**LAPORAN KEGIATAN MAGANG MAHASISWA**

**SISTEM INFORMASI PERGUDANGAN**

**BAHAN MAKANAN KERING**

Diajukan Untuk Memenuhi Sebagian Persyaratan Memperoleh Kelulusan

Mata Kuliah Kegiatan Magang Mahasiswa



Disusun oleh :

**MUTHIA AMESTHI SAHARANI**

**NIM. M3118061**

**PROGRAM DIPLOMA III TEKNIK INFORMATIKA**

**SEKOLAH VOKASI**

**UNIVERSITAS SEBELAS MARET**

**2021**

**HALAMAN PERSETUJUAN**

**LAPORAN KEGIATAN MAGANG MAHASISWA**

**SISTEM INFORMASI PERGUDANGAN**

**BAHAN MAKANAN KERING**

Disusun Oleh

Muthia Amesthi Saharani

NIM. M3118061

Laporan Kegiatan Magang Mahasiswa ini disetujui untuk dipresentasikan

pada Seminar KMM pada tanggal

**Pembimbing Pembimbing Lapangan**

**Taufiqurrakhman Nur Hidayat S.Kom., M.Cs** **Anisah Puji Nastiti**

NIK 1982052220180601

**HALAMAN PENGESAHAN**

Laporan Kegiatan Magang Mahasiswa (KMM) yang dilaksanakan oleh :

**Nama : Muthia Amesthi Saharani**

**NIM : M3118061**

dengan judul :

**LAPORAN KEGIATAN MAGANG MAHASISWA**

**SISTEM INFORMASI PERGUDANGAN**

**BAHAN MAKANAN KERING**

Pada bulan Januari-April 2021, dipresentasikan dan disahkan pada :

Hari :

Tanggal :

|  |
| --- |
| **Pembimbing / Penguji** |
|  |
| **Taufiqurrakhman Nur Hidayat S.Kom., M.Cs**  NIK. 1979060520180601 |

Mengetahui,

|  |  |  |
| --- | --- | --- |
| Ketua Program Diploma III Teknik Informatika FMIPA UNS |  | Direktur Trainer Refactory |
|  |  |  |
| **Hartatik. S.Si., M.Si** |  | **Maulana Prambadi** |
| NIP. 1978050320130201 |  |  |

# KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah SWT yang telah memberikan rahmat, hidayah serta nikmat-Nya, penulis dapat menyelesaikan kegiatan magang mahasiswa dan laporan ini dengan judul *“Sistem Informasi Pergudangan Bahan Makanan Kering”*.

Kegiatan magang mahasiswa ini bertujuan untuk menerapkan ilmu yang didapatkan di dalam perkuliahan serta menambah pengalaman baru yang diperoleh yang penulis lalui.

Dalam pelaksanaan kegiatan magang mahasiswa yang dimulai dari tanggal 19 Januari 2021 hingga 19 April 2021 penulis mengucapkan terima kasih atas dorongan, semangat, petunjuk, nasehat, bimbingan serta kerja sama dari berbagai pihak antara lain :

1. Ibu Hartatik. S.Si.,M.Si selaku kepala program studi Diploma III Teknik Informatika Sekolah Vokasi UNS.
2. Bapak Taufiqurrakhman Nur Hidayat S.Kom., M.Cs. selaku dosen pembimbing yang telah memberikan dorongan, nasehat serta bimbingan sehingga penulis dapat melaksanakan kegiatan magang mahasiswa dengan baik.
3. Bapak Maulana Prambadi selaku direktur trainer Refactory yang telah memberikan kesempatan penulis untuk magang di Refactory selama 3 bulan ini.
4. Ibu Anisa Puji Nastiti selaku pembimbing lapangan yang telah memberikan petunjuk serta bimbingan sehingga penulis dapat melaksanakan kegiatan magang mahasiswa dengan baik.
5. Kedua orang tua dan teman-teman yang senantiasa memberikan semangat, dorongan dan doa untuk penulis.

Surakarta, 19 April 2021

Penulis

# DAFTAR ISI

[KATA PENGANTAR iv](#_Toc77358791)

[DAFTAR ISI v](#_Toc77358792)

[DAFTAR TABEL viii](#_Toc77358793)

[DAFTAR GAMBAR x](#_Toc77358794)

[BAB I PENDAHULUAN 1](#_Toc77358795)

[A. Latar Belakang 1](#_Toc77358796)

[B. Perumusan Masalah 1](#_Toc77358797)

[C. Tujuan Kegiatan 2](#_Toc77358798)

[D. Manfaat Kegiatan 2](#_Toc77358799)

[E. Batasan Masalah 3](#_Toc77358800)

[BAB II LANDASAN TEORI 4](#_Toc77358801)

[A. Tinjauan Pustaka 4](#_Toc77358802)

[B. Landasan Teori 5](#_Toc77358803)

[1. Istilah Umum 5](#_Toc77358804)

[2. Perancangan 6](#_Toc77358805)

[3. Implementasi 10](#_Toc77358806)

[4. Pengujian 11](#_Toc77358807)

[BAB III PELAKSANAAN KEGIATAN 12](#_Toc77358808)

[A. Gambaran Instansi 12](#_Toc77358809)

[1. Profil 12](#_Toc77358810)

[2. Struktur 13](#_Toc77358811)

[3. Website 13](#_Toc77358812)

[B. Paparan Kegiatan 14](#_Toc77358813)

[C. Jadwal Kegiatan 16](#_Toc77358814)

[BAB IV HASIL DAN PEMBAHASAN 18](#_Toc77358815)

[A. Perancangan Sistem 18](#_Toc77358816)

[1. Proses Bisnis 18](#_Toc77358817)

[4. Use Case Diagram 22](#_Toc77358818)

[5. Use Case Text 24](#_Toc77358819)

[3. Activity Diagram 36](#_Toc77358820)

[4. Entity Relational Diagram 48](#_Toc77358821)

[5. Tabel Fisik 50](#_Toc77358822)

[6. Sitemap Sistem 56](#_Toc77358823)

[D. Implementasi dan Koding 57](#_Toc77358824)

[1. Tampilan Homepage 57](#_Toc77358825)

[2. Tampilan Login 58](#_Toc77358826)

[3. Tampilan Dashboard Admin 61](#_Toc77358827)

[4. Tampilan Dashboard Pegawai 63](#_Toc77358828)

[5. Tampilan Data Master 64](#_Toc77358829)

[6. Tampilan Stok Barang 72](#_Toc77358830)

[7. Tampilan Transaksi 73](#_Toc77358831)

[8. Tampilan Laporan Pertanggal 77](#_Toc77358832)

[9. Tampilan Akun 83](#_Toc77358833)

[E. Pengujian Sistem 85](#_Toc77358834)

[1. Pengujian sistem login 85](#_Toc77358835)

[2. Pengujian form input data pengguna 86](#_Toc77358836)

[3. Pengujian form input data jenis barang 87](#_Toc77358837)

[4. Pengujian form input data satuan barang 87](#_Toc77358838)

[5. Pengujian form input data barang 88](#_Toc77358839)

[6. Pengujian form input data supplier 89](#_Toc77358840)

[7. Pengujian form input data stok barang 90](#_Toc77358841)

[8. Pengujian form input data barang masuk 90](#_Toc77358842)

[9. Pengujian form input data barang keluar 91](#_Toc77358843)

[10. Pengujian form laporan pertanggal 92](#_Toc77358844)

[11. Pengujian form ubah password 93](#_Toc77358845)

[BAB V PENUTUP 96](#_Toc77358846)

[A. Kesimpulan 96](#_Toc77358847)

[B. Saran 96](#_Toc77358848)

[DAFTAR PUSTAKA 97](#_Toc77358849)

# DAFTAR TABEL

[Table 3.0.1 Jadwal kegiatan 16](#_Toc77344333)

[Table 4.0.1 Kebutuhan Fungsional 19](#_Toc77344334)

[Table 4.0.2 Use case text login 24](#_Toc77344335)

[Table 4.0.3 Use case text tambah data pengguna 25](#_Toc77344336)

[Table 4.0.4 Use case text tambah data jenis 26](#_Toc77344337)

[Table 4.0.5 Use case text tambah data satuan 27](#_Toc77344338)

[Table 4.0.6 Use case text tambah data barang 28](#_Toc77344339)

[Table 4.0.7 Use case text tambah data supplier 29](#_Toc77344340)

[Table 4.0.8 Use case text tambah data barang masuk 30](#_Toc77344341)

[Table 4.0.9 Use case text tambah data barang keluar 31](#_Toc77344342)

[Table 4.0.10 Use case text tambah data stok barang 32](#_Toc77344343)

[Table 4.0.11 Use case text cetak laporan pertanggal 33](#_Toc77344344)

[Table 4.0.12 Use case text setting akun 34](#_Toc77344345)

[Table 4.0.13 Use case text logout 35](#_Toc77344346)

[Table 4.0.14 Tabel users 50](#_Toc77344347)

[Table 4.0.15 Tabel data barang 51](#_Toc77344348)

[Table 4.0.16 Tabel jenis barang 52](#_Toc77344349)

[Table 4.0.17 Tabel satuan barang 53](#_Toc77344350)

[Table 4.0.18 Tabel barang masuk 53](#_Toc77344351)

[Table 4.0.19 Tabel barang keluar 54](#_Toc77344352)

[Table 4.0.20 Tabel stok barang 54](#_Toc77344353)

[Table 4.0.21 Tabel supplier 55](#_Toc77344354)

[Table 4.0.22 Pengujian sistem login 85](#_Toc77344355)

[Tabel 4.0.23 Pengujian form input data pengguna 86](#_Toc77344356)

[Tabel 4.0.24 Pengujian form input data jenis barang 87](#_Toc77344357)

[Table 4.0.25 Pengujian form input data satuan barang 87](#_Toc77344358)

[Table 4.0.26 Pengujian form input data jenis barang 88](#_Toc77344359)

[Table 4.0.27 Pengujian form input data supplier 89](#_Toc77344360)

[Table 4.0.28 Pengujian form input data stok barang 90](#_Toc77344361)

[Table 4.0.29 Pengujian form input data barang masuk 90](#_Toc77344362)

[Table 4.0.30 Pengujian form input data barang keluar 91](#_Toc77344363)

[Table 4.0.31 Pengujian form laporan pertanggal 92](#_Toc77344364)

[Table 4.0.32 Pengujian form ubah password 93](#_Toc77344365)

# DAFTAR GAMBAR

[Gambar 3.0.1 Kantor Gedung Refactory 12](#_Toc77241568)

[Gambar 3.0.2 Struktur Organisasi 13](#_Toc77241569)

[Gambar 3.0.3 Website 14](#_Toc77241570)

[Gambar 3.0.4 Meeting 14](#_Toc77241571)

[Gambar 3.0.5 Stand up 15](#_Toc77241572)

[Gambar 4.0.1 Bisnis proses admin 18](#_Toc77241573)

[Gambar 4.0.2 Use case diagram 23](#_Toc77241574)

[Gambar 4.0.3 Activity diagram menerima inputan login 36](#_Toc77241575)

[Gambar 4.0.4 Activity diagram data pegawai 37](#_Toc77241576)

[Gambar 4.0.5 Activity diagram data pegawai 38](#_Toc77241577)

[Gambar 4.0.6 Activity diagram data pegawai 39](#_Toc77241578)

[Gambar 4.0.7 Activity diagram data barang 40](#_Toc77241579)

[Gambar 4.0.8 Activity diagram data supplier 41](#_Toc77241580)

[Gambar 4.0.9 Activity diagram data barang masuk 42](#_Toc77241581)

[Gambar 4.0.10 Activity diagram data barang keluar 43](#_Toc77241582)

[Gambar 4.0.11 Activity diagram data stok barang 44](#_Toc77241583)

[Gambar 4.0.12 Activity diagram laporan pertanggal 45](#_Toc77241584)

[Gambar 4.0.13 Activity diagram setting akun 46](#_Toc77241585)

[Gambar 4.0.14 Activity diagram logout 47](#_Toc77241586)

[Gambar 4.0.15 Entity relational diagram 48](#_Toc77241587)

[Gambar 4.0.16 Gambar Sitemap/Navigasi atau peta sistem berdasarkan aktor 56](#_Toc77241588)

[Gambar 4.0.17 Hompage 57](#_Toc77241589)

[Gambar 4.0.18 Halaman dashboard pegawai 59](#_Toc77241590)

[Gambar 4.0.19 Halaman dashboard admin 61](#_Toc77241591)

[Gambar 4.0.20 Halaman dashboard pegawai 63](#_Toc77241592)

[Gambar 4.0.21 Halaman data pengguna 64](#_Toc77241593)

[Gambar 4.0.22 Halaman data jenis 67](#_Toc77241594)

[Gambar 4.0.23 Halaman data satuan 68](#_Toc77241595)

[Gambar 4.0.24 Halaman data barang 69](#_Toc77241596)

[Gambar 4.0.25 Halaman data supplier 71](#_Toc77241597)

[Gambar 4.0.26 Halaman data stok barang 72](#_Toc77241598)

[Gambar 4.0.27 Halaman data barang masuk 74](#_Toc77241599)

[Gambar 4.0.28 Halaman data barang keluar 75](#_Toc77241600)

[Gambar 4.0.29 Halaman laporan stok 77](#_Toc77241601)

[Gambar 4.0.30 Halaman laporan barang masuk 79](#_Toc77241602)

[Gambar 4.0.31 Halaman laporan barang keluar 81](#_Toc77241603)

[Gambar 4.0.32 Halaman akun 83](#_Toc77241604)

# BAB I PENDAHULUAN

## Latar Belakang

Sebuah toko grosir tentunya banyak transaksi disetiap harinya. Setiap penjualan satu persatu selalu harus dicatat agar pemilik toko dapat mengetahui pendapatan yang diperoleh dalam kurun waktu tertentu. Sebelum ada mesin penghitung seperti komputer, pencatatan dilakukan menggunakan cara manual, yaitu dengan kertas dan di buat tabel-tabel agar mudah pengolahannya (Muhammad, 2010).

Menurut Gigin Ginanjar pada jurnal yang berjudul *Sistem Informasi Persediaan Bahan Baku Pada Rumah Makan Sari Sunda Bandung* permasalahan yang sering terjadi pada bagian persediaan bahan baku gudang adalah tidak sesuainya perhitungan antara barang yang masuk ataupun keluar dikarenakan pencatatannya masih secara manual yaitu dituliskannya ke dalam buku pergudangan, belum adanya sistem persediaan bahan baku secara client-server atau terintergerasi dengan bagian produksi dan dapur, sehingga hal tersebut menjadi tidak terkontrolnya barang yang masuk maupun keluar.

Dengan adanya berbagai permasalahan tersebut maka sistem informasi bisa menjadi salah satu alternatif untuk membantu dalam pemecahan masalah pengelolaan data pada suatu gudang agar lebih cepat dan mudah terutama untuk memanipulasi suatu data.

## Perumusan Masalah

Dari latar belakang masalah di atas, maka solusi yang dapat diambil adalah bagaimana merancang dan membangun sebuah sistem informasi pergudangan bahan makanan pada sebuah toko.

## Tujuan Kegiatan

Judul penelitian Sistem Informasi Pergudangan Bahan Makanan Kering memiliki tujuan sebagai berikut :

1. Merancang dan membangun sebuah sistem informasi yang dapat membantu mencatat kegiatan transaksi seperti stok barang, barang masuk dan barang keluar.
2. Menghasilkan informasi berupa laporan pencatatan stok barang, barang masuk dan barang keluar guna membantu pemilik toko untuk mengelola tokonya.

## Manfaat Kegiatan

Manfaat yang dapat diperoleh dengan adanya produk sistem informasi pergudangan bahan makanan kering ini adalah sebagai berikut:

1. Untuk toko atau perusahaan

* Meningkatkan kinerja sebuah gudang pada toko.

1. Untuk pegawai Gudang

* Membantu dalam pengelolaan data gudang dalam jumlah yang relative banyak.
* Membantu dalam pengelolaan data transaksi keluar masuk barang sesuai dengan tanggalnya.
* Pegawai gudang akan lebih cepat untuk mencari data gudang yang sudah relatif lama penyimpanannya.
* Membantu proses pelaporan data gudang kepada pemilik toko.

1. Bagi Penulis :

* Untuk memenuhi sebagian persyaratan memperoleh kelulusan mata kuliah Kegiatan Magang Mahasiswa (KMM).

## Batasan Masalah

Dalam penulisan laporan Kegiatan Magang Mahasiswa ini, penulis membatasi masalah sebagai berikut :

1. Pembuatan sistem informasi pergudangan bahan makanan kering berbasis web.
2. Sistem informasi ini dapat diakses oleh admin gudang dan pegawai gudang.
3. Sistem informasi ini hanya untuk pencatatan stok barang, barang masuk, barang keluar dan pembuatan laporan stok barang, barang masuk hingga barang keluar.

# BAB II LANDASAN TEORI

## Tinjauan Pustaka

Sistem Informasi Persediaan Bahan Baku Pada Rumah Makan Sari Sunda Bandung. Sistem informasi ini dibuat oleh Gigin Ginanjar mahasiswa Sistem Informasi Fakultas Teknik dan Ilmu Komputer Universitas Komputer Indonesia Bandung. Kegunaan dari sistem ini yaitu untuk mengelola pendataan bahan baku meliputi data pembelian dan pengeluaran yang ada pada Rumah Makan Sari Sunda. Bahasa pemrograman yang digunakan pada sistem informasi ini yaitu bahasa pemrograman PHP dengan database MySQL.

Sistem Informasi Penjualan Pada Toko Do’a Ibu Grosir Sembako Di Desa Ciguha – Bogor. Sistem informasi ini dibuat oleh Muhammad Ikhsan mahasiswa Sistem Informasi, STMIK Amikom Purwokerto. Kegunaan dari sistem ini yaitu untuk proses pengolahan data dan pelaporan data penjualan dan pembelian pada toko grosir sembako Do’a Ibu di Desa Ciguha-Bogor. Sistem informasi ini dibuat menggunakan visual studio 2008 dan SQL Server 2005.

Sistem Informasi Stok Gudang Pada Platinum Hotel Berbasis Web. Sistem informasi ini dibuat oleh Tika Sari Ramadhani, Sudi Suryadi dan Deci Irmayani Akademi Manajemen Informatika Komputer Labuhan Batu. Kegunaan dari sistem ini yaitu untuk memudahkan pengelolaan stok barang pada Platinum Hotel agar lebih cepat dan akurat. Sistem ini dibuat menggunakan bahasa pemrograman PHP dan database MySQL.

Perancangan Sistem Informasi Persediaan Bahan Baku Di Ud. Xy, Tulungagung. Sistem ini dibuat oleh Retno Astuti, Reza Ashari dan Mas’ud Effendi mahasiswa Jurusan Teknologi Industri Pertanian, Fakultas Teknologi Pertanian,Universitas Brawijaya. Kegunaan sistem ini yaitu dapat mengelola data input persediaan bahan baku menjadi output laporan persediaan. Sistem ini dibuat menggunakan konsep Object Oriented Programming (OOP) dengan UML versi 2.0, database MySQL dan XAMPP sebagai local server serta bahasa pemrograman PHP.

## Landasan Teori

### Istilah Umum

* 1. Gudang

Kelancaran dalam suatu toko atau pabrik salah satunya dipengaruhi oleh gudang atau tempat penyimpanan barangnya. Gudang merupakan tempat yang dibebani untuk tempat penyimpanan barang yang akan dipergunakan untuk produksi. Ada tiga hal yang berkaitan dengan pengadaan barang di gudang, yaitu :

1. Pengawasan, yang dimaksud dengan pengawasan disini adalah mengontrol keluar masuknya barang pada sebuah gudang.
2. Pemilihan, yang dimaksud dengan pemilihan disini yaitu bagaimana aktivitas perawatan dan pemeliharaan penyimpanan barang pada sebuah gudang agar tidak rusak dan sesuai dengan penyimpanannya.

Penimbunan atau penyimpanan, yaitu berkaitan dengan stok barang agar saat barang dibutuhkan tetap tersedia selama dan sebelum proses produksi berlangsung.

* 1. Sistem Informasi

Sistem informasi merupakan bagian dari suatu organisasi yang difungsikan untuk mempertemukan kebutuhan pada pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi dan menyediakan laporan yang diperlukan kepada pihak luar. System informasi terdiri dari berbagai elemen diantaranya orang, prosedur, perangkat keras, perangkat lunak, basis data, jaringan computer dan komunikasi data.

* 1. Database

Merupakan kumpulan data atau table yang saling berhubungan atau berelasi, dimana table tersebut merupakan komponen utama dalam pembangun basis data. Relasi dalam tabel dapat ditunjukkan dengan key dari setiap tabel yang ada. Database juga merupakan kumpulan data yang menggambarkan aktivitas-aktivitas dan pelakunya dalam suatu organisasi.

### Perancangan

* 1. Proses Bisnis

Proses bisnis merupakan arus kerja dari informasi, pengetahuan dan berbagai aktivitas, kelancaran sebuah system bergantung pada proses bisnis yang dirancang dan dikoordinasikan. Untuk membuat proses bisnis memerlukan beberapa prinsip yang perlu diperhatikan diantaranya : efektivitas, efisiensi dan adaptif.

* 1. Use Case Diagram

Use Case Diagram merupakan permodelan aktivitas system informasi yang dibuat yang fungsinya untuk mengetahui fungsi apa saja yang terdapat pada system informasi tersebut dan siapa saja yang memegang dan berhak menggunakan fungsi tersebut. Symbol yang digunakan untuk membuat use case diagram diantaranya yaitu :

|  |  |
| --- | --- |
| Simbol | Keterangan |
|  | Symbol di samping dinamakan use case yang fungsinya untuk menggambarkan fungsionalitas pada system sebagai fungsi yang bertukar pesan antar fungsi yang lain dengan actor. |
| Use Case Diagram, Lengkap Studi Kasus dan Contoh Use Case - Materi Dosen | Symbol di samping merupakan symbol actor yang merupakan abstraction dari orang. Actor berinteraksi dengan use case namun tidak memiliki control terhadap use case. |
|  | Symbol di samping merupakan symbol asosiasi yang digambarkan dengan garis tanpa panah yang fungsinya sebagai asosiasi antara use case dengan actor. Mengindikasikan siapa atau apa yang berinteraksi secara langsung. |
|  | Symbol di samping merupakan asosiasi antara use case dengan actor yang menggunakan panah terbuka. Mengindikasikan bila actor berinteraksi secara pasif dengan system. |
| <<include>> | Symbol di samping dinamakan include yang merupakan di dalam use case lain (required) atau pemanggilan use case oleh use case yang lain. |
| <<extend>> | Symbol di samping dinamakan extend, yang merupakan perluasan dari use case jika kondisi atau syarat terpenuhi. |

* 1. Activity Diagram

Activity Diagram atau Diagram Aktivitas merupakan gambaran workflow atau aliran kerja sebuah sistem atau proses bisnis. Adapun symbol-simbol yang digunakan untuk membuat diagram aktivitas yaitu:

|  |  |
| --- | --- |
| Simbol | Keterangan |
|  | Simbol di samping dinamakan start point yang digunakan untuk mengawali sebuah aktivitas. |
|  | Simbol di samping dinamakan end point yang fungsinya untuk mengakhiri sebuah aktivitas. |
|  | Simbol di samping dinamakan activities yang menggambarkan suatu proses atau kegiatan sistem. |
|  | Simbol di samping dinamakan fork atau percabangan yang fungsinya untuk menunjukkan kegiatan yang dilakukan secara pararel atau untuk menggabunggkan dua kegiatan pararel menjadi satu. |
|  | Simbol di sambaing dinamakan join atau sebagai penggabung (rake) yang fungsinya untuk menunjukkan adanya dekomposisi. |
|  | Simbol di samping dinamakan decision point yang fungsinya untuk menggambarkan pilihan untuk pengambilan keputusan true atau false. |
|  | Simbol di samping dinamakan swimlane yang fungsinya untuk membagi aktivitas diagram agar dapat menunjukkan siapa untuk melakukan apa. |

* 1. ERD (Entity Relational Diagram)

ERD (Entity Relational Diagram merupakan model data diagram hubungan yang dibuat berdasarkan anggapan bahwa dunia nyata terdiri dari koleksi object-object dasar yang disebut entitas dan hubungan antar entitas-entitas tersebut. Komponen utama dalam ERD diantaranya :

1. Entitas

Entitas biasanya digambarkan dengan bentuk persegi panjang yang memiliki arti suatu objek pada dunia nyata yang dapat dibedakan antara satu dengan yang lainnya yang memiliki manfaat bagi system yang sedang dikembangkan.

1. Atribut

Atribut biasanya digambarkan dengan bentuk elips yang fungsinya untuk mendeskripsikan entitas.

1. Relasi

Relasi biasanya digambarkan dengan bentuk belah ketupat yang berarti hubungan antara beberapa entitas.

* 1. Functional Requirement

Functional Requirement merupakan apa yang harus dilakukan sistem, layanan apa saja yang harus disediakan pada sebuah sistem, bagaimana reaksi sistem terhadap input dan apa yang harus dilakukan sistem pada situasi khusus. Jenis kebutuhan Fungtional Requirement dilihat dari kacamata penggunanya.

* 1. Non Functional Requirement

Non Functional Requirement merupakan kendala yang dihadapi oleh sebuah sistem atau pelayanan seperti kendala waktu, kendala proses pengembangan, respon dan kebutuhan storage atau kemampuan peralatan I/O, representasi sistem dan lain-lain.

### Implementasi

* 1. Apache

Apache merupakan web server yang bertanggung jawab pada request-responese HTTP dan logging informasi secara detail. Sesuai dengan hasil survei yang dilakukan oleh Netcraft pada bulan Januari 2005 pengguna web server di internet jumlahnya tidak kurang dari 68%. Sehingga Apache merupakan web server yang banyak digemari oleh penggunanya. Selain itu apache merupakan web server yang kompak, modular dan mengikuti standar protocol HTTP.

* 1. MySQL

MySQL merupakan sebuah perangkat lunak system manajemen basis data SQL (Structured Query Language) yang di distribusikan secara gratis. SQL merupakan sebuah inti pengoperasian basis data agar dapat dikerjakan secara mudan dan otomatis. Pengguna dapat secara bebas menggunakan MySQL dengan catatan tidak boleh dijadikan produk turunan yang bersifat komersial.

* 1. Html

Hipertetx Markup Language atau biasa disebut HTML merupakan bahasa dasar untuk web scripting bersifat Client. HTML pada dasarnya merupakan dokumen ASCII atau teks biasa, yang dirancang untuk tidak bergantung pada suatu system operasi tertentu. HTML dibuat oleh Tim Berners-Lee dan dipopulerkan pertama kali oleh web browser Mosaic.

* 1. PHP

PHP dikembangkan pada tahun 1995 oleh Rasmus Lerdorf. Kepanjangan dari PHP yaitu Hypertext Preprocessor. Pada mulanya PHP merupakan kependekan dari Personal Home Page. PHP adalah bahasa script yang dapat disisipkan ke dalam HTML. PHP banyak digunakan untuk situs web dinamis. PHP diproses pada komputer server (server side scripting) yang berari seluruh script php diletakkan pada computer server dan diterjemahkan oleh web server terlebih dahulu sebelum dikirim ke web browser client.

### Pengujian

* 1. Debugging

Secara garis besar debugging merupakan proses penanganan error. Debugging terjadi sebagai konsekuensi testing yang berhasil. Apabila proses testing menemukan error maka debugging merupakan proses menghilangkan error. Proses debugging merupakan proses untuk mencocokkan indikasi dengan penyebab sehingga dapat mengarah pembenahan kesalahan.

* 1. Testing

Testing atau pengujian dalam sebuah system informasi itu sangat penting dan diperlukan untuk membuktikan bahwa sebuah system informasi tersebut layak digunakan. Secara umum testing software dapat dibagi menjadi tiga kategori yaitu testing saat input data, testing saat pemrosesan dan testing saat output. Testing input data meliputi Tindakan untuk menguji edit dan control dalam pemasukan data. Testing saat pemrosesan bertujuan untuk meyakinkan bahwa program telah bekerja seperti yang diharapkan. Sedangkan testing saat output bertujuan untuk meyakinkan bahwa laporan yang dihasilkan telah dibuat dengan format yang benar dan mempunyai informasi yang valid.

# BAB III PELAKSANAAN KEGIATAN

## Gambaran Instansi

### Profil

Refactory didirikan pada tahun 2016 di Surabaya dan sekarang juga terdapat di Yogyakarta yang terletak di Jl. Palagan Tentara Pelajar, Km. 9,8 Ngaglik, Kab. Sleman, DIY 55581. Gambar 3.1 merupakan gambar kantor gedung Refactory.

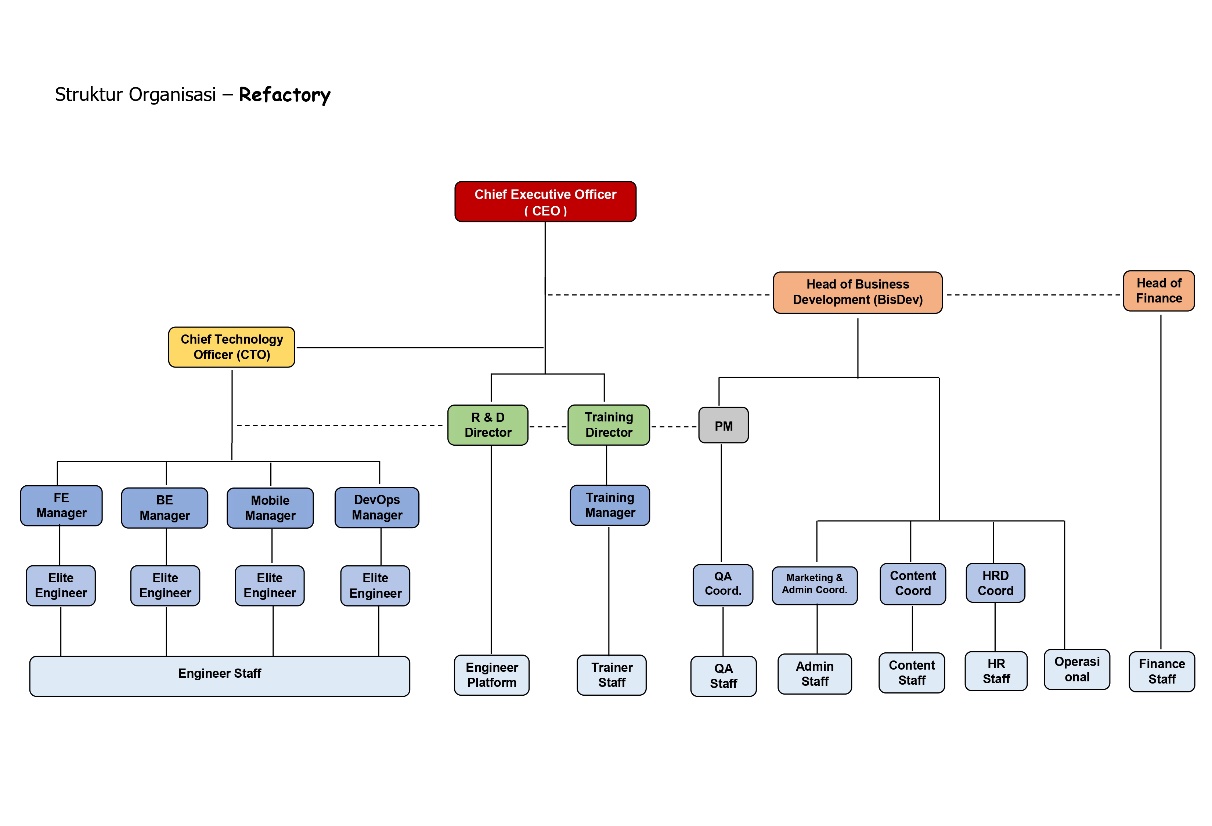


Gambar 3.. Kantor Gedung Refactory

Refactory merupakan perusahaan edukasi dan teknologi yang menyediakan layanan course maupun custom training. Course maupun custom training yang disediakan diantaranya yaitu Back-End Developer, Front-End Developer, QA Engineer dan beberapa bidang TI lainnya.

### Struktur

Struktur organisasi sangat berperan penting dalam sebuah organisasi, perusahaan maupun instansi demi memajukan dan menjamin kelancaran kerja pada sebuah perusahaan, dengan adanya struktur organisasi perusahaan akan lebih mudah dalam hal pembagian kerja atau tugas sesuai dengan kebutuhan perusahaan tersebut sehingga kegiatan operasional dalam perusahaan tersebut berjalan lancar dan terciptanya tujuan dari perusahaan. Gambar 3.2 merupakan gambar struktur organisasi Refactory.



Gambar 3.0.2 Struktur Organisasi

### Website

Refactory juga menyediakan website bagi pengguna yang ingin mengetahui dan mempelajari course-course pemrograman dengan materi yang sangat beragam. Adapaun alamat websitenya adalah <https://refactory.id/>. Gambar website refactory ditunjukkan pada gambar 3.3.



Gambar 3.. Website

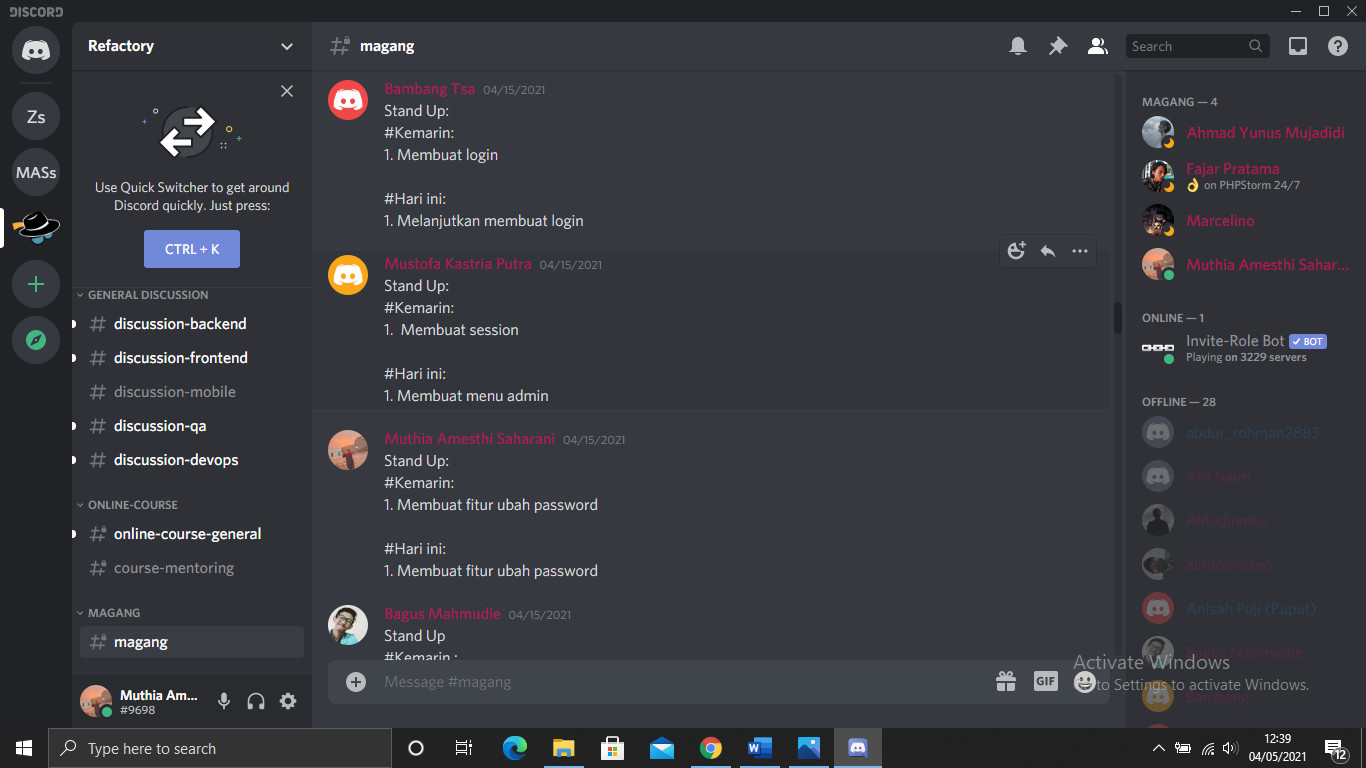
## Paparan Kegiatan

Gambar 3.4 merupakan gambar kegiatan meeting yang dilakukan satu minggu sekali yang bisanya dilakukan pada hari Jum’at.



Gambar 3.. Meeting

Gambar 3.5 merupakan gambar kegiatan stand up untuk melaporkan tugas apa yang dilakukan kemarin dan tugas apa yang akan di lakukan hari ini.



Gambar 3.. Stand up

Selama pandemic covid-19 kegiatan magang mahasiswa di Refactory dilakukan secara online atau daring. Mahasiswa yang melakukan kegiatan magang di Refactory mendapatkan banyak ilmu pengetahuan, pengalaman baru serta akses gratis seluruh materi-materi yang ada di situs web Refactory Course. Selain itu mahasiswa magang diberikan kesempatan untuk mempelajari materi-materi dan mengerjakan soal-soal dan projek-projek kecil yang ada di situs web Refactory sebelum diberikan projek akhir. Mahasiswa magang juga mempunyai kewajiban setiap hari pada pukul 09.00 – 10.00 untuk melakukan stand up di discord yang berisikan laporan kegiatan magang di hari sebelumnya dan kegiatan yang akan di lakukan hari selanjutnya. Selain itu mahasiswa magang juga wajib online discord pada waktu core time yaitu pada pukul 13.00 – 16.00 jika sewaktu-waktu diadakan presentasi dadakan. Kemudian pada hari rabu dan jum’at sore diadakan live discussion via zoom meeting yang membahas tentang hal-hal yang berkaitan dengan teknologi informasi.

## Jadwal Kegiatan

Jadwal kegiatan magang ditunjukkan pada tabel 3.1.

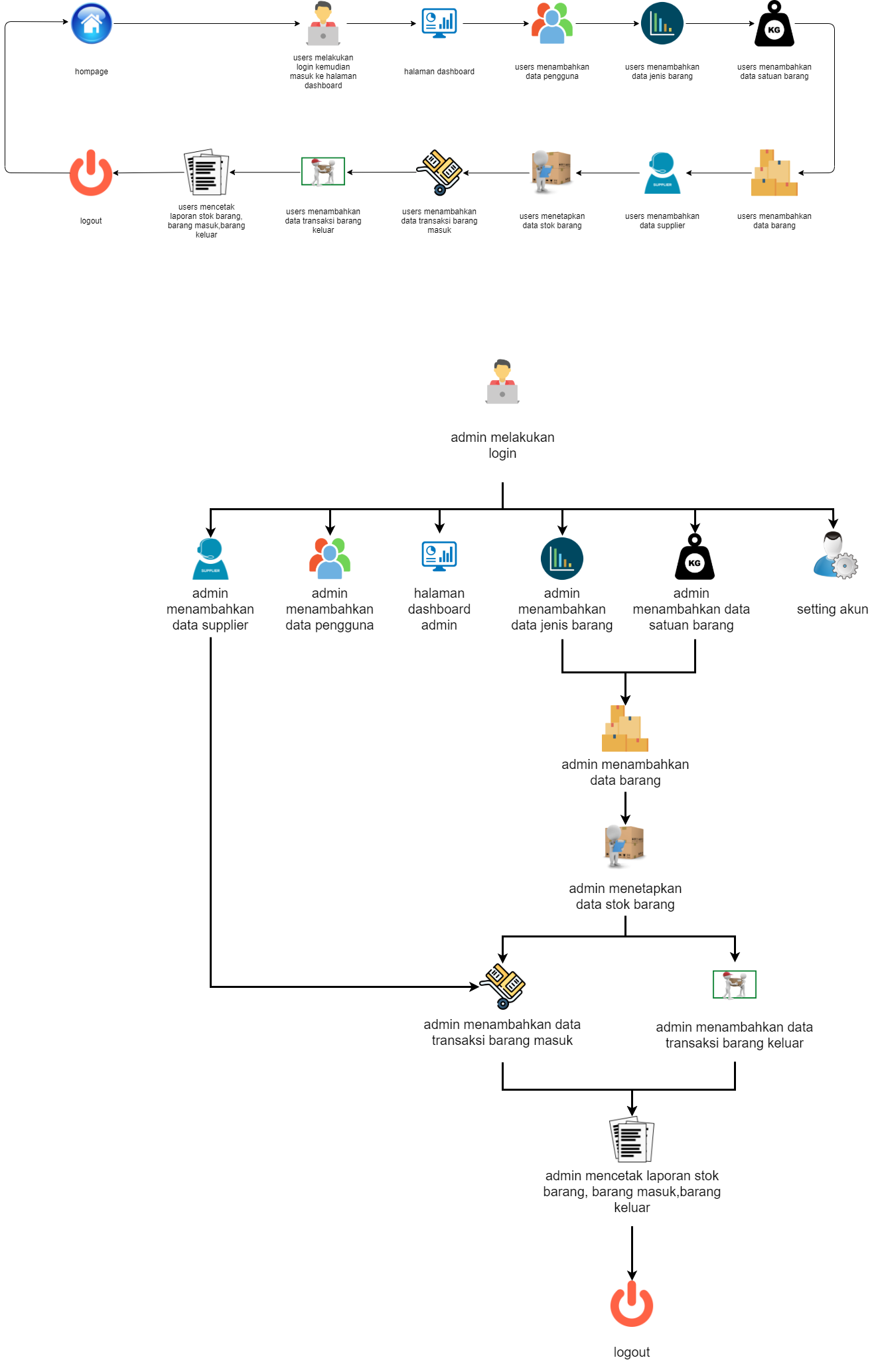
Table 3.. Jadwal kegiatan

|  |  |  |
| --- | --- | --- |
| No | Waktu Pengerjaan | Uraian Tugas |
| 1 | 18-Jan-2021 s.d 22-Jan-2021 | * Menyelesaikan fundamental course * Membuat CV di dalam github pages * Mempelajari PHP Basic Course * Mengerjakan Task Logic * Mempelajari course js introduction * Mempelajari typescript course |
| 2 | 25-Jan-2021 s.d 29-Jan-2021 | * Mengerjakan challenge typescript * Mempelajari node.js course * Mempelajari express.js course * Mempelajari express.js intermediate course * Mempelajari Laravel course * Mengerjakan challenge course Laravel for beginner |
| 3 | 02-Feb-2021 s.d 05-Feb-2021 | * Mengerjakan challenge course Laravel for beginner |
| 4 | 08-Feb-2021 s.d 16-Feb-2021 | * Mempelajari Laravel * Mempelajari login dan register pada Laravel/UI * Mempelajari blade templating Laravel * Brainstorming ide sistem informasi pergudangan bahan makanan kering |
| 5 | 17-Feb-2021 s.d 25-Feb-2021 | * Membuat alur sistem * Membuat erd * Membuat proses bisnis * Membuat migration pada Laravel * Membuat base template frontend * Membuat seeder database |
| 6 | 25-Feb-2021 s.d 19-Mar-2021 | * Mengerjakan CRUD (data pegawai,data barang, data jenis, data satuan, transaksi, data supplier) |
| 7 | 29-Mar-2021 s.d 09-Apr-2021 | * Perhitungan data stok barang * Membuat report berdasarkan tanggal * Membuat halaman dashboard berdasarkan role * Membuat landing page * Membuat menu ubah password |
| 8 | 12-Apr-2021 s.d 19-Apr-2021 | * Debugging backend dan frontend * Presentasi projek ke instansi * Membuat laporan magang |

# BAB IV HASIL DAN PEMBAHASAN

## Perancangan Sistem

### Proses Bisnis



Gambar 4.. Bisnis proses admin

Penjelasan :

Admin melakukan login kemudian jika berhasil maka akan beralih ke halaman dashboard admin. Setelah itu admin dapat menambahkan data pengguna, data jenis, data satuan, dan data supplier. Untuk menambahkan data barang admin harus menambahkan data jenis dan data satuan. Kemudian admin dapat menetapkan stok barang dan menambahkan data transaksi keluar masuk barang. Lalu admin dapat mencetak laporan stok barang, barang masuk dan barang keluar. Untuk mencetak laporan admin juga dapat mencetaknya berdasarkan range tanggal. Yang terakhir admin dapat mengubah password dan juga profil kemudian logout.

1. **Kebutuhan Fungsional**

Kebutuhan Fungsional yang terdapat pada sistem informasi pergudangan bahan makanan kering adalah sebagai berikut:

Table 4.. Kebutuhan Fungsional

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Kode** | No FR | **Jurnal** | **Aktor** | Dependensi |
| 1 | FR 01 | Sistem mampu menerima login | Admin, Pegawai |  |
| 2 | FR 02 | Sistem mampu menampilkan data pengguna | Admin | FR 01 |
| 3 | FR 03 | Sistem mampu menambahkan data pengguna | Admin | FR 01, FR 02 |
| 4 | FR 04 | Sistem mampu mengubah status pengguna aktif dan tidak aktif | Admin | FR 01, FR 02 |
| 5 | FR 05 | Sistem mampu megedit data pengguna | Admin | FR 01, FR 02 |
| 6 | FR 06 | Sistem mampu menghapus data pengguna | Admin | FR 01, FR 02 |
| 7 | FR 07 | Sistem mampu menampilkan data barang | Admin, Pegawai | FR 01 |
| 8 | FR 08 | Sistem mampu menambahkan data barang | Admin, Pegawai | FR 01, FR 07 |
| 9 | FR 09 | Sistem mampu mengedit data barang | Admin, Pegawai | FR 01, FR 07 |
| 10 | FR 10 | Sistem mampu menghapus data barang | Admin, Pegawai | FR 01, FR 07 |
| 11 | FR 11 | Sistem mampu menampilkan data jenis | Admin, Pegawai | FR 01 |
| 12 | FR 12 | Sistem mampu menambahkan data jenis | Admin, Pegawai | FR 01, FR 11 |
| 13 | FR 13 | Sistem mampu mengedit data jenis | Admin, Pegawai | FR 01, FR 11 |
| 14 | FR 14 | Sistem mampu menghapus data jenis | Admin, Pegawai | FR 01, FR 11 |
| 15 | FR 15 | Sistem mampu menampilkan data satuan | Admin, Pegawai | FR 01 |
| 16 | FR 16 | Sistem mampu menambahkan data satuan | Admin, Pegawai | FR 01, FR 15 |
| 17 | FR 17 | Sistem mampu mengedit data satuan | Admin, Pegawai | FR 01, FR 15 |
| 18 | FR 18 | Sistem mampu menghapus data satuan | Admin, Pegawai | FR 01, FR 15 |
| 19 | FR 19 | Sistem mampu menampilkan data supplier | Admin, Pegawai | FR 01 |
| 20 | FR 20 | Sistem mampu menambahkan data supplier | Admin, Pegawai | FR 01, FR 19 |
| 21 | FR 21 | Sistem mampu mengedit data supplier | Admin, Pegawai | FR 01, FR 19 |
| 22 | FR 22 | Sistem mampu menghapus data supplier | Admin, Pegawai | FR 01, FR 19 |
| 23 | FR 23 | Sistem mampu menampilkan data stok barang | Admin, Pegawai | FR 01 |
| 24 | FR 24 | Sistem mampu menambahkan data stok barang | Admin, Pegawai | FR 01, FR 23 |
| 25 | FR 25 | Sistem mampu mengedit data stok barang | Admin, Pegawai | FR 01, FR 23 |
| 26 | FR 26 | Sistem mampu menghapus data stok barang | Admin, Pegawai | FR 01, FR 23 |
| 27 | FR 27 | Sistem mampu menampilkan data barang masuk | Admin, Pegawai | FR 01 |
| 28 | FR 28 | Sistem mampu menambahkan data barang masuk | Admin, Pegawai | FR 01, FR 27 |
| 29 | FR 29 | Sistem mampu mengedit data barang masuk | Admin, Pegawai | FR 01, FR 27 |
| 30 | FR 30 | Sistem mampu menghapus data barang masuk | Admin, Pegawai | FR 01, FR 27 |
| 31 | FR 31 | Sistem mampu menampilkan data barang keluar | Admin, Pegawai | FR 01 |
| 32 | FR 32 | Sistem mampu menambahkan data barang keluar | Admin, Pegawai | FR 01, FR 31 |
| 33 | FR 33 | Sistem mampu mengedit barang keluar | Admin, Pegawai | FR 01, FR 31 |
| 34 | FR 34 | Sistem mampu menghapus barang keluar | Admin, Pegawai | FR 01, FR 31 |
| 35 | FR 35 | Sistem mampu membuat laporan stok barang berdasarkan range tanggal | Admin, Pegawai | FR 01, FR 23 |
| 36 | FR 36 | Sistem mampu membuat laporan barang masuk berdasarkan range tanggal | Admin, Pegawai | FR 01, FR 27 |
| 37 | FR 37 | Sistem mampu membuat laporan barang keluar berdasarkan range tanggal | Admin, Pegawai | FR 01, FR 31 |
| 38 | FR 38 | Sistem mampu mengubah password | Admin, Pegawai | FR 01 |
| 39 | FR 39 | Sistem mampu melakukan logout | Admin, Pegawai | FR 01 |

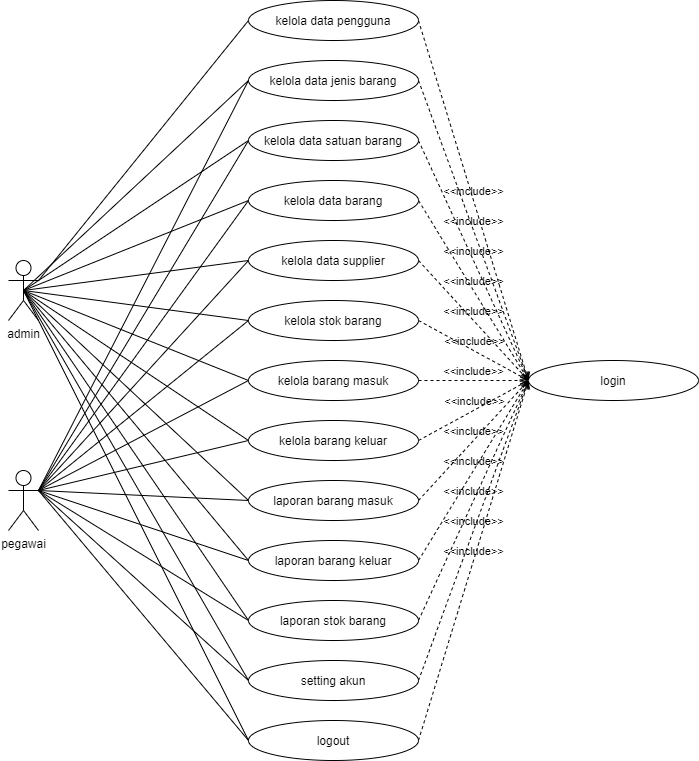
1. **Kebutuhan non Fungsional**

Dalam merancang dan membangun sistem informasi pergudangan bahan makanan kering ini dibutuhkan beberapa perangkat sebagai berikut :

1. Perangkat Keras
2. Processor A6-5200 2.0 GHz.
3. Harddisk dengan ruang kosong minimal 1GB.
4. RAM 2 GB.
5. Perangkat Lunak
6. Windows 10
7. Visual Studio Code
8. Bahasa pemrograman HTML
9. Bahasa pemrograman PHP
10. Database Mysql
11. Xampp
12. Browser
13. Ms. Visio
14. Balsamiq wireframe
15. Git bash dan github dekstop

### Use Case Diagram

Dari use case diagram yang telah dibuat tugas admin dapat mengelola data pengguna, data jenis barang, data satuan barang, data barang, data supplier, data barang masuk, data barang keluar, data stok barang hingga mencetak laporan stok barang, barang masuk dan barang keluar. Sedangkan tugas pegawai gudang yaitu sama dengan tugas admin mengelola semua data kecuali data pengguna.



Gambar 4.. Use case diagram

### Use Case Text

#### Use Case Text Login

Table 4.. Use case text login

|  |  |  |
| --- | --- | --- |
| Name Of Usecase | Login | |
| Aktor | Admin, Pegawai | |
| Kondisi Awal | Aktor berada pada halaman landing page | |
| Skenario | Actor | Sistem |
| 1. admin mengeklik button login | 1. sistem menampilkan halaman login |
| 1. admin memasukkan email dan password | 1. sistem memeriksa account pengguna apakah valid |
|  | 1. sistem menampilkan dashboard berdasarkan role nya masing-masing |
| Pengecualian | Apabila account tidak valid maka aktor akan kembali ke langkah nomor 2 | |
| Kondisi Akhir | Actor berhasil masuk ke halaman dashboard sesuai dengan role nya masing-masing | |

#### Use Case Text Tambah Data Pengguna

Table 4.. Use case text tambah data pengguna

|  |  |  |
| --- | --- | --- |
| Name Of Usecase | Tambah data pengguna | |
| Aktor | Admin | |
| Kondisi Awal | Admin berada pada halaman dashboard admin | |
| Skenario | Actor | Sistem |
| 1. admin mengeklik menu data pengguna | 1. sistem menampilkan data pengguna |
| 1. admin mengeklik button tambah data | 1. sistem menampilkan form tambah data pengguna |
| 1. admin mengisi form tambah data pegawai kemudian klik submit | 1. sistem menyimpan data pengguna ke dalam database dan menampilkan dalam tabel data pengguna |
| Pengecualian | Jika data yang diisikan tidak valid maka sistem akan tetap berada pada halaman form tambah data pengguna | |
| Kondisi Akhir | Actor berhasil menambahkan data pengguna | |

#### Use Case Text Tambah Data Jenis

Table 4.. Use case text tambah data jenis

|  |  |  |
| --- | --- | --- |
| Name Of Usecase | Tambah data jenis | |
| Aktor | Admin, Pegawai | |
| Kondisi Awal | Admin/Pegawai berada pada halaman dashboard | |
| Skenario | Aktor | Sistem |
| 1. admin/pegawai mengeklik menu data jenis | 1. sistem menampilkan data jenis |
| 1. admin/pegawai mengeklik button tambah data | 1. sistem menampilkan form tambah data jenis |
| 1. admin/pegawai mengisi form tambah data jenis kemudian klik submit | 1. sistem menyimpan data jenis ke dalam database dan menampilkan dalam tabel data jenis |
| Pengecualian | Jika data yang diisikan tidak valid maka sistem akan tetap berada pada halaman form tambah data jenis | |
| Kondisi Akhir | Admin/pegawai berhasil menambahkan data jenis | |

#### Use Case Text Tambah Data Satuan

Table 4.. Use case text tambah data satuan

|  |  |  |
| --- | --- | --- |
| Name Of Usecase | Tambah data satuan | |
| Aktor | Admin, Pegawai | |
| Kondisi Awal | Admin/Pegawai berada pada halaman dashboard | |
| Skenario | Aktor | Sistem |
| 1. admin/pegawai mengeklik menu data satuan | 1. sistem menampilkan data satuan |
| 1. admin/pegawai mengeklik button tambah data | 1. sistem menampilkan form tambah data satuan |
| 1. admin/pegawai mengisi form tambah data satuan kemudian klik submit | 1. sistem menyimpan data satuan ke dalam database dan menampilkan dalam tabel data satuan |
| Pengecualian | Jika data yang diisikan tidak valid maka sistem akan tetap berada pada halaman form tambah data satuan | |
| Kondisi Akhir | Admin/pegawai berhasil menambahkan data satuan | |

#### Use Case Text Tambah Data Barang

Table 4.. Use case text tambah data barang

|  |  |  |
| --- | --- | --- |
| Name Of Usecase | Tambah data barang | |
| Aktor | Admin, Pegawai | |
| Kondisi Awal | Admin/Pegawai berada pada halaman dashboard | |
| Skenario | Aktor | Sistem |
| 1. admin/pegawai mengeklik menu data barang | 1. sistem menampilkan data barang |
| 1. admin/pegawai mengeklik button tambah data | 1. sistem menampilkan form tambah data barang |
| 1. admin/pegawai mengisi form tambah data barang kemudian klik submit | 1. sistem menyimpan data barang ke dalam database dan menampilkan dalam tabel data barang |
| Pengecualian | Jika data yang diisikan tidak valid maka sistem akan tetap berada pada halaman form tambah data barang | |
| Kondisi Akhir | Admin/pegawai berhasil menambahkan data barang | |

#### Use Case Text Tambah Data Supplier

Table 4.. Use case text tambah data supplier

|  |  |  |
| --- | --- | --- |
| Name Of Usecase | Tambah data supplier | |
| Aktor | Admin, Pegawai | |
| Kondisi Awal | Admin/Pegawai berada pada halaman dashboard | |
| Skenario | Aktor | Sistem |
| 1. admin/pegawai mengeklik menu data supplier | 1. sistem menampilkan data supplier |
| 1. admin/pegawai mengeklik button tambah data | 1. sistem menampilkan form tambah data supplier |
| 1. admin/pegawai mengisi form tambah data supplier kemudian klik submit | 1. sistem menyimpan data supplier ke dalam database dan menampilkan dalam tabel data supplier |
| Pengecualian | Jika data yang diisikan tidak valid maka sistem akan tetap berada pada halaman form tambah data supplier | |
| Kondisi Akhir | Admin/pegawai berhasil menambahkan data supplier | |

#### Use Case Text Tambah Data Barang Masuk

Table 4.. Use case text tambah data barang masuk

|  |  |  |
| --- | --- | --- |
| Name Of Usecase | Tambah data barang masuk | |
| Aktor | Admin, Pegawai | |
| Kondisi Awal | Admin/Pegawai berada pada halaman dashboard | |
| Skenario | Aktor | Sistem |
| 1. admin/pegawai mengeklik menu data barang masuk | 1. sistem menampilkan data barang masuk |
| 1. admin/pegawai mengeklik button tambah data | 1. sistem menampilkan form tambah data barang masuk |
| 1. admin/pegawai mengisi form tambah data barang masuk kemudian klik submit | 1. sistem menyimpan data barang masuk ke dalam database dan menampilkan dalam tabel data barang masuk |
| Pengecualian | Jika data yang diisikan tidak valid maka sistem akan tetap berada pada halaman form tambah data barang masuk | |
| Kondisi Akhir | Admin/pegawai berhasil menambahkan data barang masuk | |

#### Use Case Text Tambah Data Barang Keluar

Table 4.. Use case text tambah data barang keluar

|  |  |  |
| --- | --- | --- |
| Name Of Usecase | Tambah data barang keluar | |
| Aktor | Admin, Pegawai | |
| Kondisi Awal | Admin/Pegawai berada pada halaman dashboard | |
| Skenario | Aktor | Sistem |
| 1. admin/pegawai mengeklik menu data barang keluar | 1. sistem menampilkan data barang keluar |
| 1. admin/pegawai mengeklik button tambah data | 1. sistem menampilkan form tambah data barang keluar |
| 1. admin/pegawai mengisi form tambah data barang keluar kemudian klik submit | 1. sistem menyimpan data barang keluar ke dalam database dan menampilkan dalam tabel data barang keluar |
| Pengecualian | Jika data yang diisikan tidak valid maka sistem akan tetap berada pada halaman form tambah data barang keluar | |
| Kondisi Akhir | Admin/pegawai berhasil menambahkan data barang keluar | |

#### Use Case Text Tambah Data Stok Barang

Table 4.. Use case text tambah data stok barang

|  |  |  |
| --- | --- | --- |
| Name Of Usecase | Tambah data stok barang | |
| Aktor | Admin, Pegawai | |
| Kondisi Awal | Admin/Pegawai berada pada halaman dashboard | |
| Skenario | Aktor | Sistem |
| 1. admin/pegawai mengeklik menu data stok barang | 1. sistem menampilkan data stok barang |
| 1. admin/pegawai mengeklik button tambah data | 1. sistem menampilkan form tambah data stok barang |
| 1. admin/pegawai mengisi form tambah data stok barang kemudian klik submit | 1. sistem menyimpan data stok barang ke dalam database dan menampilkan dalam tabel data stok barang |
| Pengecualian | Jika data yang diisikan tidak valid maka sistem akan tetap berada pada halaman form tambah data stok barang | |
| Kondisi Akhir | Admin/pegawai berhasil menambahkan data stok barang | |

#### Use Case Text Cetak Laporan Pertanggal

Table 4.. Use case text cetak laporan pertanggal

|  |  |  |
| --- | --- | --- |
| Name Of Usecase | Cetak Laporan Petanggal | |
| Aktor | Admin, Pegawai | |
| Kondisi Awal | Admin/Pegawai berada pada halaman dashboard | |
| Skenario | Aktor | Sistem |
| 1. admin/pegawai mengeklik menu laporan pertanggal yang terdapat pada halaman stok barang, barang masuk dan barang keluar | 1. sistem menampilkan form tanggal awal dan tanggal akhir |
| 1. admin/pegawai mengisi form dan mengeklik button cetak | 1. sistem menampilkan hasil laporan pertanggal |
| 1. admin/pegawai dapat mengunduh dan menyimpan laporan |  |
| Pengecualian | Jika tanggal awal dan tanggal akhir yang diisikan tidak sesuai dengan data maka hasil laporan akan kosong | |
| Kondisi Akhir | Admin/pegawai berhasil menampilkan, mengunduh dan menyimpan hasil laporan pertanggal | |

#### Use Case Text Setting Akun

Table 4.. Use case text setting akun

|  |  |  |
| --- | --- | --- |
| Name Of Usecase | Akun | |
| Aktor | Admin, Pegawai | |
| Kondisi Awal | Admin/Pegawai berada pada halaman dashboard | |
| Skenario | Aktor | Sistem |
| 1. admin/pegawai mengeklik menu akun | 1. sistem menampilkan form ubah password dan ubah profil |
| 1. admin/pegawai mengisi form ubah password dan ubah profil lalu mengeklik simpan | 1. sistem menampilkan alert “password berhasil diubah” dan “profil berhasil diubah” |
| Pengecualian | * password baru tidak boleh sama dengan password lama * password baru memiliki panjang minimal 6 karakter | |
| Kondisi Akhir | Admin/pegawai berhasil mengubah password dan profil | |

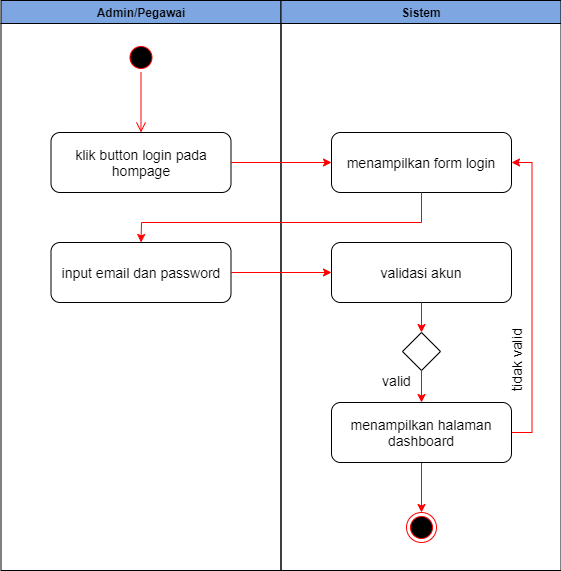
#### Use Case Text Logout

Table 4.. Use case text logout

|  |  |  |
| --- | --- | --- |
| Name Of Usecase | Logout | |
| Aktor | Admin, Pegawai | |
| Kondisi Awal | Admin/Pegawai berada pada halaman dashboard | |
| Skenario | Aktor | Sistem |
| 1. admin/pegawai mengeklik button logout | 1. sistem Kembali ke halaman homepage |
| Pengecualian | - | |
| Kondisi Akhir | Admin/pegawai berhasil melakukan logout | |

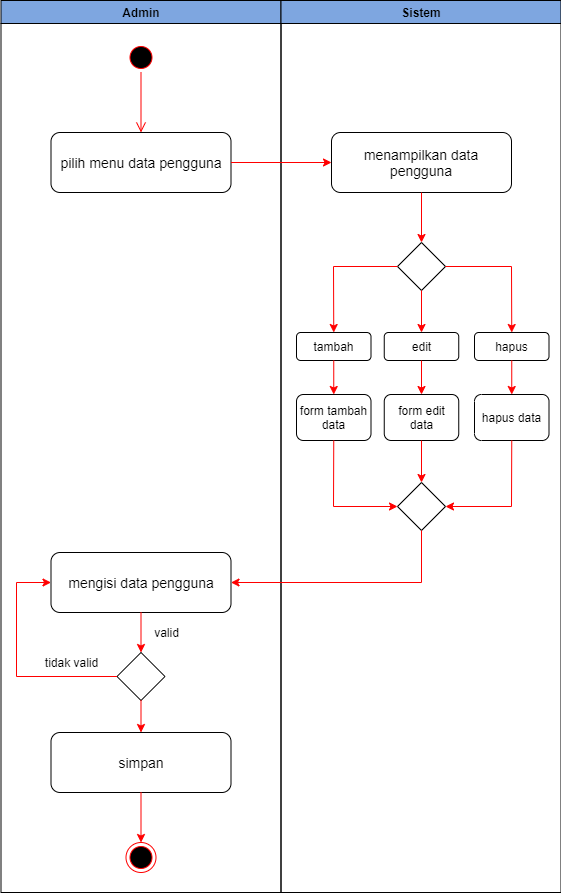
### Activity Diagram

#### Activity Diagram Menerima inputan login



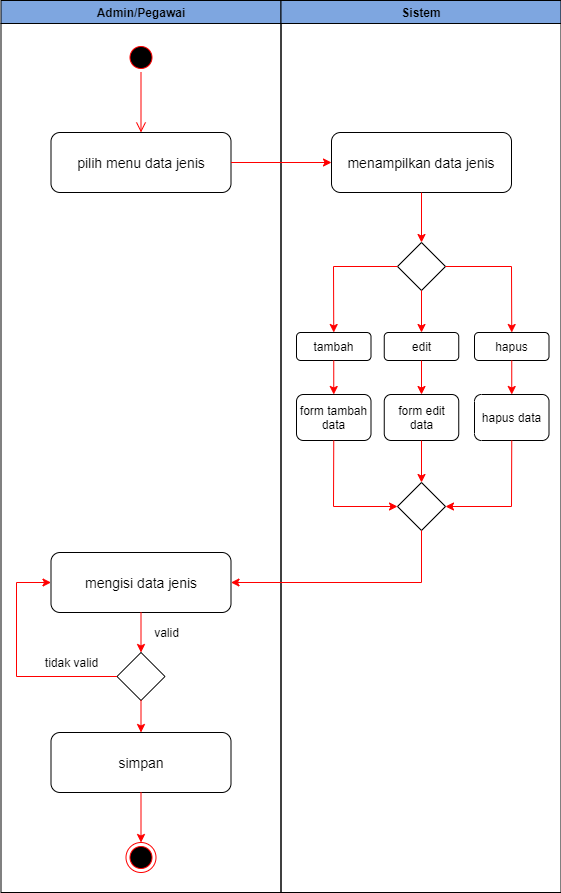
Gambar 4.. Activity diagram menerima inputan login

#### Activity Diagram Data Pengguna



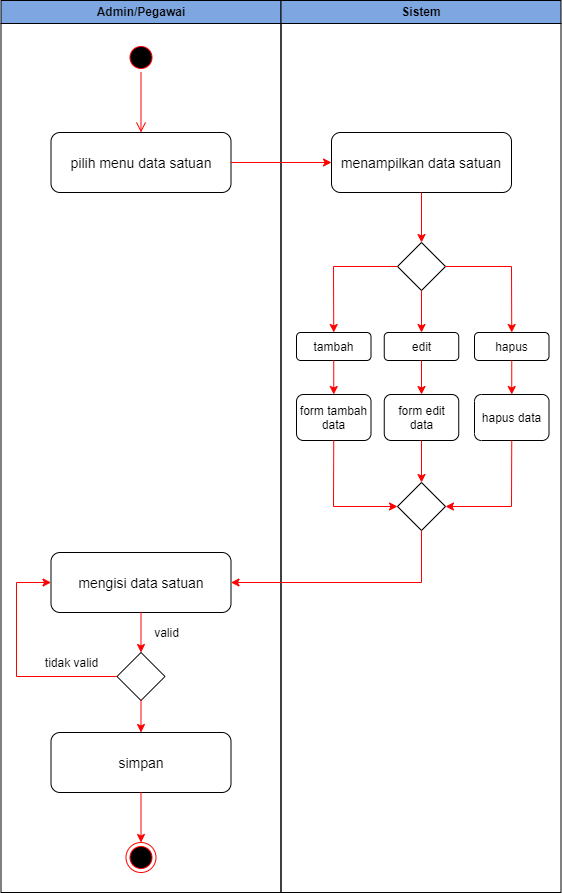
Gambar 4.. Activity diagram data pegawai

#### Activity Diagram Data Jenis Barang



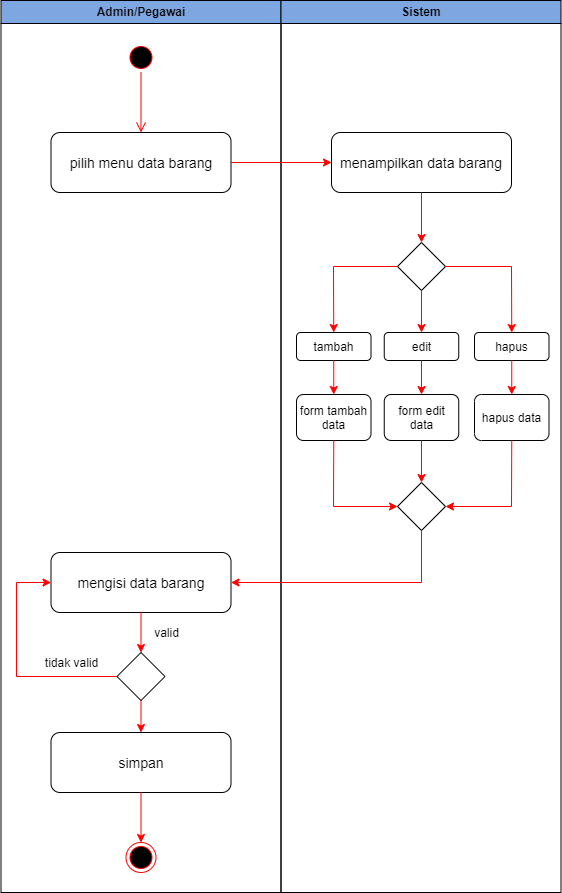
Gambar 4.. Activity diagram data pegawai

#### Activity Diagram Data Satuan Barang



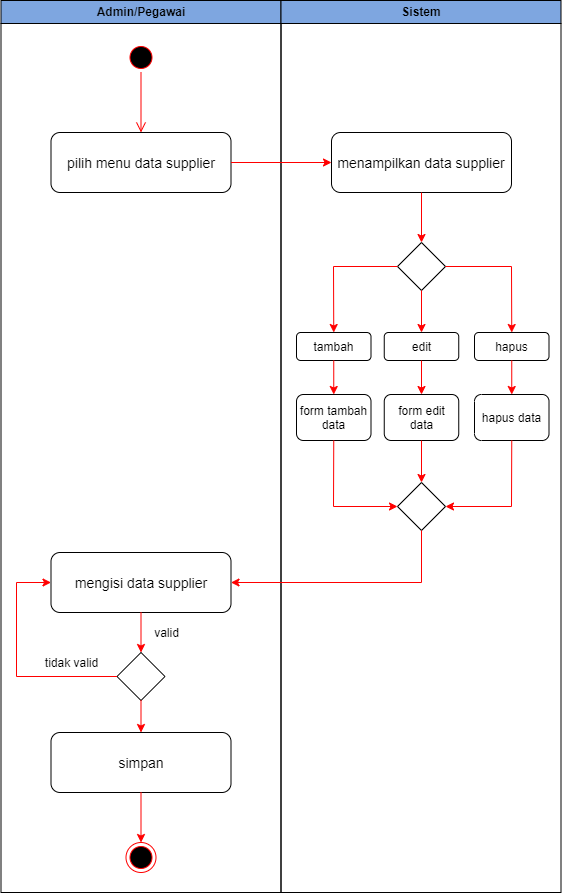
Gambar 4.. Activity diagram data pegawai

#### Activity Diagram Data Barang



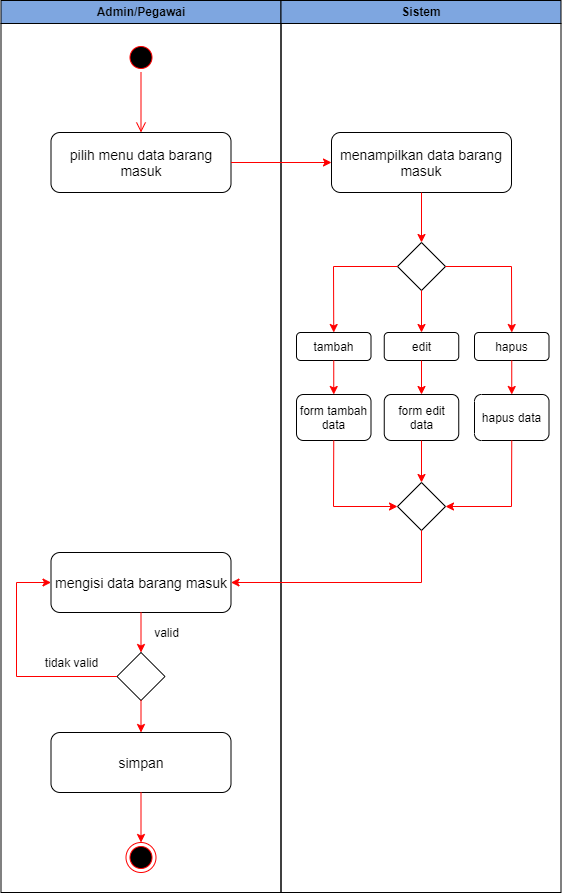
Gambar 4.. Activity diagram data barang

#### Activity Diagram Data Supplier



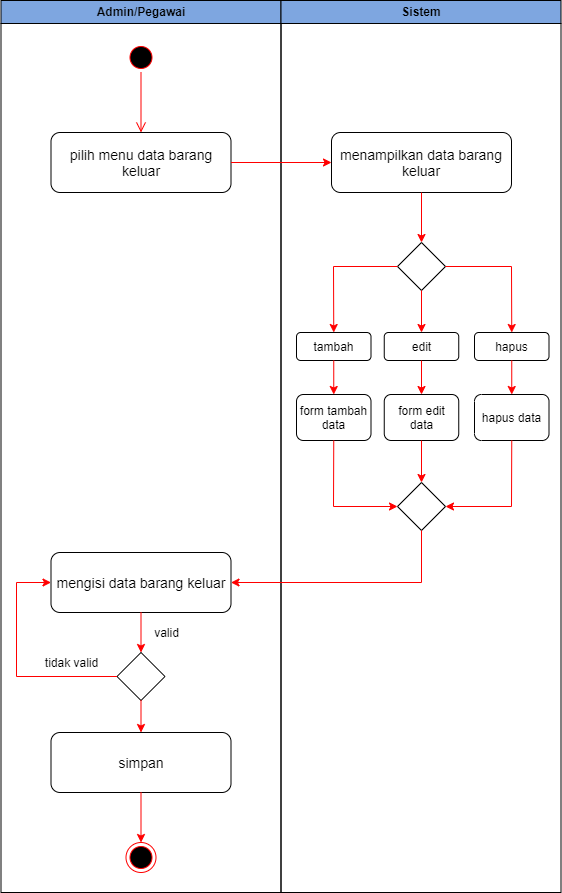
Gambar 4.. Activity diagram data supplier

#### Activity Diagram Data Barang Masuk



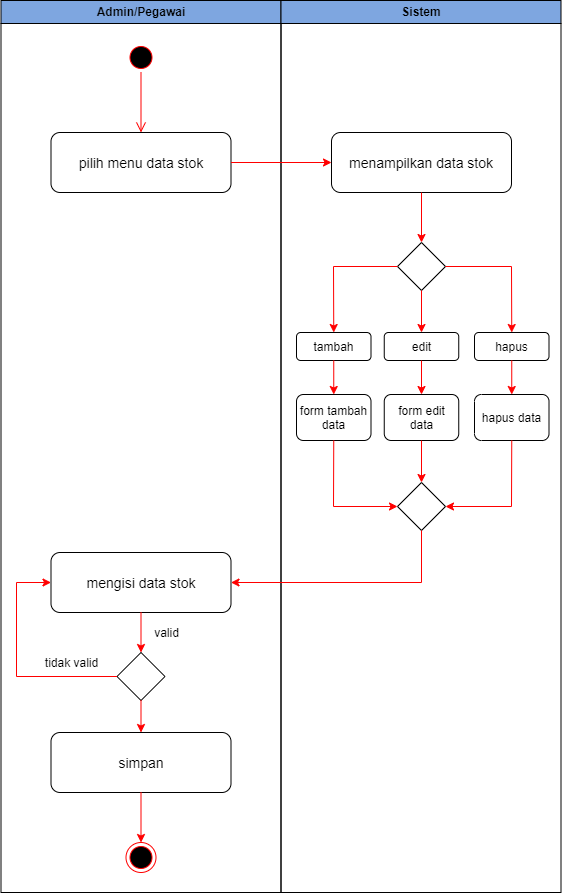
Gambar 4.. Activity diagram data barang masuk

#### Activity Diagram Data Barang Keluar



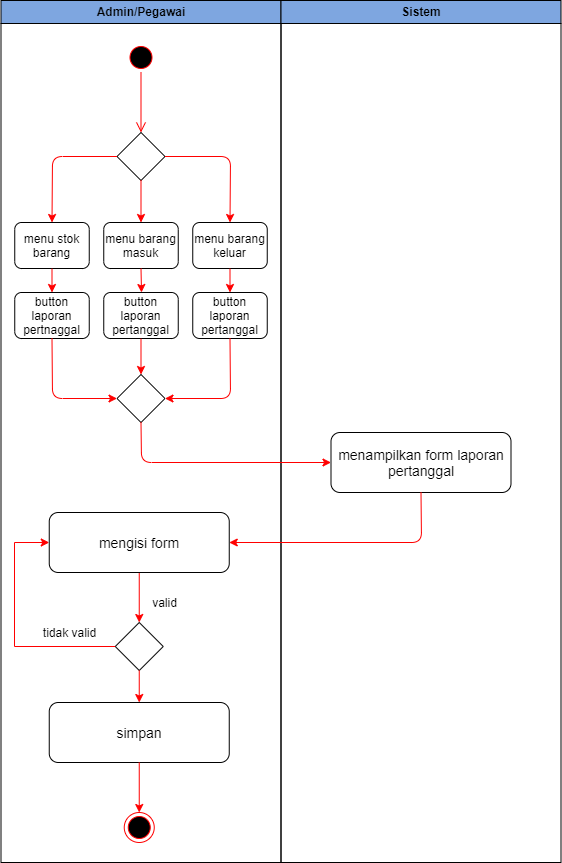
Gambar 4.. Activity diagram data barang keluar

#### Activity Diagram Data Stok Barang



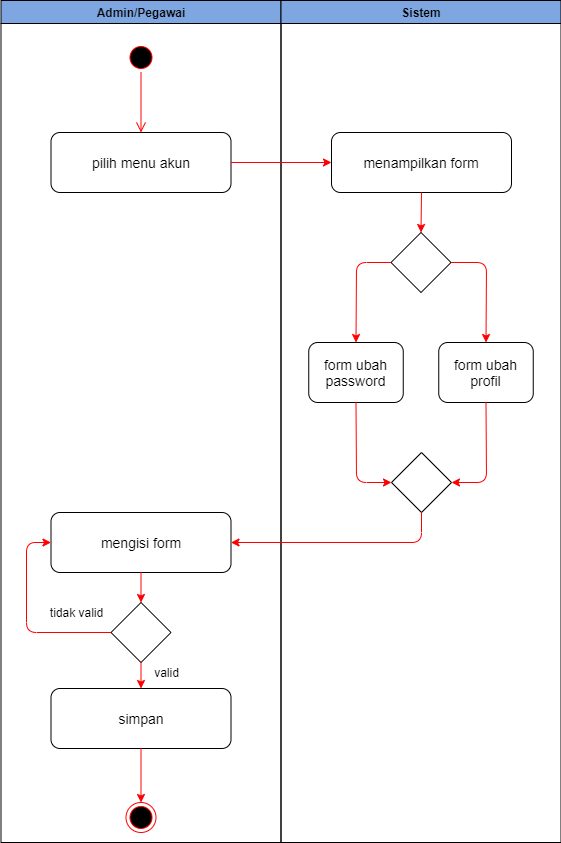
Gambar 4.. Activity diagram data stok barang

#### Activity Diagram Cetak Laporan Pertanggal



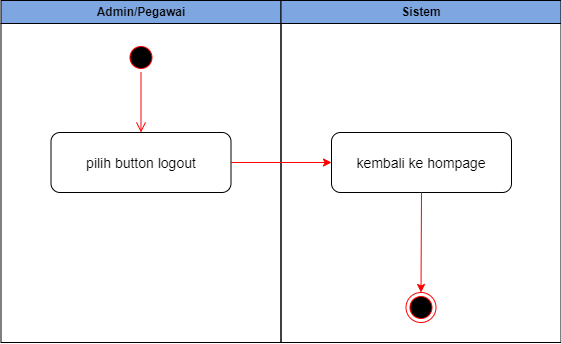
Gambar 4.. Activity diagram laporan pertanggal

#### Activity Diagram Setting Akun



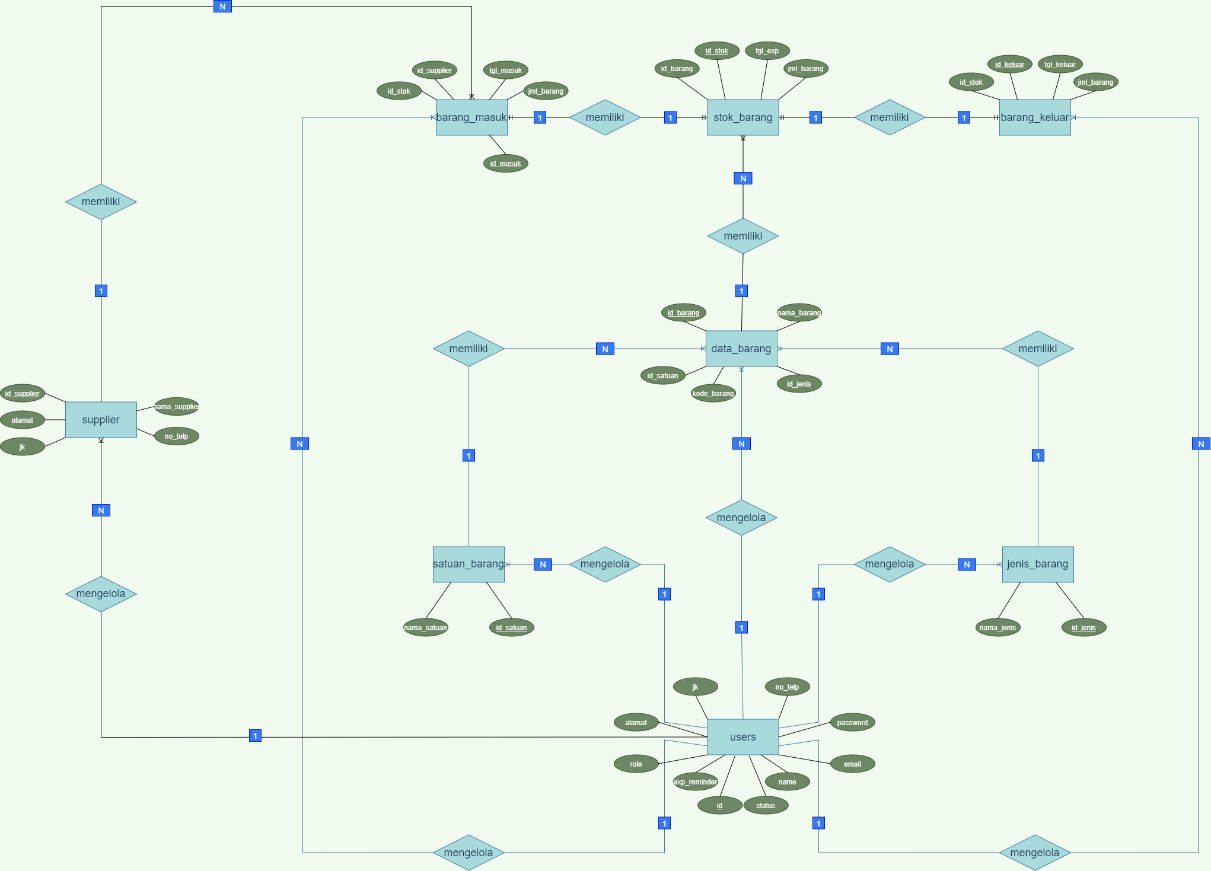
Gambar 4.. Activity diagram setting akun

#### Activity Diagram Logout



Gambar 4.. Activity diagram logout

### Entity Relational Diagram



Gambar 4.. Entity relational diagram

Dari perancangan ERD di atas didapat beberapa tabel yang saling berhubungan diantaranya yaitu tabel users, data barang, jenis barang, satuan barang, stok barang, barang masuk, barang keluar dan supplier.

1. Hubungan tabel users dengan tabel jenis barang

Ditunjukkan dengan kardinalitas one to many dimana satu users dapat mengelola banyak data jenis barang.

1. Hubungan tabel users dengan tabel satuan barang

Ditunjukkan dengan kardinalitas one to many dimana satu users dapat mengelola banyak data satuan barang.

1. Hubungan tabel users dengan tabel data barang

Ditunjukkan dengan kardinalitas one to many dimana satu users dapat mengelola banyak data barang.

1. Hubungan tabel users dengan tabel stok barang

Ditunjukkan dengan kardinalitas one to many dimana satu users dapat mengelola banyak data stok barang.

1. Hubungan tabel users dengan tabel supplier

Ditunjukkan dengan kardinalitas one to many dimana satu users dapat mengelola banyak data supplier.

1. Hubungan tabel users dengan tabel stok barang

Ditunjukkan dengan kardinalitas one to many dimana satu users dapat mengelola banyak data stok barang.

1. Hubungan tabel users dengan tabel barang masuk

Ditunjukkan dengan kardinalitas one to many dimana satu users dapat mengelola banyak data barang masuk.

1. Hubungan tabel users degan tabel barang keluar

Ditunjukkan dengan kardinalitas one to many dimana satu users dapat mengelola banyak data barang keluar.

1. Hubungan tabel data barang dengan tabel jenis barang

Ditunjukkan dengan kardnalitas one to many dimana satu jenis barang dapat dimiliki banyak data barang.

1. Hubungan tabel data barang dengan tabel satuan barang

Ditunjukkan dengan kardnalitas one to many dimana satu satuan barang dapat dimiliki banyak data barang.

1. Hubungan tabel data barang dengan tabel stok barang

Ditunjukkan dengan kardinalitas one to many dimana satu data barang dapat dimiliki banyak stok barang.

1. Hubungan tabel stok barang dengan tabel barang masuk

Ditunjukkan dengan kardinalitas one to one dimana satu jenis barang masuk hanya memasukkan satu jenis stok barang.

1. Hubungan tabel stok barang dengan tabel barang keluar

Ditunjukkan dengan kardinalitas one to one dimana satu jenis barang keluar hanya mengeluarkan satu jenis stok barang.

1. Hubungan tabel barang masuk dengan tabel supplier

Ditunjukkan dengan kardinalitas one to many dimana banyak barang masuk dimiliki satu supplier.

### Tabel Fisik

Dari perancangan ERD maka dihasilkan 8 tabel diantaranya yaitu tabel data users, tabel data barang, tabel barang masuk, tabel barang keluar, tabel jenis barang, tabel satuan barang, tabel stok barang dan tabel supplier.

#### Tabel Users

|  |  |
| --- | --- |
| Nama Tabel | : users |
| Deskripsi | : Tabel users adalah tabel yang berfungsi untuk menyimpan data users. Adapun table fisik dari table users adalah sebagai berikut. |
| Primary key | : id |
| Foregn key | : - |

Table 4.. Tabel users

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Type | Length | Key | Keterangan |
| id | bigint | 20 | Primary Key | Auto Increment |
| name | varchar | 225 |  |  |
| jk | enum(‘Laki-laki’,’Perempuan’) |  |  |  |
| alamat | varchar | 225 |  |  |
| no\_telp | varchar | 225 |  |  |
| email | varchar | 225 | unique |  |
| level\_user | Enum(‘admin’,’user’) |  |  |  |
| email\_verified\_at | timestamp |  |  |  |
| password | varchar | 225 |  |  |
| role | varchar | 225 |  |  |
| status | integer | 11 |  |  |
| remember\_token | varchar | 100 |  |  |
| created\_at | timestamp |  |  |  |
| update\_at | timestamp |  |  |  |

#### Tabel Data Barang

|  |  |
| --- | --- |
| Nama Tabel | : data\_barang |
| Deskripsi | : Tabel data\_barang adalah tabel yang berfungsi untuk menyimpan data barang. Adapun table fisik dari table data barang adalah sebagai berikut. |
| Primary key | : id\_barang |
| Foregn key | : id\_jenis, id\_satuan |

Table 4.. Tabel data barang

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Type | Length | Key | Keterangan |
| id\_barang | bigint | 20 | Primary Key | Auto Increment |
| kode\_barang | varchar | 225 |  |  |
| nama\_barang | varchar | 225 |  |  |
| id\_jenis | bigint | 20 | Foreign Key |  |
| id\_satuan | bigint | 20 | Foreign Key |  |
| created\_at | timestamp |  |  |  |
| update\_at | timestamp |  |  |  |

#### Tabel Jenis Barang

|  |  |
| --- | --- |
| Nama Tabel | : jenis\_barang |
| Deskripsi | : Tabel jenis\_barang adalah tabel yang berfungsi untuk menyimpan data jenis barang. Adapun table fisik dari table jenis barang adalah sebagai berikut. |
| Primary key | : id\_jenis |
| Foregn key | : - |

Table 4.. Tabel jenis barang

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Type | Length | Key | Keterangan |
| id\_jenis | bigint | 20 | Primary Key | Auto Increment |
| nama\_jenis | varchar | 225 |  |  |
| created\_at | timestamp |  |  |  |
| update\_at | timestamp |  |  |  |

#### Tabel Satuan Barang

|  |  |
| --- | --- |
| Nama Tabel | : satuan\_barang |
| Deskripsi | : Tabel satuan\_barang adalah tabel yang berfungsi untuk menyimpan data satuan barang. Adapun table fisik dari table satuan barang adalah sebagai berikut. |
| Primary key | : id\_satuan |
| Foregn key | : - |

Table 4.. Tabel satuan barang

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Type | Length | Key | Keterangan |
| id\_satuan | bigint | 20 | Primary Key | Auto Increment |
| nama\_satuan | varchar | 225 |  |  |
| created\_at | timestamp |  |  |  |
| update\_at | timestamp |  |  |  |

#### Tabel Barang Masuk

|  |  |
| --- | --- |
| Nama Tabel | : barang\_masuk |
| Deskripsi | : Tabel barang\_masuk adalah tabel yang berfungsi untuk menyimpan data barang masuk. Adapun table fisik dari table barang masuk adalah sebagai berikut. |
| Primary key | : id\_masuk |
| Foregn key | : id\_stok, id\_supplier |

Table 4.. Tabel barang masuk

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Type | Length | Key | Keterangan |
| id\_masuk | bigint | 20 | Primary Key | Auto Increment |
| jml\_barang | int | 11 |  |  |
| tgl\_masuk | date |  |  |  |
| id\_stok | bigint | 20 | Foreign Key |  |
| id\_supplier | bigint | 20 | Foreign Key |  |
| created\_at | timestamp |  |  |  |
| update\_at | timestamp |  |  |  |

#### Tabel Barang Keluar

|  |  |
| --- | --- |
| Nama Tabel | : barang\_keluar |
| Deskripsi | : Tabel barang\_keluar adalah tabel yang berfungsi untuk menyimpan data barang keluar. Adapun table fisik dari table barang keluar adalah sebagai berikut. |
| Primary key | : id\_keluar |
| Foregn key | : id\_stok |

Table 4.. Tabel barang keluar

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Type | Length | Key | Keterangan |
| id\_keluar | bigint | 20 | Primary Key | Auto Increment |
| jml\_barang | int | 11 |  |  |
| tgl\_keluar | date |  |  |  |
| id\_stok | bigint | 20 | Foreign Key |  |
| created\_at | timestamp |  |  |  |
| update\_at | timestamp |  |  |  |

#### Tabel Stok Barang

|  |  |
| --- | --- |
| Nama Tabel | : stok\_barang |
| Deskripsi | : Tabel stok\_barang adalah tabel yang berfungsi untuk menyimpan data stok barang. Adapun table fisik dari table stok barang adalah sebagai berikut. |
| Primary key | : id\_stok |
| Foregn key | : id\_barang |

Table 4.. Tabel stok barang

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Type | Length | Key | Keterangan |
| id\_stok | bigint | 20 | Primary Key | Auto Increment |
| jml\_barang | int | 11 |  |  |
| tgl\_exp | date |  |  |  |
| id\_barang | bigint | 20 | Foreign Key |  |
| created\_at | timestamp |  |  |  |
| update\_at | timestamp |  |  |  |

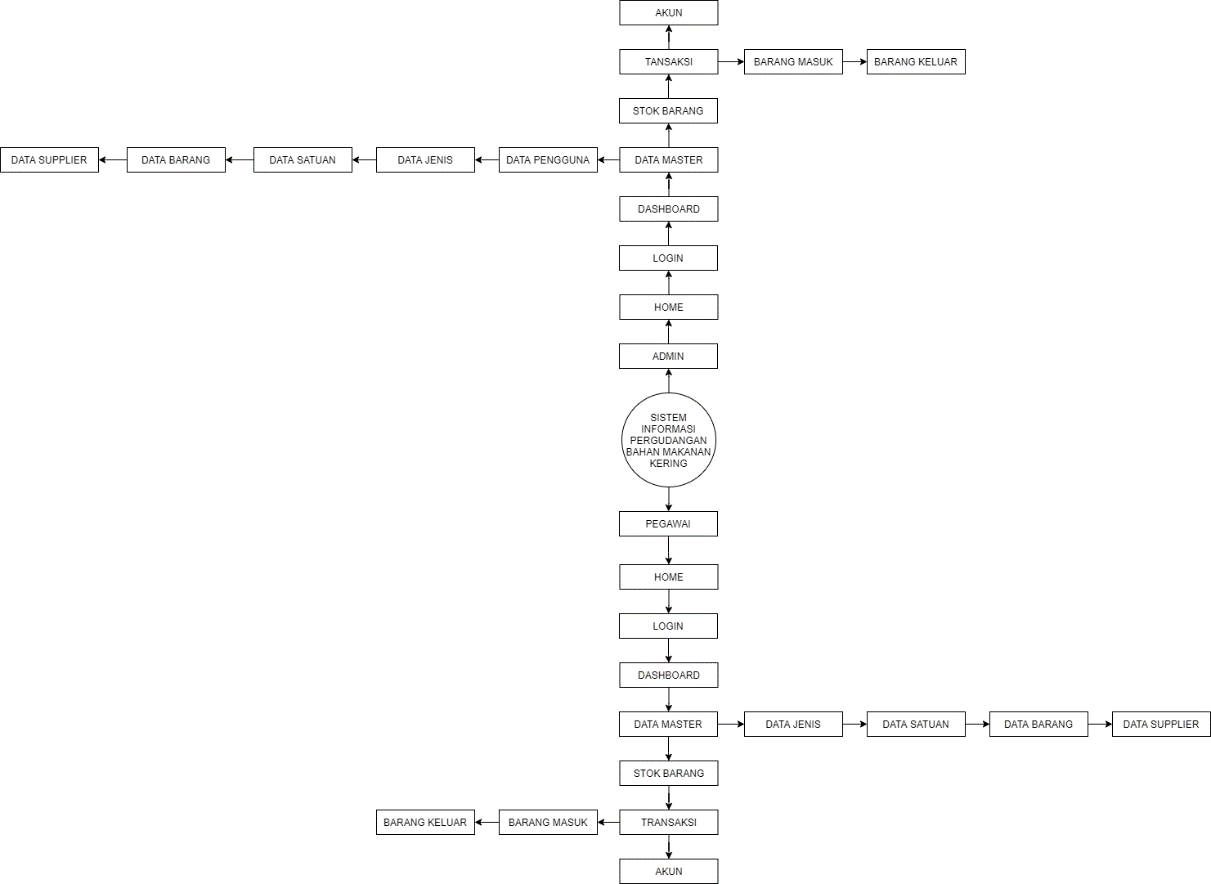
#### Tabel Supplier

|  |  |
| --- | --- |
| Nama Tabel | : supplier |
| Deskripsi | : Tabel supplier adalah tabel yang berfungsi untuk menyimpan data supplier. Adapun table fisik dari tabel supplier adalah sebagai berikut. |
| Primary key | : id\_supplier |
| Foregn key | : - |

Table 4.. Tabel supplier

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Type | Length | Key | Keterangan |
| id\_supplier | bigint | 20 | Primary Key | Auto Increment |
| nama\_supplier | varchar | 255 |  |  |
| jk | enum(‘Laki-laki’,’Perempuan’) |  |  |  |
| alamat | varchar | 225 |  |  |
| no\_telp | varchar | 225 |  |  |
| created\_at | timestamp |  |  |  |
| update\_at | timestamp |  |  |  |

### Sitemap Sistem

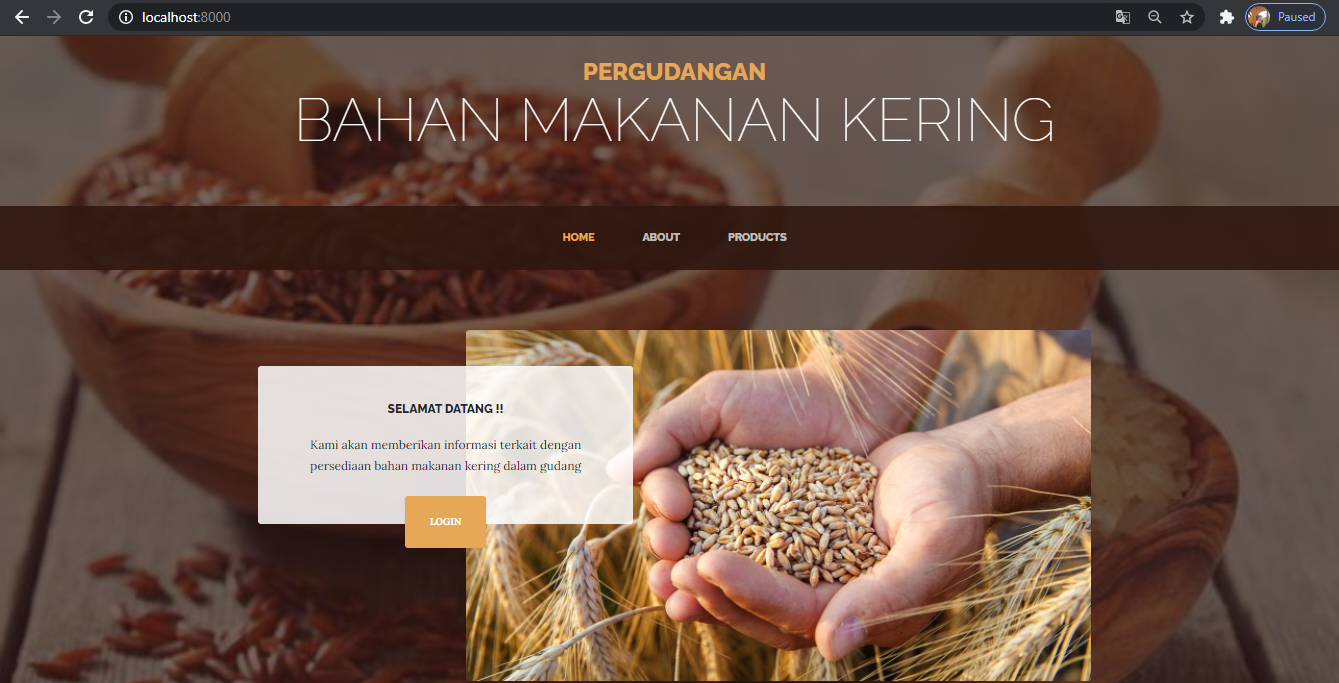


Gambar 4.. Gambar Sitemap/Navigasi atau peta sistem berdasarkan aktor

## Implementasi dan Koding

### Tampilan Homepage

Implementasi homepage ditunjukkan pada gambar 4.17.



Gambar 4.. Hompage

* 1. Script tampilan homepage

**………….**

<!-- Navigation -->

  <nav class="navbar navbar-expand-lg navbar-dark py-lg-4" id="mainNav">

    <div class="container">

      <a class="navbar-brand text-uppercase text-expanded font-weight-bold d-lg-none" href="#">Start Bootstrap</a>

      <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarResponsive" aria-controls="navbarResponsive" aria-expanded="false" aria-label="Toggle navigation">

        <span class="navbar-toggler-icon"></span>

      </button>

      <div class="collapse navbar-collapse" id="navbarResponsive">

        <ul class="navbar-nav mx-auto">

          <li class="nav-item active px-lg-4">

            <a class="nav-link text-uppercase text-expanded" href="/">Home

              <span class="sr-only">(current)</span>

            </a>

          </li>

          <li class="nav-item px-lg-4">

            <a class="nav-link text-uppercase text-expanded" href="/about">About</a>

          </li>

          <li class="nav-item px-lg-4">

            <a class="nav-link text-uppercase text-expanded" href="/produk">Products</a>

          </li>

        </ul>

      </div>

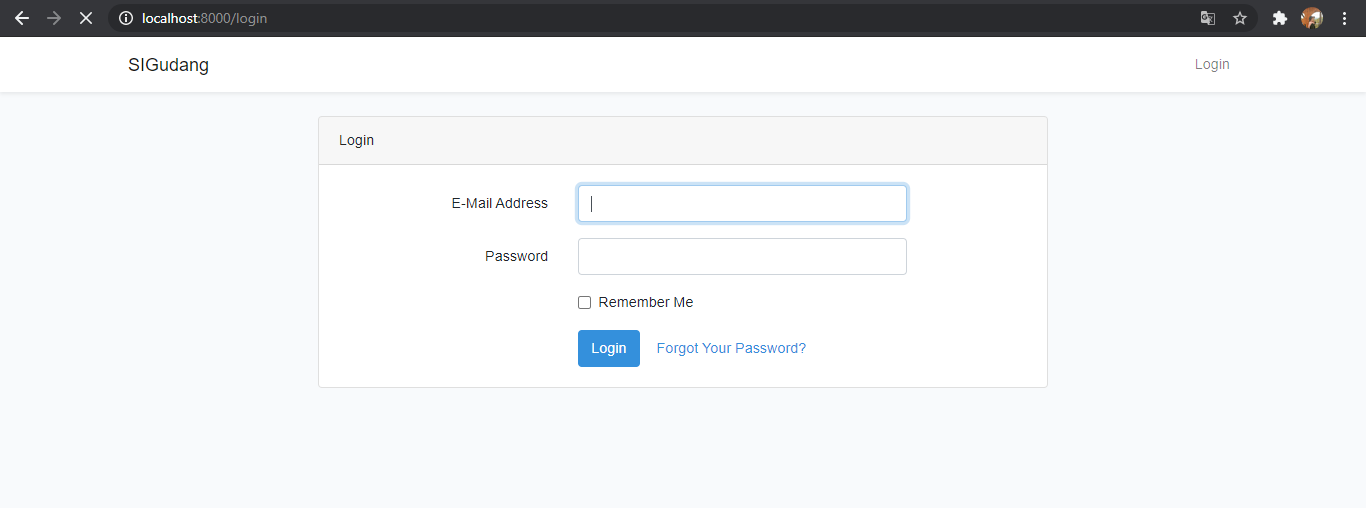
    </div>

  </nav>

**………..**

### Tampilan Login

Gambar 4.18 merupakan tampilan halaman login. Form login akan muncul ketika pengguna klik button login pada halaman homepage. Pada form ini pengguna mengisikan email dan password yang telah terdaftar. Jika pegawai gudang belum memiliki email dan password maka pegawai gudang menemui admin gudang untuk menambahkan hak akses sebagai pegawai pada halaman tambah data pegawai.



Gambar 4.. Halaman dashboard pegawai

* 1. Script tampilan login

**……**

<form method="POST" action="{{ route('login') }}">

@csrf

<div class="form-group row">

<label for="email" class="col-md-4 col-form-label text-md-right">{{ \_\_('E-Mail Address') }}</label>

<div class="col-md-6">

<input id="email" type="email" class="form-control @error('email') is-invalid @enderror" name="email" value="{{ old('email') }}" required autocomplete="email" autofocus>

@error('email')

<span class="invalid-feedback" role="alert">

<strong>{{ $message }}</strong>

</span>

@enderror

</div>

</div>

<div class="form-group row">

<label for="password" class="col-md-4 col-form-label text-md-right">{{ \_\_('Password') }}</label>

<div class="col-md-6">

<input id="password" type="password" class="form-control @error('password') is-invalid @enderror" name="password" required autocomplete="current-password">

@error('password')

<span class="invalid-feedback" role="alert">

<strong>{{ $message }}</strong>

</span>

@enderror

</div>

</div>

<div class="form-group row">

<div class="col-md-6 offset-md-4">

<div class="form-check">

<input class="form-check-input" type="checkbox" name="remember" id="remember" {{ old('remember') ? 'checked' : '' }}>

<label class="form-check-label" for="remember">

{{ \_\_('Remember Me') }}

</label>

</div>

</div>

</div>

<div class="form-group row mb-0">

<div class="col-md-8 offset-md-4">

<button type="submit" class="btn btn-primary">

{{ \_\_('Login') }}

</button>

@if (Route::has('password.request'))

<a class="btn btn-link" href="{{ route('password.request') }}">

{{ \_\_('Forgot Your Password?') }}

</a>

@endif

</div>

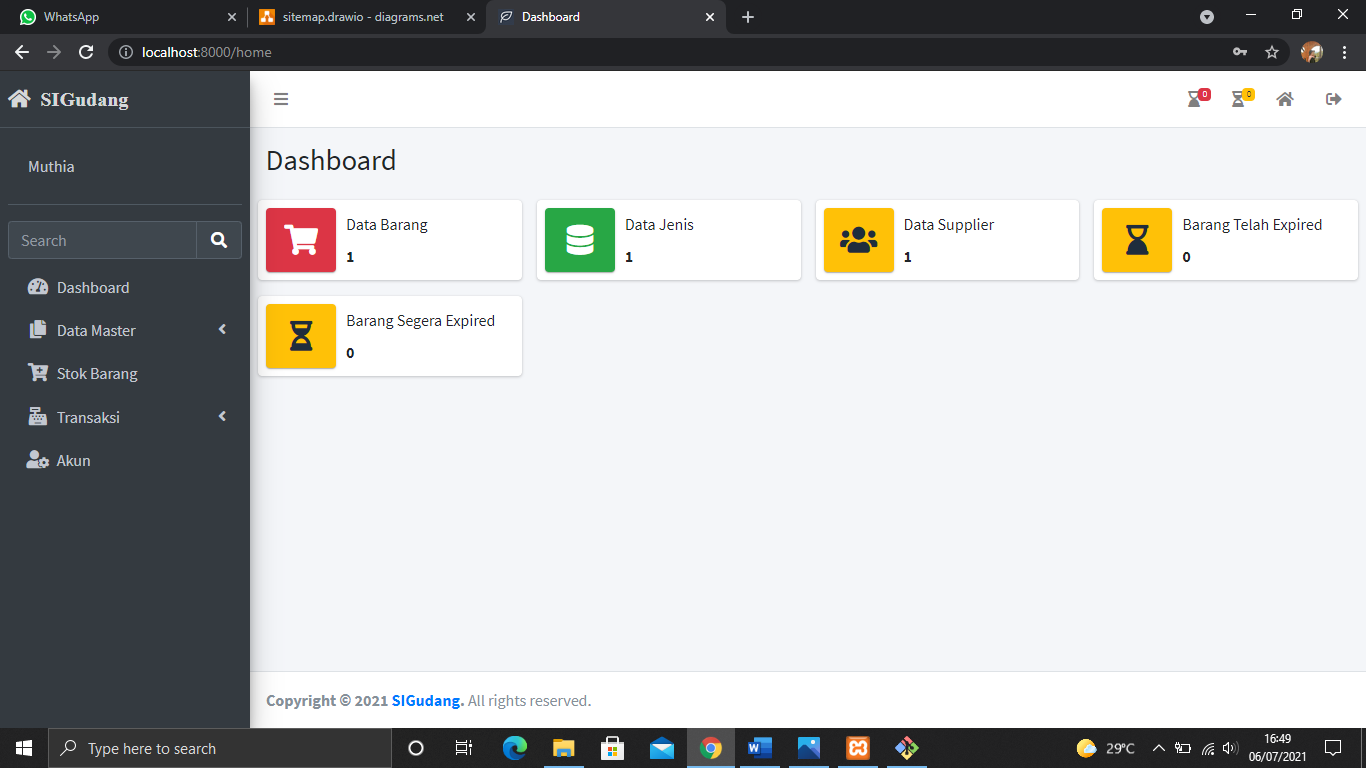
</div>

</form>

**……**

### Tampilan Dashboard Admin

Gambar 4. 19 merupakan tampilan halaman dashboard admin yang hanya dapat diakses oleh admin saja. Perbedaan hak akses admin dengan pegawai gudang yaitu admin memiliki hak untuk mengelola data pegawai gudang.



Gambar 4.. Halaman dashboard admin

* 1. Script tampilan dashboard admin

**……**

<div class="col-12 col-sm-6 col-md-3">

<div class="info-box mb-3">

<span class="info-box-icon bg-warning elevation-1"><i class="fas fa-hourglass-end"></i></span>

<div class="info-box-content">

<span class="info-box-text">Barang Telah Expired</span>

<span class="info-box-number">{{ $expired }}</span>

</div>

<!-- /.info-box-content -->

</div>

<!-- /.info-box -->

</div>

<div class="col-12 col-sm-6 col-md-3">

<div class="info-box mb-3">

<span class="info-box-icon bg-warning elevation-1"><i class="fas fa-hourglass-half"></i></span>

<div class="info-box-content">

<span class="info-box-text">Barang Segera Expired</span>

<span class="info-box-number">{{ $reminder }}</span>

</div>

<!-- /.info-box-content -->

</div>

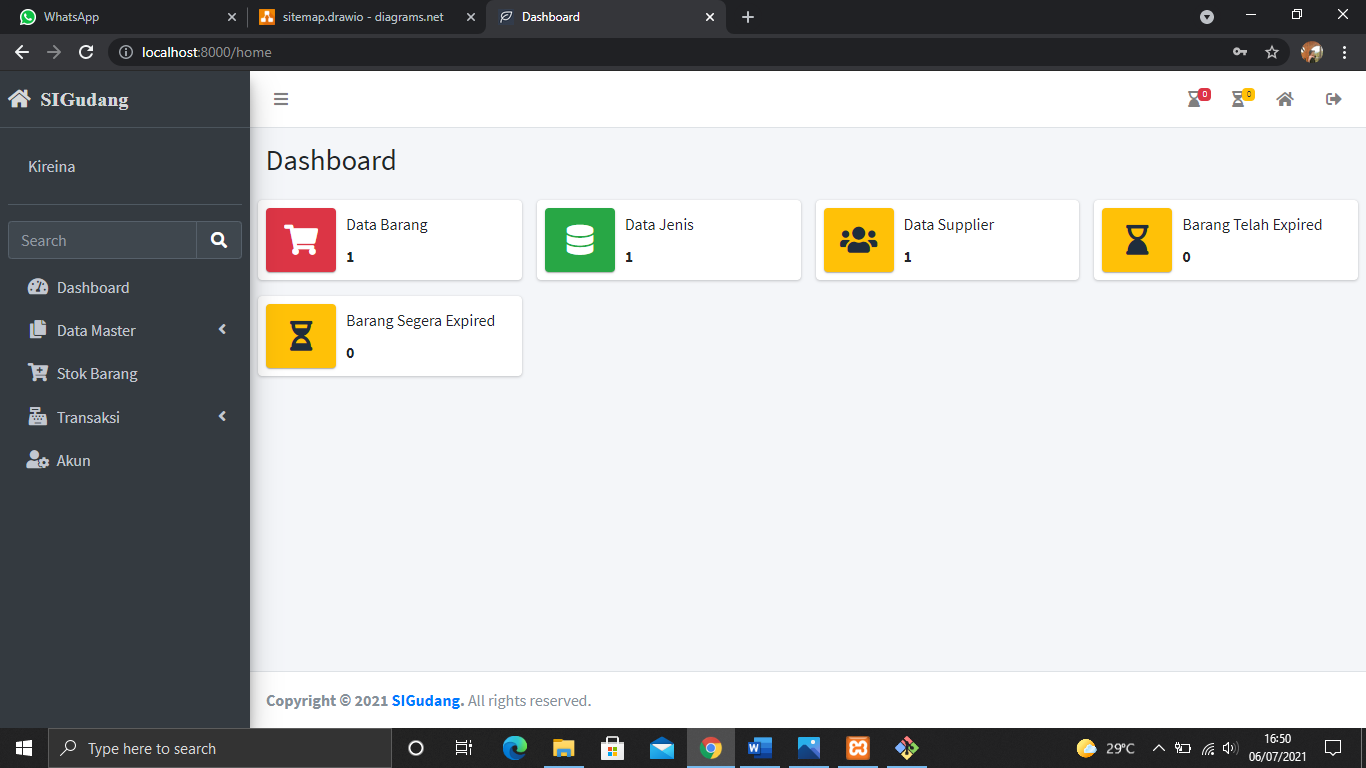
<!-- /.info-box -->

</div>

**……**

### Tampilan Dashboard Pegawai

Gambar 4.20 merupakan tampilan halaman dashboard pegawai yang hanya dapat diakses oleh pegawai saja. Pegawai memiliki hak akses untuk mengelola semua data kecuali data pegawai.



Gambar 4.. Halaman dashboard pegawai

* 1. Script tampilan dashboard pegawai

**……**

@if (auth()->user()->role == 'admin')

<div class="col-12 col-sm-6 col-md-3">

<div class="info-box">

<span class="info-box-icon bg-info elevation-1"><i class="fas fa-user"></i></span>

<div class="info-box-content">

<span class="info-box-text">Data Pegawai</span>

<span class="info-box-number">

{{ \App\Models\User::count() }}

</span>

</div>

<!-- /.info-box-content -->

</div>

<!-- /.info-box -->

</div>

@endif

**……**

### Tampilan Data Master

1. Data Pengguna

Gambar 4.21 merupakan tampilan halaman data pengguna dimana terdapat tabel yang berisi nama, alamat, jenis kelamin, no telp, email, role, expired reminder dan status. Kemudian terdapat buttin tambah data pengguna, button untuk mengaktifkan dan me non aktifkan pengguna, button edit dan button hapus. Menu ini hanya dapat diakses oleh admin gudang saja.



Gambar 4.. Halaman data pengguna

* 1. Script halaman data pengguna

**……**

<!-- Main content -->

<section class="content">

<table class="table data-table dt-head-center table-sm table-bordered table-hover table-striped">

<thead class="text-center">

<tr>

<th scope="col">No</th>

<th scope="col">Nama</th>

<th scope="col">Alamat</th>

<th scope="col">Jenis Kelamin</th>

<th scope="col">No Telp</th>

<th scope="col">Email</th>

<th scope="col">Role</th>

<th scope="col">EXP Reminder</th>

<th scope="col">Status</th>

<th scope="col">Action</th>

</tr>

</thead>

<tbody>

@foreach ($user as $usr)

<tr>

<th scope="row">{{ $loop->iteration }}</th>

<td>{{ $usr->name }}</td>

<td>{{ $usr->alamat }}</td>

<td>{{ $usr->jk }}</td>

<td>{{ $usr->no\_telp }}</td>

<td>{{ $usr->email }}</td>

<td>{{ $usr->role }}</td>

<td>{{ $usr->exp\_reminder }} hari</td>

<td><span class="badge {{ ($usr->status == 1) ? 'bg-success' : 'bg-secondary' }}">{{ ($usr->status == 1) ? 'Aktif' : 'Tidak Aktif' }}</span></td>

<td class="text-center">

@if ($usr->status == 1)

<a href="/user/status/{{ $usr->id }}" class="btn btn-sm btn-secondary"><i class="fas fa-times-circle"></i></a>

@else

<a href="/user/status/{{ $usr->id }}" class="btn btn-sm btn-success"><i class="fas fa-check-circle"></i></a>

@endif

<a href="/user/{{ $usr->id }}/edit" class="btn btn-warning btn-sm"><i class="fas fa-edit"></i></a>

<form action="/user/{{ $usr->id }}" method="POST" class="d-inline" onsubmit="return confirm('Yakin Hapus Data?')">

@method('delete')

@csrf

<button type="submit" class="btn btn-danger btn-sm"><i class="fas fa-trash"></i></button>

</form>

</td>

</tr>

@endforeach

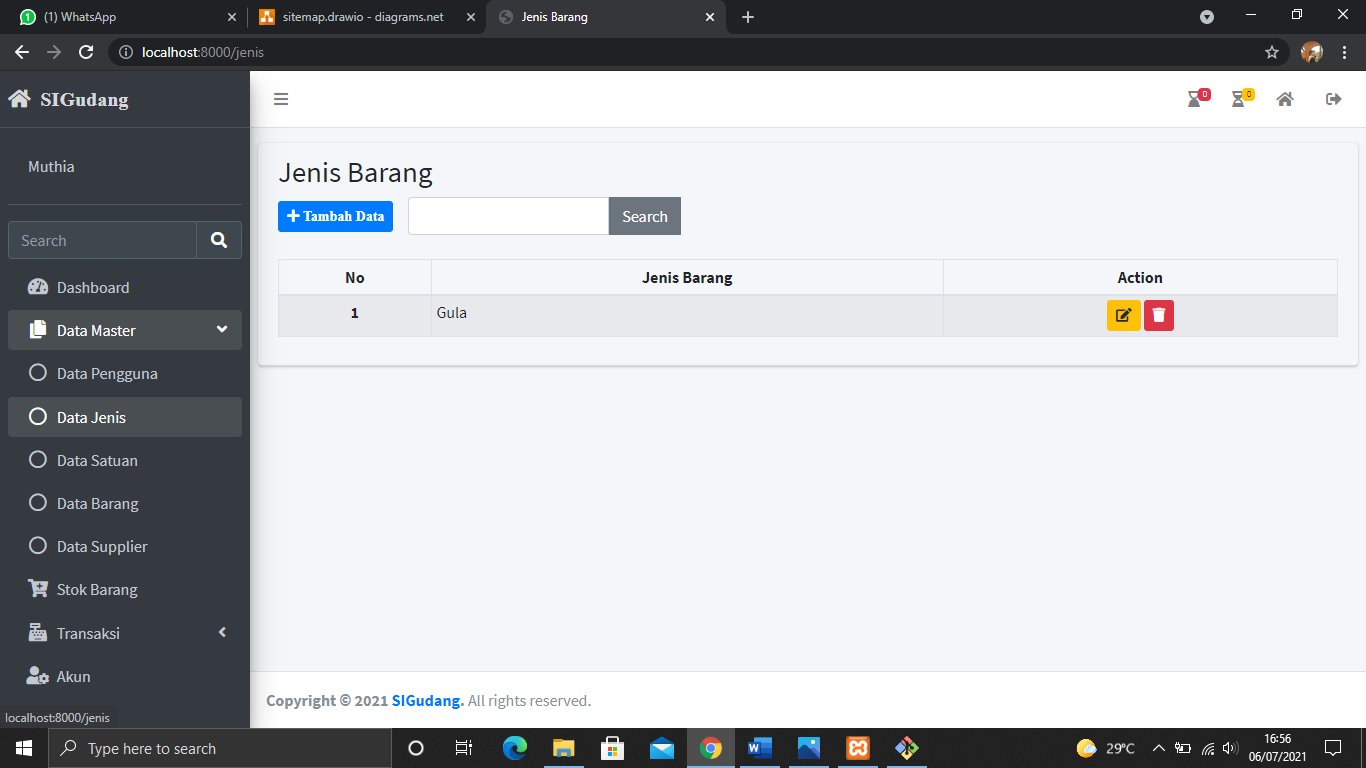
</tbody>

</table>

</section>

1. Data Jenis Barang

Gambar 4.22 merupakan tampilan halaman data jenis barang dimana pengguna dapat menambahkan, mengedit dan menghapus data jenis barang sesuai dengan kebutuhan. Menu ini dapat diakses oleh admin ataupun pegawai gudang.



Gambar 4.. Halaman data jenis

* 1. Script halaman data jenis

**……**

@foreach ($jenisBarang as $jenis)

<tr>

<th scope="row" class="text-center">{{ $loop->iteration }}</th>

<td>{{ $jenis->nama\_jenis }}</td>

<td class="text-center">

<a href="/jenis/{{ $jenis->id\_jenis }}/edit" class="btn btn-warning btn-sm"><i class="fas fa-edit"></i></a>

<form action="/jenis/{{ $jenis->id\_jenis }}" method="POST" class="d-inline" onsubmit="return confirm('Yakin Hapus Data?')">

@method('delete')

@csrf

<button type="submit" class="btn btn-danger btn-sm"><i class="fas fa-trash"></i></button>

</form>

</td>

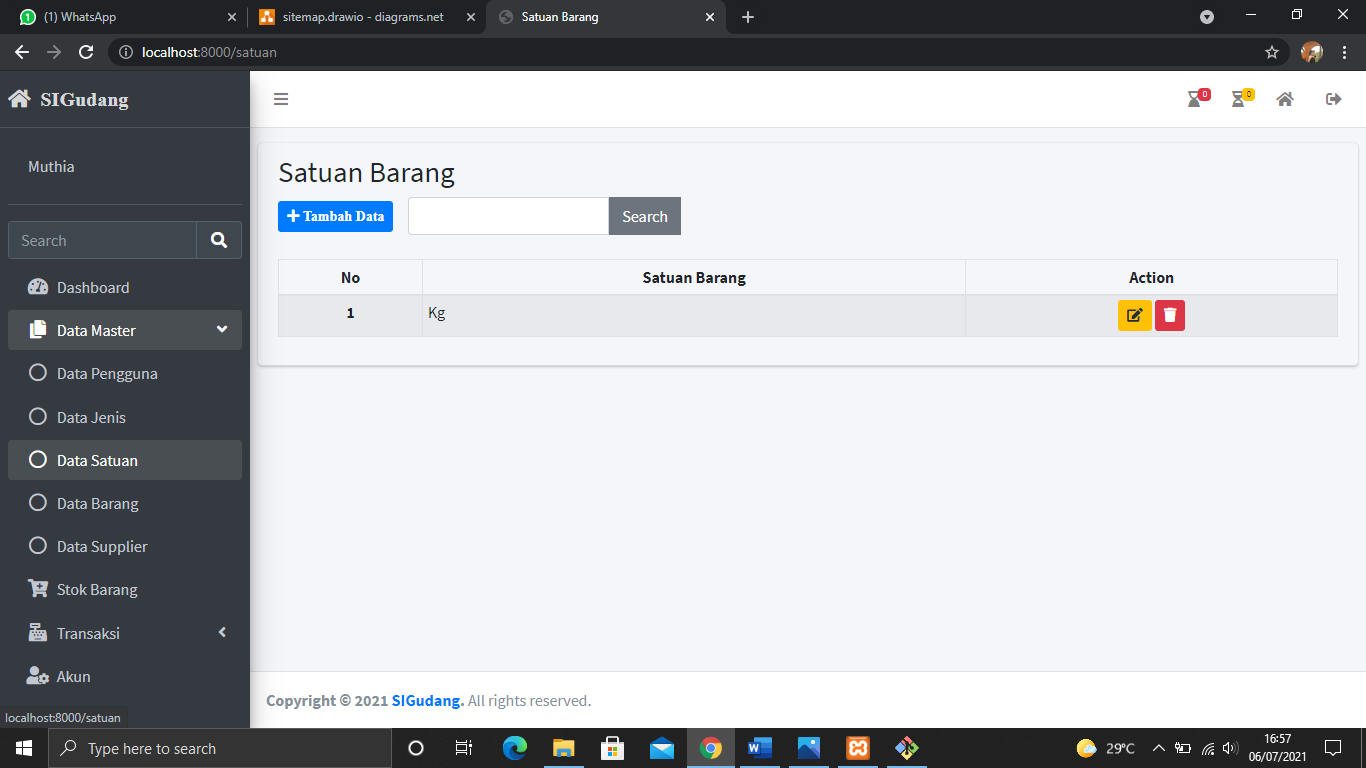
</tr>

@endforeach

**……**

1. Data Satuan Barang

Gambar 4.23 merupakan tampilan halaman data satuan barang dimana pengguna dapat menambahkan, mengedit dan menghapus data satuan barang sesuai dengan kebutuhan. Menu ini dapat diakses oleh admin ataupun pegawai gudang.



Gambar 4.. Halaman data satuan

* 1. Script halaman data satuan

**……**

@foreach ($satuanBarang as $satuan)

<tr>

<th scope="row" class="text-center">{{ $loop->iteration }}</th>

<td>{{ $satuan->nama\_satuan }}</td>

<td class="text-center">

<a href="/satuan/{{ $satuan->id\_satuan }}/edit" class="btn btn-warning btn-sm"><i class="fas fa-edit"></i></a>

<form action="/satuan/{{ $satuan->id\_satuan }}" method="POST" class="d-inline" onsubmit="return confirm('Yakin Hapus Data?')">

@method('delete')

@csrf

<button type="submit" class="btn btn-danger btn-sm"><i class="fas fa-trash"></i></button>

</form>

</td>

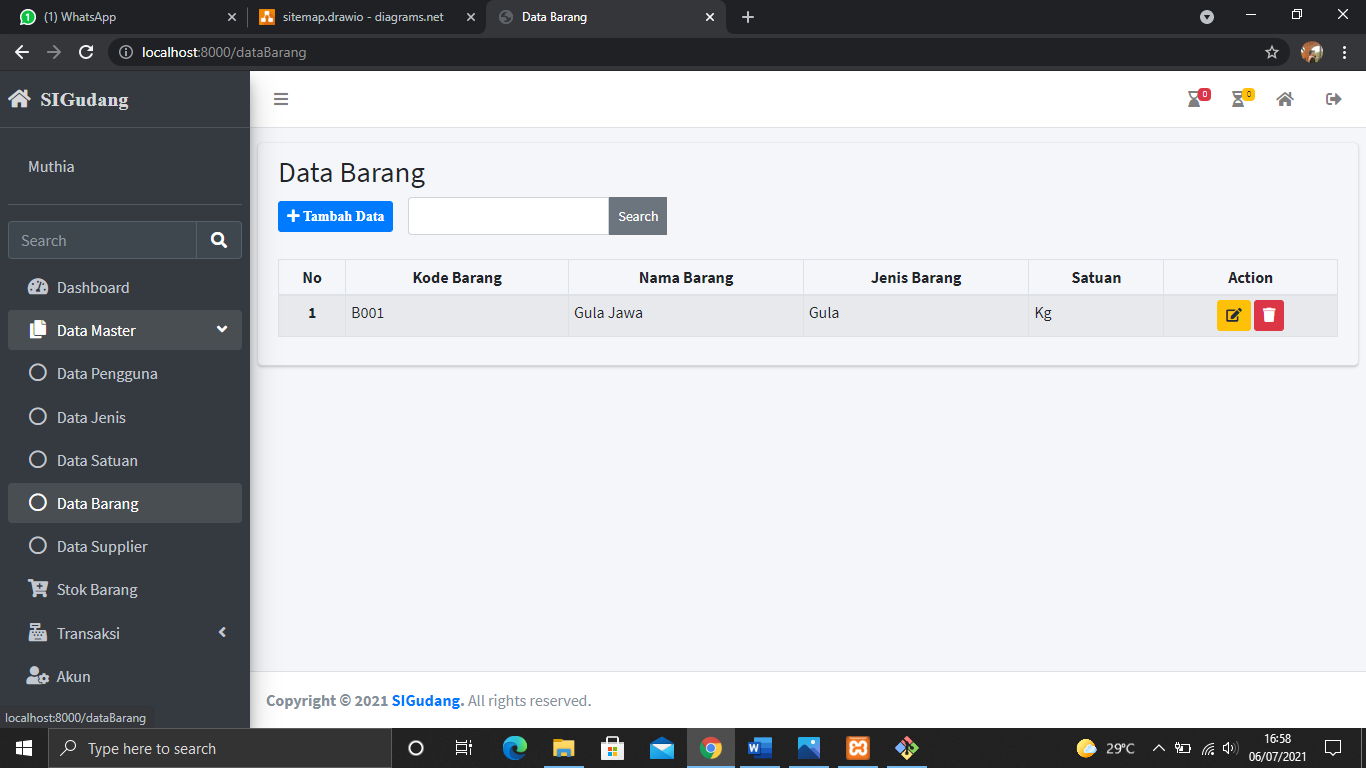
</tr>

@endforeach

**……**

1. Data Barang

Gambar 4.24 merupakan tampilan halaman data barang dimana pengguna dapat menambahkan, mengedit dan menghapus data barang sesuai dengan kebutuhan. Menu ini dapat diakses oleh admin ataupun pegawai gudang.



Gambar 4.. Halaman data barang

* 1. Script halaman data barang

**……**

@foreach ($dataBarang as $dtBarang)

<tr>

<th scope="row" class="text-center">{{ $loop->iteration }}</th>

<td>{{ $dtBarang->kode\_barang }}</td>

<td>{{ $dtBarang->nama\_barang }}</td>

<td>{{ $dtBarang->jenis->nama\_jenis }}</td>

<td>{{ $dtBarang->satuan->nama\_satuan }}</td>

<td class="text-center">

<a href="/dataBarang/{{ $dtBarang->id\_barang }}/edit" class="btn btn-warning btn-sm"><i class="fas fa-edit"></i></a>

<form action="/dataBarang/{{ $dtBarang->id\_barang }}" method="POST" class="d-inline" onsubmit="return confirm('Yakin Hapus Data?')">

@method('delete')

@csrf

<button type="submit" class="btn btn-danger btn-sm"><i class="fas fa-trash"></i></button>

</form>

</td>

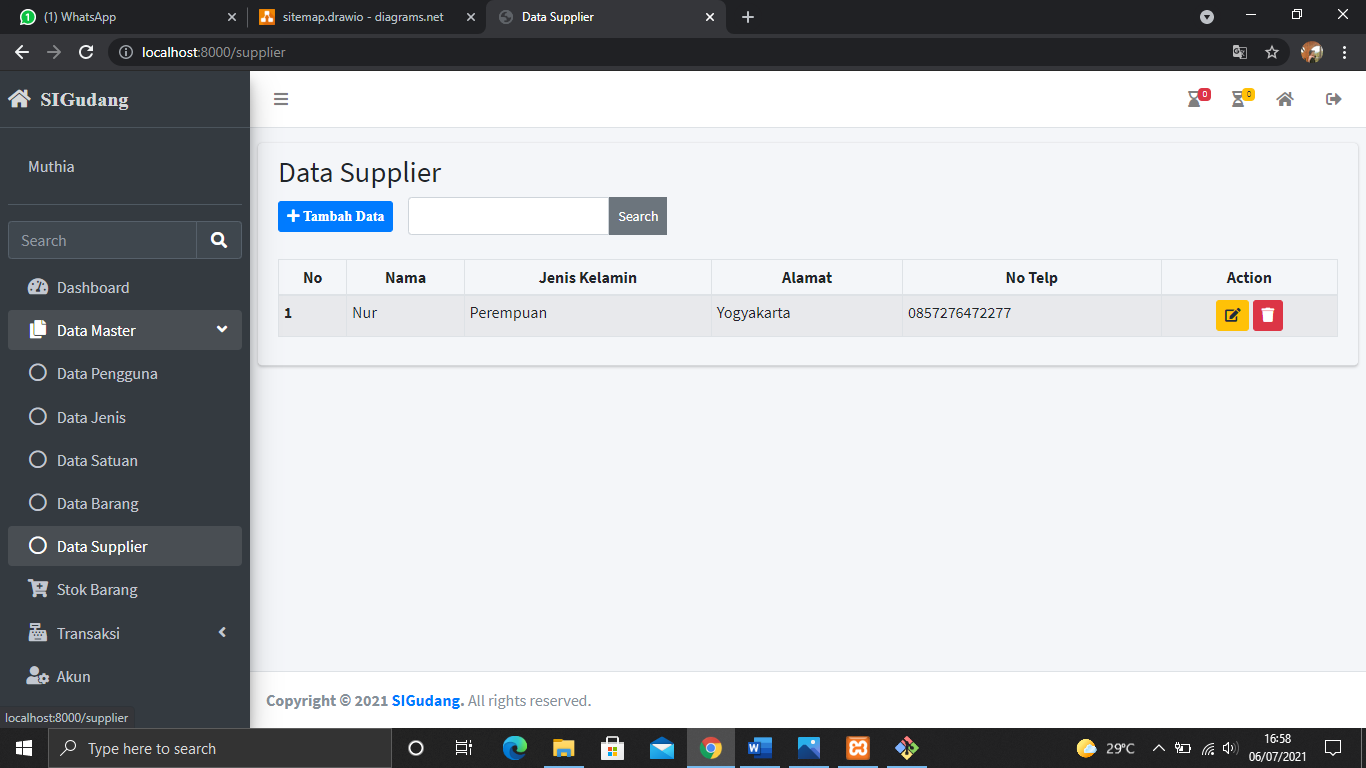
</tr>

@endforeach

**……**

1. Data Supplier

Gambar 4.25 merupakan tampilan halaman data supplier dimana pengguna dapat menambahkan, mengedit dan menghapus data supplier sesuai dengan kebutuhan. Menu ini dapat diakses oleh admin ataupun pegawai gudang.



Gambar 4.. Halaman data supplier

* 1. Script halaman data supplier

**……**

@foreach ($supplier as $sup)

<tr>

<th scope="row">{{ $loop->iteration }}</th>

<td>{{ $sup->nama\_supplier }}</td>

<td>{{ $sup->jk }}</td>

<td>{{ $sup->alamat }}</td>

<td>{{ $sup->no\_telp }}</td>

<td class="text-center">

<a href="/supplier/{{ $sup->id\_supplier }}/edit" class="btn btn-warning btn-sm"><i class="fas fa-edit"></i></a>

<form action="/supplier/{{ $sup->id\_supplier }}" method="POST" class="d-inline" onsubmit="return confirm('Yakin Hapus Data?')">

@method('delete')

@csrf

<button type="submit" class="btn btn-danger btn-sm"><i class="fas fa-trash"></i></button>

</form>

</td>

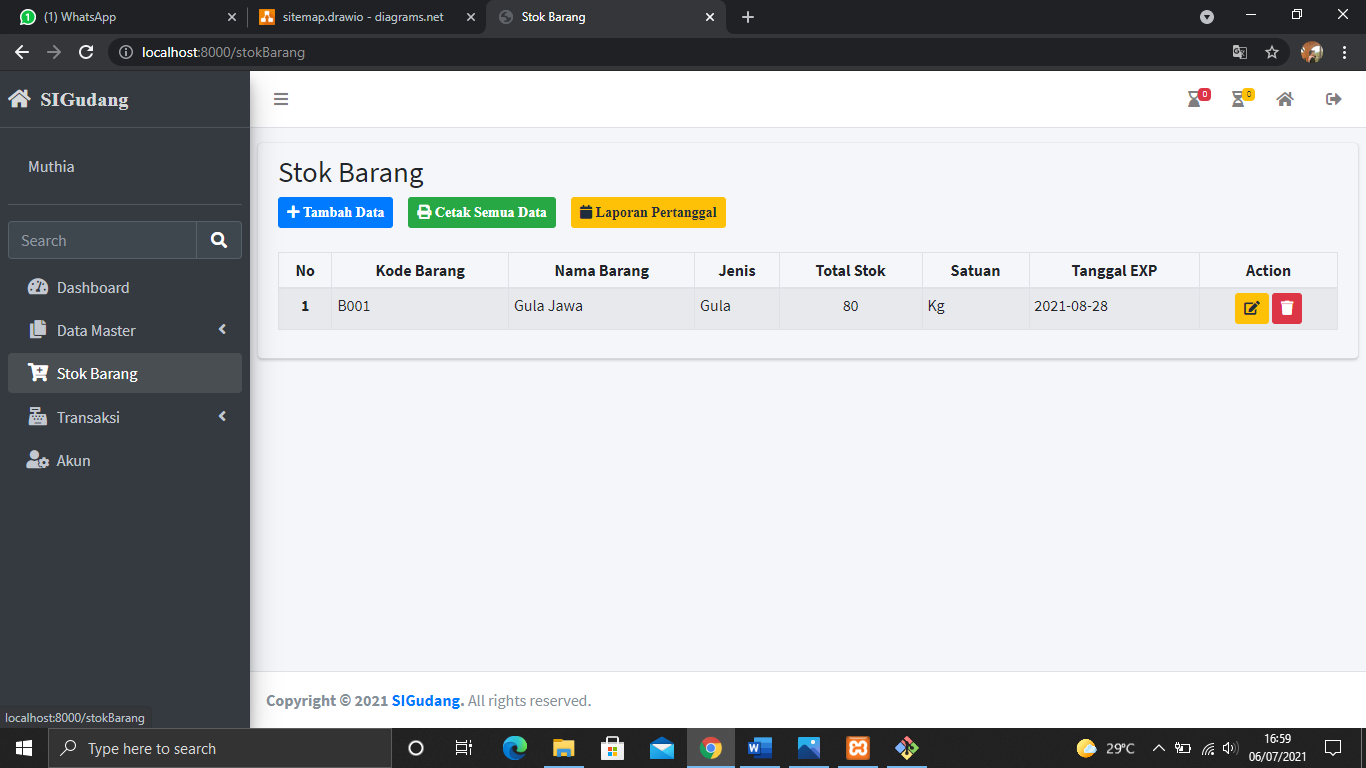
</tr>

@endforeach

**……**

### Tampilan Stok Barang

Gambar 4.26 merupakan tampilan halaman data stok barang dimana pengguna dapat menambahkan, mengedit dan menghapus data stok barang sesuai dengan kebutuhan. Menu ini dapat diakses oleh admin ataupun pegawai gudang.



Gambar 4.. Halaman data stok barang

* 1. Script halaman stok barang

**……**

@foreach ($stokBarang as $stok)

<tr>

<th scope="row" class="text-center">{{ $loop->iteration }}</th>

<td>{{ $stok->dataBarang->kode\_barang }}</td>

<td>{{ $stok->dataBarang->nama\_barang }}</td>

<td>{{ $stok->dataBarang->jenis->nama\_jenis }}</td>

<td class="text-center">{{ $stok->jml\_barang }}</td>

<td>{{ $stok->dataBarang->satuan->nama\_satuan }}</td>

<td>{{ $stok->tgl\_exp }}</td>

<td class="text-center">

<a href="/stokBarang/{{ $stok->id\_stok }}/edit" class="btn btn-warning btn-sm"><i class="fas fa-edit"></i></a>

<form action="/stokBarang/{{ $stok->id\_stok }}" method="POST" class="d-inline" onsubmit="return confirm('Yakin Hapus Data?')">

@method('delete')

@csrf

<button type="submit" class="btn btn-danger btn-sm"><i class="fas fa-trash"></i></button>

</form>

</td>

</tr>

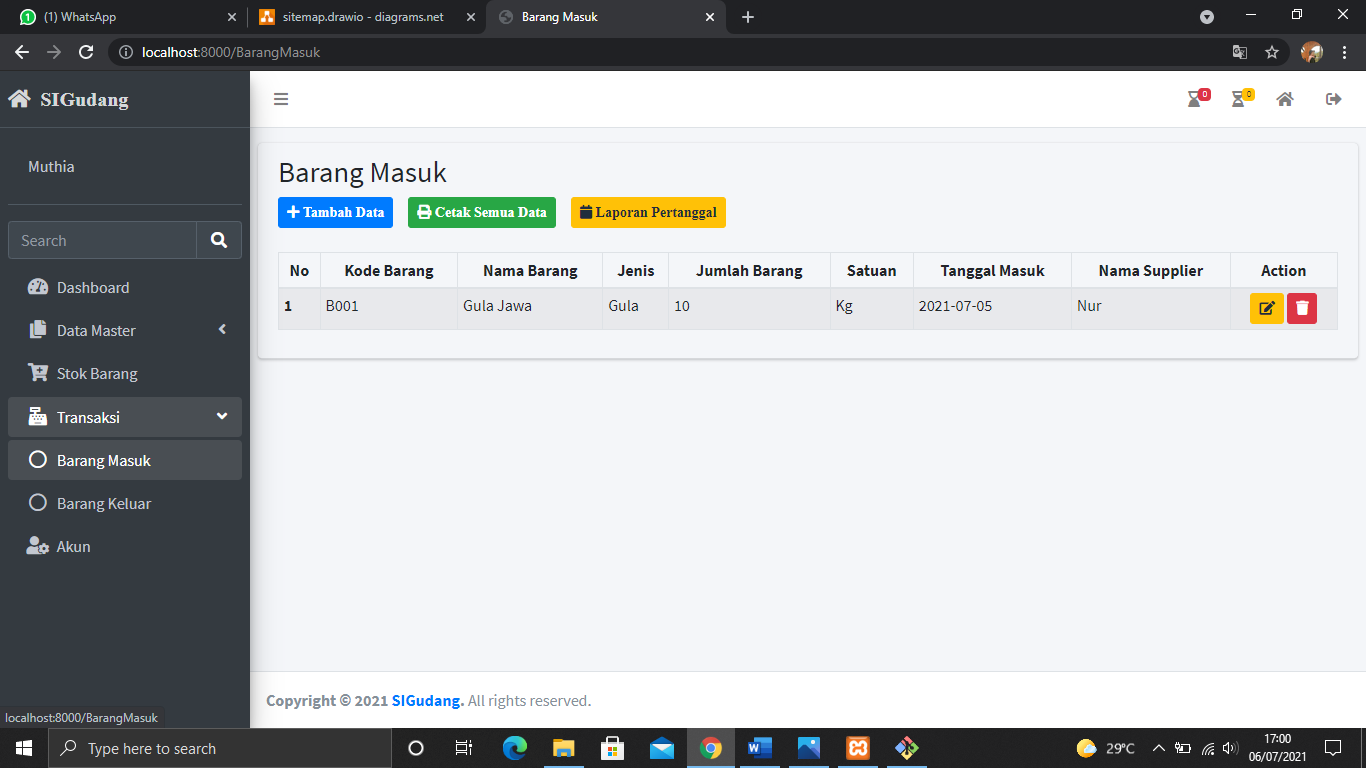
@endforeach

**……**

### Tampilan Transaksi

1. Barang Masuk

Gambar 4.27 merupakan tampilan halaman data barang masuk dimana pengguna dapat menambahkan, mengedit dan menghapus data barang masuk sesuai dengan kebutuhan. Menu ini dapat diakses oleh admin ataupun pegawai gudang.



Gambar 4.. Halaman data barang masuk

* 1. Script halaman data barang masuk

**……**

@foreach ($barangMasuk as $brgMasuk)

<tr>

<th scope="row">{{ $loop->iteration }}</th>

<td>{{ $brgMasuk->stokBarang->dataBarang->kode\_barang }}</td>

<td>{{ $brgMasuk->stokBarang->dataBarang->nama\_barang }}</td>

<td>{{ $brgMasuk->stokBarang->dataBarang->jenis->nama\_jenis }}</td>

<td>{{ $brgMasuk->jml\_barang }}</td>

<td>{{ $brgMasuk->stokBarang->dataBarang->satuan->nama\_satuan }}</td>

<td>{{ $brgMasuk->tgl\_masuk }}</td>

<td>{{ $brgMasuk->supplier->nama\_supplier }}</td>

<td class="text-center">

<a href="/BarangMasuk/{{ $brgMasuk->id\_masuk }}/edit" class="btn btn-warning btn-sm"><i class="fas fa-edit"></i></a>

<form action="/BarangMasuk/{{ $brgMasuk->id\_masuk }}" method="POST" class="d-inline" onsubmit="return confirm('Yakin Hapus Data?')">

@method('delete')

@csrf

<button type="submit" class="btn btn-danger btn-sm"><i class="fas fa-trash"></i></button>

</form>

</td>

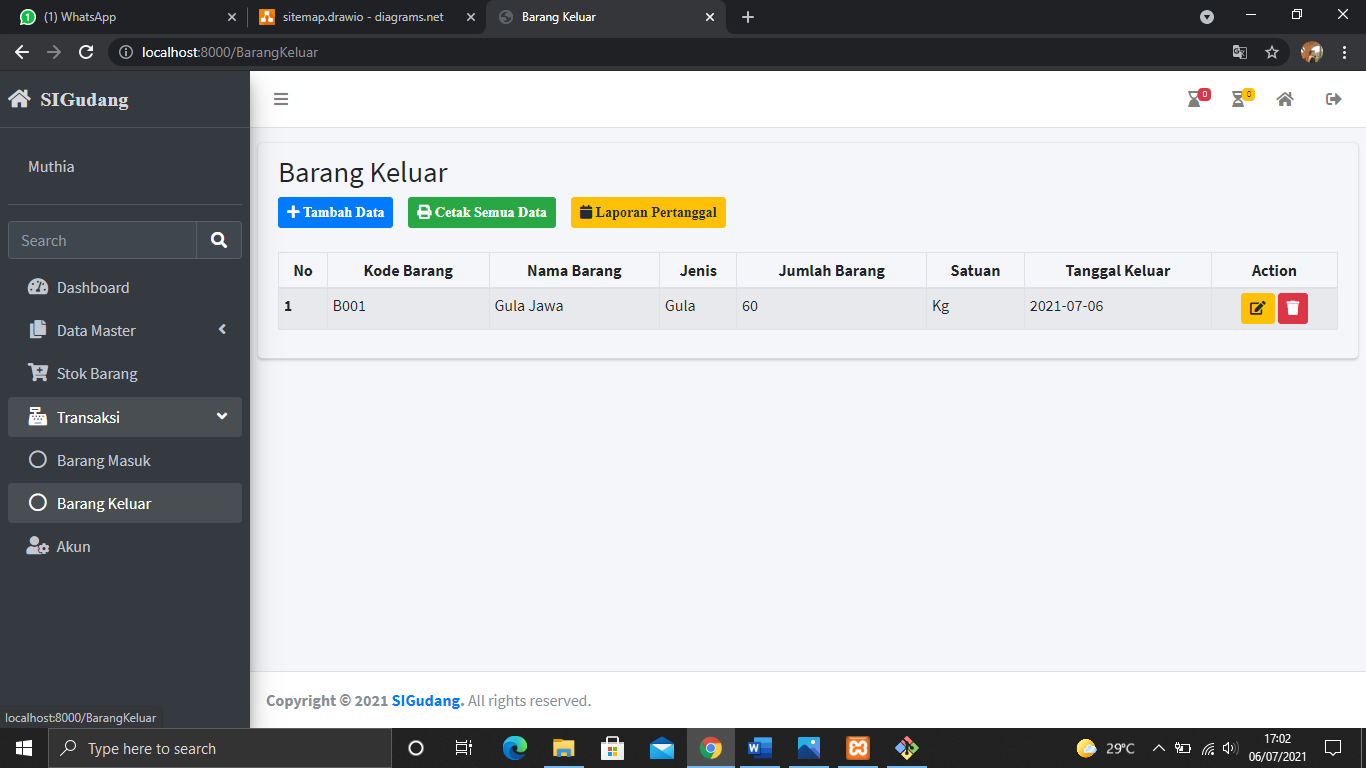
</tr>

@endforeach

**……**

1. Barang Keluar

Gambar 4.28 merupakan tampilan halaman data barang keluar dimana pengguna dapat menambahkan, mengedit dan menghapus data barang keluar sesuai dengan kebutuhan. Menu ini dapat diakses oleh admin ataupun pegawai gudang.



Gambar 4.. Halaman data barang keluar

* 1. Script halaman data barang keluar

**……**

@foreach ($barangKeluar as $brgKeluar)

<tr>

<th scope="row">{{ $loop->iteration }}</th>

<td>{{ $brgKeluar->stokBarang->dataBarang->kode\_barang }}</td>

<td>{{ $brgKeluar->stokBarang->dataBarang->nama\_barang }}</td>

<td>{{ $brgKeluar->stokBarang->dataBarang->jenis->nama\_jenis }}</td>

<td>{{ $brgKeluar->jml\_barang }}</td>

<td>{{ $brgKeluar->stokBarang->dataBarang->satuan->nama\_satuan }}</td>

<td>{{ $brgKeluar->tgl\_keluar }}</td>

<td class="text-center">

<a href="/BarangKeluar/{{ $brgKeluar->id\_keluar }}/edit" class="btn btn-warning btn-sm"><i class="fas fa-edit"></i></a>

<form action="/BarangKeluar/{{ $brgKeluar->id\_keluar }}" method="POST" class="d-inline" onsubmit="return confirm('Yakin Hapus Data?')">

@method('delete')

@csrf

<button type="submit" class="btn btn-danger btn-sm"><i class="fas fa-trash"></i></button>

</form>

</td>

</tr>

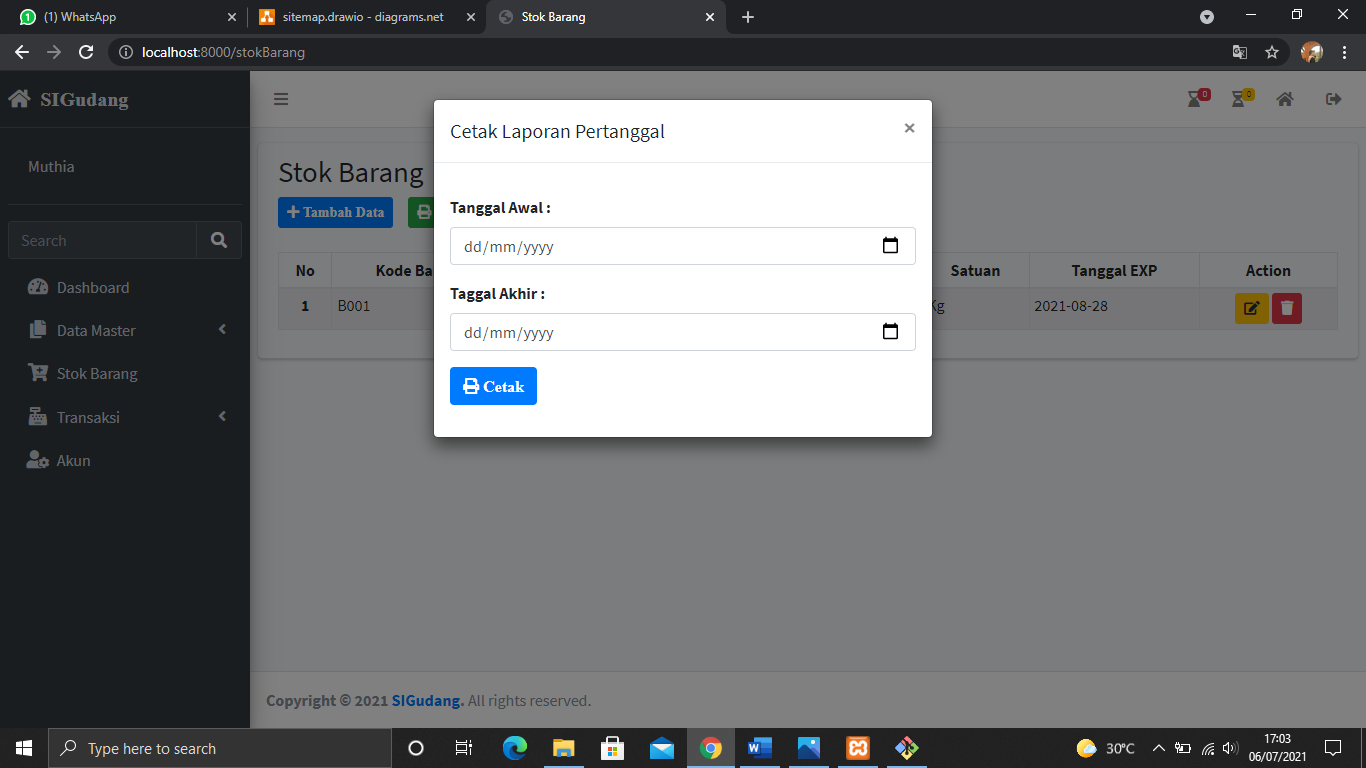
@endforeach

**……**

### Tampilan Laporan Pertanggal

1. Stok Barang

Gambar 4.29 merupakan tampilan halaman laporan data stok barang pertanggal dimana terdapat form untuk mengisikan tanggal awal dan tanggal akhir yang berguna untuk melakukan pelaporan stok barang yang ada dalam gudang berdasarkan range tanggal sesuai dengan keinginan. Menu ini dapat diakses oleh admin ataupun pegawai gudang.



Gambar 4.. Halaman laporan stok

* 1. Script halaman laporan stok

**……**

<div class="modal fade" id="laporanStok" role="dialog" aria-labelledby="modalLabel" aria-hidden="true">

<div class="modal-dialog" role="document">

<div class="modal-content">

<div class="modal-header">

<h5 class="modal-title">Cetak Laporan Pertanggal</h5>

<button type="button" class="close" data-dismiss="modal" aria-label="Close">

<span aria-hidden="true">&times;</span>

</button>

</div>

<div class="modal-body">

<form>

<div class="form-group">

<div class="my-3">

<label for="label">Tanggal Awal : </label>

<input type="date" name="tglawal" id="tglawal" class="form-control">

</div>

<div class="my-3">

<label for="label">Taggal Akhir : </label>

<input type="date" name="tglakhir" id="tglakhir" class="form-control">

</div>

<div class="my-3">

<a href="" onclick="this.href='/laporanStokPertanggal/'+document.getElementById('tglawal').value + '/'

+ document.getElementById('tglakhir').value" target="\_blank" class="btn btn-primary"><i class="fas fa-print"> Cetak</i></a>

</div>

</div>

</form>

</div>

</div>

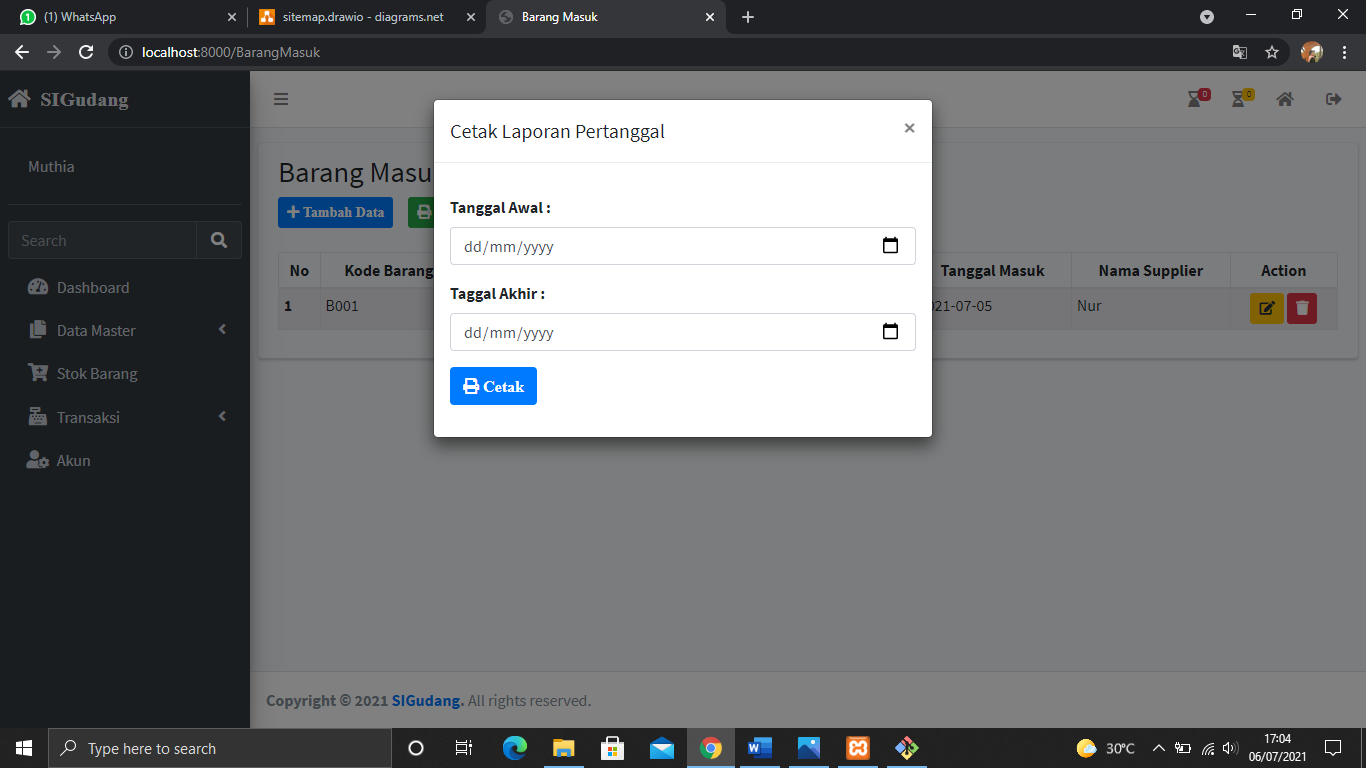
</div>

</div>

**……**

1. Barang Masuk

Gambar 4.30 merupakan tampilan halaman laporan data barang masuk pertanggal dimana terdapat form untuk mengisikan tanggal awal dan tanggal akhir yang berguna untuk melakukan pelaporan barang masuk berdasarkan range tanggal sesuai dengan keinginan. Menu ini dapat diakses oleh admin ataupun pegawai gudang.



Gambar 4.. Halaman laporan barang masuk

* 1. Script halaman laporan barang masuk

**……**

<div class="col-auto">

<!-- Button trigger modal -->

<button type="button" class="btn btn-warning btn-sm" data-toggle="modal" data-target="#laporanStok">

<i class="fas fa-calendar"> Laporan Pertanggal</i>

</button>

<!-- Modal -->

<div class="modal fade" id="laporanStok" role="dialog" aria-labelledby="modalLabel" aria-hidden="true">

<div class="modal-dialog" role="document">

<div class="modal-content">

<div class="modal-header">

<h5 class="modal-title">Cetak Laporan Pertanggal</h5>

<button type="button" class="close" data-dismiss="modal" aria-label="Close">

<span aria-hidden="true">&times;</span>

</button>

</div>

<div class="modal-body">

<form>

<div class="form-group">

<div class="my-3">

<label for="label">Tanggal Awal : </label>

<input type="date" name="tglawal" id="tglawal" class="form-control">

</div>

<div class="my-3">

<label for="label">Tanggal Akhir : </label>

<input type="date" name="tglakhir" id="tglakhir" class="form-control">

</div>

<div class=" my-3">

<a href="" onclick="this.href='/laporanMasukPertanggal/'+document.getElementById('tglawal').value + '/'

+ document.getElementById('tglakhir').value" target="\_blank" class="btn btn-primary"><i class="fas fa-print"> Cetak</i></a>

</div>

</div>

</form>

</div>

</div>

</div>

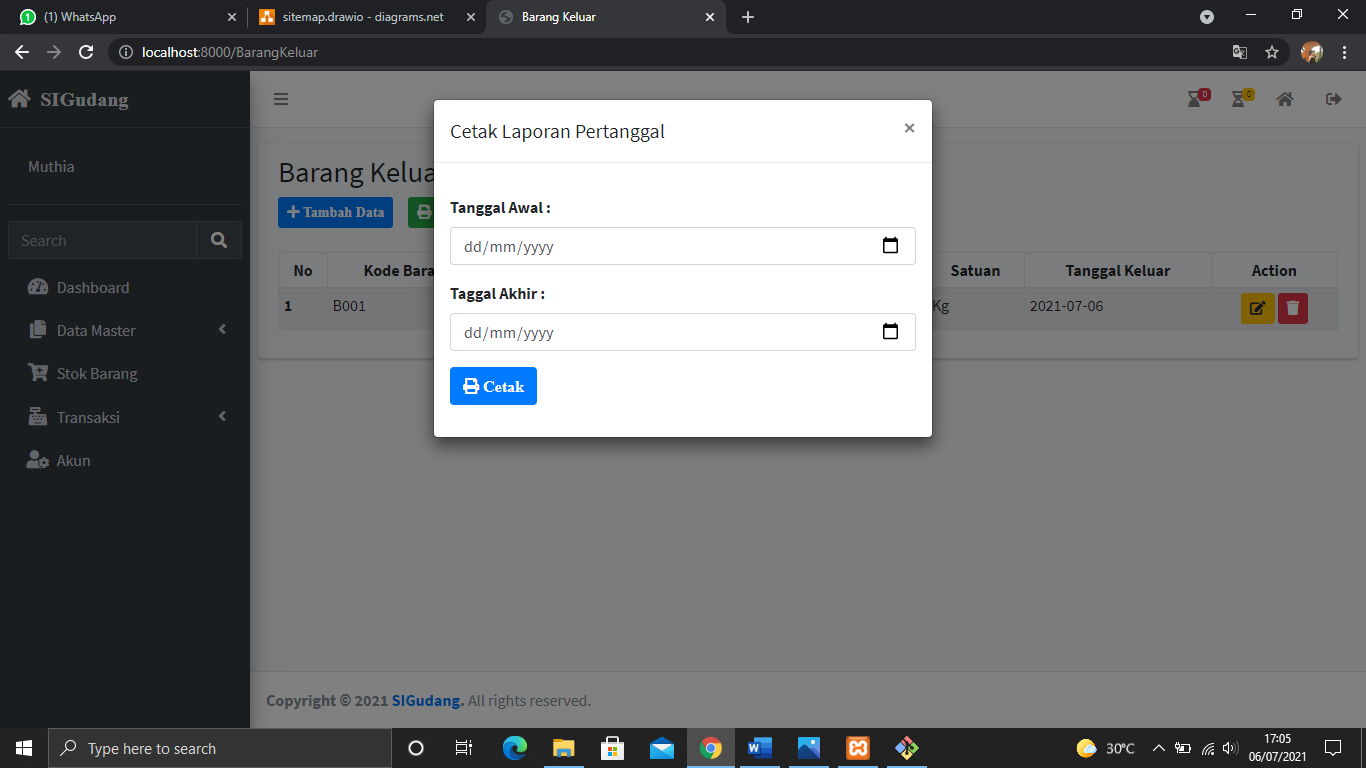
</div>

</div>

**……**

1. Barang Keluar

Gambar 4.31 merupakan tampilan halaman laporan data barang keluar pertanggal dimana terdapat form untuk mengisikan tanggal awal dan tanggal akhir yang berguna untuk melakukan pelaporan barang keluar berdasarkan range tanggal sesuai dengan keinginan. Menu ini dapat diakses oleh admin ataupun pegawai gudang.



Gambar 4.. Halaman laporan barang keluar

* 1. Script halaman laporan barang keluar

**……**

<div class="modal fade" id="laporanStok" role="dialog" aria-labelledby="modalLabel" aria-hidden="true">

<div class="modal-dialog" role="document">

<div class="modal-content">

<div class="modal-header">

<h5 class="modal-title">Cetak Laporan Pertanggal</h5>

<button type="button" class="close" data-dismiss="modal" aria-label="Close">

<span aria-hidden="true">&times;</span>

</button>

</div>

<div class="modal-body">

<form>

<div class="form-group">

<div class="my-3">

<label for="label">Tanggal Awal : </label>

<input type="date" name="tglawal" id="tglawal" class="form-control">

</div>

<div class="my-3">

<label for="label">Taggal Akhir : </label>

<input type="date" name="tglakhir" id="tglakhir" class="form-control">

</div>

<div class=" my-3">

<a href="" onclick="this.href='/laporanKeluarPertanggal/'+document.getElementById('tglawal').value + '/'

+ document.getElementById('tglakhir').value" target="\_blank" class="btn btn-primary"><i class="fas fa-print"> Cetak</i></a>

</div>

</div>

</form>

</div>

</div>

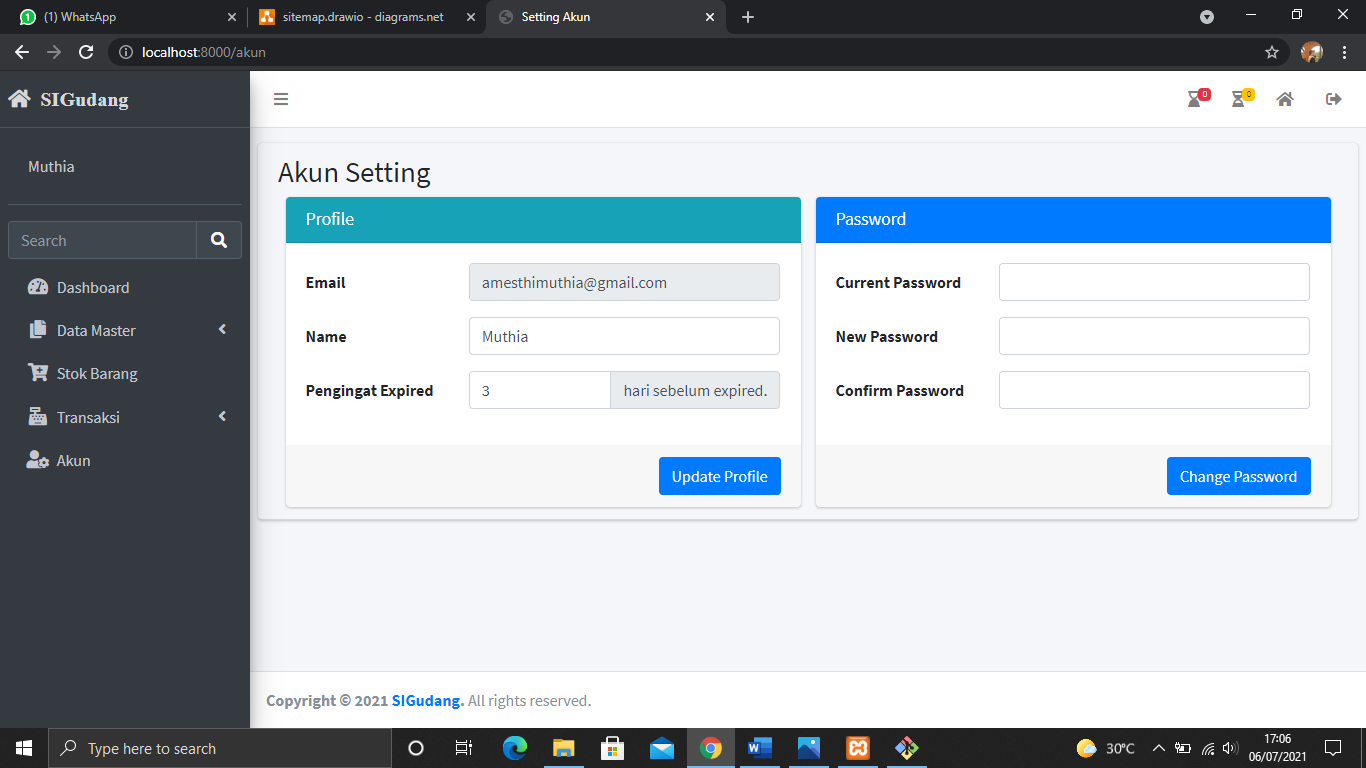
</div>

</div>

**……**

### Tampilan Akun

Gambar 4.32 merupakan tampilan setting akun yang berfungsi untuk mengganti password yang sudah terdaftar dengan password baru dan juga berfungsi untuk mengupdate username, email dan juga reminder expired sesuai dengan yang diinginkan. Terdapat dua form yaittu form untuk mengubah password dan juga mengubah profil. Dalam form ubah password user diminta untuk memasukkan password lama, password baru dan konfirmasi password baru yang telah diubah. Password baru tidak boleh sama dengan password lama. Menu ini dapat diakses oleh admin maupun pegawai gudang.



Gambar 4.. Halaman akun

* 1. Script halaman akun

**……**

<form role="form" id="update" action="{{ route('setting.akun.update') }}" method="post">

@csrf

<div class="form-group row">

<label class="col-sm-4 col-form-label">{{ \_\_('Email') }}</label>

<div class="col-sm-8">

<input type="text" class="form-control" id="email" value="{{ Auth::user()->email }}" disabled>

</div>

</div>

<div class="form-group row">

<label for="name" class="col-sm-4 col-form-label">{{ \_\_('Name') }}</label>

<div class="col-sm-8">

<input type="text" class="form-control" id="name" name="name" value="{{ Auth::user()->name }}">

</div>

</div>

<div class="form-group row">

<label for="exp\_reminder" class="col-sm-4 col-form-label">{{ \_\_('Pengingat Expired') }}</label>

<div class="col-sm-8">

<div class="input-group">

<input type="number" class="form-control" id="exp\_reminder" name="exp\_reminder" min="1" value="{{ Auth::user()->exp\_reminder }}"/>

<div class="input-group-append">

<div class="input-group-text">hari sebelum expired.</div>

</div>

</div>

</div>

</div>

</form>

## Pengujian Sistem

### Pengujian sistem login

Table 4.. Pengujian sistem login

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Skenario | Test Case | Harapan | Hasil | Validasi |
| 1 | Mengosongkan semua *field* pada *form login* kemudian menekan tombol ‘Login’ | Email : (kosong)  Password : (kosong) | Sistem menolak untuk masuk ke halaman dashboard dan muncul pemberitahuan bahwa username dan password harus diisi. | Sesuai harapan | Valid |
| 2 | Mengosongkan email dan mengisikan password kemudian menekan tombol ‘Login’ | Email : (kosong)  Password : (password) | Sistem menolak untuk masuk ke halaman dashboard dan muncul pemberitahuan bahwa email harus diisi. | Sesuai harapan | Valid |
| 3 | Mengisikan email dan mengosongkan password kemudian menekan tombol ‘Login’ | Email : (email)  Password : (kosong) | Sistem menolak untuk masuk ke halaman dashboard dan muncul pemberitahuan bahwa password harus diisi. | Sesuai harapan | Valid |
| 4 | Mengisikan semua *field* pada *form login* kemudian menekan tombol ‘Login’ | Email : (email)  Password : (password) | Sistem masuk ke halaman dashboard. | Sesuai harapan | Valid |
| 5 | Mengetikkan Email dan/atau password tidak sesuai, kemudian klik tombol ‘Login’ | Email : (tidak sesuai)  Password : (tidak sesuai) | Sistem akan menolak dan menampilkan pesan “Email atau Password yang anda masukan salah” | Sesuai harapan | Valid |

### Pengujian form input data pengguna

Tabel 4.0.23 Pengujian form input data pengguna

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Skenario | Test Case | Harapan | Hasil | Validasi |
| 1 | Nama  Jenis kelamin  Alamat  No telp  Email  Password  Role  (tidak diisi) kemudian klik button submit | Mengklik submit tanpa mengisi data atau hanya sebagian data diisi | Sistem akan menolak dan menampilkan pesan “The field is required” | Sesuai harapan | Valid |
| 2 | Mengisikan Nama  Jenis kelamin  Alamat  No telp  Email  Password  Role kemudian klik button submit | Mengklik submit dengan mengisi data lengkap dan benar | Sistem menerima akses input pegawai dan kemudian akan muncul alert “data pegawai berhasil ditambahkan” | Sesuai harapan | Valid |

### Pengujian form input data jenis barang

Tabel 4.0.24 Pengujian form input data jenis barang

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Skenario | Test Case | Harapan | Hasil | Validasi |
| 1 | Nama jenis barang (tidak diisi) kemudian klik button submit | Mengklik submit tanpa mengisi data | Sistem akan menolak dan menampilkan pesan “The field is required” | Sesuai harapan | Valid |
| 2 | Mengisikan Nama jenis barang kemudian klik button submit | Mengklik submit dengan mengisi data lengkap dan benar | Sistem menerima akses input data jenis barang dan kemudian akan muncul alert “jenis barang berhasil ditambahkan” | Sesuai harapan | Valid |

### Pengujian form input data satuan barang

Table 4.. Pengujian form input data satuan barang

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Skenario | Test Case | Harapan | Hasil | Validasi |
| 1 | Nama satuan barang (tidak diisi) kemudian klik button submit | Mengklik submit tanpa mengisi data | Sistem akan menolak dan menampilkan pesan “The field is required” | Sesuai harapan | Valid |
| 2 | Mengisikan Nama satuan barang kemudian klik button submit | Mengklik submit dengan mengisi data lengkap dan benar | Sistem menerima akses input data satuan barang dan kemudian akan muncul alert “satuan barang berhasil ditambahkan” | Sesuai harapan | Valid |

### Pengujian form input data barang

Table 4.. Pengujian form input data jenis barang

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Skenario | Test Case | Harapan | Hasil | Validasi |
| 1 | Kode barang  Nama barang  Jenis barang  Satuan barang (tidak diisi) kemudian klik button submit | Mengklik submit tanpa mengisi data atau hanya sebagian data diisi | Sistem akan menolak dan menampilkan pesan “The field is required” | Sesuai harapan | Valid |
| 2 | Mengisikan Kode barang (sama dengan yang sudah ada)  Nama barang  Jenis barang  Satuan barang kemudian klik button submit | Mengklik submit dengan mengisi kode barang sama dengan yang sudah ada | Sistem akan menolak dan kemudian akan muncul alert “The kode barang has already been taken.” | Sesuai harapan | Valid |
| 3 | Mengisikan  Kode barang  Nama barang  Jenis barang  Satuan barang kemudian klik button submit | Mengklik submit dengan mengisi data lengkap dan benar | Sistem menerima akses input data barang dan kemudian akan muncul alert “data barang berhasil ditambahkan” | Sesuai harapan | Valid |

### Pengujian form input data supplier

Table 4.. Pengujian form input data supplier

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Skenario | Test Case | Harapan | Hasil | Validasi |
| 1 | Nama supplier  Jenis kelamin  Alamat  No telp  (tidak diisi) kemudian klik button submit | Mengklik submit tanpa mengisi data atau hanya sebagian data diisi | Sistem akan menolak dan menampilkan pesan “The field is required” | Sesuai harapan | Valid |
| 2 | Mengisikan  Nama supplier  Jenis kelamin  Alamat  No telp kemudian klik button submit | Mengklik submit dengan mengisi data lengkap dan benar | Sistem menerima akses input data barang dan kemudian akan muncul alert “supplier berhasil ditambahkan” | Sesuai harapan | Valid |

### Pengujian form input data stok barang

Table 4.. Pengujian form input data stok barang

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Skenario | Test Case | Harapan | Hasil | Validasi |
| 1 | Kode barang  Nama barang  Jenis  Jumlah barang  Satuan  Tanggal EXP  (tidak diisi) kemudian klik button submit | Mengklik submit tanpa mengisi data atau hanya sebagian data diisi | Sistem akan menolak dan menampilkan pesan “The field is required” | Sesuai harapan | Valid |
| 2 | Mengisikan  Kode barang  Nama barang  Jenis  Jumlah barang  Satuan  Tanggal EXP kemudian klik button submit | Mengklik submit dengan mengisi data lengkap dan benar | Sistem menerima akses input data barang dan kemudian akan muncul alert “stok barang berhasil ditambahkan” | Sesuai harapan | Valid |

### Pengujian form input data barang masuk

Table 4.. Pengujian form input data barang masuk

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Skenario | Test Case | Harapan | Hasil | Validasi |
| 1 | Kode barang  Nama barang  Jenis  Jumlah barang  Satuan  Tanggal masuk  Nama supplier  (tidak diisi) kemudian klik button submit | Mengklik submit tanpa mengisi data atau hanya sebagian data diisi | Sistem akan menolak dan menampilkan pesan “The field is required” | Sesuai harapan | Valid |
| 2 | Mengisikan  Kode barang  Nama barang  Jenis  Jumlah barang  Satuan  Tanggal masuk  Nama supplier kemudian klik button submit | Mengklik submit dengan mengisi data lengkap dan benar | Sistem menerima akses input data barang kemudian akan muncul alert “barang masuk berhasil ditambahkan” dan otomatis jumlah barang pada tabel stok barang akan berubah jumlahnya. | Sesuai harapan | Valid |

### Pengujian form input data barang keluar

Table 4.. Pengujian form input data barang keluar

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Skenario | Test Case | Harapan | Hasil | Validasi |
| 1 | Kode barang  Nama barang  Jenis  Jumlah barang  Satuan  Tanggal keluar  (tidak diisi) kemudian klik button submit | Mengklik submit tanpa mengisi data atau hanya sebagian data diisi | Sistem akan menolak dan menampilkan pesan “The field is required” | Sesuai harapan | Valid |
| 2 | Mengisikan  Kode barang  Nama barang  Jenis  Jumlah barang  Satuan  Tanggal keluar kemudian klik button submit | Mengklik submit dengan mengisi data lengkap dan benar | Sistem menerima akses input data barang kemudian akan muncul alert “barang keluar berhasil ditambahkan” dan otomatis jumlah barang pada tabel stok barang akan berubah jumlahnya. | Sesuai harapan | Valid |

### Pengujian form laporan pertanggal

Table 4.. Pengujian form laporan pertanggal

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Skenario | Test Case | Harapan | Hasil | Validasi |
| 1 | Tanggal awal  Tanggal akhir  (tidak diisi) kemudian klik button cetak | Mengklik cetak tanpa mengisi data atau hanya sebagian data diisi | Sistem menampilkan “page not found” | Sesuai harapan | Valid |
| 2 | Mengisikan Tanggal awal  Tanggal akhir  kemudian klik button cetak | Mengklik submit dengan mengisi data lengkap dan benar | Sistem akan menampilkan laporan stok barang/barang masuk/barang keluar berdasarkan range tanggal yang dibutuhkan kemudian hasil laporan dapat disimpan dalam bentuk pdf. | Sesuai harapan | Valid |

### Pengujian form ubah password

Table 4.. Pengujian form ubah password

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Skenario | Test Case | Harapan | Hasil | Validasi |
| 1 | Password lama  Password baru  Konformasi password  (tidak diisi) kemudian klik button simpan | Mengklik simpan tanpa mengisi data atau hanya sebagian data diisi | Sistem akan menolak dan menampilkan pesan “The field is required” | Sesuai harapan | Valid |
| 2 | Mengisikan  Password lama  Password baru (kurang dari 6 karakter)  Konformasi password kemudian klik button simpan | Mengklik simpan dengan mengisi password baru kurang dari 6 karakter | Sistem akan menolak dan menampilkan pesan “Panjang password baru minimal 6 karakter” | Sesuai harapan | Valid |
| 3 | Mengisikan  Password lama  Password baru (sama dengan password lama)  Konformasi password kemudian klik button simpan | Mengklik simpan dengan mengisi password baru sama dengan password lama | Sistem akan menolak dan menampilkan pesan “password baru tidak boleh sama” | Sesuai harapan | Valid |
| 4 | Mengisikan  Password lama  Password baru  Konformasi password (berbeda dengan password baru) kemudian klik button simpan | Mengklik simpan dengan mengisi konfirmasi password berbeda dengan password baru | Sistem akan menolak dan menampilkan pesan “password tidak valid” | Sesuai harapan | Valid |
| 5 | Mengisikan  Password lama  Password baru  Konformasi password kemudian klik button simpan | Mengklik simpan dengan mengisi data lengkap dan benar | Sistem akan menampilkan alert “password berhasil diubah” kemudian password baru sudah dapat digunakan untuk login | Sesuai harapan | Valid |

# BAB V PENUTUP

## Kesimpulan

Dari hasil perancangan dan pengujian sistem diatas maka diambil kesimpulan sebagi berikut :

1. Sistem dapat melakukan pengelolaan data pengguna, data jenis barang, data satuan barang, data barang, data supplier, data stok barang, data transaksi barang masuk dan barang keluar dan juga pengelolaan akun seperti mengupdate profil dan mengubah password. Sistem juga dapat membantu membuat pelaporan transaksi barang dari semua data hingga membuat pelaporan berdasarkan range tanggal.
2. Berdasarkan pengujian sistem yang ditunjukkan, sistem dapat berjalan sesuai dengan harapan.

## Saran

Saran yang dapat penulis sampaikan untuk pengembangan sistem informasi pergudangan bahan makanan kering ini agar lebih efisien dan dapat membantu pekerjaan pegawai gudang adalah diharapkan jika sistem ini dikembangkan agar dapat menghitung budgeting dan menambahkan harga barang agar lebih detail dalam transaksi barangnya dikarenakan sistem informasi ini hanya dapat mengelola data stok gudang dan transaksi barang saja.

# DAFTAR PUSTAKA

Arifianto, A. (2014). *SISTEM INFORMASI PERSEDIAAN BARANG PT. KARYA AGUNG.* Yogyakarta.

Hermanto, & Siti, K. (2020). Rancang Bangun Sistem Informasi Pergudangan Berbasis Website. *Jurnal Ilmu Komputer dan Desain Komunikasi Visual*, 22-34.

M, H. B., & Bramantiyo, E. P. (2018). Analisis Perancangan Sistem Informasi Pergudangan di CV. Karya Nugraha . *Jurnal Media Teknik & Sistem Industri*, 20-29.

Muhammad, I. (2010). SISTEM INFORMASI PENJUALAN PADA TOKO DO’A IBU GROSIR SEMBAKO DI DESA CIGUHA – BOGOR. *Jurnal Pro Bisnis*, 12-25.

Wahyudin, M. (2019). *ANALISA WAREHOUSE MANAGEMENT SYSTEM (WMS) GUDANG BAKU PT NUTRIFOOD INDONESIA.* Tegal: pelitabangsa.ac.id.