplier')."';</script>";  }  } else {  echo "<script>window.location='".site\_url('supplier')."';</script>";  }  }  public function process()  {  if(isset($\_POST['add'])) {  $data = $this->input->post(null, TRUE);  $this->supplier->add($data);  } else if(isset($\_POST['edit'])) {  $data = $this->input->post(null, TRUE);  $this->supplier->edit($data);  }  if($this->db->affected\_rows() == 1) {  echo "<script>alert('Data berhasil disimpan');  window.location='".site\_url('supplier')."';</script>";  } else {  redirect('supplier');  }  }  public function del($id = null)  {  $this->supplier->del($id);    if($this->db->affected\_rows() == 1) {  echo "<script>alert('Data berhasil dihapus');  window.location='".site\_url('supplier')."';</script>";  } else {  echo "<script>window.location='".site\_url('supplier')."';</script>";  }  }  } |

|  |
| --- |
| Controller Satuan |
| <?php  defined('BASEPATH') or exit('No direct script access allowed');  class Unit extends CI\_Controller  {  var $title = "units";  function \_\_construct()  {  parent::\_\_construct();  check\_login\_session();  $this->load->model('unit\_m', 'unit');  }  public function index()  {  $data['title'] = $this->title;  $data['row'] = $this->unit->get()->result();  $this->template->load('\_template', 'product/unit/unit\_data', $data);  }  public function add()  {  $unit = new stdClass();  $unit->unit\_id = null;  $unit->uname = null;  $data = array(  'title' => $this->title,  'page' => 'add',  'row' => $unit  );  $this->template->load('\_template', 'product/unit/unit\_form', $data);  }  public function edit($id = null)  {  if ($id != null) {  $query = $this->unit->get($id);  if ($query->num\_rows() > 0) {  $unit = $query->row();  $data = array(  'title' => $this->title,  'page' => 'edit',  'row' => $unit  );  $this->template->load('\_template', 'product/unit/unit\_form', $data);  } else {  echo "<script>alert('Data tidak ditemukan');  window.location='" . site\_url('unit') . "';</script>";  }  } else {  echo "<script>window.location='" . site\_url('unit') . "';</script>";  }  }  public function process()  {  if (isset($\_POST['add'])) {  $data = $this->input->post(null, TRUE);  $this->unit->add($data);  } else if (isset($\_POST['edit'])) {  $data = $this->input->post(null, TRUE);  $this->unit->edit($data);  }  if ($this->db->affected\_rows() == 1) {  echo "<script>alert('Data berhasil disimpan');  window.location='" . site\_url('unit') . "';</script>";  } else {  redirect('unit');  }  }  public function del($id = null)  {  $this->unit->del($id);  if ($this->db->affected\_rows() == 1) {  echo "<script>alert('Data berhasil dihapus');  window.location='" . site\_url('unit') . "';</script>";  } else {  echo "<script>window.location='" . site\_url('unit') . "';</script>";  }  }  } |

|  |
| --- |
| Controller *User* |
| <?php  defined('BASEPATH') or exit('No direct script access allowed');  class user extends CI\_Controller  {  var $title = "users";  function \_\_construct()  {  parent::\_\_construct();  check\_login\_session();  $this->load->model('user\_m', 'user');  }  public function index()  {  $data['title'] = $this->title;  $data['row'] = $this->user->get()->result();  $this->template->load('\_template', 'user/user\_data', $data);  }  public function add()  {  if (isset($\_POST['add'])) {  $this->load->library('form\_validation');  $this->form\_validation->set\_rules('name', 'Nama', 'required');  $this->form\_validation->set\_rules('username', 'Username', 'required|min\_length[5]|is\_unique[user.username]');  $this->form\_validation->set\_rules('password', 'Password', 'required|min\_length[5]');  $this->form\_validation->set\_rules('passconf', 'Konfirmasi Password', 'required|matches[password]');  $this->form\_validation->set\_rules('level', 'Level', 'required');  $this->form\_validation->set\_message('required', '{field} masih kosong, silakan diisi');  $this->form\_validation->set\_message('min\_length', '{field} minimal 5 karakter');  $this->form\_validation->set\_message('is\_unique', '%s ini sudah dipakai, ganti yang lain');  $this->form\_validation->set\_message('matches', '%s tidak sesuai dengan password');  $this->form\_validation->set\_error\_delimiters('<span class="help-block">', '</span>');  if ($this->form\_validation->run() == FALSE) {  $this->template->load('\_template', 'user/user\_form\_add', array('title' => $this->title));  } else {  $data = $this->input->post(null, TRUE);  $this->user->add($data);  if ($this->db->affected\_rows() == 1) {  echo "<script>alert('Data berhasil disimpan'); window.location='" . site\_url('user') . "';</script>";  } else {  redirect('user');  }  }  } else {  $this->template->load('\_template', 'user/user\_form\_add', array('title' => $this->title));  }  }  public function edit($id)  {  if (isset($\_POST['edit'])) {  $data = $this->input->post(null, TRUE);  $check\_username = $this->db->query("SELECT \* FROM user  WHERE username = '$data[username]'  AND user\_id != '$data[user\_id]'");  if ($check\_username->num\_rows() > 0) {  echo "<script>alert('Username ini sudah dipakai, ganti yang lain'); window.location='';</script>";  } else {  $this->user->edit();  if ($this->db->affected\_rows() == 1) {  echo "<script>alert('Data berhasil disimpan'); window.location='" . site\_url('user') . "';</script>";  } else {  redirect('user');  }  }  } else {  $query = $this->user->get($id);  if ($query->num\_rows() > 0) {  $user = $query->row();  $data = array(  'title' => $this->title,  'row' => $user  );  $this->template->load('\_template', 'user/user\_form\_edit', $data);  } else {  echo "<script>alert('Data tidak ditemukan');  window.location='" . site\_url('user') . "';</script>";  }  }  }  public function del($id = null)  {  $this->user->del($id);  if ($this->db->affected\_rows() == 1) {  echo "<script>alert('Data berhasil dihapus');  window.location='" . site\_url('user') . "';</script>";  } else {  echo "<script>window.location='" . site\_url('user') . "';</script>";  }  }  } |

|  |
| --- |
| *View csutomer\_data* |
| <section class="content-header">  <h1><?= ucwords($title) ?>  <small>Pelanggan</small>  </h1>  <ol class="breadcrumb">  <li><a><i class="fa fa-dashboard"></i></a></li>  <li class="active"><?= ucwords($title) ?></li>  </ol>  </section>  <section class="content">  <!-- <div class="row">  <div class="col-lg-12"> -->  <div class="box">  <div class="box-header">  <h3 class="box-title">Data Pelanggan</h3>  <div class="pull-right">  <a href="<?= site\_url('customer/add') ?>" class="btn btn-flat btn-primary">  <i class="fa fa-plus"></i> Tambah Pelanggan  </a>  </div>  </div>  <div class="box-body table-responsive">  <table id="table1" class="table table-bordered table-striped">  <thead>  <tr>  <th>No</th>  <th>Nama</th>  <th>Jenis Kelamin</th>  <th>No.Telepon</th>  <th>Alamat</th>  <th>Aksi</th>  </tr>  </thead>  <tbody>  <?php $no = 1;  foreach ($row as $r => $data) { ?>  <tr>  <td width="35px"><?= $no++ ?>.</td>  <td><?= $data->name ?></td>  <td><?= $data->gender ?></td>  <td><?= $data->phone ?></td>  <td><?= $data->address ?></td>  <td class="text-center" width="160px">  <a href="<?= site\_url('customer/edit/' . $data->customer\_id) ?>" class="btn btn-xs btn-primary">  <i class="fa fa-pencil"></i> Ubah  </a>  <a href="<?= site\_url('customer/del/' . $data->customer\_id) ?>" onclick="return confirm('Apakah Anda yakin?')" class="btn btn-xs btn-danger">  <i class="fa fa-trash"></i> Hapus  </a>  </td>  </tr>  <?php } ?>  </tbody>  </table>  </div>  </div>  <!-- </div>  </div> -->  </section> |

|  |
| --- |
| *View customer\_form\_data* |
| <section class="content-header">  <h1><?= ucwords($title) ?>  <small>Pelanggan</small>  </h1>  <ol class="breadcrumb">  <li><a><i class="fa fa-dashboard"></i></a></li>  <li class="active"><?= ucwords($title) ?></li>  </ol>  </section>  <section class="content">  <div class="box">  <div class="box-header">  <h3 class="box-title"><?= ucfirst($page) ?> Pelanggan</h3>  <div class="pull-right">  <a href="<?= site\_url('customer') ?>" class="btn btn-flat btn-warning btn-sm"><i class="fa fa-undo"></i> Back</a>  </div>  </div>  <div class="box-body">  <div class="row">  <div class="col-md-4 col-md-offset-4">  <form action="<?= site\_url('customer/process') ?>" method="post" autocomplete="off">  <div class="form-group">  <label for="name">Nama Pelanggan \*</label>  <input type="hidden" name="id" value="<?= $row->customer\_id ?>">  <input type="text" name="name" id="name" value="<?= $row->name ?>" class="form-control" required>  </div>  <div class="form-group">  <label for="gender">Jenis Kelamin \*</label>  <select name="gender" id="gender" class="form-control" required>  <option value="">- Pilih -</option>  <option value="Laki-laki" <?= $row->gender == 'Laki-laki' ? 'selected' : null ?>>Laki-laki</option>  <option value="Perempuan" <?= $row->gender == 'Perempuan' ? 'selected' : null ?>>Perempuan</option>  </select>  </div>  <div class="form-group">  <label for="phone">No Telepon \*</label>  <input type="number" name="phone" id="phone" value="<?= $row->phone ?>" class="form-control" required>  </div>  <div class="form-group">  <label for="address">Alamat \*</label>  <textarea name="addr" id="address" class="form-control" required><?= $row->address ?></textarea>  </div>  <div class="form-group pull-right">  <button type="submit" name="<?= $page ?>" class="btn btn-flat btn-success"><i class="fa fa-paper-plane"></i> Save</button>  <button type="reset" class="btn btn-flat">Reset</button>  </div>  </form>  </div>  </div>  </div>  </div>  </section> |

|  |
| --- |
| *View category\_data* |
| <section class="content-header">  <h1><?= ucwords($title) ?>  <small>Kategori Barang</small>  </h1>  <ol class="breadcrumb">  <li><a><i class="fa fa-dashboard"></i></a></li>  <li><a>Products</a></li>  <li class="active"><?= ucwords($title) ?></li>  </ol>  </section>  <section class="content">  <div class="box">  <div class="box-header">  <h3 class="box-title">Data Categories</h3>  <div class="pull-right">  <a href="<?= site\_url('category/add') ?>" class="btn btn-flat btn-primary">  <i class="fa fa-plus"></i> Tambah Kategori  </a>  </div>  </div>  <div class="box-body">  <table id="table1" class="table table-bordered table-striped">  <thead>  <tr>  <th>#</th>  <th>Nama</th>  <th>Aksi</th>  </tr>  </thead>  <tbody>  <?php $no = 1;  foreach ($row->result() as $r => $data) { ?>  <tr>  <td width="35px"><?= $no++ ?>.</td>  <td><?= $data->cname ?></td>  <td class="text-center" width="160px">  <a href="<?= site\_url('category/edit/' . $data->category\_id) ?>" class="btn btn-xs btn-primary">  <i class="fa fa-pencil"></i> Ubah  </a>  <a href="<?= site\_url('category/del/' . $data->category\_id) ?>" onclick="return confirm('Apakah Anda yakin?')" class="btn btn-xs btn-danger">  <i class="fa fa-trash"></i> Hapus  </a>  </td>  </tr>  <?php } ?>  </tbody>  </table>  </div>  </div>  </section> |

|  |
| --- |
| *View category\_form* |
| <section class="content-header">  <h1><?= ucwords($title) ?>  <small>Kategori Barang</small>  </h1>  <ol class="breadcrumb">  <li><a><i class="fa fa-dashboard"></i></a></li>  <li><a>Products</a></li>  <li class="active"><?= ucwords($title) ?></li>  </ol>  </section>  <section class="content">  <div class="box">  <div class="box-header">  <h3 class="box-title"><?= ucfirst($page) ?> Category</h3>  <div class="pull-right">  <a href="<?= site\_url('category') ?>" class="btn btn-flat btn-warning btn-sm"><i class="fa fa-undo"></i> Back</a>  </div>  </div>  <div class="box-body">  <div class="row">  <div class="col-md-4 col-md-offset-4">  <form action="<?= site\_url('category/process') ?>" method="post" autocomplete="off">  <div class="form-group">  <label for="name">Nama Kategori \*</label>  <input type="hidden" name="id" value="<?= $row->category\_id ?>">  <input type="text" name="cname" id="name" value="<?= $row->cname ?>" class="form-control" required>  </div>  <div class="form-group pull-right">  <button type="submit" name="<?= $page ?>" class="btn btn-flat btn-success"><i class="fa fa-paper-plane"></i> Save</button>  <button type="reset" class="btn btn-flat">Reset</button>  </div>  </form>  </div>  </div>  </div>  </div>  </section> |

|  |
| --- |
| *View item\_data* |
| <section class="content-header">  <h1><?= ucwords($title) ?>  <small>Data Barang</small>  </h1>  <ol class="breadcrumb">  <li><a><i class="fa fa-dashboard"></i></a></li>  <li><a>Products</a></li>  <li class="active"><?= ucwords($title) ?></li>  </ol>  </section>  <section class="content">  <div class="box">  <div class="box-header">  <h3 class="box-title">Data Items</h3>  <div class="pull-right">  <a href="<?= site\_url('item/add') ?>" class="btn btn-flat btn-primary">  <i class="fa fa-plus"></i> Tambah Product Item  </a>  </div>  </div>  <div class="box-body table-responsive">  <table id="table1" class="table table-bordered table-striped">  <thead>  <tr>  <th>No</th>  <th>Barcode</th>  <th>Name</th>  <th>Kategori</th>  <th>Satuan</th>  <th>Harga</th>  <th>Persediaan</th>  <th>Aksi</th>  </tr>  </thead>  <tbody>  <?php $no = 1;  foreach ($row as $r => $data) { ?>  <tr>  <td width="35px"><?= $no++ ?>.</td>  <td><?= $data->barcode ?></td>  <td><?= $data->name ?></td>  <td><?= $data->category\_name ?></td>  <td><?= $data->unit\_uname ?></td>  <td class="text-right">Rp.<?= indo\_currency($data->price) ?> </td>  <td class="text-right"><?= $data->stock ?></td>  <td class="text-center" width="160px">  <a href="<?= site\_url('item/edit/' . $data->item\_id) ?>" class="btn btn-xs btn-primary">  <i class="fa fa-pencil"></i> Ubah  </a>  <a href="<?= site\_url('item/del/' . $data->item\_id) ?>" onclick="return confirm('Apakah Anda yakin?')" class="btn btn-xs btn-danger">  <i class="fa fa-trash"></i> Hapus  </a>  </td>  </tr>  <?php } ?>  </tbody>  </table>  </div>  </div>  </section> |

|  |
| --- |
| *View form\_ item\_data* |
| <section class="content-header">  <h1><?= ucwords($title) ?>  <small>Data Barang</small>  </h1>  <ol class="breadcrumb">  <li><a><i class="fa fa-dashboard"></i></a></li>  <li><a>Products</a></li>  <li class="active"><?= ucwords($title) ?></li>  </ol>  </section>  <section class="content">  <div class="box">  <div class="box-header">  <h3 class="box-title"><?= ucfirst($page) ?> Item</h3>  <div class="pull-right">  <a href="<?= site\_url('item') ?>" class="btn btn-flat btn-warning btn-sm"><i class="fa fa-undo"></i> Back</a>  </div>  </div>  <div class="box-body">  <div class="row">  <div class="col-md-4 col-md-offset-4">  <form action="<?= site\_url('item/process') ?>" method="post" autocomplete="off">  <div class="form-group">  <label for="barcode">Kode Barang \*</label>  <input type="hidden" name="id" value="<?= $row->item\_id ?>">  <input type="text" name="barcode" id="barcode" value="<?= $row->barcode ?>" class="form-control" required>  </div>  <div class="form-group">  <label for="name">Nama Produk \*</label>  <input type="text" name="name" id="name" value="<?= $row->name ?>" class="form-control" required>  </div>  <div class="form-group">  <label for="category">Kategori \*</label>  <?php  echo form\_dropdown('category', $category, $selectedcategory, ['class' => 'form-control', 'id' => 'category', 'required' => 'required']);  ?>  </div>  <div class="form-group">  <label for="unit">Satuan \*</label>  <?php  echo form\_dropdown('unit', $unit, $selectedunit, ['class' => 'form-control', 'id' => 'unit', 'required' => 'required']);  ?>  </div>  <div class="form-group">  <label for="price">Harga \*</label>  <input type="number" name="price" id="price" value="<?= $row->price ?>" class="form-control" required>  </div>  <div class="form-group pull-right">  <button type="submit" name="<?= $page ?>" class="btn btn-flat btn-success"><i class="fa fa-paper-plane"></i> Save</button>  <button type="reset" class="btn btn-flat">Reset</button>  </div>  </form>  </div>  </div>  </div>  </div>  </section> |

|  |
| --- |
| *View* satuan\_data |
| <section class="content-header">  <h1><?= ucwords($title) ?>  <small>Satuan Barang</small>  </h1>  <ol class="breadcrumb">  <li><a><i class="fa fa-dashboard"></i></a></li>  <li><a>Products</a></li>  <li class="active"><?= ucwords($title) ?></li>  </ol>  </section>  <section class="content">  <div class="box">  <div class="box-header">  <h3 class="box-title">Data Units</h3>  <div class="pull-right">  <a href="<?= site\_url('unit/add') ?>" class="btn btn-flat btn-primary">  <i class="fa fa-plus"></i> Tambah Satuan  </a>  </div>  </div>  <div class="box-body">  <table id="table1" class="table table-bordered table-striped">  <thead>  <tr>  <th>No</th>  <th>Name</th>  <th>Aksi</th>  </tr>  </thead>  <tbody>  <?php $no = 1;  foreach ($row as $r => $data) { ?>  <tr>  <td width="35px"><?= $no++ ?>.</td>  <td><?= $data->uname ?></td>  <td class="text-center" width="160px">  <a href="<?= site\_url('unit/edit/' . $data->unit\_id) ?>" class="btn btn-xs btn-primary">  <i class="fa fa-pencil"></i> Ubah  </a>  <a href="<?= site\_url('unit/del/' . $data->unit\_id) ?>" onclick="return confirm('Apakah Anda yakin?')" class="btn btn-xs btn-danger">  <i class="fa fa-trash"></i> Hapus  </a>  </td>  </tr>  <?php } ?>  </tbody>  </table>  </div>  </div>  </section> |

|  |
| --- |
| *View* satuan\_*form* |
| <section class="content-header">  <h1><?= ucwords($title) ?>  <small>Satuan Barang</small>  </h1>  <ol class="breadcrumb">  <li><a><i class="fa fa-dashboard"></i></a></li>  <li><a>Products</a></li>  <li class="active"><?= ucwords($title) ?></li>  </ol>  </section>  <section class="content">  <div class="box">  <div class="box-header">  <h3 class="box-title"><?= ucfirst($page) ?> Unit</h3>  <div class="pull-right">  <a href="<?= site\_url('unit') ?>" class="btn btn-flat btn-warning btn-sm"><i class="fa fa-undo"></i> Back</a>  </div>  </div>  <div class="box-body">  <div class="row">  <div class="col-md-4 col-md-offset-4">  <form action="<?= site\_url('unit/process') ?>" method="post" autocomplete="off">  <div class="form-group">  <label for="name">Nama Satuan \*</label>  <input type="hidden" name="id" value="<?= $row->unit\_id ?>">  <input type="text" name="uname" id="name" value="<?= $row->uname ?>" class="form-control" required>  </div>  <div class="form-group pull-right">  <button type="submit" name="<?= $page ?>" class="btn btn-flat btn-success"><i class="fa fa-paper-plane"></i> Save</button>  <button type="reset" class="btn btn-flat">Reset</button>  </div>  </form>  </div>  </div>  </div>  </div>  </section> |

|  |
| --- |
| *View* laporan\_penjualan |
| <section class="content-header">  <h1><?= ucwords($title) ?>  <small>Laporan Penjualan</small>  </h1>  <ol class="breadcrumb">  <li><a><i class="fa fa-dashboard"></i></a></li>  <li><a>Laporan</a></li>  <li class="active"><?= ucwords($title) ?></li>  </ol>  </section>  <section class="content">  <div class="row">  <div class="col-lg-12">  <div class="box">  <div class="box-header with-border">  <h3 class="box-title">Filter Data</h3>  </div>  <div class="box-body">  <form action="<?= site\_url('report/sale') ?>" method="post">  <div class="row">  <div class="col-md-4">  <div class="form-horizontal">  <div class="form-group">  <label class="col-sm-3 control-label">Tanggal</label>  <div class="col-sm-9">  <input type="text" name="date1" id="date1" value="<?= @$post['date1'] ?>" class="form-control">  </div>  </div>  </div>  </div>  <div class="col-md-4">  <div class="form-horizontal">  <div class="form-group">  <label class="col-sm-3 control-label">s/d</label>  <div class="col-sm-9">  <input type="text" name="date2" id="date2" value="<?= @$post['date2'] ?>" class="form-control">  </div>  </div>  </div>  </div>  <div class="col-md-4">  <div class="form-horizontal">  <div class="form-group">  <label class="col-sm-3 control-label">Nota</label>  <div class="col-sm-9">  <input type="text" name="invoice" value="<?= @$post['invoice'] ?>" class="form-control">  </div>  </div>  </div>  </div>  </div>  <div class="row">  <div class="col-md-12">  <div class="pull-right">  <button type="submit" name="reset" class="btn btn-flat">  Reset  </button>  <button type="submit" name="filter" class="btn btn-info btn-flat">  <i class="fa fa-search"></i> Filter  </button>  </div>  </div>  <div>  </form>  </div>  </div>  </div>  </div>  <div class="row">  <div class="col-lg-12">  <div class="box">  <div class="box-header with-border">  <h3 class="box-title">Data Penjualan</h3>  </div>  <div class="box-body">  <table class="table table-bordered table-striped">  <thead>  <tr>  <th>No</th>  <th>Nota</th>  <th>Tanggal</th>  <th>Pelanggan</th>  <th>Total</th>  <th>Discount</th>  <th>Grand Total</th>  <th>Detail</th>  </tr>  </thead>  <tbody>  <?php $no = $this->uri->segment(3) ? $this->uri->segment(3) + 1 : 1;  foreach ($row->result() as $r => $data) { ?>  <tr>  <td width="35px"><?= $no++ ?>.</td>  <td><?= $data->invoice ?></td>  <td><?= indo\_date($data->date) ?></td>  <td><?= $data->customer\_id == null ? "Umum" : $data->customer\_name ?></td>  <td class="text-right"><?= indo\_currency($data->total\_price) ?></td>  <td class="text-right"><?= indo\_currency($data->discount) ?></td>  <td class="text-right"><?= indo\_currency($data->final\_price) ?></td>  <td class="text-center" width="200px">  <a href="<?= site\_url('sale/cetak/' . $data->sale\_id) ?>" target="\_blank" class="btn btn-xs btn-info">  <i class="fa fa-print"></i> Cetak  </a>  <a href="<?= site\_url('sale/del/' . $data->sale\_id) ?>" onclick="return confirm('Apakah Anda yakin?')" class="btn btn-xs btn-danger">  <i class="fa fa-trash"></i> Hapus  </a>  </td>  </tr>  <?php } ?>  </tbody>  </table>  </div>  <div class="box-footer clearfix">  <ul class="pagination pagination-sm no-margin pull-right">  <?php echo $pagination ?>  </ul>  </div>  </div>  </div>  </div>  </section>  <div class="modal fade" id="modal-detail">  <div class="modal-dialog">  <div class="modal-content">  <div class="modal-header">  <button type="button" class="close" data-dismiss="modal" aria-label="Close">  <span aria-hidden="true">&times;</span>  </button>  <h4 class="modal-title">Sales Report Detail</h4>  </div>  <div class="modal-body table-responsive">  <table class="table table-bordered no-margin">  <tbody>  <tr>  <th style="width:20%">Invoice</th>  <td style="width:30%"><span id="invoice"></span></td>  <th style="width:20%">Customer</th>  <td style="width:30%"><span id="cust"></span></td>  </tr>  <tr>  <th>Date Time</th>  <td><span id="datetime"></span></td>  <th>Cashier</th>  <td><span id="cashier"></span></td>  </tr>  <tr>  <th>Total</th>  <td><span id="total"></alspan>  </td>  <th>Cash</th>  <td><span id="cash"></span></td>  </tr>  <tr>  <th>Discount</th>  <td><span id="discount"></span></td>  <th>Change</th>  <td><span id="change"></span></td>  </tr>  <tr>  <th>Grand Total</th>  <td><span id="grandtotal"></span></td>  <th>Note</th>  <td><span id="note"></span></td>  </tr>  <tr>  <th>Product</th>  <td colspan="3"><span id="product"></span></td>  </tr>  </tbody>  </table>  </div>  </div>  </div>  </div>  <script>  $(document).ready(function() {  $('#customer').select2()  $('#date1, #date2').datepicker({  format: 'dd/mm/yyyy',  autoclose: true  })  $(document).on('click', '#dtl', function() {  $('#invoice').text($(this).data('invoice'));  $('#datetime').text($(this).data("date") + ' ' + $(this).data('time'));  $('#cust').text($(this).data('customer'));  $('#total').text($(this).data('total'));  $('#discount').text($(this).data('discount'));  $('#grandtotal').text($(this).data('grandtotal'));  $('#cash').text($(this).data('cash'));  $('#change').text($(this).data('change'));  $('#note').text($(this).data('note'));  $('#cashier').text($(this).data('cashier'));  var product = '<table class="table no-margin"><tr><th>Item</th><th>Price</th><th>Qty</th><th>Disc</th><th>Total</th></tr>';  $.getJSON('<?= site\_url('report/sale\_product/') ?>' + $(this).data('saleid'), function(data) {  $.each(data, function(key, val) {  product += '<tr><td style="width:30%">' + val.name + '</td><td>' + val.price + '</td><td style="width:10%">' + val.qty + '</td><td>' + val.discount\_item + '</td><td>' + val.total + '</td></tr>';  });  product += '</table>';  $('#product').html(product);  });  })  })  </script> |

|  |
| --- |
| *View* laporan\_*stock\_*barang |
| <section class="content-header">  <h1><?= ucwords($title) ?>  <small>Laporan Stock Barang</small>  </h1>  <ol class="breadcrumb">  <li><a><i class="fa fa-dashboard"></i></a></li>  <li><a>Reports</a></li>  <li class="active"><?= ucwords($title) ?></li>  </ol>  </section>  <section class="content">  <div class="row">  <div class="col-lg-12">  <div class="box">  <div class="box-header with-border">  <h3 class="box-title">Filter Data</h3>  </div>  <div class="box-body">  <form action="<?= site\_url('report/stock') ?>" method="post">  <div class="row">  <div class="col-md-6">  <div class="form-horizontal">  <div class="form-group">  <label class="col-sm-3 control-label">category</label>  <div class="col-sm-9">  <select name="kategori" id="category" class="form-control">  <option value="">- All -</option>  <option value="null" <?= @$post['kategori'] == 'null' ? 'selected' : null ?>>Umum</option>  <?php foreach ($category as $cst => $data) { ?>  <option value="<?= $data->category\_id ?>" <?= @$post['kategori'] == $data->category\_id ? 'selected' : null ?>><?= $data->cname ?></option>  <?php } ?>  </select>  </div>  </div>  </div>  </div>  <div class="col-md-6">  <div class="form-horizontal">  <div class="form-group">  <label class="col-sm-3 control-label">name</label>  <div class="col-sm-9">  <input type="text" name="name" value="<?= @$post['name'] ?>" class="form-control">  </div>  </div>  </div>  </div>  </div>  <div class="row">  <div class="col-md-12">  <div class="pull-right">  <button type="submit" name="reset" class="btn btn-flat">  Reset  </button>  <button type="submit" name="filter" class="btn btn-info btn-flat">  <i class="fa fa-search"></i> Filter  </button>  </div>  </div>  <div>  </form>  </div>  </div>  </div>  </div>  <div class="row">  <div class="col-lg-12">  <div class="box">  <div class="box-header with-border">  <h3 class="box-title">Data Penjualan</h3>  </div>  <div class="box-body">  <table class="table table-bordered table-striped">  <thead>  <tr>  <th class="text-center">No</th>  <th class="text-center">Barcode</th>  <th class="text-center">Nama Barang</th>  <th class="text-center">Satuan</th>  <th class="text-center">Kategori</th>  <th class="text-center">Stock</th>  <th class="text-center">Harga</th>  <th class="text-center">Aksi</th>  </tr>  </thead>  <tbody>  <?php $no = $this->uri->segment(3) ? $this->uri->segment(3) + 1 : 1;  foreach ($row->result() as $r => $data) { ?>  <tr>  <td width="35px"><?= $no++ ?>.</td>  <td class="text-center"><?= $data->barcode ?></td>  <td class="text-center"><?= $data->name ?></td>  <td class="text-center"><?= $data->uname ?></td>  <td><?= $data->category\_id == null ? "Umum" : $data->category\_cname ?></td>  <td class="text-center"><?= $data->stock ?></td>  <td class="text-center"><?= indo\_currency($data->price) ?></td>  <td class="text-center">  <a href="<?= site\_url('stock/cetak/' . $data->item\_id) ?>" target="\_blank" class="btn btn-xs btn-info">  <i class="fa fa-print"></i> Cetak  </a>  </td>  </tr>  <?php } ?>    </tbody>  </table>  </div>  <div class="box-footer clearfix">  <ul class="pagination pagination-sm no-margin pull-right">  <?php echo $pagination ?>  </ul>  </div>  </div>  </div>  </div>  </section>  <div class="modal fade" id="modal-detail">  <div class="modal-dialog">  <div class="modal-content">  <div class="modal-header">  <button type="button" class="close" data-dismiss="modal" aria-label="Close">  <span aria-hidden="true">&times;</span>  </button>  <h4 class="modal-title">Laporan Stock Barang</h4>  </div>  <div class="modal-body table-responsive">  <table class="table table-bordered no-margin">  <tbody>  <tr>  <th style="width:20%">name</th>  <td style="width:30%"><span id="name"></span></td>  <th style="width:20%">category</th>  <td style="width:30%"><span id="cust"></span></td>  </tr>  <tr>  <th>Total</th>  <td><span id="total"></alspan>  </td>  <th>Cash</th>  <td><span id="qty"></span></td>  </tr>  <tr>  <th>uname</th>  <td><span id="uname"></span></td>  <th>price</th>  <td><span id="price"></span></td>  </tr>  <tr>  <th>Grand Total</th>  <td><span id="cname"></span></td>  </tr>  <tr>  <th>Product</th>  <td colspan="3"><span id="product"></span></td>  </tr>  </tbody>  </table>  </div>  </div>  </div>  </div>  <script>  $(document).ready(function() {  $('#category').select2()  $('#date1, #date2').datepicker({  format: 'dd/mm/yyyy',  autoclose: true  })  $(document).on('click', '#dtl', function() {  $('#name').text($(this).data('name'));  $('#cust').text($(this).data('category'));  $('#barcode').text($(this).data('barcode'));  $('#uname').text($(this).data('uname'));  $('#cname').text($(this).data('cname'));  $('#qty').text($(this).data('qty'));  $('#price').text($(this).data('price'));  var product = '<table class="table no-margin"><tr><th>Item</th><th>Price</th><th>Qty</th><th>Disc</th><th>Total</th></tr>';  $.getJSON('<?= site\_url('report/sale\_product/') ?>' + $(this).data('saleid'), function(data) {  $.each(data, function(key, val) {  product += '<tr><td style="width:30%">' + val.name + '</td><td>' + val.price + '</td><td style="width:10%">' + val.qty + '</td><td>' + val.uname\_item + '</td><td>' + val.total + '</td></tr>';  });  product += '</table>';  $('#product').html(product);  });  })  })  </script> |

|  |
| --- |
| *View supplier\_data* |
| <section class="content-header">  <h1>Supplier  <small>Supplier Barang</small>  </h1>  <ol class="breadcrumb">  <li><a><i class="fa fa-dashboard"></i></a></li>  <li class="active"><?= ucwords($title) ?></li>  </ol>  </section>  <section class="content">  <div class="box">  <div class="box-header">  <h3 class="box-title">Data Supplier</h3>  <div class="pull-right">  <a href="<?= site\_url('supplier/add') ?>" class="btn btn-flat btn-primary">  <i class="fa fa-plus"></i> Tanmbah Supplier  </a>  </div>  </div>  <div class="box-body table-responsive">  <table id="table1" class="table table-bordered table-striped">  <thead>  <tr>  <th>#</th>  <th>Nama Supplier</th>  <th>Alamat</th>  <th>No Telepon</th>  <th>Deskripsi</th>  <th>Aksi</th>  </tr>  </thead>  <tbody>  <?php $no = 1;  foreach ($row as $r => $data) { ?>  <tr>  <td width="35px"><?= $no++ ?>.</td>  <td><?= $data->name ?></td>  <td><?= $data->address ?></td>  <td><?= $data->phone ?></td>  <td><?= $data->description ?></td>  <td class="text-center" width="160px">  <a href="<?= site\_url('supplier/edit/' . $data->supplier\_id) ?>" class="btn btn-xs btn-primary">  <i class="fa fa-pencil"></i> Ubah  </a>  <a href="<?= site\_url('supplier/del/' . $data->supplier\_id) ?>" onclick="return confirm('Apakah Anda yakin?')" class="btn btn-xs btn-danger">  <i class="fa fa-trash"></i> Hapus  </a>  </td>  </tr>  <?php } ?>  </tbody>  </table>  </div>  </div>  </section> |

|  |
| --- |
| *View supplier\_form* |
| <section class="content-header">  <h1><?= ucwords($title) ?>  <small>Supplier Barang</small>  </h1>  <ol class="breadcrumb">  <li><a><i class="fa fa-dashboard"></i></a></li>  <li class="active"><?= ucwords($title) ?></li>  </ol>  </section>  <section class="content">  <div class="box">  <div class="box-header">  <h3 class="box-title"><?= ucfirst($page) ?> Supplier</h3>  <div class="pull-right">  <a href="<?= site\_url('supplier') ?>" class="btn btn-flat btn-warning btn-sm"><i class="fa fa-undo"></i> kembali</a>  </div>  </div>  <div class="box-body">  <div class="row">  <div class="col-md-4 col-md-offset-4">  <form action="<?= site\_url('supplier/process') ?>" method="post" autocomplete="off">  <div class="form-group">  <label for="name">Nama Supplier \*</label>  <input type="hidden" name="id" value="<?= $row->supplier\_id ?>">  <input type="text" name="name" id="name" value="<?= $row->name ?>" class="form-control" required>  </div>  <div class="form-group">  <label for="phone">No Telephon \*</label>  <input type="number" name="phone" id="phone" value="<?= $row->phone ?>" class="form-control" required>  </div>  <div class="form-group">  <label for="address">Alamat \*</label>  <textarea name="addr" id="address" class="form-control" required><?= $row->address ?></textarea>  </div>  <div class="form-group">  <label for="desc">Deskripsi</label>  <textarea name="desc" id="desc" class="form-control"><?= $row->description ?></textarea>  </div>  <div class="form-group pull-right">  <button type="submit" name="<?= $page ?>" class="btn btn-flat btn-success"><i class="fa fa-paper-plane"></i> Save</button>  <button type="reset" class="btn btn-flat">Reset</button>  </div>  </form>  </div>  </div>  </div>  </div>  </section> |

|  |
| --- |
| *View print\_*laporan |
| <html moznomarginboxes mozdisallowselectionprint>  <head>  <title>Nota Penjualan</title>  <style type="text/css">  html {  font-family: "Verdana, Arial";  }  .content {  width: 80mm;  font-size: 12px;  padding: 5px;  }  .title {  text-align: center;  font-size: 13px;  padding-bottom: 5px;  border-bottom: 1px dashed;  }  .head {  margin-top: 5px;  margin-bottom: 10px;  padding-bottom: 10px;  border-bottom: 1px solid;  }  table {  width: 100%;  font-size: 12px;  }  .thanks {  margin-top: 10px;  padding-top: 10px;  text-align: center;  border-top: 1px dashed;  }  @media print {  @page {  width: 80mm;  margin: 0mm;  }  }  </style>  </head>  <body onload="window.print()">  <div class="content">  <div class="title">  <b>Toko Banguanan Satria jaya </b>  <br>  <p style="text-align: center;">Jln. Aria Wiratanudatar Kampung Sindanglaka Kabupaten Cianjur</p>  </div>  <div class="head">  <table cellspacing="0" cellpadding="0">  <tr>  <td style="width:200px">  <?php  echo Date("d/m/Y", strtotime($sale->date)) . " " . Date("H:i", strtotime($sale->sale\_created));  ?>  </td>  <td>Kasir:</td>  <td style="text-align:center; width:10px">:</td>  <td style="text-align:right">  <?= ucfirst($sale->user\_name) ?>  </td>  </tr>  <tr>  <td>  <?= $sale->invoice ?>  </td>  <td>Pelanggan</td>  <td style="text-align:center">:</td>  <td style="text-align:right">  <?= $sale->customer\_id == null ? "Umum" : $sale->customer\_name ?>  </td>  </tr>  </table>  </div>  <div class="transaction">  <table class="transaction-table" cellspacing="0" cellpadding="0">  <?php  $arr\_discount = array();  foreach ($sale\_detail as $key => $value) { ?>  <tr>  <td style="width:165px"><?= $value->name ?></td>  <td><?= $value->qty ?></td>  <td style="text-align:right; width:60px"><?= indo\_currency($value->price) ?></td>  <td style="text-align:right; width:60px"><?= indo\_currency(($value->price - $value->discount\_item) \* $value->qty) ?></td>  </tr>  <?php  if ($value->discount\_item > 0) {  $arr\_discount[] = $value->discount\_item;  }  }  foreach ($arr\_discount as $key => $value) { ?>  <tr>  <td></td>  <td colspan="2" style="text-align:right">Disc. <?= ($key + 1) ?></td>  <td style="text-align:right"><?= indo\_currency($value) ?></td>  </tr>  <?php  } ?>  <tr>  <td colspan="4" style="border-bottom:1px dashed; padding-top:5px"></td>  </tr>  <tr>  <td colspan="2"></td>  <td style="text-align:right; padding-top:5px">Sub Total</td>  <td style="text-align:right; padding-top:5px">  <?= indo\_currency($sale->total\_price) ?>  </td>  </tr>  <?php if ($sale->discount > 0) { ?>  <tr>  <td colspan="2"></td>  <td style="text-align:right; padding-bottom:5px">Disc. Sale</td>  <td style="text-align:right; padding-bottom:5px"><?= indo\_currency($sale->discount) ?></td>  </tr>  <?php  } ?>  <tr>  <td colspan="2"></td>  <td style="border-top:1px dashed; text-align:right; padding:5px 0">Grand Total</td>  <td style="border-top:1px dashed; text-align:right; padding:5px 0">  <?= $sale->final\_price ?>  </td>  </tr>  <tr>  <td colspan="2"></td>  <td style="border-top:1px dashed; text-align:right; padding-top:5px">Bayar</td>  <td style="border-top:1px dashed; text-align:right; padding-top:5px"><?= $sale->cash ?></td>  </tr>  <tr>  <td colspan="2"></td>  <td style="text-align:right">Kembali</td>  <td style="text-align:right"><?= $sale->change ?></td>  </tr>  </table>  </div>  <div class="thanks">  Hormat Kami  Satria Jaya  </div>  </div>  </body>  </html> |

|  |
| --- |
| *View form\_*penjualan |
| <section class="content-header">  <h1><?=ucwords($title)?>  <small>Penjualan</small>  </h1>  <ol class="breadcrumb">  <li><a><i class="fa fa-dashboard"></i></a></li>  <li><a>Transaction</a></li>  <li class="active"><?=ucwords($title)?></li>  </ol>  </section>  <section class="content">  <div class="row">  <div class="col-lg-0">    <table width="0%">  <style>  p {  font-size: 50px;  min-height: 0px;  margin-left: 370px;  }  </style>  <div class="row">  <div class="col-lg-9">  <p style="font-size: 35px;text-align:center; margin-left:35%; margin-top:20px">Selamat Datang Di System Kasir <?= $this->fungsi->user\_login()->name ?> </p>  </div>  </div>  <tr>  <td style="vertical-align:top">  <!-- <label for="date">Date</label> -->  </td>  <td>  <div class="form-group">  <input type="hidden" id="date" value="<?=date('Y-m-d')?>" class="form-control">  </div>  </td>  </tr>  <tr>  <td style="vertical-align:top; width:30%">  <!-- <label for="user">Kasir</label> -->  </td>  <td>  <div class="form-group">  <input type="hidden" id="user" value="<?=$this->fungsi->user\_login()->name?>" class="form-control" readonly>  </div>  </td>  </tr>    </table>    </div>  <div class="col-lg-7">  <div class="box box-widget">  <div class="box-body">  <div class="col-lg-7">  <div class="box box-widget">  <div class="box-body">  <div align="right">  <h4>Nota <b><span id="invoice"><?= $invoice ?></span></b></h4>  <h1><b><span id="grand\_total2" style="font-size:50pt">0</span></b></h1>  </div>  </div>  </div>  </div>  <table width="100%">  <tr>  <td style="vertical-align:top; width:30%">  <label for="barcode">Barcode</label>  </td>  <td>  <div class="form-group input-group">  <input type="hidden" id="item\_id">  <input type="hidden" id="price">  <input type="hidden" id="stock">  <input type="text" id="barcode" class="form-control" autofocus>  <span class="input-group-btn">  <button type="button" class="btn btn-info btn-flat" data-toggle="modal" data-target="#modal-item">  <i class="fa fa-search"></i>  </button>  </span>  </div>  </td>  </tr>  <tr>  <td style="vertical-align:top">  <label for="qty">Qty</label>  </td>  <td>  <div class="form-group">  <input type="number" id="qty" value="1" min="1" class="form-control">  </div>  </td>  </tr>  <tr>  <td style="vertical-align:top">  <label for="customer">Customer</label>  </td>  <td>  <div>  <select id="customer" class="form-control">  <option value="">Umum</option>  <?php foreach($customer as $cst => $data) {  echo '<option value="'.$data->customer\_id.'">'.$data->name.'</option>';  } ?>  </select>  </div>  </td>  </tr>  <tr>  <td></td>  <td>  <br>  <div>  <button type="button" id="add\_cart" class="btn btn-primary">  <i class="fa fa-cart-plus"></i> Add  </button>  </div>  </td>  </tr>  </table>  </div>  </div>  </div>    <div class="col-lg-5">  <div class="box box-widget">  <div class="box-body">  <tr>  <td style="vertical-align:top; width:30%">  <label for="cash">Pembayaran</label>  </td>  <td>  <div class="form-group">  <input type="number" id="cash" value="0" min="0" class="form-control">  </div>  </td>  </tr>  </div>  </div>  </div>  <div class="row">  <div class="col-lg-5">  <div class="box box-widget">  <div class="box-body">  <table width="100%">  <tr>  <td style="vertical-align:top">  <label for="change">Kembalian</label>  </td>  <td>  <div>  <input type="number" id="change" class="form-control" readonly>  </div>  <br>  </td>  </tr>  <tr>  <td style="vertical-align:top">  <label for="discount">Potongan Harga</label>  </td>  <td>  <div class="form-group">  <input type="number" id="discount" value="0" min="0" class="form-control">  </div>  </td>  </tr>  </table>  </div>  </div>  </div>  <div class="col-lg-5">  <div class="box box-widget">  <div class="box-body">  <tr>  <td style="vertical-align:top">  <label for="note">Catatan</label>  </td>  <td>  <div>  <textarea id="note" rows="3" class="form-control"></textarea>  </div>  </td>  </tr>  </div>  </div>  </div>  </div>  </div>  <div class="row">  <div class="col-lg-12">  <div class="box box-widget">  <div class="box-body table-responsive">  <table class="table table-bordered table-striped">  <thead>  <tr>  <th>#</th>  <th>Barcode</th>  <th>Product Item</th>  <th>Price</th>  <th>Qty</th>    <th width="15%">Total</th>  <th>Actions</th>  </tr>  </thead>  <tbody id="cart\_table">  <?php $this->view('transaction/sale/cart\_data'); ?>    <?php $subtotal = 0;  foreach ($cart->result() as $c => $data) {  $subtotal += $data->total;  } ?>  </tbody>  </table>  </div>  </div>  </div>  </div>  <div class="row">  <div class="col-lg-3">  <div>  <div>  <table width="100%">  <tr>  <!-- <td style="vertical-align:top; width:30%">  <label for="sub\_total">Sub Total</label>  </td> -->  <td>  <div class="form-group">  <input type="hidden" id="sub\_total" value="<?= $subtotal ?>" class="form-control" readonly>  </div>  </td>  </tr>  <tr>  <!-- <td style="vertical-align:top">  <label for="grand\_total">Grand Total</label>  </td> -->  <td>  <div class="form-group">  <input type="hidden" id="grand\_total" class="form-control" readonly>  </div>  </td>  </tr>  </table>  </div>  </div>  </div>  <div class="col-lg-7">  <div>  <div>  <table width="100%">  </table>  </div>  </div>  </div>  <div class="col-lg-2">  <div>  <button id="cancel\_payment" class="btn btn-flat btn-warning">  <i class="fa fa-refresh"></i> Cancel  </button><br><br>  <button id="process\_payment" class="btn btn-flat btn-lg btn-success">  <i class="fa fa-paper-plane-o"></i> Process Payment  </button>  </div>  </div>  </div>    </section>  <div class="modal fade" id="modal-item">  <div class="modal-dialog">  <div class="modal-content">  <div class="modal-header">  <button type="button" class="close" data-dismiss="modal" aria-label="Close">  <span aria-hidden="true">&times;</span>  </button>  <h4 class="modal-title">Add Product Item</h4>  </div>  <div class="modal-body table-responsive">  <table id="table1" class="table table-bordered table-striped">  <thead>  <tr>  <th>Barcode</th>  <th>Name</th>  <th>Unit</th>  <th>Price</th>  <th>Stock</th>  <th>Actions</th>  </tr>  </thead>  <tbody>  <?php  foreach ($item as $i => $data) { ?>  <tr>  <td><?=$data->barcode?></td>  <td><?=$data->name?></td>  <td><?=$data->unit\_uname?></td>  <td class="text-right"><?=indo\_currency($data->price)?> IDR</td>  <td class="text-right"><?=$data->stock?></td>  <td class="text-center">  <button id="select"  data-barcode="<?=$data->barcode?>"  data-itemid="<?=$data->item\_id?>"  data-price="<?=$data->price?>"  data-stock="<?=$data->stock?>"  class="btn btn-xs btn-info">  <i class="fa fa-check"></i> Select  </button>  </td>  </tr>  <?php } ?>  </tbody>  </table>  </div>  </div>  </div>  </div>  <div class="modal fade" id="modal-item-edit">  <div class="modal-dialog modal-sm">  <div class="modal-content">  <div class="modal-header">  <button type="button" class="close" data-dismiss="modal" aria-label="Close">  <span aria-hidden="true">&times;</span>  </button>  <h4 class="modal-title">Update Product Item</h4>  </div>  <div class="modal-body">  <div class="form-group">  <label for="product\_item">Product Item</label>  <input type="hidden" id="cartid\_item">  <input type="text" id="barcode\_item" class="form-control" readonly>  </div>  <div class="form-group">  <input type="text" id="product\_item" class="form-control" readonly>  </div>  <div class="form-group">  <label for="price\_item">Price</label>  <input type="number" id="price\_item" min="0" class="form-control">  </div>  <div class="form-group">  <label for="qty\_item">Qty</label>  <input type="number" id="qty\_item" min="1" class="form-control">  </div>  <div class="form-group">  <!-- <label for="discount\_item">Discount Item</label> -->  <input type="hidden"" id="discount\_item" min="0" class="form-control">  </div>  <div class="form-group">  <label for="total\_item">Total</label>  <input type="number" id="total\_item" class="form-control" readonly>  </div>  </div>  <div class="modal-footer">  <div class="pull-right">  <button type="button" id="edit\_cart" class="btn btn-flat btn-success"><i class="fa fa-paper-plane"></i> Save</button>  </div>  </div>  </div>  </div>  </div>  <script>  function count() {  var subtotal = 0  $('#cart\_table tr').each(function() {  subtotal += parseInt($(this).find('#total').text())  })  isNaN(subtotal) ? $("#sub\_total").val(0) : $("#sub\_total").val(subtotal)  var discount = $('#discount').val()  var grand\_total = subtotal - discount  if(isNaN(grand\_total)) {  $('#grand\_total').val(0)  $('#grand\_total2').text(0)  } else {  $('#grand\_total').val(grand\_total)  $('#grand\_total2').text(grand\_total)  }  var cash = $('#cash').val()  cash != 0 ? $('#change').val(cash - grand\_total) : $('#change').val(0)  }  $(document).ready(function() {  $('#customer').select2()  count()  })  $(document).on('keyup mouseup', '#discount, #cash', function() {  count()  })  // choose item  $(document).on('click', '#select', function() {  var barcode = $(this).data('barcode')  $('#barcode').val(barcode)  $('#item\_id').val($(this).data('itemid'))  $('#price').val($(this).data('price'))  $('#stock').val($(this).data('stock'))  $('#modal-item').modal('hide')  })  // add item to cart table  $(document).on('click', '#add\_cart', function() {  var item\_id = $('#item\_id').val()  var stock = $('#stock').val()  var price = $('#price').val()  var qty = $('#qty').val()  if(item\_id == '') {  alert('Product belum dipilih')  $('#barcode').focus()  } else if(stock < 1) {  alert('Stock tidak mencukupi')  $('#item\_id').val('')  $('#barcode').val('')  $('#barcode').focus()  } else {  $.ajax({  type: 'POST',  url: '<?=site\_url('sale/process')?>',  data: {'add\_cart': true, 'item\_id': item\_id, 'price': price, 'qty': qty},  dataType: 'json',  success: function(result) {  if(result.success == true) {  $('#cart\_table').load('<?=site\_url('sale/cart\_data')?>', function() {  count()  })  $('#item\_id').val('')  $('#barcode').val('')  $('#barcode').focus()  } else {  alert('Gagal tambah item cart')  }  }  })  }  })  // del cart item  $(document).on('click', '#del\_cart', function() {  if(confirm('Apakah Anda yakin?')) {  var cart\_id = $(this).data('cartid')  $.ajax({  type: 'POST',  url: '<?=site\_url('sale/cart\_del')?>',  data: {'cart\_id': cart\_id},  dataType: 'json',  success: function(result) {  if(result.success == true) {  $('#cart\_table').load('<?=site\_url('sale/cart\_data')?>', function() {  count()  })  } else {  alert('Gagal hapus item cart')  }  }  })  }  })  // on modal cart  $(document).on('click', '#update\_cart', function() {  $('#cartid\_item').val($(this).data('cartid'))  $('#barcode\_item').val($(this).data('barcode'))  $('#product\_item').val($(this).data('product'))  $('#price\_item').val($(this).data('price'))  $('#qty\_item').val($(this).data('qty'))  $('#discount\_item').val($(this).data('discount'))  $('#total\_item').val($(this).data('total'))  })  function count\_on\_modal() {  var price = $('#price\_item').val()  var qty = $('#qty\_item').val()  var discount = $('#discount\_item').val()  total = (price - discount) \* qty  $('#total\_item').val(total)  }  // edit cart item  $(document).on('click', '#edit\_cart', function() {  var cart\_id = $('#cartid\_item').val()  var price = $('#price\_item').val()  var qty = $('#qty\_item').val()  var discount = $('#discount\_item').val()  var total = $('#total\_item').val()  if(price == '') {  alert('Harga tidak boleh kosong')  $('#price\_item').focus()  } else if(qty == '') {  alert('Qty tidak boleh kosong')  $('#qty\_item').focus()  } else if(discount == '') {  alert('Diskon tidak boleh kosong')  $('#discount\_item').focus()  } else {  $.ajax({  type: 'POST',  url: '<?=site\_url('sale/process')?>',  data: {'edit\_cart': true, 'cart\_id': cart\_id, 'price': price, 'qty': qty, 'discount': discount, 'total': total},  dataType: 'json',  success: function(result) {  if(result.success == true) {  $('#cart\_table').load('<?=site\_url('sale/cart\_data')?>', function() {  count()  })  $('#modal-item-edit').modal('hide')  } else {  alert('Gagal update item cart')  }  }  })  }  })  $(document).on('keyup mouseup', '#price\_item, #qty\_item, #discount\_item', function() {  count\_on\_modal()  })  // cancel payment  $(document).on('click', '#cancel\_payment', function() {  if(confirm('Apakah Anda yakin?')) {  $.ajax({  type: 'POST',  url: '<?=site\_url('sale/cart\_del')?>',  dataType: 'json',  success: function(result) {  if(result.success == true) {  $('#cart\_table').load('<?=site\_url('sale/cart\_data')?>', function() {  count()  })  }  }  })  $('#discount').val(0)  $('#cash').val(0)  $('#customer').val(0).change()  $('#barcode').val('')  $('#barcode').focus()  }  })  // process payment  $(document).on('click', '#process\_payment', function() {  if($('#sub\_total').val() < 1) {  alert('Belum ada product item yang dipilih')  $('#barcode').focus()  } else if($('#cash').val() < 1) {  alert('Jumlah uang cash belum diinput')  $('#cash').focus()  } else {  if(confirm('Yakin proses transaksi ini?')) {  var customer\_id = $('#customer').val()  var subtotal = $('#sub\_total').val()  var discount = $('#discount').val()  var grandtotal = $('#grand\_total').val()  var cash = $('#cash').val()  var change = $('#change').val()  var note = $('#note').val()  var date = $('#date').val()  $.ajax({  type: 'POST',  url: '<?=site\_url('sale/process')?>',  data: {'process\_payment': true, 'customer\_id': customer\_id, 'subtotal': subtotal, 'discount': discount, 'grandtotal': grandtotal, 'cash': cash, 'change': change, 'note': note, 'date': date},  dataType: 'json',  success: function(result) {  if(result.success == true) {  alert('Transaksi berhasil')  window.open('<?=site\_url('sale/cetak/')?>'+result.sale\_id, '\_blank')  } else {  alert('Transaksi gagal')  }  location.href='<?=site\_url('sale')?>'  }  })  }  }  })  </script> |

|  |
| --- |
| *View form\_print\_stock* |
| <!DOCTYPE html>  <html>  <head>  <strong>  <p style="margin-bottom: 15px;font-size:30px;text-align:center;"> Toko Bangunan Satria Jaya</p>  </strong>  <p style="text-align: center;">Jln. Aria Wiratanudatar Kampung Sindanglaka Kabupaten Cianjur</p>  <hr>  <p style="margin-left:65%">Cianjur <?php echo format\_indo(date('Y-m-d')); ?>  </p>  <p style="margin-left:55px;font-size:20px">Laporan Persediaan Barang</p>  <style type="text/css">  #outtable {  padding: 20px;  border: 1px solid #e3e3e3;  width: 600px;  border-radius: 5px;  margin-left: 50px;  font-size: 15px;  padding: 5px;  }  .short {  width: 50px;  }  .normal {  width: 150px;  }  table {  border-collapse: collapse;  font-family: arial;  color: #5E5B5C;  text-align: center;  }  thead th {  text-align: left;  padding: 10px;  }  tbody td {  border-top: 1px solid #e3e3e3;  padding: 10px;  }  tbody tr:nth-child(even) {  background: #F6F5FA;  }  tbody tr:hover {  background: #EAE9F5  }  </style>  </head>  <body onload="window.print()">  <div id="outtable">  <table>  <thead>  <tr>  <th class="short">No</th>  <th class="normal">Nama Barang</th>  <th class="normal">Satuan</th>  <th class="normal">Kategori</th>  <th class="normal">Harga</th>  <th class="normal">Stock</th>  </tr>  </thead>  <tbody>  <?php  $no = 1;  foreach ($stock as $s) : ?>  <tr>  <td><?= $no++ ?></td>  <td><?= $s->name ?></td>  <td><?= $s->uname ?></td>  <td><?= $s->cname ?></td>  <td><?= $s->price ?></td>  <td><?= $s->stock ?></td>  </tr>  <?php ?>  <?php endforeach; ?>  </tbody>  </table>  </div>  </body>  </html> |

|  |
| --- |
| *View stock\_in\_data* |
| <!DOCTYPE html>  <html>  <head>  <strong>  <p style="margin-bottom: 15px;font-size:30px;text-align:center;"> Toko Bangunan Satria Jaya</p>  </strong>  <p style="text-align: center;">Jln. Aria Wiratanudatar Kampung Sindanglaka Kabupaten Cianjur</p>  <hr>  <p style="margin-left:65%">Cianjur <?php echo format\_indo(date('Y-m-d')); ?>  </p>  <p style="margin-left:55px;font-size:20px">Laporan Persediaan Barang</p>  <style type="text/css">  #outtable {  padding: 20px;  border: 1px solid #e3e3e3;  width: 600px;  border-radius: 5px;  margin-left: 50px;  font-size: 15px;  padding: 5px;  }  .short {  width: 50px;  }  .normal {  width: 150px;  }  table {  border-collapse: collapse;  font-family: arial;  color: #5E5B5C;  text-align: center;  }  thead th {  <section class="content-header">  <h1><?= ucwords($title) ?>  <small>Barang Masuk / Pembelian</small>  </h1>  <ol class="breadcrumb">  <li><a><i class="fa fa-dashboard"></i></a></li>  <li><a>Transaction</a></li>  <li class="active"><?= ucwords($title) ?></li>  </ol>  </section>  <section class="content">  <div class="box">  <div class="box-header">  <h3 class="box-title">Data Stock In</h3>  <div class="pull-right">  <a href="<?= site\_url('stock/in/add') ?>" class="btn btn-flat btn-primary">  <i class="fa fa-plus"></i> Tambah barang masuk  </a>  </div>  </div>  <div class="box-body">  <table id="table1" class="table table-bordered table-striped">  <thead>  <tr>  <th>No</th>  <th>Kode Barang</th>  <th>Product Item</th>  <th>Jumlah</th>  <th>Tanggal</th>  <th>Aksi</th>  </tr>  </thead>  <tbody>  <?php $no = 1;  foreach ($row as $r => $data) { ?>  <tr>  <td width="35px"><?= $no++ ?>.</td>  <td><?= $data->barcode ?></td>  <td><?= $data->item\_name ?></td>  <td class="text-center"><?= $data->qty ?></td>  <td class="text-center"><?= indo\_date($data->date) ?></td>  <td class="text-center" width="160px">  <a href="<?= site\_url('stock/in/del/' . $data->item\_id . '/' . $data->stock\_id) ?>" onclick="return confirm('Apakah Anda yakin?')" class="btn btn-xs btn-danger">  <i class="fa fa-trash"></i> Hapus  </a>  </td>  </tr>  <?php } ?>  </tbody>  </table>  </div>  </div>  </section>  <div class="modal fade" id="modal-detail">  <div class="modal-dialog modal-sm">  <div class="modal-content">  <div class="modal-header">  <button type="button" class="close" data-dismiss="modal" aria-label="Close">  <span aria-hidden="true">&times;</span>  </button>  <h4 class="modal-title">Stock In Detail</h4>  </div>  <div class="modal-body table-responsive">  <table class="table table-bordered no-margin">  <tbody>  <tr>  <th style="width:35%">Kode Barang</th>  <td><span id="barcode"></span></td>  </tr>  <tr>  <th>Item Name</th>  <td><span id="item\_name"></span></td>  </tr>  <tr>  <th>Detail</th>  <td><span id="detail"></span></td>  </tr>  <tr>  <th>Supplier</th>  <td><span id="supplier\_name"></span></td>  </tr>  <tr>  <th>Jumlah</th>  <td><span id="qty"></span></td>  </tr>  <tr>  <th>Tanggal</th>  <td><span id="date"></span></td>  </tr>  </tbody>  </table>  </div>  </div>  </div>  </div>  <script>  $(document).ready(function() {  $(document).on("click", "#dtl", function() {  var barcode = $(this).data("barcode");  var item\_name = $(this).data("itemname");  var detail = $(this).data("detail");  var supplier\_name = $(this).data("suppliername");  var qty = $(this).data("qty");  var date = $(this).data("date");  $("#barcode").text(barcode);  $("#item\_name").text(item\_name);  $("#detail").text(detail);  $("#supplier\_name").text(supplier\_name);  $("#qty").text(qty);  $("#date").text(date);  })  })  </script> |

|  |
| --- |
| *View stock\_in\_form* |
| <section class="content-header">  <h1><?= ucwords($title) ?>  <small>Barang Masuk / Pembelian</small>  </h1>  <ol class="breadcrumb">  <li><a><i class="fa fa-dashboard"></i></a></li>  <li><a>Transaction</a></li>  <li class="active"><?= ucwords($title) ?></li>  </ol>  </section>  <section class="content">  <div class="box">  <div class="box-header">  <h3 class="box-title">Add Stock In</h3>  <div class="pull-right">  <a href="<?= site\_url('stock/in') ?>" class="btn btn-flat btn-warning btn-sm"><i class="fa fa-undo"></i> Back</a>  </div>  </div>  <div class="box-body">  <div class="row">  <div class="col-md-4 col-md-offset-4">  <form action="<?= site\_url('stock/process') ?>" method="post" autocomplete="off">  <div class="form-group">  <label for="date">Tanggal \*</label>  <input type="date" name="date" id="date" value="<?= date('Y-m-d') ?>" class="form-control" required>  </div>  <div>  <label for="barcode">Kode Barang \*</label>  </div>  <div class="form-group input-group">  <input type="hidden" name="item\_id" id="item\_id">  <input type="text" name="barcode" id="barcode" class="form-control" required autofocus>  <span class="input-group-btn">  <button type="button" class="btn btn-info btn-flat" data-toggle="modal" data-target="#modal-item">  <i class="fa fa-search"></i>  </button>  </span>  </div>  <div class="form-group">  <label for="item\_name">Nama Item</label>  <input type="text" name="item\_name" id="item\_name" value="-" class="form-control" readonly>  </div>  <div class="form-group">  <div class="row">  <div class="col-md-8">  <label for="unit\_name">Satuan</label>  <input type="text" name="unit\_name" id="unit\_name" value="-" class="form-control" readonly>  </div>  <div class="col-md-4">  <label for="stock">Stock Awal</label>  <input type="text" name="stock" id="stock" value="-" class="form-control" readonly>  </div>  </div>  </div>  <div class="form-group">  <label for="detail">Detail \*</label>  <input type="text" name="detail" id="detail" class="form-control" placeholder="ex. Kulakan / Tambahan / etc" required>  </div>  <div class="form-group">  <label for="supplier">Supplier</label>  <select name="supplier" id="supplier" class="form-control">  <option value="">- Pilih -</option>  <?php  foreach ($supplier as $spl => $data) {  echo '<option value="' . $data->supplier\_id . '">' . $data->name . '</option>';  } ?>  </select>  </div>  <div class="form-group">  <label for="qty">Jumlah \*</label>  <input type="number" name="qty" id="qty" min="1" class="form-control" required>  </div>  <div class="form-group pull-right">  <button type="submit" name="<?= $page ?>" class="btn btn-flat btn-success"><i class="fa fa-paper-plane"></i> Save</button>  <button type="reset" class="btn btn-flat">Reset</button>  </div>  </form>  </div>  </div>  </div>  </div>  </section>  <div class="modal fade" id="modal-item">  <div class="modal-dialog">  <div class="modal-content">  <div class="modal-header">  <button type="button" class="close" data-dismiss="modal" aria-label="Close">  <span aria-hidden="true">&times;</span>  </button>  <h4 class="modal-title">Tambah Barang</h4>  </div>  <div class="modal-body table-responsive">  <table id="table1" class="table table-bordered table-striped">  <thead>  <tr>  <th>Kode Barang</th>  <th>Nama</th>  <th>Satuan</th>  <th>Harga</th>  <th>Persediaan</th>  <th>Aksi</th>  </tr>  </thead>  <tbody>  <?php  foreach ($item as $i => $data) { ?>  <tr>  <td><?= $data->barcode ?></td>  <td><?= $data->name ?></td>  <td><?= $data->unit\_uname ?></td>  <td><?= $data->price ?></td>  <td><?= $data->stock ?></td>  <td class="text-center">  <button id="select" data-id="<?= $data->item\_id ?>" data-barcode="<?= $data->barcode ?>" data-name="<?= $data->name ?>" data-unit="<?= $data->unit\_uname ?>" data-stock="<?= $data->stock ?>" class="btn btn-xs btn-info">  <i class="fa fa-check"></i> Select  </button>  </td>  </tr>  <?php } ?>  </tbody>  </table>  </div>  </div>  </div>  </div>  <script>  $(document).ready(function() {  $('#supplier').select2();  $(document).on('click', '#select', function() {  var item\_id = $(this).data('id');  var barcode = $(this).data('barcode');  var name = $(this).data('name');  var unit\_name = $(this).data('unit');  var stock = $(this).data('stock');  $('#item\_id').val(item\_id);  $('#barcode').val(barcode);  $('#item\_name').val(name);  $('#unit\_name').val(unit\_name);  $('#stock').val(stock);  $('#modal-item').modal('hide');  });  })  </script> |

|  |
| --- |
| *View stock\_out\_data* |
| <section class="content-header">  <h1><?= ucwords($title) ?>  <small>Barang Keluar</small>  </h1>  <ol class="breadcrumb">  <li><a><i class="fa fa-dashboard"></i></a></li>  <li><a>Transaction</a></li>  <li class="active"><?= ucwords($title) ?></li>  </ol>  </section>  <section class="content">  <div class="box">  <div class="box-header">  <h3 class="box-title">Data Stock Out</h3>  <div class="pull-right">  <a href="<?= site\_url('stock/out/add') ?>" class="btn btn-flat btn-primary">  <i class="fa fa-plus"></i> Tambah Barang Keluar  </a>  </div>  </div>  <div class="box-body">  <table id="table1" class="table table-bordered table-striped">  <thead>  <tr>  <th>No</th>  <th>Kode Barang</th>  <th>Product Item</th>  <th>Jumlah</th>  <th>Detail</th>  <th>Tanggal</th>  <th>Aksi</th>  </tr>  </thead>  <tbody>  <?php $no = 1;  foreach ($row as $r => $data) { ?>  <tr>  <td width="35px"><?= $no++ ?>.</td>  <td><?= $data->barcode ?></td>  <td><?= $data->name ?></td>  <td class="text-right"><?= $data->qty ?></td>  <td><?= $data->detail ?></td>  <td class="text-center"><?= indo\_date($data->date) ?></td>  <td class="text-center" width="160px">  <a href="<?= site\_url('stock/out/del/' . $data->item\_id . '/' . $data->stock\_id) ?>" onclick="return confirm('Apakah Anda yakin?')" class="btn btn-xs btn-danger">  <i class="fa fa-trash"></i> Hapus  </a>  </td>  </tr>  <?php } ?>  </tbody>  </table>  </div>  </div>  </section> |

|  |
| --- |
| *View stock\_out\_form* |
| <section class="content-header">  <h1><?= ucwords($title) ?>  <small>Barang Keluar</small>  </h1>  <ol class="breadcrumb">  <li><a><i class="fa fa-dashboard"></i></a></li>  <li><a>Transaction</a></li>  <li class="active"><?= ucwords($title) ?></li>  </ol>  </section>  <section class="content">  <div class="box">  <div class="box-header">  <h3 class="box-title">Add Stock Out</h3>  <div class="pull-right">  <a href="<?= site\_url('stock/out') ?>" class="btn btn-flat btn-warning btn-sm"><i class="fa fa-undo"></i> Back</a>  </div>  </div>  <div class="box-body">  <div class="row">  <div class="col-md-4 col-md-offset-4">  <form action="<?= site\_url('stock/process') ?>" method="post" autocomplete="off">  <div class="form-group">  <label for="date">Tanggal \*</label>  <input type="date" name="date" id="date" value="<?= date('Y-m-d') ?>" class="form-control" required>  </div>  <div>  <label for="barcode">Kode Barang \*</label>  </div>  <div class="form-group input-group">  <input type="hidden" name="item\_id" id="item\_id">  <input type="text" name="barcode" id="barcode" class="form-control" required autofocus>  <span class="input-group-btn">  <button type="button" class="btn btn-info btn-flat" data-toggle="modal" data-target="#modal-item">  <i class="fa fa-search"></i>  </button>  </span>  </div>  <div class="form-group">  <label for="item\_name">Nama Item</label>  <input type="text" name="item\_name" id="item\_name" value="-" class="form-control" readonly>  </div>  <div class="form-group">  <div class="row">  <div class="col-md-8">  <label for="unit\_name">Satuan</label>  <input type="text" name="unit\_name" id="unit\_name" value="-" class="form-control" readonly>  </div>  <div class="col-md-4">  <label for="stock">Stock Awal</label>  <input type="text" name="stock" id="stock" value="-" class="form-control" readonly>  </div>  </div>  </div>  <div class="form-group">  <label for="detail">Detail \*</label>  <input type="text" name="detail" id="detail" class="form-control" placeholder="ex. Hilang / Rusak / Kedaluwarsa / etc" required>  </div>  <div class="form-group">  <label for="qty">Jumlah \*</label>  <input type="number" name="qty" id="qty" min="1" class="form-control" required>  </div>  <div class="form-group pull-right">  <button type="submit" name="<?= $page ?>" class="btn btn-flat btn-success"><i class="fa fa-paper-plane"></i> Save</button>  <button type="reset" class="btn btn-flat">Reset</button>  </div>  </form>  </div>  </div>  </div>  </div>  </section>  <div class="modal fade" id="modal-item">  <div class="modal-dialog">  <div class="modal-content">  <div class="modal-header">  <button type="button" class="close" data-dismiss="modal" aria-label="Close">  <span aria-hidden="true">&times;</span>  </button>  <h4 class="modal-title">Add Product Item</h4>  </div>  <div class="modal-body table-responsive">  <table id="table1" class="table table-bordered table-striped">  <thead>  <tr>  <th>Kode Barang</th>  <th>Nama Barang</th>  <th>Satuan</th>  <th>Harga</th>  <th>Persediaan</th>  <th>Aksi</th>  </tr>  </thead>  <tbody>  <?php  foreach ($item as $i => $data) { ?>  <tr>  <td><?= $data->barcode ?></td>  <td><?= $data->name ?></td>  <td><?= $data->unit\_uname ?></td>  <td><?= $data->price ?></td>  <td><?= $data->stock ?></td>  <td class="text-center">  <button id="select" data-id="<?= $data->item\_id ?>" data-barcode="<?= $data->barcode ?>" data-name="<?= $data->name ?>" data-unit="<?= $data->unit\_uname ?>" data-stock="<?= $data->stock ?>" class="btn btn-xs btn-info">  <i class="fa fa-check"></i> Select  </button>  </td>  </tr>  <?php } ?>  </tbody>  </table>  </div>  </div>  </div>  </div>  <script>  $(document).ready(function() {  $(document).on('click', '#select', function() {  var item\_id = $(this).data('id');  var barcode = $(this).data('barcode');  var name = $(this).data('name');  var unit\_name = $(this).data('unit');  var stock = $(this).data("stock");  $('#item\_id').val(item\_id);  $('#barcode').val(barcode);  $('#item\_name').val(name);  $('#unit\_name').val(unit\_name);  $('#stock').val(stock);  $('#modal-item').modal('hide');  });  })  </script> |

|  |
| --- |
| *View user\_data* |
| <section class="content-header">  <h1><?= ucwords($title) ?>  <small>Pengguna / Karyawan</small>  </h1>  <ol class="breadcrumb">  <li><a><i class="fa fa-dashboard"></i></a></li>  <li class="active"><?= ucwords($title) ?></li>  </ol>  </section>  <section class="content">  <div class="box">  <div class="box-header">  <h3 class="box-title">Data users</h3>  <div class="pull-right">  <a href="<?= site\_url('user/add') ?>" class="btn btn-flat btn-primary">  <i class="fa fa-user-plus"></i> Tambah Pengguna  </a>  </div>  </div>  <div class="box-body table-responsive">  <table id="table1" class="table table-bordered table-striped">  <thead>  <tr>  <th>No</th>  <th>Username</th>  <th>Nama</th>  <th>Alamat</th>  <th>Level</th>  <th>Aksi</th>  </tr>  </thead>  <tbody>  <?php $no = 1;  foreach ($row as $r => $data) { ?>  <tr>  <td width="35px"><?= $no++ ?>.</td>  <td><?= $data->username ?></td>  <td><?= $data->name ?></td>  <td><?= $data->address ?></td>  <td><?= $data->level == 1 ? '<span class="label label-default">Admin</span>' : '<span class="label label-default">Kasir</span>' ?></td>  <td class="text-center" width="160px">  <a href="<?= site\_url('user/edit/' . $data->user\_id) ?>" class="btn btn-xs btn-primary">  <i class="fa fa-pencil"></i> Ubah  </a>  <?php if ($data->user\_id != $this->session->userdata('userid')) { ?>  <a href="<?= site\_url('user/del/' . $data->user\_id) ?>" onclick="return confirm('Apakah Anda yakin?')" class="btn btn-xs btn-danger">  <i class="fa fa-trash"></i> Hapus  </a>  <?php  } ?>  </td>  </tr>  <?php } ?>  </tbody>  </table>  </div>  </div>  </section> |

|  |
| --- |
| *View user\_form\_add* |
| <section class="content-header">  <h1>Users  <small>Pengguna / Karyawan</small>  </h1>  <ol class="breadcrumb">  <li><a><i class="fa fa-dashboard"></i></a></li>  <li class="active">Users</li>  </ol>  </section>  <section class="content">  <div class="box">  <div class="box-header">  <h3 class="box-title">Add User</h3>  <div class="pull-right">  <a href="<?= site\_url('user') ?>" class="btn btn-flat btn-warning btn-sm"><i class="fa fa-undo"></i> Back</a>  </div>  </div>  <div class="box-body">  <div class="row">  <div class="col-md-4 col-md-offset-4">  <?php // echo validation\_errors()  ?>  <form action="" method="post" autocomplete="off">  <div class="form-group <?= form\_error('name') ? 'has-error' : null ?>">  <label for="name">Name \*</label>  <input type="text" name="name" id="name" value="<?= set\_value('name') ?>" class="form-control">  <?= form\_error('name') ?>  </div>  <div class="form-group <?= form\_error('username') ? 'has-error' : null ?>">  <label for="username">Username \*</label>  <input type="text" name="username" id="username" value="<?= set\_value('username') ?>" class="form-control">  <?= form\_error('username') ?>  </div>  <div class="form-group <?= form\_error('password') ? 'has-error' : null ?>">  <label for="password">Password \*</label>  <input type="password" name="password" id="password" value="<?= set\_value('password') ?>" class="form-control">  <?= form\_error('password') ?>  </div>  <div class="form-group <?= form\_error('passconf') ? 'has-error' : null ?>">  <label for="passconf">Repeat Confirmation \*</label>  <input type="password" name="passconf" id="passconf" value="<?= set\_value('passconf') ?>" class="form-control">  <?= form\_error('passconf') ?>  </div>  <div class="form-group">  <label for="address">Address</label>  <textarea name="address" id="address" class="form-control"><?= set\_value('address') ?></textarea>  </div>  <div class="form-group <?= form\_error('level') ? 'has-error' : null ?>">  <label for="level">Level \*</label>  <select name="level" id="level" class="form-control">  <option value="">- Pilih -</option>  <option value="1" <?= set\_value('level') == 1 ? 'selected' : null ?>>Admin</option>  <option value="2" <?= set\_value('level') == 2 ? 'selected' : null ?>>Kasir</option>  </select>  <?= form\_error('level') ?>  </div>  <div class="form-group pull-right">  <button type="submit" name="add" class="btn btn-flat btn-success"><i class="fa fa-paper-plane"></i> Save</button>  <button type="reset" class="btn btn-flat">Reset</button>  </div>  </form>  </div>  </div>  </div>  </div>  </section> |

|  |
| --- |
| *View user\_form\_edit* |
| <section class="content-header">  <h1>Users  <small>Pengguna / Karyawan</small>  </h1>  <ol class="breadcrumb">  <li><a><i class="fa fa-dashboard"></i></a></li>  <li class="active">Users</li>  </ol>  </section>  <section class="content">  <div class="box">  <div class="box-header">  <h3 class="box-title">Edit User</h3>  <div class="pull-right">  <a href="<?= site\_url('user') ?>" class="btn btn-flat btn-warning btn-sm"><i class="fa fa-undo"></i> Back</a>  </div>  </div>  <div class="box-body">  <div class="row">  <div class="col-md-4 col-md-offset-4">  <form action="" method="post" autocomplete="off">  <div class="form-group">  <label for="name">Name \*</label>  <input type="hidden" name="user\_id" value="<?= $row->user\_id ?>">  <input type="text" name="name" id="name" value="<?= $row->name ?>" class="form-control" required>  </div>  <div class="form-group">  <label for="username">Username \*</label>  <input type="text" name="username" id="username" value="<?= $row->username ?>" class="form-control" required>  </div>  <div class="form-group">  <label for="password">Password</label>  <input type="password" name="password" id="password" class="form-control">  </div>  <div class="form-group">  <label for="address">Address</label>  <textarea name="address" id="address" class="form-control"><?= $row->address ?></textarea>  </div>  <div class="form-group">  <label for="level">Level \*</label>  <select name="level" id="level" class="form-control" required>  <option value="">- Pilih -</option>  <option value="1" <?= $row->level == 1 ? 'selected' : null ?>>Admin</option>  <option value="2" <?= $row->level == 2 ? 'selected' : null ?>>Kasir</option>  </select>  </div>  <div class="form-group pull-right">  <button type="submit" name="edit" class="btn btn-flat btn-success"><i class="fa fa-paper-plane"></i> Save</button>  <button type="reset" class="btn btn-flat">Reset</button>  </div>  </form>  </div>  </div>  </div>  </div>  </section> |

|  |
| --- |
| *View admin\_lte* |
| <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8">  <meta http-equiv="X-UA-Compatible" content="IE=edge">  <title><?= ucwords($title) ?> | Satria Jaya</title>  <meta content="width=device-width, initial-scale=1, maximum-scale=1, user-scalable=no" name="viewport">  <link rel="stylesheet" href="<?= base\_url() ?>assets/bower\_components/bootstrap/dist/css/bootstrap.min.css">  <link rel="stylesheet" href="<?= base\_url() ?>assets/bower\_components/font-awesome/css/font-awesome.min.css">  <link rel="stylesheet" href="<?= base\_url() ?>assets/bower\_components/Ionicons/css/ionicons.min.css">  <link rel="stylesheet" href="<?= base\_url() ?>assets/plugins/pace/pace.min.css">  <link rel="stylesheet" href="<?= base\_url() ?>assets/bower\_components/datatables.net-bs/css/dataTables.bootstrap.min.css">  <link rel="stylesheet" href="<?= base\_url() ?>assets/bower\_components/select2/dist/css/select2.min.css">  <link rel="stylesheet" href="<?= base\_url() ?>assets/dist/css/AdminLTE.min.css">  <link rel="stylesheet" href="<?= base\_url() ?>assets/dist/css/skins/\_all-skins.min.css">  <link rel="stylesheet" href="<?= base\_url() ?>assets/bower\_components/bootstrap-datepicker/dist/css/bootstrap-datepicker.min.css">  <!--[if lt IE 9]>  <script src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script>  <script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>  <![endif]-->  <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,600,700,300italic,400italic,600italic">  </head>  <?php if ($this->session->userdata('level') == 2) { ?>  <body class="hold-transition skin-blue-light <?= $this->uri->segment(1) == 'sale' ? 'sidebar-collapse' : null ?>">  <?php } ?>  <?php if ($this->session->userdata('level') == 1) { ?>  <body class="hold-transition skin-red <?= $this->uri->segment(1) == 'sale' ? 'sidebar-collapse' : null ?>">  <?php } ?>  <div class="wrapper">  <header class="main-header">  <!-- Logo -->  <a href="<?= base\_url() ?>" class="logo">  <span class="logo-mini">S<b>J</b></span>  <span class="logo-lg">Aplikasi Satria Jaya</span>  </a>  <nav class="navbar navbar-static-top">  <a href="#" class="sidebar-toggle" data-toggle="push-menu" role="button">  <span class="sr-only">Toggle navigation</span>  <span class="icon-bar"></span>  <span class="icon-bar"></span>  <span class="icon-bar"></span>  </a>  <div class="navbar-custom-menu">  <ul class="nav navbar-nav">  <!-- Notifications: style can be found in dropdown.less -->  <li class="dropdown notifications-menu">  <a href="#" class="dropdown-toggle" data-toggle="dropdown">  <i class="fa fa-bell-o"></i>  <span class="label label-warning">2</span>  </a>  <ul class="dropdown-menu">  <li class="header">You have 2 notifications</li>  <li>  <ul class="menu">  <li>  <a href="#">  <i class="fa fa-th-list"></i> 5 new items added today  </a>  </li>  </ul>  </li>  <li class="footer"><a href="#">View all</a></li>  </ul>  </li>  <!-- User Account: style can be found in dropdown.less -->  <li class="dropdown user user-menu">  <a href="#" class="dropdown-toggle" data-toggle="dropdown">  <img src="<?= base\_url() ?>assets/dist/img/user.png" class="user-image">  <span class="hidden-xs"><?= $this->fungsi->user\_login()->username ?></span>  </a>  <ul class="dropdown-menu">  <li class="user-header">  <img src="<?= base\_url() ?>assets/dist/img/user.png">  <p>  <?= $this->fungsi->user\_login()->name ?>  <small><?= $this->fungsi->user\_login()->address ?></small>  </p>  </li>  <li class="user-footer">  <div class="pull-left">  <a href="<?= site\_url('dashboard') ?>" class="btn btn-default btn-flat">Profile</a>  </div>  <div class="pull-right">  <a href="<?= site\_url('auth/logout') ?>" onclick="return confirm('Apakah Anda yakin?')" class="btn btn-md btn-danger" class="btn btn-flat bg-red">Logout</a>  </div>  </li>  </ul>  </li>  </ul>  </div>  </nav>  </header>  <!-- Left side column. contains the sidebar -->  <aside class="main-sidebar">  <section class="sidebar">  <div class="user-panel">  <div class="pull-left image">  <img src="<?= base\_url() ?>assets/dist/img/user.png" class="img-circle">  </div>  <div class="pull-left info">  <p><?= ucfirst($this->fungsi->user\_login()->username) ?></p>  <a href="#"><i class="fa fa-circle text-success"></i> Online</a>  </div>  </div>  <form action="#" method="get" class="sidebar-form">  <div class="input-group">  <input type="text" name="q" class="form-control" placeholder="Search...">  <span class="input-group-btn">  <button type="submit" name="search" id="search-btn" class="btn btn-flat"><i class="fa fa-search"></i>  </button>  </span>  </div>  </form>  <!-- sidebar menu: : style can be found in sidebar.less -->  <ul class="sidebar-menu" data-widget="tree">  <li class="header">MAIN MENU</li>  <li class="<?= $this->uri->segment(1) == 'dashboard' ? 'active' : null ?>">  <a href="<?= site\_url('dashboard') ?>"><i class="fa fa-dashboard"></i> <span>Beranda</span></a>  </li>  <?php if ($this->session->userdata('level') == 2) { ?>  <li class="<?= $this->uri->segment(1) == 'customer' ? 'active' : null ?>">  <a href="<?= site\_url('customer') ?>">  <i class="fa fa-users"></i> <span>Pelanggan</span>  <span class="pull-right-container">  <span class="label bg-purple pull-right"><?= $this->fungsi->count\_customer() ?></span>  </span>  </a>  </li>  <li class="treeview <?= $this->uri->segment(1) == 'sale'  || $this->uri->segment(1) == 'stock' ? 'active' : null ?>">  <a href="#">  <i class="fa fa-shopping-cart"></i> <span>Transaksi</span>  <span class="pull-right-container">  <i class="fa fa-angle-left pull-right"></i>  </span>  </a>  <ul class="treeview-menu">  <li class="<?= $this->uri->segment(1) == 'sale' ? 'active' : null ?>">  <a href="<?= site\_url('sale') ?>"><i class="fa fa-circle-o text-green"></i> <b>Penjualan</b></a>  </li>  <li class="<?= ($this->uri->segment(1) == 'stock'  && $this->uri->segment(2) == 'in') ? 'active' : null ?>">  <a href="<?= site\_url('stock/in') ?>"><i class="fa fa-circle-o text-green"></i> Persediaan Masuk </a>  </li>  <li class="<?= ($this->uri->segment(1) == 'stock'  && $this->uri->segment(2) == 'out') ? 'active' : null ?>">  <a href="<?= site\_url('stock/out') ?>"><i class="fa fa-circle-o"></i> Persediaan Keluar</a>  </li>  </ul>  </li>  <li class="<?= $this->uri->segment(1) == 'supplier' ? 'active' : null ?>">  <a href="<?= site\_url('supplier') ?>">  <i class="fa fa-truck"></i> <span>Supplier</span>  <span class="pull-right-container">  <span class="label bg-purple pull-right"><?= $this->fungsi->count\_supplier() ?></span>  </span>  </a>  </li>  <li class="treeview <?= $this->uri->segment(1) == 'category'  || $this->uri->segment(1) == 'unit'  || $this->uri->segment(1) == 'item' ? 'active' : null ?>">  <a href="#">  <i class="fa fa-archive"></i> <span>Produk</span>  <span class="pull-right-container">  <i class="fa fa-angle-left pull-right"></i>  </span>  </a>  <ul class="treeview-menu">  <li class="<?= $this->uri->segment(1) == 'category' ? 'active' : null ?>">  <a href="<?= site\_url('category') ?>"><i class="fa fa-circle-o"></i> Kategori</a>  </li>  <li class="<?= $this->uri->segment(1) == 'unit' ? 'active' : null ?>">  <a href="<?= site\_url('unit') ?>"><i class="fa fa-circle-o"></i> Satuan</a>  </li>  <li class="<?= $this->uri->segment(1) == 'item' ? 'active' : null ?>">  <a href="<?= site\_url('item') ?>"><i class="fa fa-circle-o text-green"></i> <b>Items</b></a>  </li>  </ul>  </li>  <?php } ?>  <?php if ($this->session->userdata('level') == 1) { ?>  <li class="header">SETTINGS</li>  <li class="<?= $this->uri->segment(1) == 'user' ? 'active' : null ?>">  <a href="<?= site\_url('user') ?>"> <i class="fa fa-user">  </i> <span>Pengguna</span>  <span class="pull-right-container">  <span class="label label-danger pull-right"><?= $this->fungsi->count\_user() ?></span>  </span>  </a>  </li>  <li class="treeview <?= $this->uri->segment(1) == 'report' ? 'active' : null ?>">  <a href="#">  <i class="fa fa-pie-chart"></i> <span>Laporan</span>  <span class="pull-right-container">  <i class="fa fa-angle-left pull-right"></i>  </span>  </a>  <ul class="treeview-menu">  <li class="<?= $this->uri->segment(1) == 'report'  && $this->uri->segment(2) == 'sale' ? 'active' : null ?>">  <a href="<?= site\_url('report/sale') ?>"><i class="fa fa-circle-o"></i> Penjualan</a>  </li>  <li class="<?= $this->uri->segment(1) == 'report'  && $this->uri->segment(2) == 'stock' ? 'active' : null ?>">  <a href="<?= site\_url('report/stock') ?>"><i class="fa fa-circle-o"></i> Stock Barang</a>  </li>  </ul>  </li>  <?php } ?>  </ul>  </section>  </aside>  <script src="<?= base\_url() ?>assets/bower\_components/jquery/dist/jquery.min.js"></script>  <!-- Content Wrapper. Contains page content -->  <div class="content-wrapper">  <?= $contents ?>  </div>  <footer class="main-footer">  <div class="pull-right hidden-xs"><b>Version</b> 0.1</div>  <strong> <?= date('Y') ?>.</strong> All rights reserved.  </footer>  </div>  <script src="<?= base\_url() ?>assets/bower\_components/bootstrap/dist/js/bootstrap.min.js"></script>  <script src="<?= base\_url() ?>assets/bower\_components/PACE/pace.min.js"></script>  <script src="<?= base\_url() ?>assets/bower\_components/jquery-slimscroll/jquery.slimscroll.min.js"></script>  <script src="<?= base\_url() ?>assets/bower\_components/fastclick/lib/fastclick.js"></script>  <script src="<?= base\_url() ?>assets/dist/js/adminlte.min.js"></script>  <script src="<?= base\_url() ?>assets/dist/js/demo.js"></script>  <script src="<?= base\_url() ?>assets/bower\_components/datatables.net/js/jquery.dataTables.min.js"></script>  <script src="<?= base\_url() ?>assets/bower\_components/datatables.net-bs/js/dataTables.bootstrap.min.js"></script>  <script src="<?= base\_url() ?>assets/bower\_components/select2/dist/js/select2.full.min.js"></script>  <script src="<?= base\_url() ?>assets/bower\_components/bootstrap-datepicker/dist/js/bootstrap-datepicker.min.js"></script>  <script>  $(document).ready(function() {  $(document).ajaxStart(function() {  Pace.restart()  })  $('#table1').DataTable()  })  </script>  </body>  </html> |

|  |
| --- |
| *View\_dashboard* |
| <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8">  <meta http-equiv="X-UA-Compatible" content="IE=edge">  <title><?= ucwords($title) ?> | Satria Jaya</title>  <meta content="width=device-width, initial-scale=1, maximum-scale=1, user-scalable=no" name="viewport">  <link rel="stylesheet" href="<?= base\_url() ?>assets/bower\_components/bootstrap/dist/css/bootstrap.min.css">  <link rel="stylesheet" href="<?= base\_url() ?>assets/bower\_components/font-awesome/css/font-awesome.min.css">  <link rel="stylesheet" href="<?= base\_url() ?>assets/bower\_components/Ionicons/css/ionicons.min.css">  <link rel="stylesheet" href="<?= base\_url() ?>assets/plugins/pace/pace.min.css">  <link rel="stylesheet" href="<?= base\_url() ?>assets/bower\_components/datatables.net-bs/css/dataTables.bootstrap.min.css">  <link rel="stylesheet" href="<?= base\_url() ?>assets/bower\_components/select2/dist/css/select2.min.css">  <link rel="stylesheet" href="<?= base\_url() ?>assets/dist/css/AdminLTE.min.css">  <link rel="stylesheet" href="<?= base\_url() ?>assets/dist/css/skins/\_all-skins.min.css">  <link rel="stylesheet" href="<?= base\_url() ?>assets/bower\_components/bootstrap-datepicker/dist/css/bootstrap-datepicker.min.css">  <!--[if lt IE 9]>  <script src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script>  <script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>  <![endif]-->  <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,600,700,300italic,400italic,600italic">  </head>  <?php if ($this->session->userdata('level') == 2) { ?>  <body class="hold-transition skin-blue-light <?= $this->uri->segment(1) == 'sale' ? 'sidebar-collapse' : null ?>">  <?php } ?>  <?php if ($this->session->userdata('level') == 1) { ?>  <body class="hold-transition skin-red <?= $this->uri->segment(1) == 'sale' ?  <?php if ($this->session->userdata('level') == 2) { ?>  <section class="content-header">  <h1>Profil  <small>Profil Pengguna</small>  </h1>  </section>  <div class="row">  <div class="col-lg-9">  <p style="font-size: 50px; margin-left:35%; margin-top:20px">Selamat Datang <?= $this->fungsi->user\_login()->name ?> </p>  </div>  </div>  <section class="content">  <div class="box">  <div class="box-header">  <h3 class="box-title">Data Pengguna System</h3>  </div>  <div class="box-body table-responsive">  <table id="table1" class="table table-bordered table-striped">  <thead>  <tr>  <th style="text-align: center;">Foto</th>  <th style="text-align: center;">Nama</th>  <th style="text-align: center;">Alamat</th>  <th style="text-align: center;">Dibuat Pada</th>  <th style="text-align: center;">Diubah pada</th>  </tr>  </thead>  <tbody>  <tr>  <td><a href="#" class="dropdown-toggle" data-toggle="dropdown">  <img style="border-radius:100%; margin-left:10%;" src="<?= base\_url() ?>assets/dist/img/user.png" class="user-image">  </a></td>  <td style="font-size:20px;line-height: 120px;"> <?= $this->fungsi->user\_login()->username ?></td>  <td style="font-size:20px;line-height: 120px;"><?= $this->fungsi->user\_login()->address ?></td>  <td style="font-size:20px;line-height: 120px;"><?= $this->fungsi->user\_login()->created ?></td>  <td style="font-size:20px;line-height: 120px;"><?= $this->fungsi->user\_login()->updated ?></td>  </tr>  </tbody>  </table>  </div>  </div>  </section>  <?php } ?>  <?php if ($this->session->userdata('level') == 1) { ?>  <section class="content-header">  <h1>Beranda  <small>Antar Muka</small>  </h1>  <ol class="breadcrumb">  <li><a><i class="fa fa-dashboard"></i></a></li>  <li class="active"><?= $title ?></li>  </ol>  </section>  <section class="content">  <div class="row">  <div class="col-md-3 col-sm-6 col-xs-12">  <div class="info-box">  <span class="info-box-icon bg-aqua"><i class="ion ion-bag"></i></span>  <div class="info-box-content">  <span class="info-box-text">Items</span>  <span class="info-box-number"><?= $this->fungsi->count\_item() ?></span>  </div>  </div>  </div>  <div class="col-md-3 col-sm-6 col-xs-12">  <div class="info-box">  <span class="info-box-icon bg-red"><i class="fa fa-truck"></i></span>  <div class="info-box-content">  <span class="info-box-text">Suppliers</span>  <span class="info-box-number"><?= $this->fungsi->count\_supplier() ?></span>  </div>  </div>  </div>  <div class="clearfix visible-sm-block"></div>  <div class="col-md-3 col-sm-6 col-xs-12">  <div class="info-box">  <span class="info-box-icon bg-green"><i class="fa fa-users"></i></span>  <div class="info-box-content">  <span class="info-box-text">Customers</span>  <span class="info-box-number"><?= $this->fungsi->count\_customer() ?></span>  </div>  </div>  </div>  <div class="col-md-3 col-sm-6 col-xs-12">  <div class="info-box">  <span class="info-box-icon bg-yellow"><i class="fa fa-user-plus"></i></span>  <div class="info-box-content">  <span class="info-box-text">Users</span>  <span class="info-box-number"><?= $this->fungsi->count\_user() ?></span>  </div>  </div>  </div>  </div>  <div class="row">  <div class="col-md-1">  </div>  <div class="col-md-9">  <div class="box box-solid">  <div class="box-header">  <i class="fa fa-th"></i>  <h3 class="box-title">Akun Pengguna</h3>  <div class="box-tools pull-right">  <button type="button" class="btn btn-sm" data-widget="collapse">  <i class="fa fa-minus">  </i>  <section class="content">  <div class="box">  <div class="box-header">  <h3 class="box-title">Data Pengguna System</h3>  </div>  <div class="box-body table-responsive">  <table id="table1" class="table table-bordered table-striped">  <thead>  <tr>  <th style="text-align: center;">Foto</th>  <th style="text-align: center;">Nama</th>  <th style="text-align: center;">Alamat</th>  <th style="text-align: center;">Dibuat Pada</th>  <th style="text-align: center;">Diubah pada</th>  </tr>  </thead>  <tbody>  <tr>  <td><a href="#" class="dropdown-toggle" data-toggle="dropdown">  <img style="border-radius:100%; margin-left:10%;" src="<?= base\_url() ?>assets/dist/img/user.png" class="user-image">  </a></td>  <td style="font-size:20px;line-height: 120px;"> <?= $this->fungsi->user\_login()->username ?></td>  <td style="font-size:20px;line-height: 120px;"><?= $this->fungsi->user\_login()->address ?></td>  <td style="font-size:20px;line-height: 120px;"><?= $this->fungsi->user\_login()->created ?></td>  <td style="font-size:20px;line-height: 120px;"><?= $this->fungsi->user\_login()->updated ?></td>  </tr>  </tbody>  </table>  </div>  </div>  </section>  </button>  </div>  </div>  <div class="box-body">  <div id="hero-bar" class="graph"></div>  </div>  </div>  </div>  </div>  </section>  <link rel="stylesheet" href="<?= base\_url() ?>assets/bower\_components/morris.js/morris.css">  <script src="<?= base\_url() ?>assets/bower\_components/raphael/raphael.min.js"></script>  <script src="<?= base\_url() ?>assets/bower\_components/morris.js/morris.min.js"></script>  <script>  Morris.Bar({  element: 'hero-bar',  xkey: 'item',  ykeys: ['sold'],  labels: ['Sold'],  barRatio: 0.4,  xLabelAngle: 35,  hideHover: 'auto'  });  </script>  <?php } ?> |

|  |
| --- |
| *View\_login* |
| <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8">  <meta http-equiv="X-UA-Compatible" content="IE=edge">  <title>Login | Mysistem</title>  <meta content="width=device-width, initial-scale=1, maximum-scale=1, user-scalable=no" name="viewport">  <link rel="stylesheet" href="<?= base\_url() ?>assets/bower\_components/bootstrap/dist/css/bootstrap.min.css">  <link rel="stylesheet" href="<?= base\_url() ?>assets/bower\_components/font-awesome/css/font-awesome.min.css">  <link rel="stylesheet" href="<?= base\_url() ?>assets/dist/css/AdminLTE.min.css">  <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,600,700,300italic,400italic,600italic">  <style>  .hold-transition {  background-image: url(<?= base\_url('assets/dist/img/photo.png') ?>);  background-size: 100%;  }  </style>  </head>  <body class="hold-transition login-page">  <div class="login-box">  <div>  </div>  <div class="login-logo">  <a href="<?= base\_url() ?>"><b>Login</b> Satria Jaya</a>  </div>  <!-- /.login-logo -->  <div class="login-box-body">  <p class="login-box-msg">Masuk ke akun Anda</p>  <form action="<?= site\_url('auth/proses') ?>" method="post" autocomplete="off">  <div class="form-group has-feedback">  <input type="text" name="username" class="form-control" placeholder="Username" required autofocus>  <span class="glyphicon glyphicon-user form-control-feedback"></span>  </div>  <div class="form-group has-feedback">  <input type="password" name="password" class="form-control" placeholder="Password" required>  <span class="glyphicon glyphicon-lock form-control-feedback"></span>  </div>  <div class="row">  <div class="col-xs-8">  </div>  <div class="col-xs-4">  <button type="submit" name="login" class="btn btn-primary btn-block btn-flat">Login</button>  </div>  </div>  </form>  </div>  </div>  </body>  </html> |