**RANCANG BANGUN SISTEM INFORMASI INVENTORI MENGGUNAKAN *FRAMEWORK LARAVEL* 8**

**DI PT. CEMARA KWANGJIN TEKSTIL**

**SKRIPSI**

**Karya Tulis sebagai syarat memperoleh**

**Gelar Sarjana Komputer dari Fakultas Teknologi Informasi**

**Universitas Bale Bandung**

Disusun oleh :

KURNIAWAN

NPM. 302190008

Logo

Description automatically generated

PROGRAM STRATA 1

PROGRAM STUDI SISTEM INFORMASI

FAKULTAS TEKNOLOGI INFORMASI

UNIVERSITAS BALE BANDUNG

BANDUNG

2023

**LEMBAR PERSETUJUAN PEMBIMBING**

RANCANG BANGUN SISTEM INFORMASI INVENTORI

MENGGUNAKAN *FRAMEWORK LARAVEL* 8

DI PT. CEMARA KWANGJIN TEKSTIL

Disusun oleh :

KURNIAWAN

NPM. 302190008

Telah diterima dan disetujui untuk memenuhi persyaratan mencapai gelar

**SARJANA KOMPUTER**Pada

**PROGRAM STUDI SISTEM INFORMASI  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS BALE BANDUNG**

Baleendah, Agustus 2023

|  |  |  |
| --- | --- | --- |
| Disetujui oleh: | | |
| Pembimbing Utama |  | Pembimbing Pendamping |
| Denny Rusdianto, S.T., M.Kom.  NIK. 04104808094 |  | Sutiyono, S.T., M.Kom.  NIK. 01043180002 |

**LEMBAR PERSETUJUAN PENGUJI**

RANCANG BANGUN SISTEM INFORMASI INVENTORI

MENGGUNAKAN *FRAMEWORK LARAVEL* 8

DI PT. CEMARA KWANGJIN TEKSTIL

Disusun oleh :

KURNIAWAN

NPM. 302190008

Telah diterima dan disetujui untuk memenuhi persyaratan mencapai gelar

**SARJANA KOMPUTER**Pada

**PROGRAM STUDI SISTEM INFORMASI  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS BALE BANDUNG**

Baleendah, Agustus 2023

|  |  |  |
| --- | --- | --- |
| Disetujui oleh: | | |
| Penguji 1 |  | Penguji 2 |
| Rosmalina, S.T., M.Kom.  NIK. 04104808122 |  | Khilda Nistrina, S.Pd., M.Sc.  NIK. 04104820004 |

**LEMBAR PENGESAHAN PROGRAM STUDI**

RANCANG BANGUN SISTEM INFORMASI INVENTORI

MENGGUNAKAN *FRAMEWORK LARAVEL* 8

DI PT. CEMARA KWANGJIN TEKSTIL

Disusun oleh :

KURNIAWAN

NPM. 302190008

SKRIPSI ini telah diterima dan disetujui untuk memenuhi persyaratan

mencapai gelar

**SARJANA KOMPUTER**

Pada

**PROGRAM STUDI SISTEM INFORMASI  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS BALE BANDUNG**

Baleendah, Agustus 2023

Disetujui oleh :

|  |  |  |
| --- | --- | --- |
| Mengetahui,  Dekan, |  | Mengesahkan,  Ketua Program Studi |
| Yudi Herdiana, S.T., M.T.  NIK. 04104808008 |  | Rosmalina, S.T., M.Kom.  NIK. 04104808122 |

**LEMBAR PERNYATAAN**

Saya yang bertanda tangan dibawah ini :

|  |  |  |
| --- | --- | --- |
| Nama | : | KURNIAWAN |
| NPM | : | 302190008 |
| Judul | : | RANCANG BANGUN SISTEM INFORMASI INVENTORI MENGGUNAKAN *FRAMEWORK LARAVEL* 8 DI PT. CEMARA KWANGJIN TEKSTIL |

Menyatakan dengan sebenarnya bahwa penulisan skripsi ini berdasarkan hasil penelitian, pemikiran dan pemaparan asli dari saya sendiri, baik untuk naskah laporan maupun kegiatan *programing* yang tercantum sebagai bagian dari skripsi ini. Jika terdapat karya orang lain, saya mencantumkan sumber yang jelas.

Pernyataan ini saya buat dengan sesungguhnya dan apabila dikemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya tulis ini dan sanksi lain sesuai dengan peraturan yang berlaku di FAKULTAS TEKNOLOGI INFORMASI UNIVERSITAS BALE BANDUNG. Demikian surat pernyataan ini saya buat dalam keadaan sadar tanpa paksaan dari pihak manapun.

|  |
| --- |
| Baleendah, Agustus 2023  Yang membuat pernyataan  KURNIAWAN  NPM. 302190008 |

# ABSTRAK

Sistem inventori atau sistem persediaan memiliki makna pengaturan persediaan dan berkaitan dengan aktivitas logistik sebuah perusahaan. Sistem inventori berguna untuk menentukan jumlah persediaan yang optimal dengan biaya total yang minimal. Salah satunya digunakan di PT. Cemara Kwangjin Tekstil yang digunakan dalam pengelolaan inventori kain. Pencatatan untuk data stok menggunakan *Microsoft Office Excel* dan berkas pencatatan. Namun pengguna yang masih awam akan fitur yang banyak pada *Microsoft Office excel* belum diketahui dan dipahami kegunaannya dan rawan terjadi perbedaan data antara di *Ms. Excel* dan sistem berkas. Maka perlu tersinkronisasi dengan transaksi atau data yang sebenarnya untuk menghindari kesalahan inputan pada pencatatan dan inputan tersebut membutuhkan waktu yang cukup banyak karena banyaknya data yang harus diinput ulang. Rancang Bangun inventori bisa menjadi solusi untuk mengatasi permasalahan tersebut. Pada sistem inventori nantinya hanya menginput barang masuk dan barang keluar secara otomatis tercatat pada data laporan masuk dan laporan keluar. Terutama pada jumlah stok barang atau inventori secara akurat sesuai dengan barang masuk dan barang keluar. Penelitian ini menggunakan metodologi *waterfall*. Tahap pertama yaitu analisis yaitu mengumpulkan data di PT. Cemara Kwangjin Tekstil kemudian data tersebut dianalisis untuk menemukan permasalahan yang lebih detail pada fokus penelitian. Pada tahap kedua yaitu perancangan dengan menggunakan model *Unified Model Language* dengan model yang digunakan yaitu *use case diagram, activity diagram, class diagram dan sequence diagram* dengan menggunakan *software Microsoft visio dan Balsamiq mockup*. Pada tahap selanjutnya implementasi menggunakan bahasa pemrograman PHP *framework Laravel* 8 dan pada tampilannya menggunakan *framework Bootstrap* 5 dengan *tool Visual studi code* dan XAMPP. Pada tahap tahap pengujian menggunakan metode *blackbox* yaitu hanya menguji keluaran atau output sesuai dengan yang diharapkan guna untuk menemukan kekurangan pada output yang dihasilkan. Hasil dari penelitian ini yaitu aplikasi sistem informasi inventori yang langsung menampilkan grafik stok barang yang tersisa. Memudahkan dalam transaksi penjualan dan pembelian kain serta hasil laporannya. Sehingga memudahkan karyawan dalam mengetahui stok barang yang tersisa dan memudahkan dalam pencatatan penjualan dan pembelian kain.

**Kata Kunci** : Aplikasi, Inventori, *Laravel,* Sistem Informasi*, Website*

# *ABSTRACT*

*Inventory system means inventory management and is related to the logistics activities of a company. The inventory system is useful for determining the optimal amount of inventory at minimal total cost. One of them is used at PT Cemara Kwangjin Tekstil which is used in fabric inventory management. Recording for stock data using Microsoft Office Excel and recording files. However, users who are still unfamiliar with the many features in Microsoft Office excel are not yet known and understand their usefulness and are prone to data differences between Ms. Excel and the file system. So it needs to be synchronized with the actual transaction or data to avoid input errors in the recording and the input takes quite a lot of time because of the large amount of data that must be re-input. Inventory design can be a solution to overcome these problems. The inventory system will only input incoming goods and outgoing goods automatically recorded in the incoming report data and outgoing reports. Especially in the number of stock items or inventory accurately in accordance with incoming goods and outgoing goods. This research uses waterfall methodology. The first stage is analysis, namely collecting data at PT Cemara Kwangjin Tekstil then the data is analyzed to find more detailed problems in the focus of the research. In the second stage, namely design using the Unified Model Language model with the models used, namely use case diagrams, activity diagrams, class diagrams and sequence diagrams using Microsoft Visio and Balsamiq mockup software. In the next stage, the implementation uses the PHP programming language Laravel 8 framework and on the display uses the Bootstrap 5 framework with the Visual study code tool and XAMPP. At the testing stage using the blackbox method, which only tests the output or output as expected in order to find deficiencies in the resulting output. The result of this research is an inventory information system application that immediately displays a graph of the remaining stock of goods. Facilitate in fabric sales and purchase transactions and report results. Making it easier for employees to find out the remaining stock of goods and facilitate the recording of sales and purchases of fabrics.*

***Keywords****: Application, Information System, Inventory, Laravel, Website*

# KATA PENGANTAR

Dengan penuh rasa syukur, segala puji hanya bagi Allah SWT yang telah melimpahkan rahmat serta anugerah-Nya. Dengan karunia-Nya, Laporan Skripsi ini berhasil diselesaikan dengan tepat waktu.

Laporan ini dibuat dengan berbagai observasi dan beberapa bantuan dari berbagai pihak untuk membantu menyelesaikan tantangan dan hambatan selama mengerjakan laporan ini. Oleh karena itu, pada kesempatan kali ini penulis ingin mengucapkan terimakasih yang sebesar-besarnya kepada :

1. Kedua orang tua yang tidak pernah letih memberikan dukungan dan do’a untuk keberhasilan penulis.
2. Bapak Yudi Herdiana, S.T., M.T. selaku Dekan Fakultas Teknologi Informasi.
3. Ibu Rosmalina S.T, M.Kom selaku Ketua Program Studi S1 Sistem Informasi Universitas Bale Bandung.
4. Bapak Denny Rurdianto, S.T., M.Kom. selaku pembimbing utama.
5. Bapak Sutiyono, M.Kom., selaku pembimbing pendamping.
6. Atasan dan rekan kerja PT. Cemara Kwangjin Tekstil.
7. Seluruh Dosen beserta staff Fakultas Teknologi Informasi Universitas Bale Bandung.
8. Semua kawan-kawan yang selalu memberikan dukungan dan bantuannya yang tidak bisa penulis disebutkan satu per satu.

Penulis menyadari bahwa masih banyak kekurangan dalam laporan ini. Akhir kata semoga laporan skripsi ini bisa diterima dan bermanfaat bagi kita semua.

|  |
| --- |
| Bandung, Agustus 2023  Kurniawan  NPM. 302190008 |

# DAFTAR ISI

[ABSTRAK vi](#_Toc147167986)

[*ABSTRACT* vii](#_Toc147167987)

[KATA PENGANTAR viii](#_Toc147167988)

[DAFTAR ISI ix](#_Toc147167989)

[DAFTAR GAMBAR xi](#_Toc147167990)

[DAFTAR TABEL xiv](#_Toc147167991)

[DAFTAR LAMPIRAN xv](#_Toc147167992)

[BAB I PENDAHULUAN 1](#_Toc147167993)

[1.1 Latar Belakang 1](#_Toc147167994)

[1.2 Rumusan Masalah 2](#_Toc147167995)

[1.3 Batasan Masalah 2](#_Toc147167996)

[1.4 Tujuan Penelitian 3](#_Toc147167997)

[1.5 Metodologi Penelitian 3](#_Toc147167998)

[1.5.1 Metode Pengumpulan Data 3](#_Toc147167999)

[1.5.2 Metode Pengembangan Sistem 4](#_Toc147168000)

[1.6 Sistematika Penulisan 4](#_Toc147168001)

[BAB II KAJIAN PUSTAKA 5](#_Toc147168002)

[2.1 Landasan Teori 5](#_Toc147168004)

[2.2 Dasar Teori 8](#_Toc147168005)

[2.2.1 Rancang Bangun 8](#_Toc147168006)

[2.2.2 Sistem Informasi 9](#_Toc147168007)

[2.2.3 Inventori 9](#_Toc147168008)

[2.2.4 Metode SDLC *Waterfall* 10](#_Toc147168009)

[*2.2.5 Unified Modelling Language* 11](#_Toc147168010)

[2.2.6 PHP 15](#_Toc147168011)

[*2.2.7 Laravel* 16](#_Toc147168012)

[*2.2.8 Bootstrap* 17](#_Toc147168013)

[2.2.9 XAMPP 18](#_Toc147168014)

[*2.2.10 Visual Studio Code* 20](#_Toc147168015)

[*2.2.11 Google Chrome* 21](#_Toc147168016)

[*2.2.12 Microsoft Visio* 22](#_Toc147168017)

[2.2.13 *Balsamiq* *Mockup* 23](#_Toc147168018)

[BAB III METODOLOGI PENELITIAN 25](#_Toc147168019)

[3.1 Kerangka Pikir 25](#_Toc147168022)

[3.2 Deskripsi 26](#_Toc147168023)

[3.2.1 Metode Pengumpulan Data 26](#_Toc147168024)

[3.2.2 Analisis 26](#_Toc147168025)

[3.2.3 Desain 28](#_Toc147168026)

[3.2.4 Pengkodean 30](#_Toc147168027)

[3.2.5 Pengujian 30](#_Toc147168028)

[BAB IV ANALISIS DAN PERANCANGAN 31](#_Toc147168029)

[4.1 Analisis 31](#_Toc147168031)

[4.1.1 Analisis Tata Kelola PT. Cemara Kwangjin Tekstil 31](#_Toc147168032)

[4.1.2 Analisis Sistem 32](#_Toc147168033)

[4.1.3 Analisis Masalah 32](#_Toc147168034)

[4.1.4 Analisis Kebutuhan Sistem 33](#_Toc147168035)

[4.1.5 Analisis Kebutuhan 34](#_Toc147168036)

[4.1.6 Analisis Penunjang Keputusan 35](#_Toc147168037)

[4.1.7 Analisis Biaya 35](#_Toc147168038)

[4.2 Perancangan 35](#_Toc147168039)

[4.2.1 Pemodelan UML (*Unified Modeling Language*) 36](#_Toc147168040)

[4.2.2 Struktur Tabel 61](#_Toc147168041)

[4.2.3 Desain 62](#_Toc147168042)

[*4.2.4 Listing Program* 70](#_Toc147168043)

[BAB V IMPLEMENTASI DAN PENGUJIAN 75](#_Toc147168044)

[5.1 Implementasi 75](#_Toc147168046)

[5.2 Pengujian 83](#_Toc147168047)

[BAB VI KESIMPULAN DAN SARAN 85](#_Toc147168048)

[6.1 Kesimpulan 85](#_Toc147168050)

[6.2 Saran 85](#_Toc147168051)

[DAFTAR PUSTAKA 87](#_Toc147168052)

[LAMPIRAN 89](#_Toc147168053)

# DAFTAR GAMBAR

[Gambar 2.1 Model *waterfall* 10](#_Toc136295138)

[Gambar 3.1 Kerangka Pikir 25](#_Toc136295145)

[Gambar 4.1 Struktur organisasi 31](#_Toc143544178)

[Gambar 4.2 *Use case diagram* 36](#_Toc143544179)

[Gambar 4.3 *Activity diagram* login 38](#_Toc143544180)

[Gambar 4.4 *Activity diagram* tambah kain 38](#_Toc143544181)

[Gambar 4.5 *Activity diagram* edit kain 39](#_Toc143544182)

[Gambar 4.6 *Activity diagram* hapus kain 39](#_Toc143544183)

[Gambar 4.7 *Activity diagram* tambah kolega 40](#_Toc143544184)

[Gambar 4.8 *Activity diagram* edit kolega 40](#_Toc143544185)

[Gambar 4.9 *Activity diagram* hapus kolega 41](#_Toc143544186)

[Gambar 4.10 *Activity diagram* tambah penyimpanan 41](#_Toc143544187)

[Gambar 4.11 *Activity diagram* edit penyimpanan 42](#_Toc143544188)

[Gambar 4.12 *Activity diagram* hapus penyimpanan 42](#_Toc143544189)

[Gambar 4.13 *Activity diagram* tambah penjualan 43](#_Toc143544190)

[Gambar 4.14 *Activity diagram* edit penjualan 43](#_Toc143544191)

[Gambar 4.15 *Activity diagram* hapus penjualan 44](#_Toc143544192)

[Gambar 4.16 *Activity diagram* laporan penjualan 44](#_Toc143544193)

[Gambar 4.17 *Activity diagram* surat jalan penjualan 45](#_Toc143544194)

[Gambar 4.18 *Activity diagram* tambah pembelian 45](#_Toc143544195)

[Gambar 4.19 *Activity diagram* edit pembelian 46](#_Toc143544196)

[Gambar 4.20 *Activity diagram* hapus pembelian 46](#_Toc143544197)

[Gambar 4.21 *Activity diagram* laporan pembelian 47](#_Toc143544198)

[Gambar 4.22 *Activity diagram* tambah pengguna 47](#_Toc143544199)

[Gambar 4.23 *Activity diagram* edit pengguna 48](#_Toc143544200)

[Gambar 4.24 *Activity diagram* hapus pengguna 48](#_Toc143544201)

[Gambar 4.25 *Class diagram* 49](#_Toc143544202)

[Gambar 4.26 *Sequence diagram login* 50](#_Toc143544203)

[Gambar 4.27 *Sequence* *diagram* tambah kain 50](#_Toc143544204)

[Gambar 4.28 *Sequence diagram* edit kain 51](#_Toc143544205)

[Gambar 4.29 *Sequence diagram* hapus kain 51](#_Toc143544206)

[Gambar 4.30 *Sequence diagram* tambah kolega 52](#_Toc143544207)

[Gambar 4.31 *Sequence diagram* edit kolega 52](#_Toc143544208)

[Gambar 4.32 *Sequence diagram* hapus kolega 53](#_Toc143544209)

[Gambar 4.33 *Sequence diagram* tambah penyimpanan 53](#_Toc143544210)

[Gambar 4.34 *Sequence diagram* edit penyimpanan 54](#_Toc143544211)

[Gambar 4.35 *Sequence diagram* hapus penyimpanan 54](#_Toc143544212)

[Gambar 4.36 *Sequence diagram* tambah penjualan 55](#_Toc143544213)

[Gambar 4.37 *Sequence diagram* edit penjualan 55](#_Toc143544214)

[Gambar 4.38 *Sequence diagram* hapus penjualan 56](#_Toc143544215)

[Gambar 4.39 *Sequence diagram* laporan penjualan 56](#_Toc143544216)

[Gambar 4.40 *Sequence diagram* surat jalan penjualan 57](#_Toc143544217)

[Gambar 4.41 *Sequence diagram* tambah pembelian 57](#_Toc143544218)

[Gambar 4.42 *Sequence diagram* edit pembelian 58](#_Toc143544219)

[Gambar 4.43 *Sequence diagram* hapus pembelian 58](#_Toc143544220)

[Gambar 4.44 *Sequence diagram* laporan pembelian 59](#_Toc143544221)

[Gambar 4.45 *Sequence diagram* tambah pengguna 59](#_Toc143544222)

[Gambar 4.46 *Sequence diagram* edit pengguna 60](#_Toc143544223)

[Gambar 4.47 *Sequence diagram* hapus pengguna 60](#_Toc143544224)

[Gambar 4.48 Desain tampilan *login* 63](#_Toc143544225)

[Gambar 4.49 Desain tampilan *dashboard* admin 63](#_Toc143544226)

[Gambar 4.50 Desain tampilan edit kain 63](#_Toc143544227)

[Gambar 4.51 Desain tampilan tambah kain 64](#_Toc143544228)

[Gambar 4.52 Desain tampilan tabel kain 64](#_Toc143544229)

[Gambar 4.53 Desain tampilan edit kolega 64](#_Toc143544230)

[Gambar 4.54 Desain tampilan tambah kolega 65](#_Toc143544231)

[Gambar 4.55 Desain tampilan tabel kolega 65](#_Toc143544232)

[Gambar 4.56 Desain tampilan edit penyimpanan 65](#_Toc143544233)

[Gambar 4.57 Desain tampilan tambah penyimpanan 66](#_Toc143544234)

[Gambar 4.58 Desain tampilan tabel penyimpanan 66](#_Toc143544235)

[Gambar 4.59 Desain tampilan edit penjualan 66](#_Toc143544236)

[Gambar 4.60 Desain tampilan laporan penjualan 67](#_Toc143544237)

[Gambar 4.61 Desain tampilan surat jalan penjualan 67](#_Toc143544238)

[Gambar 4.62 Desain tampilan tambah penjualan 67](#_Toc143544239)

[Gambar 4.63 Desain tampilan tabel penjualan 68](#_Toc143544240)

[Gambar 4.64 Desain tampilan tambah pembelian 68](#_Toc143544241)

[Gambar 4.65 Desain tampilan laporan pembelian 68](#_Toc143544242)

[Gambar 4.66 Desain tampilan tambah pembelian 69](#_Toc143544243)

[Gambar 4.67 Desain tampilan tabel pembelian 69](#_Toc143544244)

[Gambar 4.68 Desain tampilan edit pembelian 69](#_Toc143544245)

[Gambar 4.69 Desain tampilan tambah pembelian 70](#_Toc143544246)

[Gambar 4.70 Desain tampilan tabel pembelian 70](#_Toc143544247)

[Gambar 5.1 Halaman *login* 75](#_Toc143544248)

[Gambar 5.2 Halaman *dashboard* 75](#_Toc143544249)

[Gambar 5.3 Halaman tabel kain 76](#_Toc143544250)

[Gambar 5.4 Halaman edit kain 76](#_Toc143544251)

[Gambar 5.5 Halaman tambah kain 76](#_Toc143544252)

[Gambar 5.6 Halaman tabel kolega 77](#_Toc143544253)

[Gambar 5.7 Halaman edit kolega 77](#_Toc143544254)

[Gambar 5.8 Halaman tambah kolega 77](#_Toc143544255)

[Gambar 5.9 Halaman tabel penyimpanan 78](#_Toc143544256)

[Gambar 5.10 Halaman edit penyimpanan 78](#_Toc143544257)

[Gambar 5.11 Halaman tambah penyimpanan 78](#_Toc143544258)

[Gambar 5.12 Halaman edit penjualan 79](#_Toc143544259)

[Gambar 5.13 Halaman laporan penjualan 79](#_Toc143544260)

[Gambar 5.14 Halaman surat jalan penjualan 79](#_Toc143544261)

[Gambar 5.15 Halaman tambah penjualan 80](#_Toc143544262)

[Gambar 5.16 Halaman tabel penjualan 80](#_Toc143544263)

[Gambar 5.17 Halaman edit pembelian 80](#_Toc143544264)

[Gambar 5.18 Halaman laporan pembelian 81](#_Toc143544265)

[Gambar 5.19 Halaman tambah pembelian 81](#_Toc143544266)

[Gambar 5.20 Halaman tabel pembelian 81](#_Toc143544267)

[Gambar 5.21 Halaman edit pengguna 82](#_Toc143544268)

[Gambar 5.22 Halaman tambah pengguna 82](#_Toc143544269)

[Gambar 5.23 Halaman tabel pengguna 82](#_Toc143544270)

# DAFTAR TABEL

[Tabel 2.1 Penelitian terdahulu 5](#_Toc136295183)

[Tabel 2.2 *Use case diagram* 12](#_Toc136295184)

[Tabel 2.3 *Activity diagram* 13](#_Toc136295185)

[Tabel 2.4 *Class diagram* 13](#_Toc136295186)

[Tabel 3.1 Spesifikasi perangkat keras 27](#_Toc136295194)

[Tabel 4.1 Analisis masalah 32](#_Toc143544271)

[Tabel 4.2 Analisis biaya 35](#_Toc143544272)

[Tabel 4.3 Deskripsi aktor aplikasi 36](#_Toc143544273)

[Tabel 4.4 Deskripsi *use case* Admin 37](#_Toc143544274)

[Tabel 4.5 Deskripsi *use case* Manajer 37](#_Toc143544275)

[Tabel 4.6 Deskripsi *use case* Marketing 37](#_Toc143544276)

[Tabel 4.7 Data pengguna 61](#_Toc143544277)

[Tabel 4.8 Data kain 61](#_Toc143544278)

[Tabel 4.9 Data kolega 61](#_Toc143544279)

[Tabel 4.10 Data pembelian 62](#_Toc143544280)

[Tabel 4.11 Data penjualan 62](#_Toc143544281)

[Tabel 4.12 Data penyimpanan 62](#_Toc143544282)

[Tabel 5.1 Hasil pengujian 83](#_Toc142770270)

# DAFTAR LAMPIRAN

[Lampiran 1 : Hasil Penelitian Terdahulu 89](#_Toc147167958)

[Lampiran 2 : Hasil Wawancara 94](#_Toc147167959)

[Lampiran 3 : Observasi 97](#_Toc147167960)

# PENDAHULUAN

## Latar Belakang

Sistem inventori atau sistem persediaan memiliki makna pengaturan persediaan dan berkaitan dengan aktivitas logistik sebuah perusahaan. Di mana kegiatan dari sistem tersebut termasuk dalam pengecekan dan penyediaan stok bahan baku atau barang setengah jadi ataupun barang jadi, demi kelancaran proses produksi atau pemenuhan permintaan pelanggan. Sistem inventori berguna untuk menentukan jumlah persediaan yang optimal dengan biaya total yang minimal. Persediaan atau inventori meliputi bahan mentah atau bahan baku, bahan pembantu, bahan dalam proses atau *work in process*, suku cadang, dan barang jadi atau *finished good*, karena timbulnya ketidakpastian permintaan, ketidakpastian pasokan supplier, dan ketidakpastian waktu pemesanan (Mubarok dkk., 2019).

PT. Cemara Kwangjin Tekstil yang bergerak dalam bidang industri manufaktur dalam pembuatan produk kain tekstil (Tekstil Industries). Instansi ini tentunya sudah banyak berbisnis dengan instansi lain untuk bekerja sama mengembangkan instansi masing-masing yang saling memenuhi kebutuhannya. Tentunya kerjasama ini perlunya pencatatan bukti transaksi dari terima barang ataupun keluar barang. Pencatatan untuk data stok menggunakan *Microsoft Office Excel* dan berkas pencatatan. Pengolahan data dengan menggunakan *Microsoft Office Excel* sangat membantu admin untuk mengolah data persediaan barang dibandingkan dengan menggunakan hanya sistem berkas.

Pengolahan data dengan menggunakan *Microsoft Office excel* tentunya sangat membantu dan memudahkan. Namun pengguna yang masih awam akan fitur yang banyak pada menu bar dan *toolbar Microsoft Office excel* belum diketahui dan dipahami kegunaannya dan rawan terjadi perbedaan data antara di MS. Excel dan sistem berkas. Maka perlu tersinkronisasi dengan transaksi atau data yang sebenarnya untuk menghindari kesalahan inputan pada pencatatan dan inputan tersebut membutuhkan waktu yang cukup banyak karena banyaknya data yang harus diinput ulang. Sehingga permasalahan ini perlu diatasi demi kelancaran instansi dalam pengelolaan data dan memudahkan pengambilan keputusan yang akurat.

Rancang Bangun inventori bisa menjadi solusi untuk mengatasi permasalahan tersebut. Pada sistem inventori nantinya hanya menginput barang masuk dan barang keluar secara otomatis tercatat pada data laporan masuk dan laporan keluar. Terutama pada jumlah stok barang atau inventori secara akurat sesuai dengan barang masuk dan barang keluar. Dengan aplikasi ini bisa diakses oleh berbagai perangkat. Bahkan demi mengamankan data hanya beberapa pegawai yang terdaftar pada akun aplikasi inventori yang hanya bisa mengakses aplikasi yaitu admin, marketing dan manajer.

Sehingga untuk mengatasi permasalahan tersebut penulis mengambil judul skripsi yaitu Rancang Bangun Sistem Infromasi Inventori Menggunakan *Laravel* 8 di PT. Cemara Kwangjin Tekstil. Dengan rancang bangun ini dapat membantu meringankan dan memudahkan dalam pengelolaan data dengan cepat dan akurat.

## Rumusan Masalah

Berdasarkan latar belakang penelitian ini maka dirumuskan masalah yang akan diteliti yaitu :

1. Bagaimana cara membuat sistem inventori yang mudah dan akurat ?
2. Bagaimana cara menghilangkan kesalahan input data persediaan barang ?
3. Bagaimana cara membuat laporan pembelian dan penjualan yang lebih praktis ?

## Batasan Masalah

Batasan masalah pada penelitian ini yaitu :

1. Penelitian ini berfokus pada inventori
2. Aplikasi ini menggunakan framework *Laravel* 8 dan *Bootstrap* 5 sebagai *tool* pengkodeannya.
3. Aplikasi ini menampilkan grafik inventori barang.
4. Input utama dalam penelitian ini yaitu data kain, penjualan dan pembelian.
5. Output yang menjadi fokus penelitian yaitu laporan penjualan dan pembelian.

## Tujuan Penelitian

Tujuan yang ingin dicapai untuk mengatasi permasalahan pada penelitian ini yaitu :

1. Membuat sistem inventori yang mudah dan akurat.
2. Menghilangkan kesalahan input data persediaan barang.
3. Membuat laporan pembelian dan penjualan yang lebih praktis.

## Metodologi Penelitian

Metode penelitian digunakan sebagai suatu cara untuk memperoleh hasil terkait dengan suatu permasalahan yang spesifik, yang seringkali disebut sebagai permasalahan penelitian. Untuk menyelesaikan permasalahan tersebut, dapat dilakukan dengan menggunakan metode pengumpulan data dan pengembangan sistem. Berikut metode yang digunakan dalam penelitian ini.

### Metode Pengumpulan Data

Beberapa metode yang digunakan dalam pengumpulan data untuk mempermudah dan memperlancar penelitian ini yaitu :

1. Wawancara

Wawancara yaitu suatu proses komunikasi antara dua atau lebih individu yang dilakukan secara tatap muka atau melalui media telekomunikasi, dengan tujuan untuk mendapatkan informasi atau data yang relevan terkait dengan topik atau subjek yang sedang dibahas.

1. Observasi

Observasi yaitu suatu metode penelitian yang dilakukan dengan mengamati perilaku dan kejadian yang terjadi pada subjek atau obyek yang diteliti. Observasi dapat dilakukan dengan cara langsung, yaitu mengamati secara langsung di lapangan, atau tidak langsung, yaitu melalui video atau rekaman lainnya.

1. Studi Pustaka

Studi pustaka yaitu metode penelitian yang dilakukan dengan mengumpulkan dan menganalisis data dari berbagai sumber tertulis, seperti buku, jurnal, laporan penelitian, dan lain-lain. Studi pustaka dapat dilakukan untuk mendapatkan informasi terkait dengan topik atau subjek yang sedang diteliti.

1. Studi Dokumentasi

Studi dokumentasi adalah proses penelitian atau penulisan yang melibatkan pengumpulan, analisis, dan interpretasi berbagai dokumen atau sumber informasi tertulis yang relevan terkait dengan suatu topik, masalah, atau subjek yang sedang diteliti.

### Metode Pengembangan Sistem

Metode pengembangan sistem uang akan digunakan dalam penelitian ini yaitu menggunakan metode *System Development Life Cycle* (SDLC) model *waterfall*. Tahapan metode model waterfall yang akan digunakan dalam penelitian ini yaitu analisis, perancangan, implementasi dan pengujian.

## Sistematika Penulisan

Berikut merupakan sistematika penulisan laporan penelitian ini yaitu :

**BAB I : PENDAHULUAN**

Pada bab ini berisi mengenai latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, metodologi penelitian dan sistematika penulisan.

**BAB II : TINJAUAN PUSTAKA**

Pada bab ini berisi landasan teori dan dasar teori.

**BAB III : METODOLOGI PENELITIAN**

Pada bab ini berisi mengenai kerangka pikir dan deskripsi.

**BAB IV : ANALISIS DAN PERANCANGAN**

Pada bab ini analisis dan perancangan.

**BAB V : IMPLEMENTASI DAN PENGUJIAN**

Pada bab ini berisi mengenai implementasi dan pengujian.

**BAB VI : KESIMPULAN DAN SARAN**

Pada bab ini berisi mengenai kesimpulan dan saran.

# KAJIAN PUSTAKA



## Landasan Teori

Landasan teori mengandung referensi atau sumber ilmu yang digunakan dalam penelitian, terutama yang terkait dengan topik penelitian. Beberapa judul jurnal yang digunakan dalam penelitian ini digunakan untuk memperkuat argumen dan memberikan dasar pengetahuan yang solid dalam penelitian ini. Berikut jurnal terdahulu yang menjadi referensi :

Tabel 2.1 Penelitian terdahulu

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Judul Artikel | Masalah | Metode | Solusi |
| 1 | Sistem Informasi Inventori Dalam Mengolah Bahan Kayu Menjadi Produk Kusen di PB. Mekar Laksana | Permasalahan perusahaan adalah permasalahan inventori, dimana ketidaksamaan stok bahan baku yang tersedia dengan data perusahaan. Kemudian tidak efisiennya waktu dalam pencatatan transaksi. | Metode *waterfall* dengan langkah : analisa kebutuhan perangkat lunak, desain, pengkodean, pengujian dan pemeliharaan. | Pembuatan aplikasi berbasis web dimana untuk data inventori dapat diakses dengan mudah bukan hanya secara offline saja melainkan juga secara online. |
| 2 | Sistem Informasi Inventori Menggunakan Framework Laravel Pada CV. Grace Bhakti Utama Bogor | Kesalahan saat penginputan secara manual akan menghasilkan laporan yang tidak akurat, hal ini merupakan permasalahan yang harus diselesaikan dalam perusahaan agar sistem berjalan secara efektif dan efisien. | Metode *waterfall* dengan tahap : analisis kebutuhan perangkat lunak, desain, pembuatan kode program, pengujian dan pendukung(support) | Sistem informasi dapat mempercepat waktu pencatatan penginputan data serta dapat meminimalisir kesalahan yang terjadi saat pencatatan data. Stok bisa dilihat dalam sistem, proses pencarian data barang atau stok barang juga menjadi lebih cepat, penyimpanan dan laporan barang masuk dan keluar juga menjadi lebih tertata dan efesien. |
| 3 | Rancang Bangun Sistem Informasi Inventori Pada Salon Kecantikan | Penanganan data yang masih menggunakan sistem manual mengakibatkan sering terjadinya kesalahan. Melihat ketersediaan stok barang masih manual yang sangat membutuhkan banyak waktu dan ketelitian. | Metode *waterfall* dengan langkah analisis kebutuhan, desain*, code, testing* dan *support.* | Dalam penelitian ini menghasilkan sebuah aplikasi perancangan sistem informasi inventori yang dapat menghasilkan sebuah laporan stok barang. Diharapkan dengan adanya penelitian ini, maka akan mempermudah proses monitoring stok barang masuk dan barang keluar. |

1. Ade Mubarok, Cairil M. Noor dan Bambang Sukajie(2019), menyatakan sistem inventori merupakan suatu sistem untuk mengetahui persediaan stok barang pada suatu tempat. Sistem inventori sudah banyak digunakan atau dikembangkan pada suatu tempat dengan berbagai macam teknologi dan sistem. Permasalahan pada PB. Mekar Laksana ini adalah belum tersedianya sistem inventori barang sehingga belum bisa mengontrol stok barang yang tersedia. Metode pengembangan sistem pada aplikasi ini menggunakan metode *waterfall* yang terdiri dari lima tahapan. Adapun tahapan tersebut yaitu Analisis dan kebutuhan Sistem, desain, *Code Generation, testing support*. Sistem inventori ini dapat dimanfaatkan dengan beberapa fasilitas yang disediakan yaitu melalui web. Dalam aplikasi ini Manajer dapat mengetahui data barang, data barang masuk, data barang keluar dan *suplier* yang ingin mengetahui stok barang *suplier* tersebut. Admin, manajemen dan *suplier* dapat berinteraksi dengan aplikasi ini sesuai dengan informasi yang dibutuhkan melalui menu yang tersedia(Mubarok dkk., 2019).
2. Marshel Christian, Suparni dan Lilyani Asri utami(2021) menyatakan Sistem inventori merupakan sebuah sistem yang mencatat segala perpindahan barang masuk maupun barang keluar. Banyaknya data transaksi yang harus dikelola setiap harinya mengakibatkan waktu yang diperlukan untuk menyelesaikan akan lebih lama apabila sistem yang digunakan masih secara manual. Demikian yang terjadi pada CV. Grace Bhakti Utama. Kesalahan saat penginputan secara manual akan menghasilkan laporan yang tidak akurat, hal ini merupakan permasalahan yang harus diselesaikan dalam perusahaan agar sistem berjalan secara efektif dan efisien. Penelitian ini mengusulkan perancangan suatu sistem informasi inventori menggunakan *framework Laravel* untuk memberikan kemudahan bagi pengguna dalam mengelola data inventori di CV. Grace Bhakti Utama. Metode pengembangan *system* dalam penelitian ini menggunakan metode *Waterfall*, merupakan model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengkodean, pengujian, dan tahap pendukung(Marshel Christian dkk., 2021).
3. Elly Mufida, Eva Rahmawati, Hylenarti Hertiana(2019) menyatakan perusahaan atau unit usaha memanfaatkan peranan internet untuk aktivitas perusahaan, Aplikasi pendukung aktivitas perusahaan yang di buat sudah mulai berbasis web baik itu yang berkaitan dengan administrasi, finance, marketing, dan pengelolaan barang atau inventori sehingga informasi yang di dapat lebih cepat dan mudah diakses dimana saja. Permasalahan inventori sering sekali menjadi kendala. Penangan data yang masih dilakukan dengan menggunakan sistem manual mengakibatkan sering terjadinya kesalahan yang ada. Dalam perkembangan perusahaan untuk melihat ketersediaan stok barang masih menggunakan sistem manual yang sangat membutuhkan banyak waktu dan ketelitian. Dalam penelitian ini menghasilkan sebuah aplikasi perancangan sistem informasi inventori yang dapat menghasilkan sebuah laporan stok barang. Diharapkan dengan adanya penelitian ini, maka akan mempermudah proses monitoring stok barang masuk dan barang keluar. Penelitian ini menggunakan UML sebagai tools dan PHP sebagai bahasa pemrograman dalam pembuatan rancang bangun sistem inventori salon(Mufida dkk., 2019).

## Dasar Teori

Dalam rancang bangun aplikasi inventori, tentu saja dibutuhkan penggunaan metode-metode atau teori-teori dasar yang dapat mendukung studi literatur dalam mengumpulkan data yang relevan dengan penelitian rancang bangun aplikasi sistem informasi inventori. Beberapa contoh teori dasar tersebut antara lain:

### Rancang Bangun

Rancang adalah serangkaian langkah-langkah yang digunakan untuk mengubah hasil analisis suatu sistem menjadi bahasa pemrograman dengan tujuan menggambarkan secara rinci bagaimana komponen-komponen sistem diimplementasikan. Sementara itu, bangun lebih merujuk pada kegiatan pembuatan sistem baru, serta perbaikan atau penggantian sistem yang sudah ada, baik secara keseluruhan maupun sebagian(marketing, 2021).

Dengan demikian, Rancang Bangun adalah proses yang melibatkan penerjemahan hasil analisis ke dalam bentuk perangkat lunak untuk merancang sebuah proyek kemudian dibuat, dikembangkan ataupun dibangun proyek tersebut sebagaimana rancangan yang sudah dibuat demi mencapai tujuan.

Berikut beberapa manfaat rancang bangun :

Meningkatkan efisiensi: Dengan melakukan rancang bangun sebelum memulai pembangunan atau produksi, dapat membantu mengurangi waktu dan biaya yang dibutuhkan, serta meningkatkan efisiensi dalam proses tersebut.

Menghasilkan produk atau sistem yang berkualitas: Dengan melakukan rancang bangun yang baik, produk atau sistem yang dihasilkan dapat memiliki kualitas yang lebih baik dan sesuai dengan kebutuhan pengguna.

Meminimalkan resiko kegagalan: Dalam rancang bangun, dapat dilakukan uji coba dan analisis resiko sehingga dapat meminimalkan resiko kegagalan pada produk atau sistem yang dihasilkan.

Meningkatkan daya saing: Dengan memiliki produk atau sistem yang berkualitas dan efisien, dapat meningkatkan daya saing di pasar dan memberikan keuntungan bagi perusahaan.

Memenuhi standar keselamatan: Dalam rancang bangun, dapat memperhatikan standar keselamatan dan regulasi yang berlaku sehingga dapat memastikan produk atau sistem yang dihasilkan aman dan sesuai dengan regulasi yang ada.

### Sistem Informasi

John F. Nash menyampaikan bahwa sistem informasi dapat diartikan sebagai gabungan dari elemen manusia, fasilitas, teknologi, media, prosedur, dan pengendalian yang bertujuan untuk mengatur jaringan komunikasi yang penting, proses, dan transaksi tertentu secara rutin. Sistem ini bertujuan untuk membantu manajemen dan pengguna internal maupun eksternal, serta memberikan dasar bagi pengambilan keputusan yang tepat.

Sedangkan menurut Robert A. Leitch bahwa sistem informasi adalah sebuah sistem yang ada di dalam sebuah organisasi yang bertujuan untuk memenuhi kebutuhan pengolahan transaksi harian, mendukung operasi manajerial, dan mendukung kegiatan strategis organisasi. Sistem ini juga berfungsi untuk menyediakan laporan yang diperlukan oleh pihak eksternal.

Dari pengertian tersebut dapat disimpulkan bahwa sistem informasi merupakan kombinasi dari elemen manusia, fasilitas, teknologi, media, prosedur, dan pengendalian yang bertujuan untuk mengatur jaringan komunikasi, proses, dan transaksi rutin dalam suatu organisasi. Sistem ini memiliki fungsi yang meliputi memenuhi kebutuhan pengolahan transaksi harian, mendukung operasi manajerial, mendukung kegiatan strategis organisasi, serta menyediakan laporan yang dibutuhkan oleh pihak eksternal. Tujuan utama dari sistem informasi adalah membantu manajemen dan pengguna internal maupun eksternal dalam pengambilan keputusan yang tepat.

### Inventori

Menurut Yuhendra, M.T, Dr. Eng (2013) mengungkapkan bahwa inventori adalah komponen dari proses produksi di perusahaan yang digunakan untuk memproduksi barang, serta termasuk barang jadi yang tersedia untuk memenuhi permintaan pelanggan setiap saat. Inventori ini disimpan dan dikelola dengan aturan tertentu dalam kondisi siap pakai dan tercatat dalam database(Gosal & Rustam, 2022).

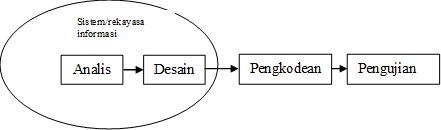
Inventori, yang juga dikenal sebagai persediaan, merujuk pada stok barang mentah, bahan baku, atau produk jadi yang disimpan untuk digunakan di masa depan atau dalam periode tertentu. Persediaan barang sangat krusial bagi sebuah perusahaan untuk menghadapi perubahan pasar produksi serta mengantisipasi fluktuasi harga yang mungkin terjadi pada permintaan barang. Inventori atau persediaan adalah kumpulan sumber daya, termasuk bahan mentah dan produk jadi, yang tersedia untuk dipakai oleh perusahaan guna memenuhi permintaan dari pelanggan. Dalam definisi lain, inventori adalah suatu teknik dalam manajemen material yang terkait dengan pengaturan persediaan(Mufida dkk., 2019).

Maka bisa disimpulkan bahwa inventori merupakan penyimpanan barang atau produk yang nantinya mudah untuk diambil kembali ketika dibutuhkan serta memudahkan dalam mengetahui stok barang yang disimpan tersebut, sehingga stok barang terorganisir dengan baik. Tentunya penggunaan inventori menjadi salah satu pengelolaan barang yang cukup penting bagi perusahaan demi mengorganisir barang atau stok dengan baik dan benar. Maka ketika barang diperlukan sudah tercatat jumlah dan posisi penyimpanan barang sehingga akan memudahkan serta mempercepat proses pemeliharaan.

### Metode SDLC *Waterfall*

Metode SDLC (*Software Development Life Cycle*) adalah pendekatan terstruktur untuk mengembangkan dan memelihara sistem perangkat lunak. Metode SDLC menguraikan tahap-tahap yang harus dilakukan dalam proses pengembangan perangkat lunak dari awal hingga selesai. Model yang digunakan pada penelitian ini yaitu model waterfall.

Model *waterfall* adalah metode pengembangan perangkat lunak yang mengikuti urutan linear dan berurutan dalam setiap tahap pengembangan perangkat lunak, dari analisis kebutuhan hingga implementasi dan pemeliharaan. Model ini mirip dengan air terjun, di mana setiap tahap harus selesai sebelum melanjutkan ke tahap berikutnya(Rosa & Shalahuddin, 2018).



Gambar 2.1 Model *waterfall*

Berikut tahapan pada metode SDLC *waterfall* :

1. Analisis

Tahap ini melibatkan pengumpulan informasi tentang kebutuhan bisnis dan kebutuhan pengguna untuk sistem perangkat lunak yang akan dikembangkan. Tim pengembang perangkat lunak akan mengumpulkan informasi dan membuat dokumen spesifikasi kebutuhan.

1. Desain

Tahap desain melibatkan membuat rancangan sistem perangkat lunak berdasarkan spesifikasi kebutuhan yang dikumpulkan pada tahap analisis kebutuhan. Rancangan ini mencakup arsitektur sistem, antarmuka pengguna, dan desain database.

1. Pengkodean

Tahap ini melibatkan pembuatan kode program berdasarkan rancangan yang telah dibuat pada tahap desain. Tim pengembang akan menggunakan bahasa pemrograman dan alat pengembangan perangkat lunak yang sesuai untuk membangun sistem perangkat lunak.

1. Pengujian

Tahap pengujian melibatkan menguji sistem perangkat lunak untuk memastikan bahwa sistem berfungsi dengan benar dan memenuhi spesifikasi kebutuhan. Pengujian mencakup pengujian fungsional, pengujian integrasi, dan pengujian keseluruhan sistem.

1. Pendukung atau pemeliharaan

Setelah sistem perangkat lunak selesai dikembangkan dan diuji, tahap terakhir adalah pemeliharaan. Tahap ini melibatkan perbaikan kesalahan, peningkatan kinerja, dan pembaruan sistem perangkat lunak untuk memenuhi kebutuhan pengguna yang berubah.

### *Unified Modelling Language*

UML (*Unified Modeling Language*) adalah bahasa pemodelan visual yang digunakan untuk memodelkan, merancang, dan memvisualisasikan sistem perangkat lunak yang kompleks. UML digunakan untuk menggambarkan berbagai aspek dari sistem, seperti fitur, struktur, dan perilaku dari suatu sistem. UML merupakan salah satu standar bahasa yang populer digunakan dalam lingkungan industri untuk menggambarkan arsitektur, melakukan analisis dan desain, serta mendefinisikan persyaratan dalam konteks pemrograman berorientasi objek (Mufida dkk., 2019). Dalam penelitian ini akan menggunakan beberapa diagram UML yaitu :

1. *Use case diagram*

*Use Case Diagram* adalah diagram yang digunakan untuk menggambarkan interaksi antara pengguna atau aktor dengan sistem yang akan dibangun. Diagram ini memberikan gambaran visual tentang fungsionalitas sistem dan bagaimana fungsionalitas tersebut terkait dengan pengguna atau aktor yang terlibat dalam interaksi tersebut. *Use Case Diagram* juga dapat membantu dalam memahami persyaratan bisnis dan kebutuhan pengguna untuk sistem yang akan dikembangkan. *Use Case Diagram* terdiri dari aktor, use case, dan hubungan antara aktor dan use case(Intern, 2021b).

Tabel 2.2 Simbol use case diagram

|  |  |
| --- | --- |
| Simbol | Keterangan |
|  | Aktor : mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan use case. |
|  | *Use case* : abstraksi dan interaksi antara sistem dan aktor |
| A picture containing letter  Description automatically generated | *Association* : abstraksi dari penghubung antara aktor dengan *use case* |
| A picture containing letter  Description automatically generated | Generalisasi : menunjukan spesialisasi aktor untuk dapat berpartisipasi dengan *use case* |
| <<include>>  A picture containing letter  Description automatically generated | *Include* : Menunjukan bahwa *use case* seluruhnya merupakan fungsionalitas dari *use case* lainnya |
| <<extend>>  A picture containing letter  Description automatically generated | *Extend* : Menunjukan bahwa suatu *use case* merupakan tambahan fungsional daru *use case* lainnya jika suatu kondisi terpenuhi |

1. *Activity diagram*

*Activity Diagram* adalah diagram yang digunakan untuk memodelkan aktivitas dalam suatu proses bisnis atau sistem. Diagram ini memberikan gambaran visual tentang urutan aktivitas, kondisi, dan tindakan yang terjadi dalam suatu proses bisnis atau sistem. *Activity Diagram* memungkinkan untuk memvisualisasikan aktivitas yang terjadi pada level yang lebih tinggi dari proses bisnis, sehingga memudahkan dalam memahami proses secara keseluruhan. Diagram ini terdiri dari aktivitas, tindakan, kondisi, dan hubungan antara aktivitas-aktivitas tersebut(Intern, 2021a).

Tabel 2.3 Simbol *activity diagram*

|  |  |
| --- | --- |
| Simbol | Keterangan |
| **Shape, arrow  Description automatically generated** | Status awal : sebuah diagram aktivitas memiliki sebuah status awal |
| Shape, arrow  Description automatically generated | Aktivitas : aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja |
| Shape, arrow  Description automatically generated | Percabangan/*decision* : percabangan dimana ada pilihan aktivitas yang lebih dari satu |
| Shape, arrow  Description automatically generated | Penggabungan/*join* : penggabungan dimana lebih dari satu aktivitas lalu digabungkan jadi satu |
| Shape, arrow  Description automatically generated | Status akhir : status akhir dilakukan sistem |
| Shape, arrow  Description automatically generated | *Swimlane* : memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi |

1. *Class diagram*

*Class Diagram* adalah diagram yang untuk memodelkan struktur sistem atau perangkat lunak yang akan dibangun. Diagram ini memberikan gambaran visual tentang kelas-kelas atau objek-objek yang ada dalam sistem, hubungan antara kelas-kelas atau objek-objek tersebut, serta atribut dan metode yang dimiliki oleh setiap kelas atau objek. *Class Diagram* memungkinkan untuk memvisualisasikan struktur sistem dengan lebih mudah dan terorganisir. Diagram ini terdiri dari kelas atau objek, atribut, metode, dan hubungan antara kelas atau objek tersebut, seperti hubungan *inheritance, association, aggregation*, dan lain sebagainya(Setiawan, 2021).

Tabel 2.4 Simbol class diagram

|  |  |
| --- | --- |
| Simbol | Keterangan |
| |  | | --- | | Nama\_kelas | | +atribut | | +operasi() | | Kelas : Kelas pada struktur sistem. |
|  | Antarmuka/*interface* : Sama dengan konsep *interface* dalam pemrograman |
|  | Asosiasi/*association* : Relasi antar class dengan arti umum, asosiasi biasanya juga disertai dengan *multiplicity* |
|  | Asosiasi berarah/*directed* *asscociation* : Relasi antar kelas dengan makna kelas yang yang lain, asosiasi berarah biasanya juga disertai dengan *multiplicity* |
|  | Generalisasi : Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus) |
|  | Kebergantungan/*dependency* : Relasi antar kelas dengan makna kebergantungan antar kelas |
|  | Agresi/*aggretion* : Relasi antar kelas dengan makna semua-bagian (*whole-part*) |

1. *Sequence Diagram*

*Sequence diagram* adalah suatu representasi visual yang digunakan untuk menggambarkan interaksi antara objek-objek di dalam sebuah sistem dengan rincian yang mendalam. Diagram ini menampilkan urutan pesan atau perintah yang dikirim antar objek, serta waktu pelaksanaannya. Biasanya, objek-objek yang terlibat dalam proses operasi diurutkan secara horizontal dari kiri ke kanan dalam diagram tersebut (Rosa & Shalahuddin, 2018).

Tabel 2. 5 Simbol *sequence diagram*

|  |  |
| --- | --- |
| Simbol | Keterangan |
|  | Aktor : user yang berinteraksi dengan sistem |
| |  | | --- | |  | | Objek : mengirim atau menerima pesan |
|  | Garis hidup objek : tanda kehiduoan objek selama urutan |
| |  | | --- | |  | | Objek sedang aktif berinteraksi : menandakan ketika objek mengirim atau menerima pesan |
| Pesan() | Pesan : objek mengirim pesan ke objek lain |
| Masukan | Masukan : suatu objek mengirimkan masukan ke objek lainnya |
| Keluaran | Keluaran : Objek menghasilkan suatu kembalian ke objek tertentu |

### PHP

PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman open-source yang umum digunakan untuk mengembangkan aplikasi web dinamis dan situs web. PHP memungkinkan pengembang untuk membuat kode yang dapat dijalankan di sisi server untuk menghasilkan halaman web dinamis yang menampilkan data dari database, menghasilkan konten dinamis, dan berinteraksi dengan pengguna melalui formulir dan tampilan web lainnya.

PHP berperan penting dalam pembangunan situs web, baik itu dalam bentuk statis maupun dinamis. Selain itu, PHP juga dapat digunakan untuk mengembangkan aplikasi berbasis web. Meskipun ada banyak bahasa pemrograman yang bisa digunakan untuk membangun sebuah website, PHP tetap menjadi pilihan yang populer, terutama bagi pemula karena mudah dipelajari. PHP dapat dijalankan di berbagai platform dan sistem operasi, serta berintegrasi dengan berbagai database, termasuk MySQL, *Oracle*, *PostgreSQL*, dan *Microsoft SQL Server*. Selain itu, PHP juga dapat digunakan dengan berbagai *framework* seperti *Laravel, CodeIgniter*, dan CakePHP(Sidik, 2017).

Berikut beberapa fungsi dari PHP:

Gratis: PHP adalah bahasa pemrograman open source yang gratis dan dapat diunduh dan digunakan oleh siapa saja.

Kompatibilitas: PHP dapat berjalan pada berbagai platform dan sistem operasi seperti *Windows*, Mac, dan *Linux*, dan juga dapat berintegrasi dengan berbagai server web seperti *Apache* dan *Nginx*.

Fleksibilitas: PHP sangat fleksibel dan dapat digunakan untuk mengembangkan berbagai jenis aplikasi web seperti blog, forum, toko online, dan lain-lain.

Mudah dipelajari: PHP memiliki sintaks yang mudah dipelajari dan dipahami, sehingga pemula dapat dengan cepat mempelajari dan mengembangkan aplikasi web.

Kinerja yang cepat: PHP memiliki kinerja yang cepat dan efisien, sehingga membuat situs web dan aplikasi web berjalan lebih cepat dan responsif.

Komunitas besar: PHP memiliki komunitas pengembang yang besar dan aktif, sehingga memungkinkan pengguna untuk memperoleh dukungan dan sumber daya yang diperlukan untuk mengembangkan aplikasi web.

### *Laravel*

*Laravel* adalah sebuah *framework* PHP yang memungkinkan pengembangan website menjadi lebih maksimal dan dinamis. *Framework* ini memberikan fitur-fitur terbaru yang lebih powerful dibandingkan dengan *framework* lainnya. Salah satu keunggulan *Laravel* adalah penggunaan struktur MVC (*Model View Controller*) yang memisahkan data dan tampilan dalam komponen aplikasi. Dengan menggunakan struktur MVC, pengguna *Laravel* dapat dengan mudah mempelajari cara kerja *framework* ini dan mempercepat proses pembuatan aplikasi berbasis website(Maksum, 2022b).

Ada pula beberapa fitur yang sudah disediakan Laravel. Berikut ini beberapa fitur Laravel:

1. *Authentication* : Fitur ini merupakan fitur yang dapat digunakan untuk membuat website dengan otentikasi. Dengan adanya fitur ini tidak perlu menulis kode dari awal untuk menambahkan fitur otentikasi di website.
2. Testing and *Debugging* : Dalam Laravel sudah disediakan fitur untuk testing dan debugging sebuah website. Sehingga tidak perlu khawatir lagi untuk melakukan testing dan debugging.
3. *Routing* : Dengan fitur ini dapat membuat aplikasi menjadi lebih mudah. Hal ini karena dalam *framework* ini, semua request akan dipetakan dengan menggunakan bantuan rute. Bisa memberikan nama routing, mengelompokan, *memfilter routing* yang ada sesuai kebutuhan aplikasi.
4. *Blade Template Engine*. *Framework* ini dilengkapi dengan *fitur blade* yang bisa digunakan untuk mendesain layout yang konsisten dengan ciri khas dari *laravel*.
5. *Composer* : Fitur ini akan membantu dalam menyempurnakan dan meningkatkan proses pengembangan website, serta mempermudah proses updatenya.

Berikut beberapa manfaat dari *framework Laravel* :

1. Mempercepat pengembangan: *Laravel* menyediakan fitur-fitur bawaan yang dapat mempercepat pengembangan aplikasi, seperti sistem routing, ORM, *template engine*, dan sistem autentikasi.
2. Kode yang lebih mudah dipelihara: Dengan struktur file yang jelas dan sintaks yang mudah dipahami, *Laravel* memudahkan pengembang dalam melakukan perbaikan dan perubahan pada kode aplikasi.
3. Keamanan yang lebih baik: *Laravel* menyediakan fitur-fitur keamanan yang kuat, seperti hashing password, proteksi terhadap serangan CSRF, dan autentikasi dua faktor.
4. Scalability yang baik: *Laravel* dirancang dengan arsitektur yang baik dan skalabel, sehingga memudahkan pengembangan aplikasi web dari skala kecil hingga besar.
5. Dukungan komunitas yang besar: *Laravel* memiliki komunitas pengembang yang besar dan aktif, sehingga memudahkan pengembang dalam mencari solusi atau dukungan dalam pengembangan aplikasi.

### *Bootstrap*

*Bootstrap* adalah kerangka kerja CSS yang digunakan untuk membuat desain website secara virtual saat proses pembuatan. Kerangka kerja ini bersifat open source dan biasanya digunakan dalam pemrograman berbasis HTML dan CSS. Bootstrap juga tersedia dalam bentuk template. Selain itu, *Bootstrap* juga berguna dalam pengembangan banyak fitur yang dibutuhkan dalam aplikasi dan situs web, seperti tipografi, tombol, navigasi, dan komponen lain yang relevan. Bahkan beberapa element *bootstrap* sudah dikombinasi dengan JavaScript(Kaban, 2019).

HTML (Hypertext Markup Language) adalah bahasa markup yang digunakan untuk menentukan elemen-elemen seperti teks, gambar, audio, video, dan elemen media lainnya pada halaman web(Sidiq & Pohan, 2014). CSS digunakan untuk mengatur tampilan dan gaya visual dari elemen yang ada pada halaman web, termasuk pengaturan teks dan layout. Dengan menggunakan CSS, pengguna tidak perlu menulis ulang kode untuk tampilan yang sama. JavaScript digunakan untuk menciptakan keinteraktifan halaman web dan memungkinkan pengguna untuk menangani event dan interaksi pengguna, membuat animasi, efek visual, menghubungkan halaman web dengan database serta sumber daya server lainnya(Darmawan & Hendra Permana, 2016).

Dengan menggunakan *bootstrap* tentunya programmer akan terbantu karena fungsi CSS yang biasa digunakan di HTML sudah siap sedia digunakan tanpa perlu membuat ulang dari awal. Fitur-fitur yang sering digunakan seperti tombol, form, card dan lain sebagainya sudah siap digunakan. Bahkan penggunaannya pun sangat mudah sekali tanpa perlu mendownload, hanya tinggal memanggil atau menggunakan link *bootstrap* secara online otomatis *framework* *bootstrap* terpanggil dan siap digunakan.

Berikut beberapa manfaat dari *bootstrap* :

1. *Responsif: Bootstrap* dirancang untuk mendukung tampilan yang *responsif* di berbagai perangkat, seperti desktop, tablet, dan *smartphone*, sehingga website akan terlihat baik di semua perangkat.
2. Konsistensi: *Bootstrap* menyediakan serangkaian komponen UI yang konsisten dan mudah digunakan, seperti tombol, kotak dialog, dan menu navigasi, sehingga memudahkan pengembangan website yang konsisten.
3. Mudah dikustomisasi: *Bootstrap* menyediakan banyak opsi pengaturan dan tema yang dapat disesuaikan, sehingga mudah untuk menyesuaikan tampilan website sesuai kebutuhan.
4. Dokumentasi lengkap: *Bootstrap* dilengkapi dengan dokumentasi yang lengkap dan mudah dipahami, sehingga memudahkan pengembang dalam mempelajari dan menggunakannya.
5. Kompatibilitas lintas browser: *Bootstrap* didukung oleh sebagian besar browser modern, sehingga *website* yang dibangun dengan *Bootstrap* dapat diakses dengan baik di berbagai browser.

### XAMPP

XAMPP adalah singkatan dari "*Cross-Platform* (X), *Apache* (A), MariaDB (M), PHP (P), and Perl (P)". XAMPP adalah aplikasi web server yang bersifat *open-source* yang digunakan untuk mengembangkan dan menjalankan aplikasi web di lingkungan lokal atau pada mesin pengembangan. XAMPP menyediakan paket lengkap yang berisi server web Apache, basis data MariaDB (sebelumnya MySQL), bahasa pemrograman PHP, dan Perl, serta komponen lain seperti phpMyAdmin, OpenSSL, dan FileZilla FTP server. XAMPP dapat diinstall pada sistem operasi *Windows, Linux*, dan macOS. XAMPP merupakan solusi yang mudah digunakan bagi pengembang web untuk melakukan pengembangan dan pengujian aplikasi web secara lokal sebelum di-deploy ke server web yang sebenarnya(Maksum, 2022a).

Pada XAMPP ini tentunya sudah terpasang dengan basis data yaitu MySQL. MySQL adalah sebuah sistem manajemen database sumber terbuka (*open-source*) yang menggunakan bahasa pemrograman *structured query language* (SQL) sebagai perintah dasarnya. MySQL sangat populer dalam dunia teknologi dan digunakan sebagai database. Dalam konteks pemrograman, SQL adalah bahasa yang digunakan untuk mengambil data dari database relasional atau database yang terstruktur. Oleh karena itu, MySQL dapat dianggap sebagai sistem manajemen database yang menggunakan bahasa SQL sebagai antarmuka antara perangkat lunak aplikasi dan server database(K, 2022).

XAMPP tersusun dari singkatan dari program-program yang ada di dalamnya, antara lain:

1. X (*Cross platform*) merupakan kode penanda dari software cross platform. Artinya, XAMPP dapat dijalankan diberbagai sistem operasi yang umum digunakan. Seperti *Windows, Linux,* serta *Mac OS*.
2. A (*Apache*) merupakan aplikasi web server yang dapat digunakan secara gratis (bersifat *open source*). Dengan menggunakan web server, seseorang dapat menjalankan file yang berisi kode (bahasa pemrograman PHP) di localhost.
3. M (MySQL / MariaDB) merupakan salah satu aplikasi database server yang menggunakan bahasa pemrograman SQL (*Structured Query Language*). Fungsinya adalah untuk mengelola data secara terstruktur dan sistematis. MySQL bisa digunakan di localhost tanpa memerlukan koneksi internet, sehingga developer dan programmer dapat membuat aplikasi berbasis website di komputernya.
4. P (PHP) merupakan salah satu bahasa pemrograman yang dijalankan pada sisi server yang dapat digunakan untuk mengelola konten dinamis dan database. Dengan menggunakan PHP, website menjadi lebih dinamis.
5. P (*Perl*) merupakan salah satu bahasa pemrograman tingkat tinggi yang bisa difungsikan untuk segala kebutuhan (*cross platform*). Saat ini perl banyak digunakan untuk keperluan pengembangan aplikasi hingga web server.

### *Visual Studio Code*

*Visual Studio Code* adalah perangkat lunak penyunting kode lintas platform yang ringan tetapi memiliki kemampuan yang kuat. Perangkat lunak ini dapat digunakan oleh siapa saja untuk membangun aplikasi web. Perangkat lunak ini memiliki dukungan bawaan untuk *JavaScript*, skrip, dan Node.js dan memiliki berbagai macam ekstensi yang tersedia untuk bahasa pemrograman lain, seperti C++, C#, Python, dan PHP. *Visual Studio Code* menggunakan *open source* .NET untuk memberikan dukungan ASP. Perangkat lunak ini memiliki antarmuka pengguna yang mudah digunakan karena didasarkan pada gaya penjelajah umum, dengan panel di sebelah kiri menampilkan semua file dan folder serta panel di sebelah kanan menunjukkan isi dari file yang sedang dibuka. Visual Studio Code memiliki fitur yang baik, termasuk *intellisense* dan *autocomplete* yang bekerja dengan baik untuk JSON, CSS, HTML, Node.js, dan lainnya (Tasari, 2021).

Manfaat *visual studio code* yaitu :

Gratis dan *Open Source: Visual Studio Code* adalah editor kode sumber gratis dan *open source* yang dapat digunakan oleh siapa saja.

Lingkungan pengembangan yang kuat: *Visual Studio Code* menyediakan lingkungan pengembangan yang kuat dan lengkap, termasuk *fitur debugging*, pengelolaan paket, integrasi Git, dan banyak lagi.

*Multi-platform: Visual Studio Code* tersedia untuk *Windows*, macOS, dan *Linux*, sehingga dapat digunakan pada berbagai sistem operasi.

Kemampuan pengayaan: *Visual Studio Code* memiliki banyak ekstensi dan pengayaan yang dapat memperluas fungsionalitas editor dan membantu pengguna untuk meningkatkan produktivitas.

Mudah digunakan: *Visual Studio Code* dirancang untuk mudah digunakan, dengan antarmuka yang intuitif dan sederhana, sehingga dapat digunakan oleh pengguna dengan berbagai tingkat pengalaman.

Dukungan bahasa pemrograman: *Visual Studio Code* mendukung berbagai bahasa pemrograman, termasuk *JavaScript*, Python, C++, dan banyak lagi.

### *Google Chrome*

*Google Chrome* adalah browser web yang dikembangkan oleh *Google*, dapat digunakan di berbagai *platform*. Diluncurkan pada 2 September 2008, awalnya hanya tersedia untuk sistem operasi *Microsoft Windows*, tetapi kemudian di porting ke Linux, macOS, iOS, dan Android. *Google Chrome* terkenal karena kecepatannya dalam mencari informasi, dan menjadi pilihan populer bagi pengguna dari berbagai latar belakang dan profesi dengan berbagai fitur yang siap membantu pengguna mencari informasi secara efektif.

Bagi pengembang web, *Google Chrome* sangat membantu karena mendukung aplikasi web sehingga digunakan sebagai flatform yang akan digunakan untuk menjalankan aplikasi program yang sudah dibuat dengan exstensi software yang terupdate. *Interface*-nya berbasis pada gaya explorer, dan dilengkapi dengan fungsi yang baik seperti *intellisense* dan autocomplete, terutama untuk JSON, CSS, HTML, Node.js, dan bahasa pemrograman lainnya(Bintara, 2020b).

Manfaat *google chrome* yaitu :

1. Kecepatan: *Google Chrome* diketahui sebagai browser web yang sangat cepat dalam melakukan rendering halaman web, memuat konten, dan membuka tab baru.
2. Keamanan: *Google Chrome* dilengkapi dengan fitur keamanan yang kuat seperti Sandbox dan fitur pembaruan otomatis, sehingga pengguna dapat merasa lebih aman saat menjelajahi internet.
3. Kompatibilitas: *Google Chrome* memiliki kemampuan yang baik dalam memproses situs web modern yang menggunakan teknologi terbaru seperti HTML5, CSS3, dan *JavaScript*.
4. Integrasi *Google: Google Chrome* terintegrasi dengan berbagai layanan Google seperti *Google Drive, Gmail,* dan *Google Translate*, sehingga pengguna dapat dengan mudah mengakses layanan Google ini dari browser mereka.
5. Kustomisasi: *Google Chrome* memungkinkan pengguna untuk mengkustomisasi browser mereka dengan menambahkan tema, ekstensi, dan pengaya untuk meningkatkan produktivitas dan efisiensi.
6. Sinkronisasi: *Google Chrome* dapat disinkronkan dengan akun Google pengguna, sehingga pengguna dapat dengan mudah mengakses riwayat *browsing*, *bookmark*, dan preferensi penggunaan lainnya dari berbagai perangkat.
7. Penghematan data: *Google Chrome* dilengkapi dengan fitur penghematan data yang dapat menghemat penggunaan data saat browsing, sehingga dapat membantu pengguna menghemat biaya data internet mereka.

### *Microsoft Visio*

*Microsoft Visio*, atau dikenal sebagai *Microsoft Office Visio*, adalah sebuah perangkat lunak komputer yang umumnya digunakan untuk membuat diagram alir, skema jaringan, brainstorming, dan diagram lainnya. Sebagai bagian dari paket *Microsoft Office*, bersama dengan *Word, Excel*, dan *PowerPoint*, *Visio* menggunakan grafik vektor untuk membuat diagram. Software ini memiliki tampilan antarmuka yang mudah digunakan dan fitur lengkap dalam merancang sistem. Bahkan bagi pemula, penggunaannya sangat mudah, cukup dengan men-drag and drop bentuk diagram yang dibutuhkan sesuai kebutuhan.

Dengan menggunakan *software* ini, pengguna dapat meningkatkan kinerjanya dengan mempersiapkan berbagai jenis diagram seperti DFD, ERD, UML, jaringan, rancangan antarmuka pengguna, dan lain sebagainya. Selain itu, tersedia banyak template yang dapat membantu pengguna dalam membuat diagram dengan mudah, intuitif, dan profesional(Bintara, 2020a).

Manfaat *Microsoft Visio* yaitu :

1. Kemudahan penggunaan: *Microsoft Visio* memiliki antarmuka pengguna yang mudah digunakan dan intuitif, sehingga pengguna dapat dengan cepat dan mudah membuat diagram dan bagan.
2. Fleksibilitas: *Microsoft Visio* memiliki banyak template diagram yang berbeda dan dapat dikustomisasi, sehingga pengguna dapat membuat diagram yang sesuai dengan kebutuhan mereka.
3. Kolaborasi: *Microsoft Visio* memungkinkan pengguna untuk berkolaborasi dalam membuat diagram dan bagan dengan pengguna lain, sehingga memudahkan pekerjaan tim dalam menyelesaikan proyek.
4. Visualisasi: *Microsoft Visio* membantu pengguna dalam memvisualisasikan data dan informasi yang kompleks dalam bentuk diagram dan bagan, sehingga memudahkan pemahaman dan pengambilan keputusan.
5. Integrasi: *Microsoft Visio* dapat diintegrasikan dengan aplikasi *Microsoft Office* lainnya seperti *Excel, PowerPoint,* dan *Word*, sehingga memudahkan pengguna dalam mengimpor dan mengekspor data dan informasi.
6. Kompatibilitas: *Microsoft Visio* dapat membaca dan menulis berbagai format file seperti XML, PDF, dan DWG, sehingga memungkinkan pengguna untuk berbagi diagram dan bagan dengan pengguna lain dari platform dan aplikasi yang berbeda.

### *Balsamiq* *Mockup*

Menurut Bubble, *Balsamiq* adalah sebuah alat yang difokuskan pada struktur dan konten wireframe. Software ini dianggap sangat cocok bagi pemula karena tidak memerlukan kemampuan coding untuk mengoperasikannya. Penggunaannya sangatlah mudah, hanya dengan melakukan drag and drop pada elemen desain yang diperlukan. Dengan begitu, proses desain dapat menjadi lebih cepat. Fitur ini dapat membantu desainer dalam memahami usability desain yang dibuat, sehingga revisi dapat dikurangi pada tahapan selanjutnya. *Balsamiq* dapat digunakan oleh semua kalangan, bahkan tim perancang situs web atau aplikasi dapat bekerja sama dalam penggunaan alat ini(Rahmalia, 2020).

Berikut beberapa manfaat software *Balsamiq* *Mockup* :

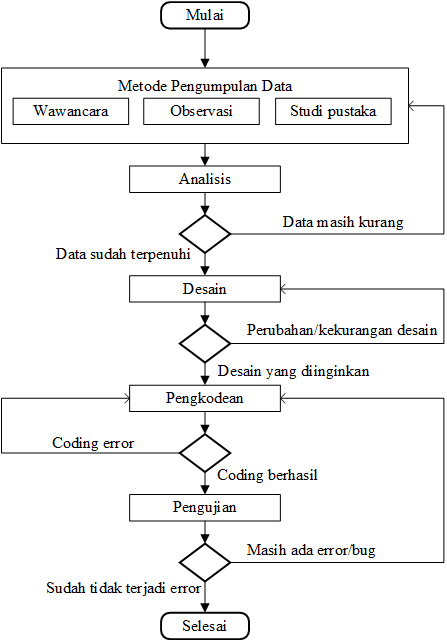
1. Mempercepat Proses Desain: Dengan *Balsamiq* *Mockup*, desainer atau pengembang dapat dengan cepat membuat prototipe tampilan antarmuka pengguna yang dapat diuji coba dan direvisi dalam waktu yang relatif singkat.
2. Meningkatkan Kualitas Desain: Dengan menggunakan *Balsamiq* *Mockup*, desainer dapat dengan mudah mengevaluasi desain dan menyempurnakannya sebelum memulai pengembangan aplikasi atau situs web.
3. Mudah Digunakan: *Balsamiq* *Mockup* memiliki antarmuka yang mudah dipahami dan digunakan oleh desainer atau pengembang, bahkan untuk mereka yang tidak memiliki pengalaman dalam membuat *Mockup* atau prototipe tampilan antarmuka pengguna.
4. Kolaborasi yang Lebih Baik: *Balsamiq* *Mockup* memungkinkan beberapa pengguna untuk bekerja bersama dalam membuat dan merevisi *Mockup* atau prototipe tampilan antarmuka pengguna.
5. Hemat Biaya: Dengan menggunakan *Balsamiq* *Mockup*, perusahaan atau pengembang dapat menghemat biaya dalam proses desain dan pengembangan aplikasi atau situs web, karena dapat mempercepat waktu pengembangan dan meningkatkan kualitas desain sebelum memulai pengembangan yang lebih lanjut.

# METODOLOGI PENELITIAN



## Kerangka Pikir

Dalam kerangka pikir ini, terdapat urutan langkah-langkah penelitian yang dimulai dari metode pengumpulan data, analisis data, perancangan, implementasi, dan pengujian. Deskripsi selanjutnya akan menjelaskan tahapan-tahapan dalam kerangka pikir tersebut secara terperinci dari awal hingga akhir.



Gambar 3.1 Kerangka Pikir

## Deskripsi

Deskripsi adalah sebuah aturan yang berkaitan dengan usaha pengolahan data menjadi sebuah informasi yang dapat dijelaskan dan diungkapkan dengan jelas dan tepat guna untuk mencapai tujuan tertentu, sehingga nantinya dapat dipahami dan dimengerti oleh pembaca.

### Metode Pengumpulan Data

Wawancara

Pada tahap ini, dilakukan wawancara secara tatap muka dengan proses tanya-jawab antara peneliti dan pihak yang terlibat mengenai stok barang seperti kepada karyawan bagian marketing yang selalu melakukan pengelolaan stok kain perusahaan.

Observasi

Pada tahap observasi, dilakukan pengamatan langsung pada lokasi penelitian yaitu di PT. Cemara Kwangjin Tekstil. Dalam tahap ini, data-data yang relevan untuk memenuhi kebutuhan dalam penelitian akan dikumpulkan terutama sesuai dengan objek penelitian.

Studi pustaka

Penulis melakukan studi pustaka dengan cara mencari sumber-sumber yang terkait dengan penelitian, baik itu dari buku, jurnal, maupun internet.

Studi dokumentasi

Pengumpulan, analisis, dan interpretasi berbagai dokumen atau sumber informasi tertulis yang sesuai dengan penelitian di PT. Cemara Kwangjin Tekstil.

### Analisis

Setelah berhasil mengumpulkan data yang diperlukan, langkah selanjutnya adalah melakukan analisis terhadap data tersebut. Tujuan dari tahap analisis data ini adalah untuk mengidentifikasi dan mengevaluasi permasalahan, kesempatan, hambatan, serta kebutuhan yang terkait dengan penelitian. Dengan melakukan analisis data yang baik, diharapkan penelitian dapat berjalan lancar dan efektif.

Adapun analisis kebutuhan yang akan digunakan dalam penelitian ini yaitu :

1. Analisis kebutuhan fungsional sistem
2. Mengelola data kain

Menginput data kain yang disediakan oleh perusahaan meliputi jenis kain dan stok kain yang disediakan. Sehingga untuk ouputnya terdata jenis kain yang disediakan.

1. Mengelola data kolega

Menginput data kolega yang menyediakan sumber kain mentah sehingga bekerjasama dengan perusahaan. Ouputnya terdata mengenai kolega yang saat ini sedang menjalin kerjasama sebagai penyedia kain mentah.

1. Mengelola data penyimpanan

Menginput data penyimpanan sebagai tempat menyimpannya stok kain yang masih tersedia. Sehingga untuk ouputnya terdata lokasi penyimpanan kain.

1. Mengelola data pembelian

Menginput data transaksi pembelian kain mentah yang akan di olah oleh perusahaan dari kolega. Sehingga untuk ouputnya terdata pembelian kain dari kolega dan menjadi laporan pembelian.

1. Mengelola data penjualan

Menginput data transaksi penjualan kain yang dijual oleh perusahaan kepada pembeli. Sehingga untuk ouputnya terdata penjualan kain kepada pembeli dan menjadi laporan penjualan.

1. Kebutuhan perangkat keras

Tabel 3.1 Spesifikasi perangkat keras

|  |  |
| --- | --- |
| Prosessor | AMD A9-9420 RADEON R5, 5 COMPUTE CORES 2C+3G (2CPUs),~3.0GHz |
| RAM | 4GB RAM |
| HDD | 1 TB |

1. Kebutuhan perangkat lunak
2. *Laravel*, digunakan sebagai kerangka siap pakai *framework back-end* pembuatan aplikasi dengan bahasa pemrograman PHP.
3. *Bootstrap*, digunakan sebagai *framework front-end* tampilan aplikasi yang berisi elemen atau template siap pakai.
4. Database MySQL, digunakan untuk menyimpan basis data.
5. XAMPP, digunakan sebagai server.
6. *Visual Studio Code*, digunakan sebagai aplikasi kode editor
7. *Balsamiq* *Mockup*s, digunakan sebagai tool perancangan desain tampilan.
8. *Microsoft Visio*, digunakan sebagai aplikasi perancangan UML.

### Desain

Desain adalah tahap lanjutan setelah melakukan analisis data. Hal ini diperlukan untuk mempersiapkan rancang bangun yang akan diimplementasikan dengan menggambarkan bagaimana suatu sistem akan dibentuk, seperti perencanaan, pembuatan sketsa, atau pengaturan beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh. Dalam tahap desain, penggunaan UML (*Unified Modeling Language*) dengan bantuan *software* *Microsoft Visio* dan *Balsamiq* *Mockup* akan membantu dalam merancang desain tampilan user interface. Beberapa jenis UML yang digunakan meliputi use case diagram, activity diagram, dan class diagram.

1. Desain model

Desain model menggunakan UML untuk dibuat dengan beberapa jenis diagram UML yaitu :

1. *Use Case Diagram*

Digunakan untuk mendeskripsikan hubungan *actor* dan *use case* yang terlibat dalam penelitian ini yaitu :

1. *Actor* : Admin, Marketing dan Manajer.
2. *Use case* : Login, mengelola data kain, pengelola data kolega, mengelola data penyimpanan, mengelola data pembelian, mengelola data penjualan dan mengelola data pengguna.
3. *Activity Diagram*

Digunakan untuk memodelkan alur kerja sistem dan aktivitas dari aktor. Adapun diagramnya yang akan dibuat yaitu :

1. Proses login
2. Proses menampilkan data kain, tambah data, edit data, hapus data.
3. Proses menampilkan data kolega, tambah data, edit data, hapus data.
4. Proses menampilkan data penyimpanan, tambah data, edit data, dan hapus data.
5. Proses menampilkan data pembelian, tambah data, edit data, hapus data dan cetak data.
6. Proses menampilkan data penjualan, tambah data, edit data, hapus data dan cetak data.
7. Proses menampilkan data pengguna, tambah data, edit data, hapus data.
8. *Class Diagram*

Dengan alur database yang saling terhubung pada sistem yang akan dibangun yaitu :

1. Mengelola data login,
2. Mengelola data kain,
3. Mengelola data kolega,
4. Mengelola data penyimpanan,
5. Mengelola data pembelian,
6. Mengelola data penjualan,
7. Mengelola data pengguna,
8. *Sequence diagram*

*Sequence diagram* pada aplikasi sistem informasi inventori yaitu :

1. Mengelola data login,
2. Mengelola data kain,
3. Mengelola data kolega,
4. Mengelola data penyimpanan,
5. Mengelola data pembelian,
6. Mengelola data penjualan,
7. Mengelola data pengguna,
8. Desain tampilan

Perancangan tampilan merupakan bagian desain tampilan pada aplikasi sistem informasi inventori dengan menggunakan *mockup*. Perancangan tampilan *mockup* yaitu :

1. *Mockup* halaman login,
2. *Mockup* halaman dashboard,
3. *Mockup* halaman kain,
4. *Mockup* halaman kolega,
5. *Mockup* halaman penyimpanan,
6. *Mockup* halaman pembelian,
7. *Mockup* halaman penjualan,
8. *Mockup* halaman pengguna,

### Pengkodean

Dalam tahapan ini, dilakukan implementasi dari pengkodean yang telah dibuat ke dalam bentuk kode pemrograman. Pengkodean tersebut berfungsi untuk menciptakan fungsi-fungsi yang dibutuhkan oleh program dalam merancang bangun aplikasi inventori. Pada tahap implementasi atau pengkodean menggunakan *framework* *Laravel* dengan bahasa pemrograman PHP dan *Bootstrap* yang telah mencakup pemrograman HTML, CSS, *JavaScript*, dan PHP. Tahap ini dilakukan sesuai dengan hasil perancangan yang telah dibuat sebelumnya.

### Pengujian

Pada tahapan akhir, dilakukan pengujian dan evaluasi terhadap aplikasi inventori yang telah dirancang dan diimplementasikan. Tahap ini dilakukan setelah aplikasi selesai dibuat dan memenuhi kebutuhan sesuai dengan data, analisis, dan perancangan. Pengujian ini menggunakan metode *blackbox* yaitu hanya menguji keluaran atau output sesuai dengan yang diharapkan. Selain itu, tahapan ini dilakukan untuk menemukan kekurangan pada aplikasi yang telah dikembangkan, dan jika ada kekurangan, dilakukan perbaikan pada aplikasi tersebut.

# ANALISIS DAN PERANCANGAN



## Analisis

### Analisis Tata Kelola PT. Cemara Kwangjin Tekstil

PT. Cemara Kwangjin Tekstil merupakan sebuah perusahaan yang bergerak di bidang industri manufaktur dalam pembuatan produk bahan tekstil (*Textile* Industri) seperti Spinning (Pembuatan Bahan Kapas Menjadi Benang), *Weaving*, (Pembuatan Bahan Benang Menjadi Kain), *Dyeing* (Pembatan dan Pewarnaan kain jadi sampai *finishing*), dan sebagainya.

PT. Cemara Abadi Tekstil didirikan pada bulan September 1978 di kawasan industri Leuwi Gajah, Cimahi. Memulai konsentrasi bisnis pada pembuatan komponen bahan kain mentah yang kemudian berkembang dalam pembuatan kain jadi. Kemudian pada Bulan Desember tahun 1997 PT. Cemara Kwangjin Tekstil menambah plant baru beralamat di jalan Kawasan Industri Cimareme, Cimahi - Bandung, Jawa Barat. Main Customer PT. Cemara Kwangjin Tekstil meliputi ekspor ke negara timur tengah, benua Australia, negara Malaysia, Singapura dan sebagainya.

A diagram of a company

Description automatically generated with low confidence

Gambar 4.1 Struktur organisasi

### Analisis Sistem

Analisis sistem diperlukan untuk mengatasi masalah yang terjadi di PT. Cemara Kwangjin Tekstil. Tujuan dari analisis ini adalah untuk menganalisis sistem yang sedang digunakan saat ini terutama pada bagian inventori kain yang digunakan sebagai bahan baku perusahaan. Selain itu, perancangan desain sistem bertujuan untuk memberikan pandangan yang terperinci mengenai proses perancangan sistem dari awal hingga akhir penelitian. Pengguna sistem meliputi admin dan marketing yang nantinya akan mengakses sistem yang dibuat.

### Analisis Masalah

Berdasarkan hasil wawancara dan observasi di PT. Cemara Kwangjin Tekstil, ditemukan bahwa sistem inventori kain yang digunakan masih semi manual menggunakan Microsoft Excel dan pencatatan sistem berkas. Kendala-kendala yang muncul akibat penggunaan sistem tersebut yaitu rawannya perbedaan data dari transaksi yang sebenarnya dengan inputan data pada *software Microsoft Excel* sehingga cara untuk mengatasinya di sinkronisasi ulang antara transaksi yang sebenarnya dengan data yang sudah dan input, tentunya hal tersebut kurang praktis dan membutuhkan waktu yang cukup lama. Selain itu, kurang jelasnya informasi mengenai stok kain yang masih tersedia dengan berbagai macam jenis dan lokasi penyimpanan kain yang disimpan. Untuk menganalisis masalah tersebut, akan digunakan metode PIECES.

Tabel 4.1 Analisis masalah

|  |  |  |
| --- | --- | --- |
| **PIECES** | **Hasil analisis terhadap sistem lama** | **Hasil yang diharapkan pada sistem baru** |
| *Performance* (Kinerja) | Pendataan stok kain masih manual dengan *Ms. Excel* dan pencatatan sistem berkas. Dalam mencegah kesalahan data di *Excel* dan berkas maka dilakukan sinkronisasi yang membutuhkan waktu dan ketelitian | Pendataan kain menjadi lebih mudah dan praktis serta ketepatan yang akurat |
| *Information* (informasi) | Informasi mengenai stok kain harus dilihat ke gudang penyimpanan.  Informasi mengenai transaksi harus dilihat ulang diberkas yang dicatat secara manual atau hasil dari *excel* jika kedua data tersebut sinkron | Memudahkan pengguna ketika ingin mengetahui stok kain yang tersedia dan laporan hasil transaksi.  Meminimalisir biaya pembelian kertas untuk berkas pencatatan transaksi |
| *Control* (Kontrol) | Petugas memiliki kendala dalam penyimpanan data transaksi yang nantinya akan menjadi laporan. Terutama berkas pendataan transaksi yang harus disinkronisasi ulang untuk menghindari kesalahan pendataan antara di *Ms. Excel* dan berkas sehingga itu memerlukan waktu banyak | Memudahkan pengguna dalam mencari data transaksi yang akan menjadi laporan yang bisa dibuat secara otomatis dengan aplikasinya berdasarkan data yang diinputkan. |
| *Efficiency* (Efisiensi) | Sinkronisasi data memerlukan waktu dan ketelitian oleh petugas terutama mengenai stok kain yang masih tersedia yang berubah ketika transaksi berlangsung | Meminimalisir waktu dan tenaga petugas dalam pendataan transaksi terutama laporan dan juga keakuratan stok kain yang masih tersedia berdasarkan transaksi yang telah berlangsung. |
| *Service* (Pelayanan) | Dari segi pelayanan masih belum optimal karena informasi mengenai stok kain masih manual dalam mengetahui stok kain yang masih tersimpan terutama mengetahui tempat penyimpanannya secara lebih rinci. | Pelayanan lebih terorganisir terutama mengenai stok yang masih tersisa sehingga informasi tersebut dengan mudah bisa didapatkan |

### Analisis Kebutuhan Sistem

Analisis kebutuhan sistem digunakan untuk mempermudah menganalisis sebuah sistem yang dibutuhkan, berikut kebutuhan sistem:

1. Sistem dapat mengelola data kain yaitu lihat, tambah, edit dan hapus.
2. Sistem dapat mengelola data tempat penyimpanan yaitu lihat, tambah, edit dan hapus.
3. Sistem dapat mengelola data kolega yaitu lihat, tambah, edit dan hapus.
4. Sistem dapat mengelola data penjualan kain yaitu lihat, tambah, edit, hapus dan cetak.
5. Sistem dapat mengelola data tempat pembelian kain yaitu lihat, tambah, edit, hapus dan cetak.
6. Sistem dapat mengelola data pengguna yaitu lihat, tambah, edit dan hapus.

### Analisis Kebutuhan

Analisis kebutuhan bertujuan untuk menentukan perangkat yang diperlukan dalam merancang bangun sistem informasi inventori. Dengan melakukan analisis kebutuhan, diharapkan sistem yang akan dirancang bangun dapat diuraikan menjadi komponen-komponen dasar yang memenuhi kebutuhan objek penelitian. Analisis kebutuhan di bagi dua yaitu :

Analisis kebutuhan fungsional

Adapun beberapa kebutuhan fungsional pada aplikasi sistem informasi inventori ini yaitu :

1. Form login.
2. Form input data kain, kolega, penjualan, pembelian, tempat penyimpanan dan pengguna.
3. Laporan penjualan kain dan laporan pembelian kain.
4. Surat jalan pengantaran barang.

Analisis kebutuhan non-fungsional

Adapun kebutuhan non-fungsional pada aplikasi sistem informasi inventori ini dibagi menjadi dua yaitu :

1. Kebutuhan perangkat keras
2. Laptop : Acer
3. *Processor* : AMD A9-9420 RADEON R5,
4. RAM : 4 Gb DDR4
5. *Hardisk* : 1 Tb HDD
6. Kebutuhan perangkat lunak
7. XAMPP
8. *Laravel* 8
9. *Bootstrap* 5
10. *Visual Studio Code*
11. *Balsamiq Mockup*

### Analisis Penunjang Keputusan

Dilihat dari kebutuhan sistem informasi dalam pengambilan keputusan bagi owner PT. Cemara Kwangjin Tekstil adalah sebagai berikut:

Sistem informasi inventori ini dapat membantu dalam menyampaikan laporan transaksi penjualan kain kepada konsumen dan transaksi pembelian kain dari kolega dengan akurat dan praktis. Sehingga dengan laporan transaksi ini memudahkan dalam mengambil keputusan kedepannya mengenai strategi perusahaan dalam memproduksi kain.

Sistem informasi ini membantu bagian marketing dalam mengecek stok kain yang tersedia dan jumlah sisa kain yang sudah menipis serta memudahkan untuk menemukan kain yang disimpan pada palet yang sesuai dengan mudah tanpa perlu dicari secara langsung digudang penyimpanan.

### Analisis Biaya

Pada penelitian sistem informasi inventori tentunya diperlukan alokasi tenaga dan sumber daya finansial sebagai penunjang penulis dalam menyelesaikan penelitian. Berikut adalah rincian biaya yang harus dikeluarkan untuk merancang bangun inventori ini :

Tabel 4.2 Analisis biaya

|  |  |
| --- | --- |
| Jenis kebutuhan | Biaya |
| Komputer | Rp. 4.000.000 |
| Jaringan internet | Rp. 500.000 |
| ATK | Rp. 1.000.000 |
| Programer | Rp. 1.000.000 |
| Transportasi | Rp. 500.000 |
| Total | Rp. 7.000.000 |

## Perancangan

Perancangan sistem informasi inventory yang akan dibangun ini menggunakan UML(*Unified Modeling Language*) sebagai model dalam merancang bangun alur sistem dan merancang desain tampilan aplikasi. berikut perancangan yang akan dibuat menjadi aplikasi.

### Pemodelan UML (*Unified Modeling Language*)

Diagram UML yang akan digunakan dalam perancangan inventory ini ada tiga macam yaitu *use case diagram, activity diagram, class diagram dan sequence diagram*.

1. *Use case diagram*

Berikut deskripsi mengenai perancangan *use case diagram* pada rancang bangun sistem informasi inventori yaitu :

A diagram of people connected to each other

Description automatically generated

Gambar 4.2 *Use case diagram*

Berikut deskripsi mengenai *use case diagram* dengan tiga aktor yaitu Admin, Manajer dan Marketing :

Tabel 4.3 Deskripsi aktor aplikasi

|  |  |
| --- | --- |
| Aktor | Deskripsi |
| Admin | Pengguna yang bisa mengakses semua menu pada aplikasi inventori |
| Marketing | Pengguna yang hanya mengakses bagian yang berhubungan dengan marketing |
| Manajer | Pengguna yang hanya bisa mengakses untuk memantau perusahaan |

Tabel 4.4 Deskripsi *use case* Admin

|  |  |
| --- | --- |
| *Use case* | Deskripsi |
| Login | Melakukan login untuk mengakses aplikasi |
| Mengelola data kain | Mengelola data kain yang bisa melakukan lihat, tambah, edit dan hapus data |
| Mengelola data penyimpanan | Mengelola data penyimpanan yang bisa melakukan lihat, tambah, edit dan hapus data |
| Mengelola data kolega | Mengelola data kolega yang bisa melakukan lihat, tambah, edit dan hapus data |
| Mengelola data pembelian | Mengelola data pembelian yang bisa melakukan lihat, tambah, edit, hapus dan cetak data |
| Mengelola data penjualan | Mengelola data penjualan yang bisa melakukan lihat, tambah, edit, hapus dan cetak data |
| Mengelola data pengguna | Mengelola data kain yang bisa melakukan lihat, tambah, edit dan hapus data |

Tabel 4.5 Deskripsi *use case* Manajer

|  |  |
| --- | --- |
| Use case | Deskripsi |
| Login | Melakukan login untuk mengakses aplikasi |
| Mengelola data kain | Mengelola data kain yang hanya bisa melakukan lihat data |
| Mengelola data kolega | Mengelola data kolega yang hanya bisa melakukan lihat data |
| Mengelola data pembelian | Mengelola data pembelian yang hanya bisa melakukan lihat dan cetak data |
| Mengelola data penjualan | Mengelola data penjualan yang hanya bisa melakukan lihat dan cetak data |

Tabel 4.6 Deskripsi *use case* Marketing

|  |  |
| --- | --- |
| *Use case* | Deskripsi |
| Login | Melakukan login untuk mengakses aplikasi |
| Mengelola data kain | Mengelola data kain yang bisa melakukan lihat, tambah, edit dan hapus data |
| Mengelola data penyimpanan | Mengelola data kolega yang bisa melakukan lihat, tambah, edit dan hapus data |
| Mengelola data kolega | Mengelola data kolega yang bisa melakukan lihat, tambah, edit dan hapus data |
| Mengelola data pembelian | Mengelola data pembelian yang bisa melakukan lihat, tambah, edit dan hapus data |
| Mengelola data penjualan | Mengelola data penjualan yang bisa melakukan lihat, tambah, edit dan hapus data |

1. *Activity diagram*
2. Activity diagram *login*

A picture containing text, screenshot, diagram, line

Description automatically generated

Gambar 4.3 *Activity diagram* login

Pada Gambar 4.3 mendeskripsikan aktor yang harus login terlebih dahulu untuk menggunakan aplikasi. Aktor haru menginput username dan password dengan benar sesuai akun yang sudah terdaftar.

1. *Activity diagram* pengelolaan data kain

A picture containing text, screenshot, diagram

Description automatically generated

Gambar 4.4 *Activity diagram* tambah kain

Gambar 4.4 mendeskripsikan tambah data yaitu aktor mengklik menu kain lalu klik tombol tambah data kain, maka masuk ke halaman tambah. Selanjutnya aktor menginput data kain dan klik tombol simpan maka data tersimpan.

A diagram of a company

Description automatically generated

Gambar 4.5 *Activity diagram* edit kain

Pada Gambar 4.5 mendeskripsikan edit data yaitu aktor mengklik menu kain kemudian klik ikon edit pada baris data yang dipilih, maka akan memasuki halaman edit. Kemudian aktor mengubah data sesuai keperluan dan mengklik tombol update.

A diagram of a company

Description automatically generated

Gambar 4.6 *Activity diagram* hapus kain

Pada Gambar 4.6 mendeskripsikan hapus data yaitu aktor mengklik menu kain kemudian klik ikon hapus pada baris yang dipilih. Maka akan ada konfirmasi untuk menghapus data atau batal, jika klik tombol setuju maka data kain terhapus.

1. *Activity diagram* pengelolaan data kolega

A diagram of a company

Description automatically generated

Gambar 4.7 *Activity diagram* tambah kolega

Gambar 4.7 mendeskripsikan tambah data yaitu aktor mengklik menu kolega lalu klik tombol tambah data, maka masuk ke halaman tambah kolega. Selanjutnya aktor menginput data kolega dan klik tombol simpan maka data tersimpan.

A diagram of a company

Description automatically generated

Gambar 4.8 *Activity diagram* edit kolega

Pada Gambar 4.5 mendeskripsikan edit data yaitu aktor mengklik menu kolega kemudian klik ikon edit pada baris data yang dipilih, maka akan memasuki halaman edit. Kemudian aktor mengubah data sesuai keperluan dan mengklik tombol update.

A diagram of a company

Description automatically generated

Gambar 4.9 *Activity diagram* hapus kolega

Pada Gambar 4.9 mendeskripsikan hapus data yaitu aktor mengklik menu kolega kemudian klik ikon hapus pada baris yang dipilih. Maka akan ada konfirmasi untuk menghapus data atau batal, jika klik tombol setuju maka data terhapus.

1. *Activity diagram* pengelolaan data penyimpanan

A diagram of a company

Description automatically generated

Gambar 4.10 *Activity diagram* tambah penyimpanan

Gambar 4.10 mendeskripsikan tambah data yaitu aktor mengklik menu penyimpanan lalu klik tombol tambah data, maka masuk ke halaman tambah. Selanjutnya aktor menginput data dan klik tombol simpan maka data tersimpan.

A diagram of a company

Description automatically generated

Gambar 4.11 *Activity diagram* edit penyimpanan

Pada Gambar 4.11 mendeskripsikan edit data yaitu aktor mengklik menu penyimpanan kemudian klik ikon edit pada baris data yang dipilih, maka akan memasuki halaman edit. Kemudian aktor mengubah data sesuai keperluan dan mengklik tombol update maka data akan terupdate.

A diagram of a company

Description automatically generated

Gambar 4.12 *Activity diagram* hapus penyimpanan

Pada Gambar 4.12 mendeskripsikan hapus data yaitu aktor mengklik menu penyimpanan kemudian klik ikon hapus pada baris yang dipilih. Maka akan ada konfirmasi menghapus data atau batal, jika klik tombol setuju maka data terhapus.

1. *Activity diagram* pengelolaan data penjualan

A diagram of a company

Description automatically generated

Gambar 4.13 *Activity diagram* tambah penjualan

Gambar 4.13 mendeskripsikan tambah data yaitu aktor mengklik menu penjualan lalu klik tombol tambah data, maka masuk ke halaman tambah. Selanjutnya aktor menginput data dan klik tombol simpan maka data tersimpan.

A diagram of a company

Description automatically generated

Gambar 4.14 *Activity diagram* edit penjualan

Pada Gambar 4.14 mendeskripsikan edit data yaitu aktor mengklik menu penjualan kemudian klik ikon edit pada baris data yang dipilih, maka akan memasuki halaman edit. Kemudian aktor mengubah data sesuai keperluan dan mengklik tombol update maka data akan terupdate.

A diagram of a company

Description automatically generated

Gambar 4.15 *Activity diagram* hapus penjualan

Pada Gambar 4.15 mendeskripsikan hapus data yaitu aktor mengklik menu penjualan kemudian klik ikon hapus pada baris yang dipilih. Maka akan ada konfirmasi untuk menghapus data, jika klik tombol setuju maka data terhapus.

A diagram of a company

Description automatically generated

Gambar 4.16 *Activity diagram* laporan penjualan

Pada Gambar 4.16 mendeskripsikan cetak laporan data yaitu aktor mengklik menu penjualan kemudian mengklik tombol cetak maka akan muncul jendela form cetak untuk menginput waktu, selanjutnya klik iya maka data tercetak.

A diagram of a company

Description automatically generated

Gambar 4.17 *Activity diagram* surat jalan penjualan

Pada Gambar 4.17 mendeskripsikan cetak surat jalan yaitu aktor mengklik menu penjualan kemudian mengklik tombol surat jalan maka akan muncul jendela form untuk data surat jalan, selanjutnya klik iya maka data surat jalan tercetak

1. *Activity diagram* pengelolaan data pembelian

A diagram of a diagram

Description automatically generated

Gambar 4.18 *Activity diagram* tambah pembelian

Gambar 4.18 mendeskripsikan tambah data yaitu aktor mengklik menu pembelian lalu klik tombol tambah data, maka masuk ke halaman tambah. Selanjutnya aktor menginput data dan klik tombol simpan maka data tersimpan.

A diagram of a company

Description automatically generated

Gambar 4.19 *Activity diagram* edit pembelian

Pada Gambar 4.19 mendeskripsikan edit data yaitu aktor mengklik menu pembelian kemudian klik ikon edit pada baris data yang dipilih, maka akan memasuki halaman edit. Kemudian aktor mengubah data sesuai keperluan dan mengklik tombol update maka data akan terupdate.

A diagram of a company

Description automatically generated

Gambar 4.20 *Activity diagram* hapus pembelian

Pada Gambar 4.20 mendeskripsikan hapus data yaitu aktor mengklik menu pembelian kemudian klik ikon hapus pada baris yang dipilih. Maka akan ada konfirmasi untuk menghapus data, jika klik tombol setuju maka data terhapus.

A diagram of a company

Description automatically generated

Gambar 4.21 *Activity diagram* laporan pembelian

Pada Gambar 4.21 mendeskripsikan cetak laporan data yaitu aktor mengklik menu pembelian kemudian mengklik tombol cetak maka akan muncul jendela form cetak untuk menginput waktu, selanjutnya klik iya maka data tercetak

1. *Activity diagram* pengelolaan data pengguna

A diagram of a company

Description automatically generated

Gambar 4.22 *Activity diagram* tambah pengguna

Gambar 4.4 mendeskripsikan tambah data yaitu aktor mengklik menu pengguna lalu klik tombol tambah data, maka masuk ke halaman tambah. Selanjutnya aktor menginput data dan klik tombol simpan maka data tersimpan.

A diagram of a company

Description automatically generated

Gambar 4.23 *Activity diagram* edit pengguna

Pada Gambar 4.23 mendeskripsikan edit data yaitu aktor mengklik menu pengguna kemudian klik ikon edit pada baris data yang dipilih, maka akan memasuki halaman edit. Kemudian aktor mengubah data sesuai keperluan dan mengklik tombol update maka data akan terupdate.

A diagram of a company

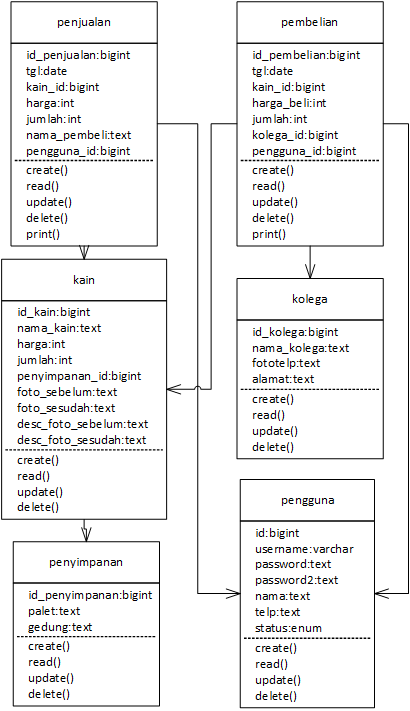
Description automatically generated

Gambar 4.24 *Activity diagram* hapus pengguna

Pada Gambar 4.24 mendeskripsikan hapus data yaitu aktor mengklik menu pengguna kemudian klik ikon hapus pada baris yang dipilih. Maka akan ada konfirmasi untuk menghapus data, jika klik tombol setuju maka data kain terhapus.

1. *Class diagram*

*Class diagram* ini menggambarkan konsep tabel pada basis data aplikasi inventori dan alur relasi yang saling menyambung dan membutuhkan untuk kelengkapan informasi yang akan disampaikan kepada pengguna aplikasi.



Gambar 4.25 *Class diagram*

1. *Sequence diagram*
2. *Sequence diagram login*

A diagram of a computer

Description automatically generated with low confidence

Gambar 4.26 *Sequence diagram login*

Pada Gambar 4.26 mendeskripsikan proses login. Aktor menginput kemudian *controller* memasukan data ke database. Jika data benar maka sistem akan masuk ke halaman dashboard, namun jika inputan salah maka kembali ke halaman yang sama dengan menampilkan pemberitahuan gagal.

1. *Sequence diagram* pengelolaan data kain

A diagram of a data flow

Description automatically generated

Gambar 4.27 *Sequence* *diagram* tambah kain

Pada Gambar 4.27 mendeskripsikan tambah kain. Aktor menginput data kain yang baru kemudian *controller* memvalidasi. Jika inputan benar maka akan dikirimkan untuk disimpan ke database dan menampilkan pemberitahuan berhasil ke layar, namun jika tidak valid maka akan kembali ke halaman yang sama dengan menampilkan pemberitahuan gagal.

A diagram of a data flow

Description automatically generated

Gambar 4.28 *Sequence diagram* edit kain

Pada Gambar 4.28 mendeskripsikan edit kain. Aktor menginput data kain yang baru kemudian *controller* memvalidasi. Jika inputan benar maka akan dikirimkan kedatabase untuk diupdate merubah data sebelumnya dan menampilkan pemberitahuan berhasil ke layar, namun jika tidak valid maka akan kembali ke halaman yang sama dengan menampilkan pemberitahuan gagal.

A diagram of a software application

Description automatically generated

Gambar 4.29 *Sequence diagram* hapus kain

Pada Gambar 4.29 mendeskripsikan hapus data. Aktor mengklik ikon hapus pada baris data yang dipilih. Maka *controller* menampilkan konfirmasi hapus. Jika aktor mengklik iya maka akan dikirim ke database dan data tersebut terhapus dan menampilkan pemberitahuan data terhapus. Jika aktor mengklik tidak maka akan kembali ke tabel data.

1. *Sequence diagram* pengelolaan data kolega

A diagram of a data flow

Description automatically generated

Gambar 4.30 *Sequence diagram* tambah kolega

Pada Gambar 4.30 mendeskripsikan tambah kolega. Aktor menginput data kain yang baru kemudian *controller* memvalidasi. Jika inputan benar maka akan dikirimkan untuk disimpan ke database dan menampilkan pemberitahuan berhasil ke layar, namun jika tidak valid maka akan kembali ke halaman yang sama dengan menampilkan pemberitahuan gagal.

A diagram of a data system

Description automatically generated

Gambar 4.31 *Sequence diagram* edit kolega

Pada Gambar 4.31 mendeskripsikan edit kolega. Aktor menginput data yang baru kemudian *controller* memvalidasi. Jika inputan benar maka akan dikirimkan ke database untuk diupdate merubah data sebelumnya dan menampilkan pemberitahuan berhasil ke layar, namun jika tidak valid maka akan kembali ke halaman yang sama dengan menampilkan pemberitahuan gagal.

A diagram of a software application

Description automatically generated

Gambar 4.32 *Sequence diagram* hapus kolega

Pada Gambar 4.32 mendeskripsikan hapus data. Aktor mengklik ikon hapus pada baris data yang dipilih. Maka *controller* menampilkan konfirmasi hapus. Jika aktor mengklik iya maka akan dikirim ke database dan data tersebut terhapus dan menampilkan pemberitahuan data terhapus. Jika aktor mengklik tidak maka akan kembali ke tabel data.

1. *Sequence diagram* pengelolaan data penyimpanan

A diagram of a data flow

Description automatically generated

Gambar 4.33 *Sequence diagram* tambah penyimpanan

Pada Gambar 4.33 mendeskripsikan tambah penyimpanan. Aktor menginput data kain yang baru kemudian *controller* memvalidasi. Jika inputan benar maka akan dikirimkan untuk disimpan ke database dan menampilkan pemberitahuan berhasil ke layar, namun jika tidak valid maka akan kembali ke halaman yang sama dengan menampilkan pemberitahuan gagal.

A diagram of a data flow

Description automatically generated

Gambar 4.34 *Sequence diagram* edit penyimpanan

Pada Gambar 4.34 mendeskripsikan edit penyimpanan. Aktor menginput data yang baru kemudian *controller* memvalidasi. Jika inputan benar maka akan dikirimkan ke database untuk diupdate merubah data sebelumnya dan menampilkan pemberitahuan berhasil ke layar, namun jika tidak valid maka akan kembali ke halaman yang sama dengan menampilkan pemberitahuan gagal.

A diagram of a software application

Description automatically generated

Gambar 4.35 *Sequence diagram* hapus penyimpanan

Pada Gambar 4.35 mendeskripsikan hapus data. Aktor mengklik ikon hapus pada baris data yang dipilih. Maka *controller* menampilkan konfirmasi hapus. Jika aktor mengklik iya maka akan dikirim ke database dan data tersebut terhapus dan menampilkan pemberitahuan data terhapus. Jika aktor mengklik tidak maka akan kembali ke tabel data.

1. *Sequence diagram* pengelolaan data penjualan

A diagram of a data flow

Description automatically generated

Gambar 4.36 *Sequence diagram* tambah penjualan

Pada Gambar 4.36 mendeskripsikan tambah penjualan. Aktor menginput data kain yang baru kemudian *controller* memvalidasi. Jika inputan benar maka akan dikirimkan untuk disimpan ke database dan menampilkan pemberitahuan berhasil ke layar, namun jika tidak valid maka akan kembali ke halaman yang sama dengan menampilkan pemberitahuan gagal.

A diagram of a data flow

Description automatically generated

Gambar 4.37 *Sequence diagram* edit penjualan

Pada Gambar 4.37 mendeskripsikan edit penjualan. Aktor menginput data yang baru kemudian *controller* memvalidasi. Jika inputan benar maka akan dikirimkan ke database untuk diupdate merubah data sebelumnya dan menampilkan pemberitahuan berhasil ke layar, namun jika tidak valid maka akan kembali ke halaman yang sama dengan menampilkan pemberitahuan gagal.

A diagram of a software application

Description automatically generated

Gambar 4.38 *Sequence diagram* hapus penjualan

Pada Gambar 4.38 mendeskripsikan hapus data. Aktor mengklik ikon hapus pada baris data yang dipilih. Maka *controller* menampilkan konfirmasi hapus. Jika aktor mengklik iya maka akan dikirim ke database dan data tersebut terhapus dan menampilkan pemberitahuan data terhapus. Jika aktor mengklik tidak maka akan kembali ke tabel data.

A diagram of a data flow

Description automatically generated

Gambar 4.39 *Sequence diagram* laporan penjualan

Pada Gambar 4.39 mendeskripsikan laporan penjualan. Aktor mengklik tombol laporan kemudian muncul form untuk menginput tanggal sesuai keinginan. Setelah itu *controller* akan mengirimkan inputan tersebut ke database. Sehingga data tersebut difilter dan menampilkan data sesuai inputan dan siap untuk di cetak.

A diagram of a data flow

Description automatically generated

Gambar 4.40 *Sequence diagram* surat jalan penjualan

Pada Gambar 4.40 mendeskripsikan surat jalan. Aktor mengklik tombol surat jalan dan muncul form untuk menginput isi surat jalan. Kemudian *controller* mengirimkan inputan ke database dan menampilkan data sesuai inputan ke layar. Maka halaman surat jalan siap dicetak.

1. *Sequence diagram* pengelolaan data pembelian

A diagram of a data flow

Description automatically generated

Gambar 4.41 *Sequence diagram* tambah pembelian

Pada Gambar 4.41 mendeskripsikan tambah pembelian. Aktor menginput data kain yang baru kemudian *controller* memvalidasi. Jika inputan benar maka akan dikirimkan untuk disimpan ke database dan menampilkan pemberitahuan berhasil ke layar, namun jika tidak valid maka akan kembali ke halaman yang sama dengan menampilkan pemberitahuan gagal.

A diagram of a data flow

Description automatically generated

Gambar 4.42 *Sequence diagram* edit pembelian

Pada Gambar 4.42 mendeskripsikan edit pembelian. Aktor menginput data yang baru kemudian *controller* memvalidasi. Jika inputan benar maka akan dikirimkan ke database untuk diupdate merubah data sebelumnya dan menampilkan pemberitahuan berhasil ke layar, namun jika tidak valid maka akan kembali ke halaman yang sama dengan menampilkan pemberitahuan gagal.

A diagram of a software application

Description automatically generated

Gambar 4.43 *Sequence diagram* hapus pembelian

Pada Gambar 4.43 mendeskripsikan hapus data. Aktor mengklik ikon hapus pada baris data yang dipilih. Maka *controller* menampilkan konfirmasi hapus. Jika aktor mengklik iya maka akan dikirim ke database dan data tersebut terhapus dan menampilkan pemberitahuan data terhapus. Jika aktor mengklik tidak maka akan kembali ke tabel data.

A diagram of a data flow

Description automatically generated

Gambar 4.44 *Sequence diagram* laporan pembelian

Pada Gambar 4.44 mendeskripsikan laporan penjualan. Aktor mengklik tombol laporan kemudian menginput tanggal sesuai keinginan. Maka *controller* akan mengirimkan inputan tersebut ke database. Sehingga data tersebut difilter dan menampilkan data sesuai inputan dan siap untuk di cetak.

1. *Sequence diagram* pengelolaan data pengguna

A diagram of a data flow

Description automatically generated

Gambar 4.45 *Sequence diagram* tambah pengguna

Pada Gambar 4.45 mendeskripsikan tambah pengguna. Aktor menginput data kain yang baru kemudian *controller* memvalidasi. Jika inputan benar maka akan dikirimkan untuk disimpan ke database dan menampilkan pemberitahuan berhasil ke layar, namun jika tidak valid maka akan kembali ke halaman yang sama dengan menampilkan pemberitahuan gagal.

A diagram of a data flow

Description automatically generated

Gambar 4.46 *Sequence diagram* edit pengguna

Pada Gambar 4.46 mendeskripsikan edit pengguna. Aktor menginput data yang baru kemudian *controller* memvalidasi. Jika inputan benar maka akan dikirimkan ke database untuk diupdate merubah data sebelumnya dan menampilkan pemberitahuan berhasil ke layar, namun jika tidak valid maka akan kembali ke halaman yang sama dengan menampilkan pemberitahuan gagal.

A diagram of a software application

Description automatically generated

Gambar 4.47 *Sequence diagram* hapus pengguna

Pada Gambar 4.47 mendeskripsikan hapus data. Aktor mengklik ikon hapus pada baris data yang dipilih. Maka *controller* menampilkan konfirmasi hapus. Jika aktor mengklik iya maka akan dikirim ke database dan data tersebut terhapus dan menampilkan pemberitahuan data terhapus. Jika aktor mengklik tidak maka akan kembali ke tabel data.

### Struktur Tabel

Berikut tabel basis data yang akan dirancang untuk membangun sistem inventory. Rancangan basis data mencakup *field, type, size, indeks*, dan keterangan. Pada kolom index bagian tertentu akan berisi PK(*Primary Key*) atau FK(*Foreign Key*). Berikut adalah rancangan basis data untuk aplikasi inventori :

Tabel 4.7 Data pengguna

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Field* | *Type* | *Size* | *Index* | Keterangan |
| Id | Bigint | 20 | PK | Id pengguna |
| username | Varchar | 30 |  | Username |
| password | Text |  |  | Password |
| password2 | Text |  |  | Password ke 2 |
| nama | Text |  |  | Nama lengkap |
| telp | Text |  |  | No. telepon |
| status | enum |  |  | Status pengguna |

Pada Tabel 4.7 berisi mengenai tabel yang akan digunakan sebagai data akun yang hanya bisa mengakses aplikasi dan aplikasi lebih aman agar tidak diakses oleh sembarang orang terutama menjaga data perusahaan.

Tabel 4.8 Data kain

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Field* | *Type* | *Size* | *Index* | Keterangan |
| id\_kain | Bigint | 20 | PK | Id kain |
| nama\_kain | Text |  |  | Nama kain |
| Harga | Int | 11 |  | Harga jual kain |
| Jumlah | Int | 11 |  | Jumlah stok kain yang tersedia |
| penyimpanan\_id | Bigint | 20 | FK | Id tempat penyimpanan kain |
| foto\_sebelum | Text |  |  | Foto kain sebelum diolah |
| foto\_sesudah | Text |  |  | Foto kain sesudah diolah |
| desc\_foto\_sebelum | Text |  |  | Deskripsi foto kain sebelum |
| desc\_foto\_sesudah | Text |  |  | Deskripsi foto kain sesudah |

Pada Tabel 4.8 berisi data kain yang akan digunakan dalam mengatur inventori atau stok data kain dan menampilkan jenis kain yang tersedia.

Tabel 4.9 Data kolega

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Field* | *Type* | *Size* | *Index* | Keterangan |
| id\_kolega | Bigint | 20 | PK | Id kolega |
| nama\_kolega | Text |  |  | Nama kolega |
| Telp | Text |  |  | No telepon kolega |
| Alamat | Text |  |  | Alamat kolega |

Pada Tabel 4.9 berisi mengenai data kolega yang bekerjasama dengan perusahaan dan menyuplai jenis kain yang akan diolah dan diperdagangkan oleh perusahaan.

Tabel 4.10 Data pembelian

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Field* | *Type* | *Size* | *Index* | Keterangan |
| id\_pembelian | Bigint | 20 | PK | Id pembelian |
| Tgl | Date |  |  | Tanggal pembelian |
| kain\_id | Bigint | 20 | FK | Id kain |
| harga\_beli | Int | 11 |  | Harga beli |
| Jumlah | Int | 11 |  | Jumlah kain yang dibeli |
| kolega\_id | Bigint | 20 | FK | Id kolega |
| pengguna\_id | Bigint | 20 | FK | Id pengguna |

Pada Tabel 4.10 berisi mengenai data transaksi pembelian kain mentah dari kolega dan berguna untuk transaksi laporan pembelian kain serta bertambahnya stok kain yang diolah dan diperdagangkan oleh perusahaan.

Tabel 4.11 Data penjualan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Field* | *Type* | *Size* | *Index* | Keterangan |
| id\_penjualan | Bigint | 20 | PK | Id penjualan |
| Tgl | Date |  |  | Tanggal penjualan |
| kain\_id | Bigint | 20 | FK | Id kain |
| Harga | Int | 11 |  | Harga jual |
| Jumlah | Int | 11 |  | Jumlah kain yang dijual |
| nama\_pembeli | Text |  |  | Nama pembeli |
| pengguna\_id | Bigint | 20 | FK | Id pengguna |

Tabel 4.11 berisi mengenai data transaksi penjualan kain kepada konsumen. Sehingga transaksi tercatat yang berguna untuk laporan penjualan dan stok kain pun akan berkurang.

Tabel 4.12 Data penyimpanan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Field* | *Type* | *Size* | *Index* | Keterangan |
| id\_penyimpanan | Bigint | 20 | PK | Id penyimpanan |
| Palet | Text |  |  | Palet penyimpanan |
| Gedung | Text |  |  | Tempat penyimpanan |

Pada Tabel 4.12 berisi mengenai data tempat penyimpanan kain. Data ini berguna untuk pengambilan kain disimpan sehingga menjadi lebih mudah da.

### Desain

Desain *user interface* pada sistem inventori menggunakan *Balsamiq Mockup*. Dengan desain ini memudahkan dalam mengimplementasikan tampilan aplikasi sesuai dengan kebutuhan. Sehingga tahap implementasi akan lebih cepat tanpa memikirkan desain, posisi, tombol dan lain sebagainya. Adapun User interface yang dibuat dari sistem inventory diantaranya:

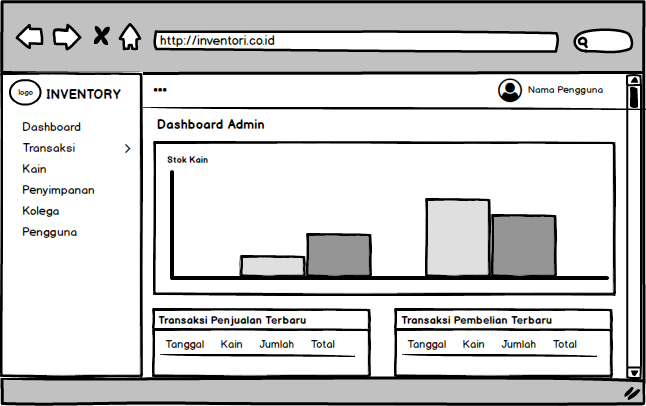
Desain tampilan *login*

A screenshot of a login box

Description automatically generated

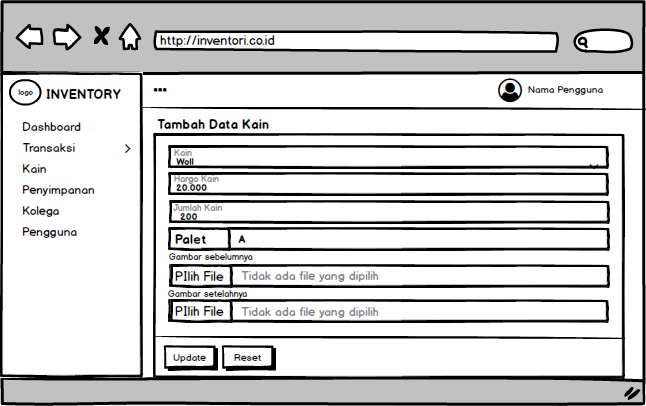
Gambar 4.48 Desain tampilan *login*

Desain tampilan *dashboard*

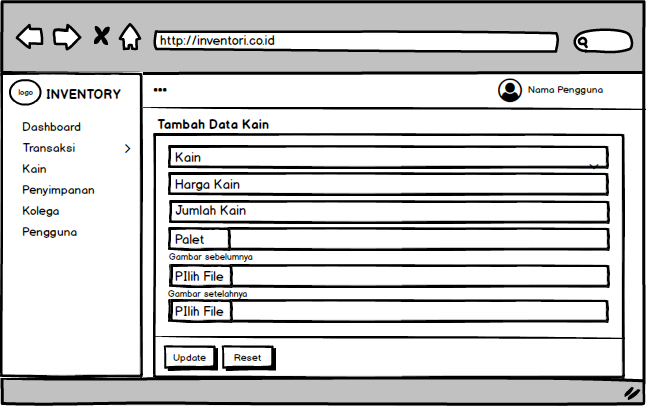


Gambar 4.49 Desain tampilan *dashboard* admin

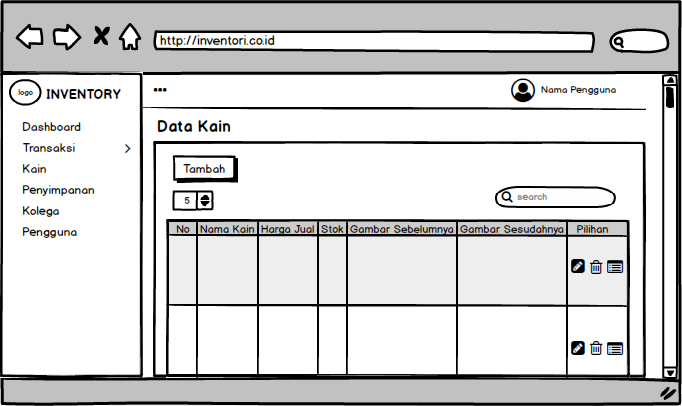
Desain tampilan kain



Gambar 4.50 Desain tampilan edit kain

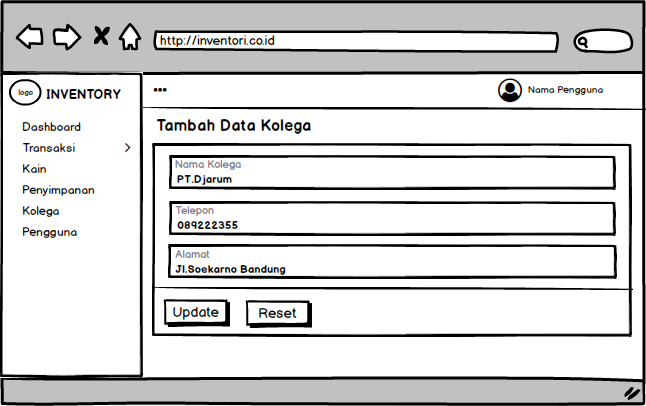


Gambar 4.51 Desain tampilan tambah kain

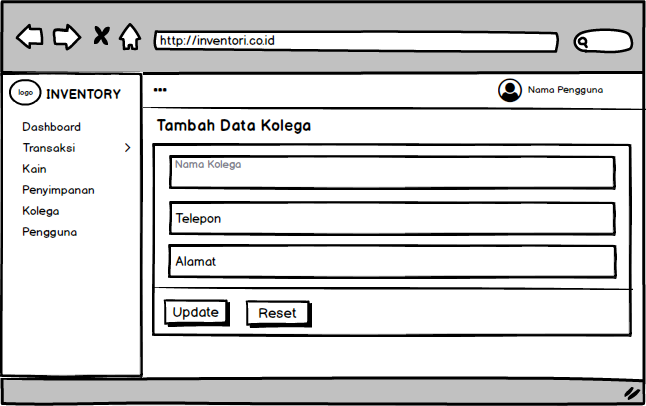


Gambar 4.52 Desain tampilan tabel kain

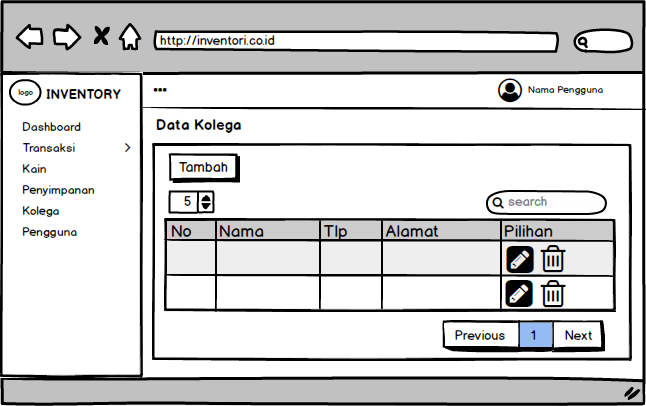
Desain tampilan kolega



Gambar 4.53 Desain tampilan edit kolega

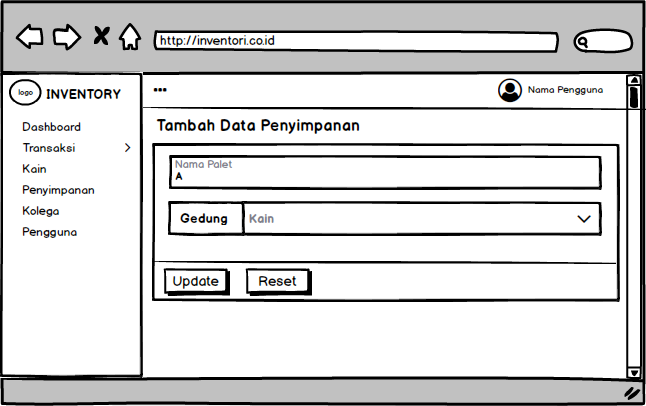


Gambar 4.54 Desain tampilan tambah kolega

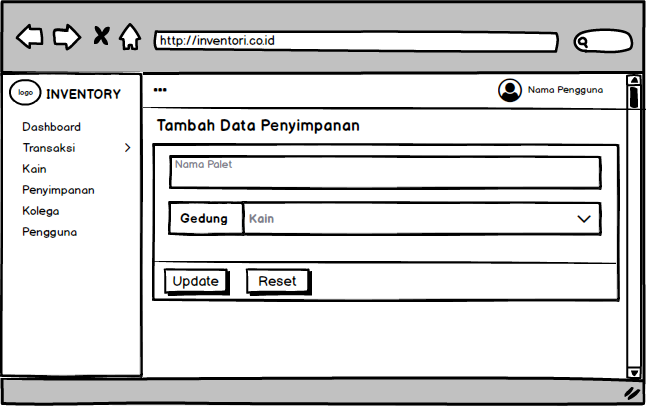


Gambar 4.55 Desain tampilan tabel kolega

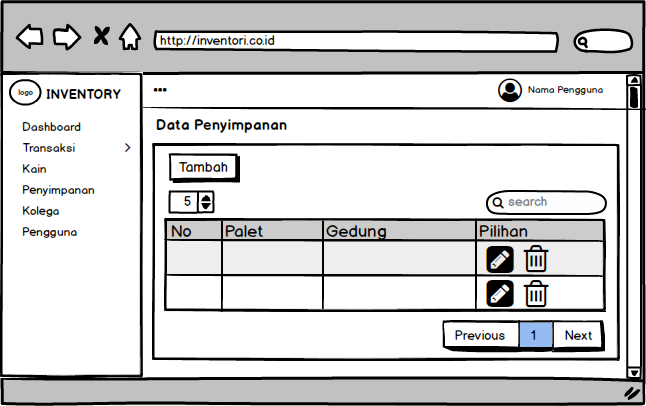
Desain tampilan tempat penyimpanan



Gambar 4.56 Desain tampilan edit penyimpanan

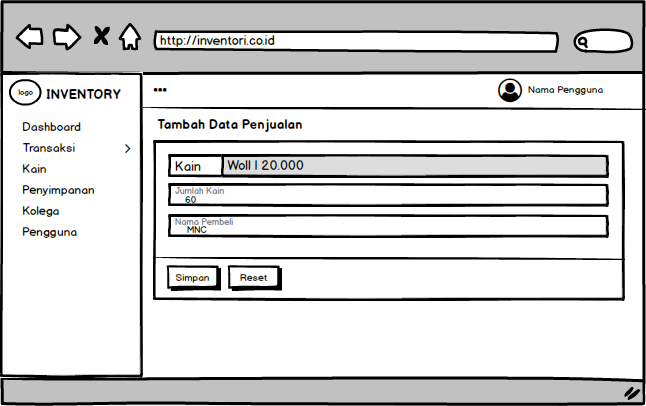


Gambar 4.57 Desain tampilan tambah penyimpanan



Gambar 4.58 Desain tampilan tabel penyimpanan

Desain tampilan penjualan



Gambar 4.59 Desain tampilan edit penjualan

A screenshot of a computer

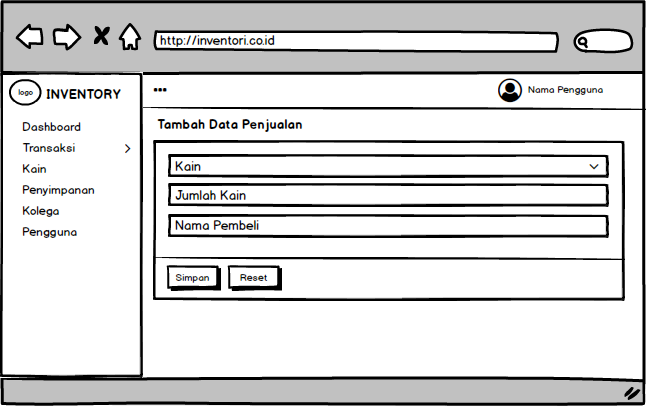
Description automatically generated

Gambar 4.60 Desain tampilan laporan penjualan

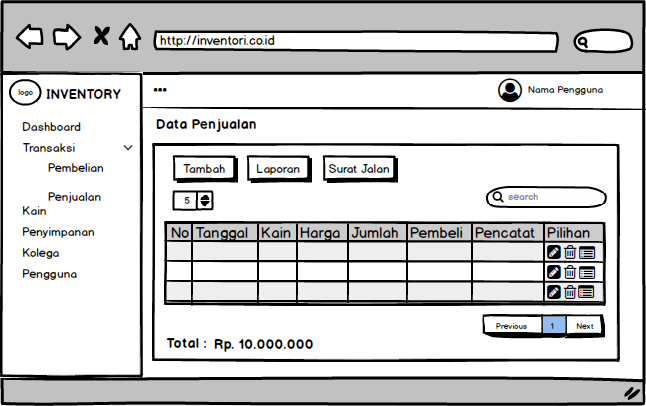
A screenshot of a computer screen

Description automatically generated

Gambar 4.61 Desain tampilan surat jalan penjualan

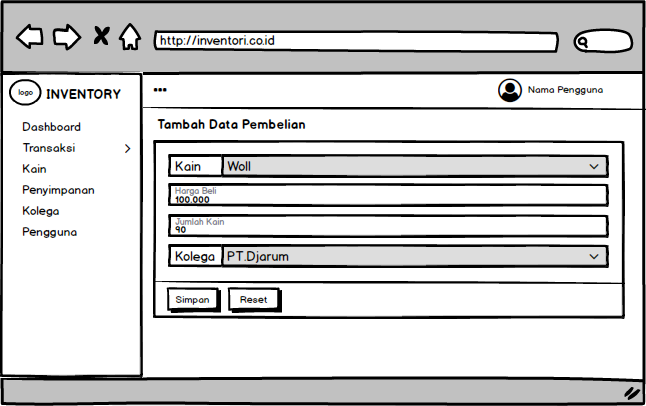


Gambar 4.62 Desain tampilan tambah penjualan



Gambar 4.63 Desain tampilan tabel penjualan

Desain tampilan pembelian

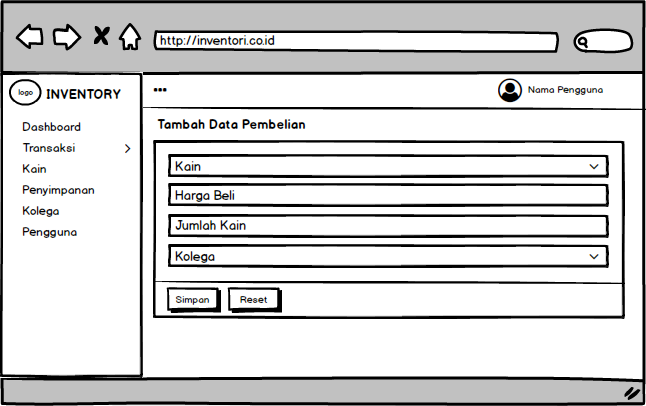


Gambar 4.64 Desain tampilan edit pembelian

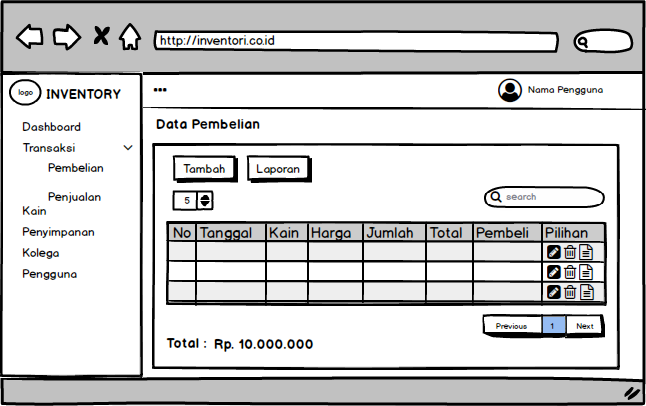
A screenshot of a computer

Description automatically generated

Gambar 4.65 Desain tampilan laporan pembelian

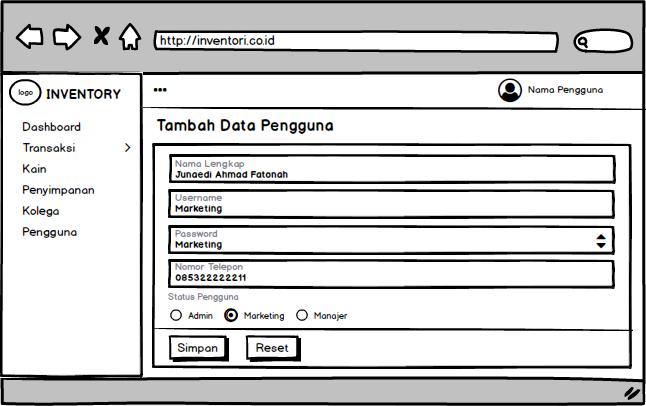


Gambar 4.66 Desain tampilan tambah pembelian

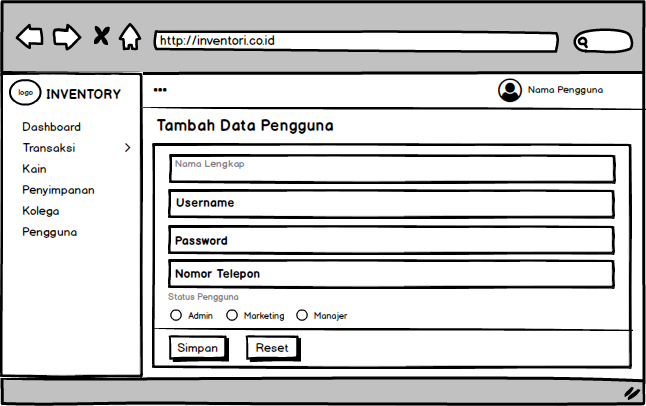


Gambar 4.67 Desain tampilan tabel pembelian

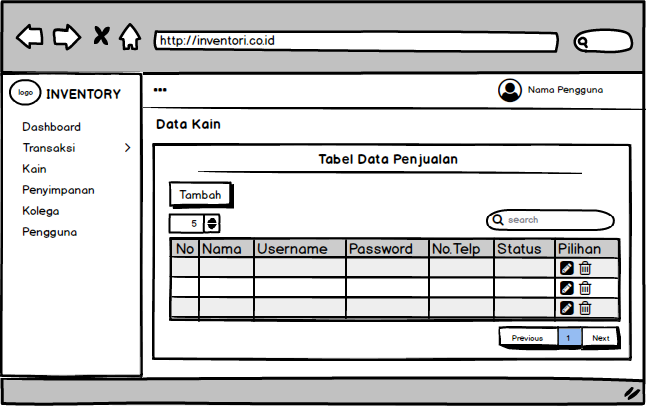
Desain tampilan pengguna



Gambar 4.68 Desain tampilan edit prngguna



Gambar 4.69 Desain tampilan tambah pengguna



Gambar 4.70 Desain tampilan tabel pengguna

### *Listing Program*

*Listing program* merupakan susunan dari beberapa struktur data / *computer codes* yang digunakan dalam merancang bangun sistem informasi terutama dalam mengatasi masalah yang sedang diteliti. Berikut list program dalam mengatasi masalah :

1. *Controller* kain

|  |
| --- |
| <?php  namespace App\Http\Controllers;  use Illuminate\Http\Request;  use App\Models\MKain;  use App\Models\MPenyimpanan;  use Illuminate\Support\Facades\Storage;  class CKain extends Controller  {  public function index()  {    $data = [    'title' => 'Data kain',    'kain' => MKain::with('penyimpanan')->get(),    ];  *return* view('home.kain.kain', $data);  }  public function create()  {  *$this*->authorize('admin');    $data = [    'title' => 'Tambah kain',    'penyimpanan' => MPenyimpanan::all(),    ];  *return* view('home.kain.kainTambah', $data);  }  public function store(Request $request)  {    $validasi = $request->validate([    'nama\_kain' => 'max:50|min:2',    'harga' => 'max:50',    'jumlah' => 'max:20',    'penyimpanan\_id' => 'required',    'foto\_sebelum' => 'image|file',    'foto\_sesudah' => 'image|file',    ]);  *if* ($request->file('foto\_sebelum')) {    $validasi['foto\_sebelum'] = $request->file('foto\_sebelum')->store('gambar-sebelum');    } *else* {    $validasi['foto\_sebelum'] = 'default-kain.jpg';    }  *if* ($request->file('foto\_sesudah')) {    $validasi['foto\_sesudah'] = $request->file('foto\_sesudah')->store('gambar-sesudah');    } *else* {    $validasi['foto\_sesudah'] = 'default-kain.jpg';    }    MKain::create($validasi);  *return* redirect('/kain')->with('pesan', 'Data kain berhasil di tambah');  }  public function edit(MKain $kain)  {  *$this*->authorize('admin');    $data = [    'title' => 'Edit kain',    'kain' => $kain,    'penyimpanan' => MPenyimpanan::all(),    ];  *return* view('home/kain/kainEdit', $data);  }  public function update(Request $request, $idKain)  {    $validasi = $request->validate([    'nama\_kain' => 'max:50',    'harga' => 'max:50',    'jumlah' => 'max:20',    'penyimpanan\_id' => 'required',    'foto\_sebelum' => 'image|file',    'foto\_sesudah' => 'image|file',    ]);  *if* ($request->file('foto\_sebelum')) {  *if* ($request->oldFotoSebelum != 'default-kain.jpg') {  Storage::delete($request->oldFotoSebelum);    }    $validasi['foto\_sebelum'] = $request->file('foto\_sebelum')->store('gambar-sebelum');    }  *if* ($request->file('foto\_sesudah')) {  *if* ($request->oldFotoSesudah != 'default-kain.jpg') {  Storage::delete($request->oldFotoSesudah);    }    $validasi['foto\_sesudah'] = $request->file('foto\_sesudah')->store('gambar-sesudah');    }    MKain::where('id\_kain', $idKain)->update($validasi);  *return* redirect('/kain')->with('pesan', 'Data kain berhasil di update');  }  public function destroy(MKain $kain)  {  *if* ($kain->foto\_sebelum != 'default-kain.jpg')    Storage::delete($kain->foto\_sebelum);  *if* ($kain->foto\_sesudah != 'default-kain.jpg')    Storage::delete($kain->foto\_sesudah);    MKain::destroy($kain->id\_kain);  *return* redirect('/kain')->with('pesan', 'Data kain berhasil di hapus');  }  } |

1. *Model* kain

|  |
| --- |
| <?php  namespace App\Models;  use Illuminate\Database\Eloquent\Factories\HasFactory;  use Illuminate\Database\Eloquent\Model;  class MKain extends Model  {  use HasFactory;  protected $table = 'tb\_kain';  protected $primaryKey = 'id\_kain';  protected $guard = 'id\_kain';  protected $fillable = ['nama\_kain', 'harga', 'jumlah', 'penyimpanan\_id', 'foto\_sebelum', 'foto\_sesudah'];  public function penyimpanan()  {  *return* *$this*->belongsTo(MPenyimpanan::class, 'penyimpanan\_id');  }  } |

1. *View* data kain

|  |
| --- |
| <x-header>  @slot('title')  {{ $title }}  @endslot  </x-header>  <div *class*="page-breadcrumb">  <div *class*="row">  <div *class*="col-12 d-flex no-block align-items-center">  <h2 *class*="page-title">Data Kain</h2>  </div>  </div>  </div>  <div *class*="container-fluid">  <div *class*="row">  <div *class*="col-12">  <div *class*="card rounded-3 shadow-sm">    <div *class*="card-body">    <h3 *class*="text-center">Tabel Data Kain</h3>    <div *class*=" border-top border-secondary pb-3"></div>    @can('admin')    <a *class*="btn btn-primary mb-3" *title*="Tambah data" *href*="/kain/create" *role*="button"><i  *class*="bi bi-bookmark-plus"></i> Tambah</a>    @endcan    <table *id*="autoTabel" *class*="table table-striped table-bordered border-dark">    <thead *class*="bg-primary text-white">  <tr>  <th *width*="3%" *class*="text-white">No</th>  *{{-- <th width="10%" class="text-white">Update</th> --}}*  <th *width*="15%" *class*="text-white">Nama kain</th>  <th *width*="15%" *class*="text-white">Harga jual</th>  <th *width*="10%" *class*="text-white">Stok</th>  <th *width*="10%" *class*="text-white">Palet</th>  <th *width*="10%" *class*="text-white">Gedung</th>  <th *width*="20%" *class*="text-white">Gambar sebelumnya</th>  <th *width*="20%" *class*="text-white">Gambar sesudahnya</th>  @can('admin')  <th *width*="12%" *class*="text-white">Pilihan</th>  @endcan  </tr>    </thead>    <tbody>  @foreach ($kain as $k)  <tr>  <td>{{ $loop->iteration }}</td>  *{{-- <td>{{ tanggal($k->updated\_at) }}</td> --}}*  <td>{{ $k->nama\_kain }}</td>  <td *class*="text-end">{{ rupiah($k->harga) }} /m<sup>2</sup></td>  <td *class*="text-end">{{ $k->jumlah }} m<sup>2</sup></td>  <td>{{ $k->penyimpanan->palet }}</td>  <td>{{ $k->penyimpanan->gedung }}</td>  <td>  <img *src*="{{ asset('storage/' . $k->foto\_sebelum) }}"  *alt*="{{ $k->foto\_sebelum }}" *width*='150'>  </td>  <td>  <img *src*="{{ asset('storage/' . $k->foto\_sesudah) }}"  *alt*="{{ $k->foto\_sesudah }}" *width*="150">  </td>  @can('admin')  <td>    <a *href*="/kain/{{ $k->id\_kain }}/edit" *title*="Edit data"  *class*="btn btn-warning "><i *class*="bi bi-pencil-square"></i></a>    <form *action*="/kain/{{ $k->id\_kain }}" *method*="post" *class*="d-inline">    @method('delete')    @csrf    <button *title*="Hapus data" *class*="btn btn-danger show\_confirm"><i  *class*="bi bi-trash-fill"></i></button>    </form>  </td>  @endcan  </tr>  @endforeach    </tbody>    </table>    </div>  </div>  *<!-- Card -->*  </div>  </div>  </div>  <x-footer></x-footer> |

# IMPLEMENTASI DAN PENGUJIAN



## Implementasi

Setelah perancangan aplikasi selesai, langkah berikutnya adalah melaksanakan implementasinya. Implementasi merupakan tahap di mana sistem baru diterapkan dan aplikasi siap digunakan. Proses implementasi bertujuan untuk mengimplementasikan berbagai modul yang telah dirancang sebelumnya.

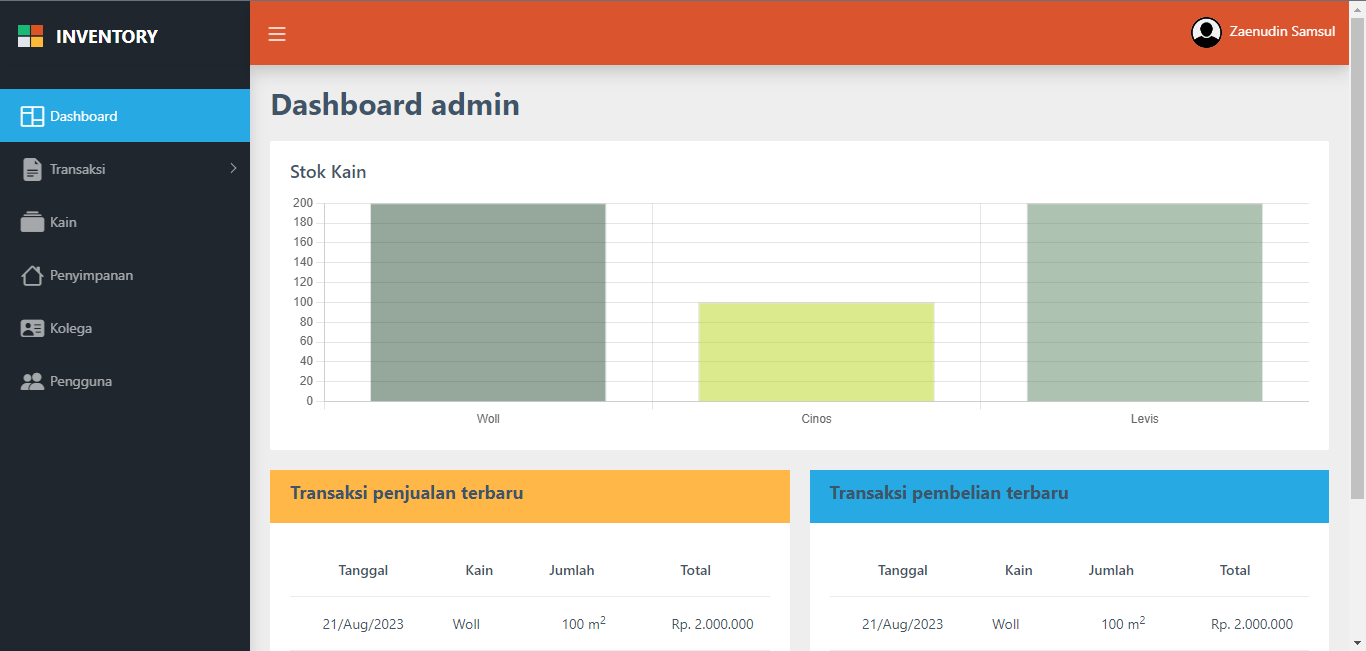
Halaman *login*

A login screen with white text and green and white text

Description automatically generated

Gambar 5.1 Halaman *login*

Halaman *dashboard*

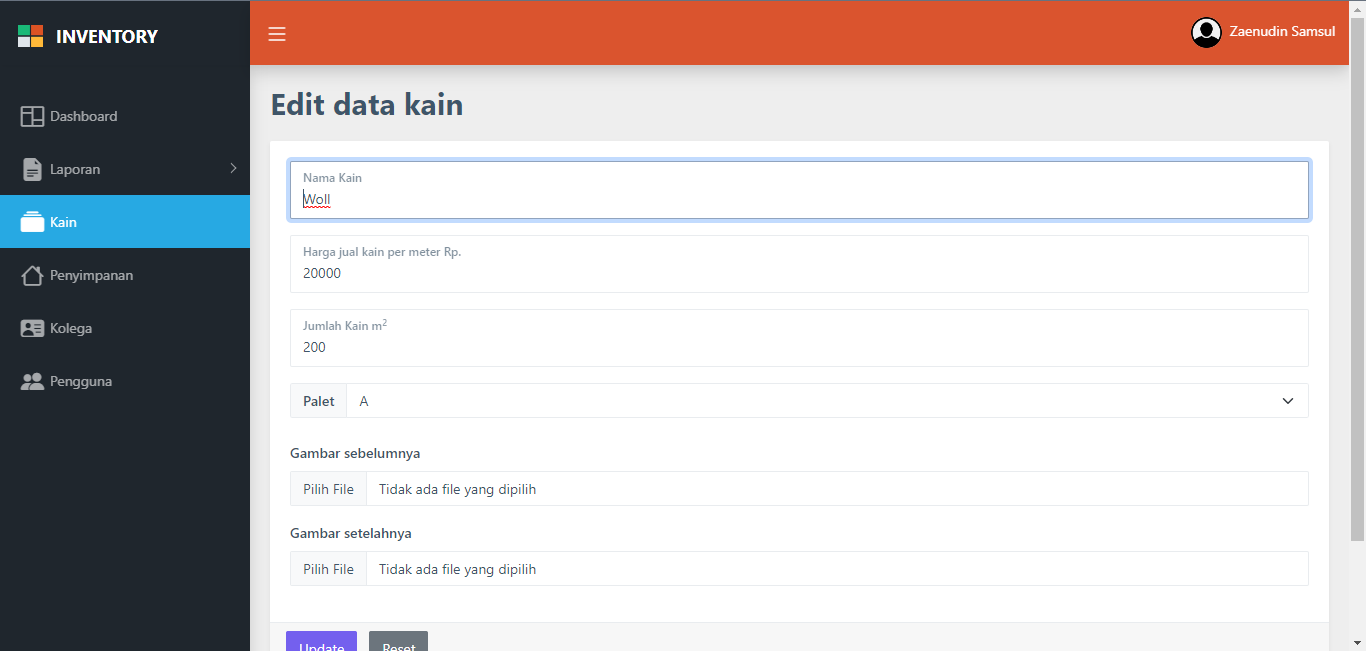


Gambar 5.2 Halaman *dashboard*

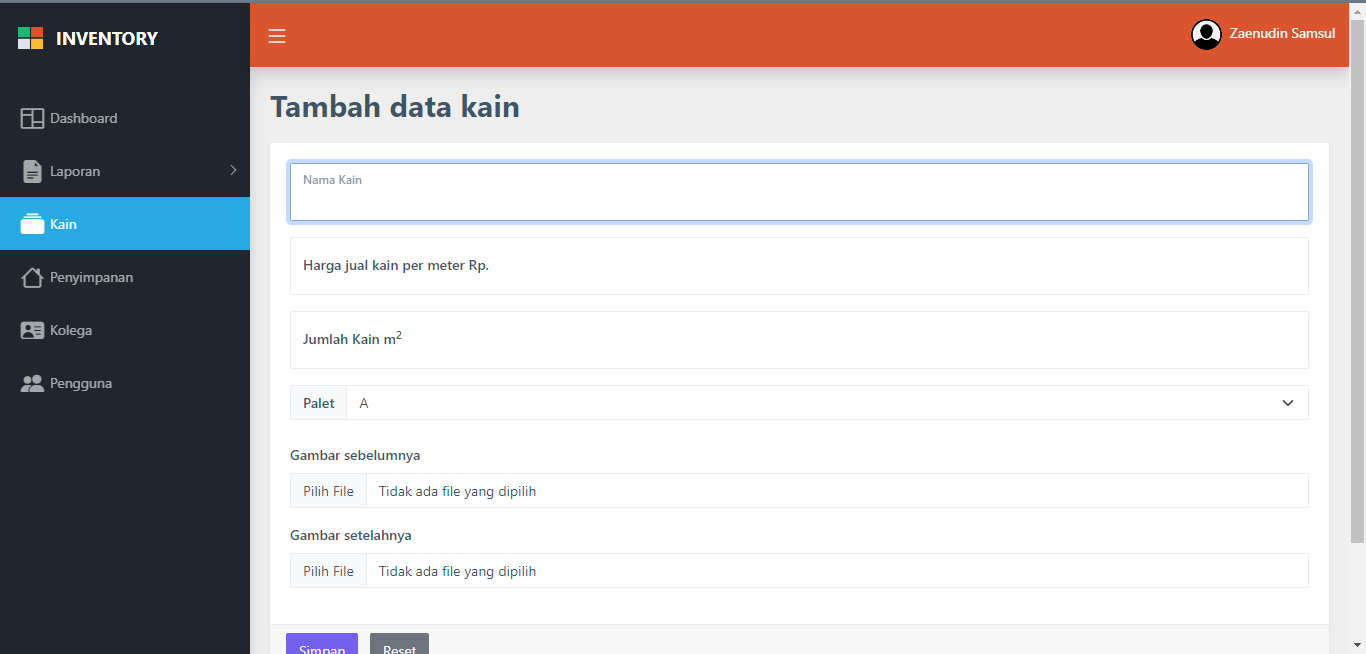
Halaman kain



Gambar 5.3 Halaman tabel kain

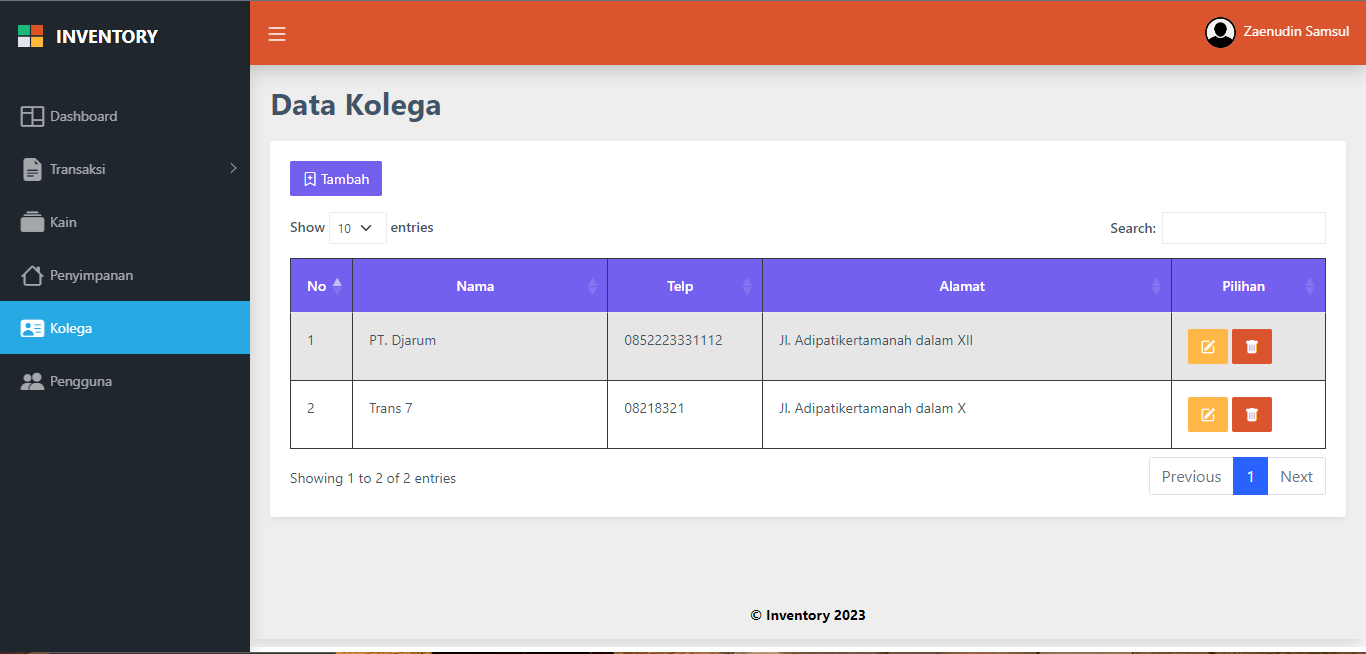


Gambar 5.4 Halaman edit kain



Gambar 5.5 Halaman tambah kain

Halaman kolega



Gambar 5.6 Halaman tabel kolega

A screenshot of a computer

Description automatically generated

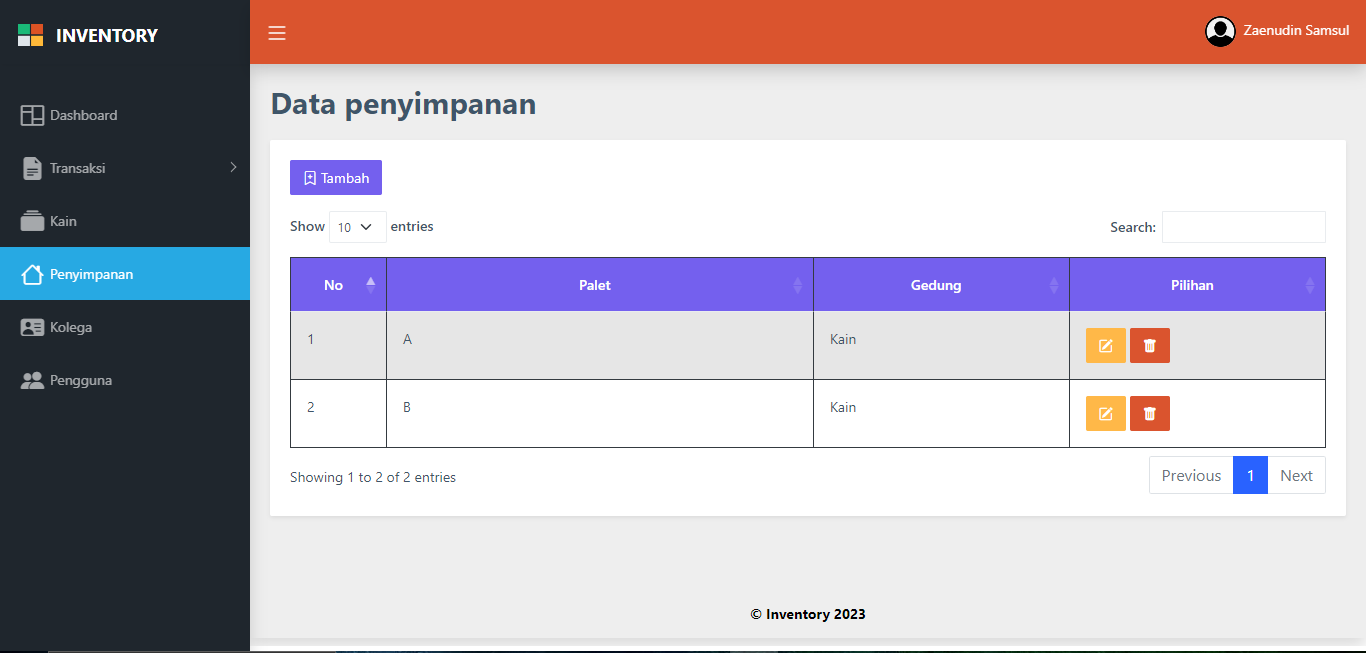
Gambar 5.7 Halaman edit kolega

A screenshot of a computer

Description automatically generated

Gambar 5.8 Halaman tambah kolega

Halaman tempat penyimpanan



Gambar 5.9 Halaman tabel penyimpanan

A screenshot of a computer

Description automatically generated

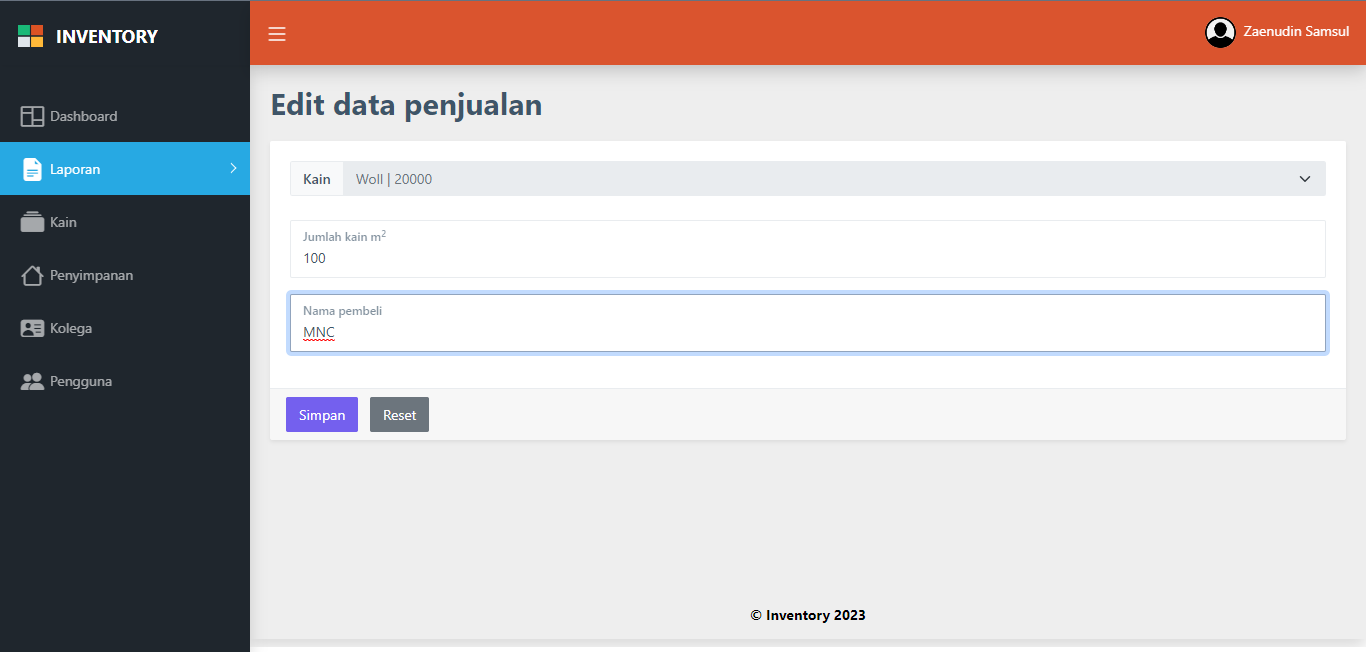
Gambar 5.10 Halaman edit penyimpanan

A screenshot of a computer

Description automatically generated

Gambar 5.11 Halaman tambah penyimpanan

Halaman penjualan



Gambar 5.12 Halaman edit penjualan

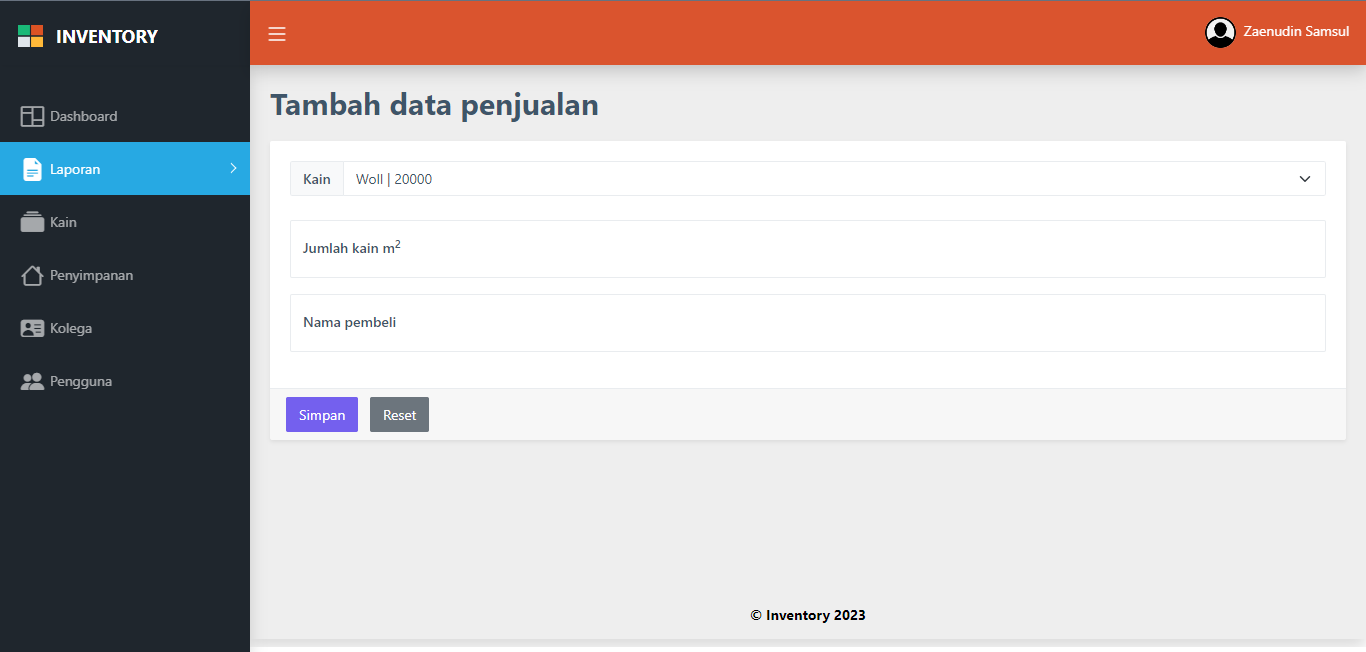


Gambar 5.13 Halaman laporan penjualan

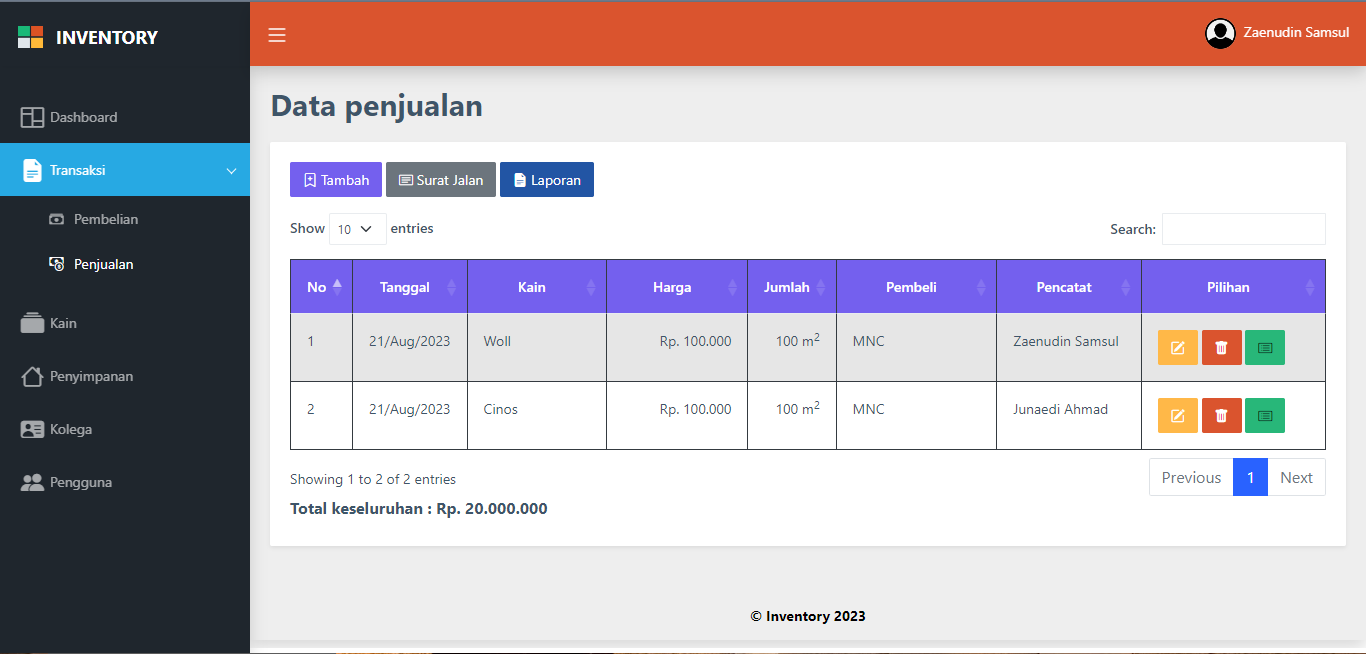
A screenshot of a computer

Description automatically generated

Gambar 5.14 Halaman surat jalan penjualan

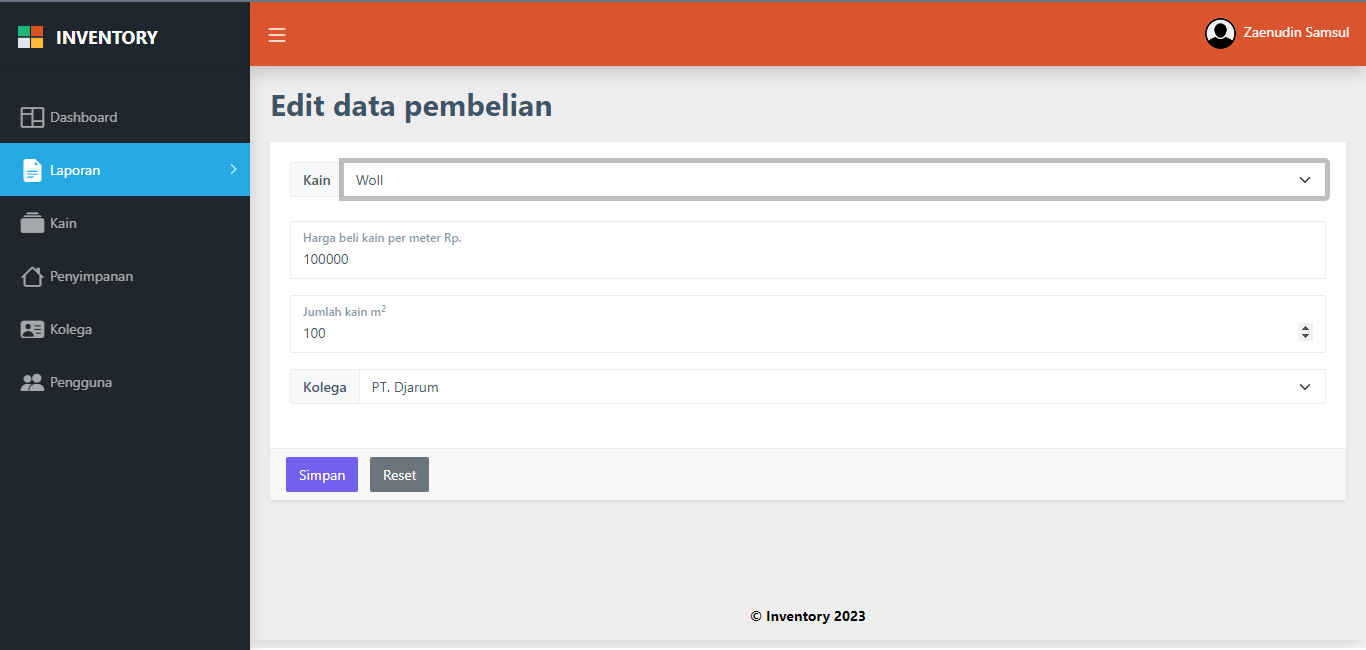


Gambar 5.15 Halaman tambah penjualan

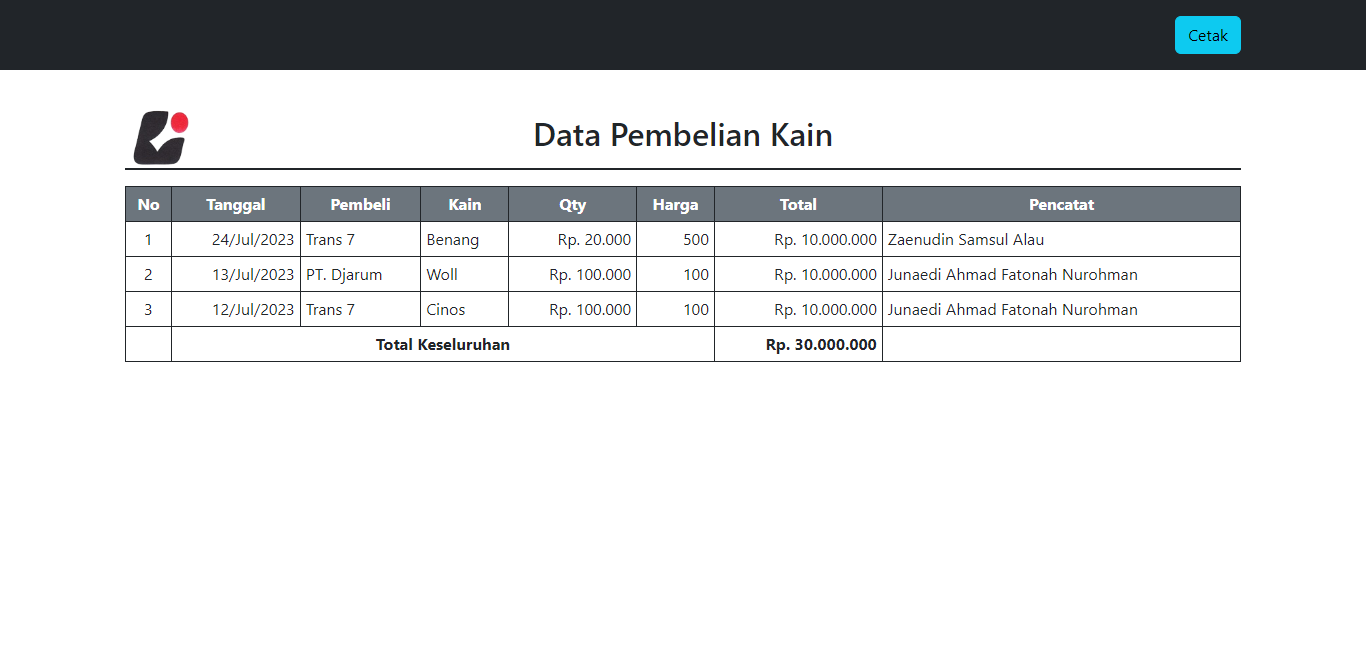


Gambar 5.16 Halaman tabel penjualan

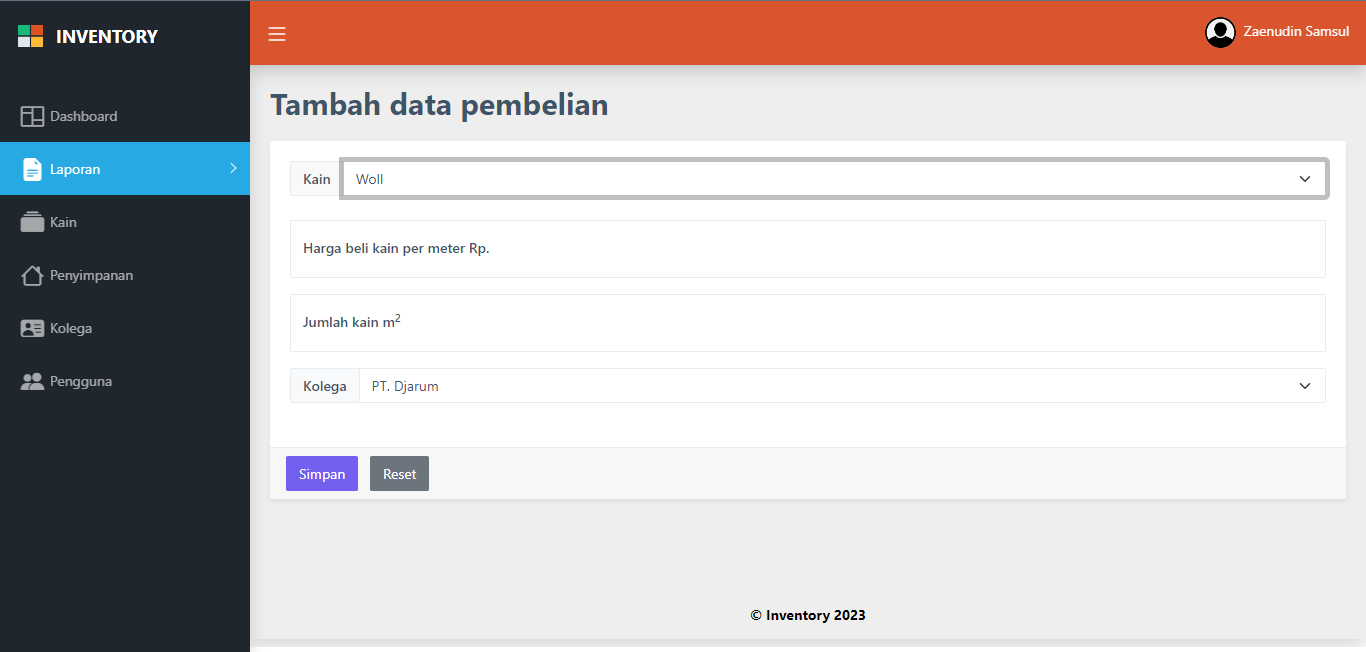
Halaman pembelian



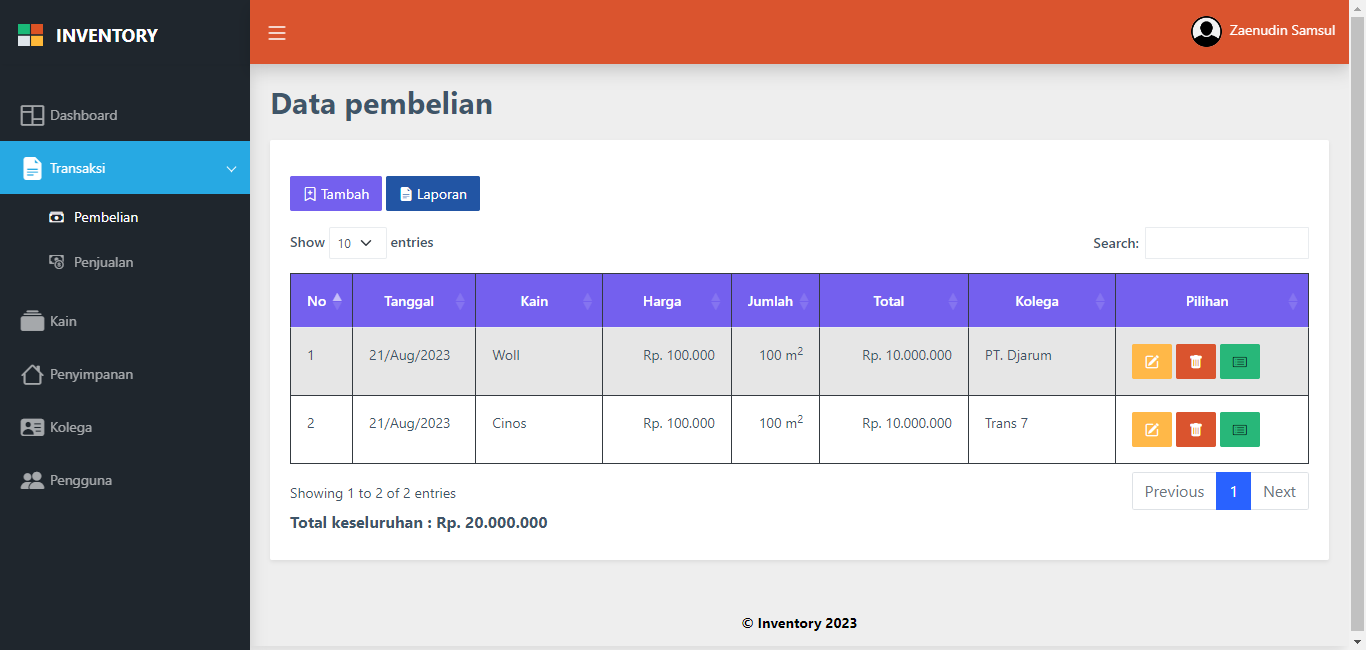
Gambar 5.17 Halaman edit pembelian



Gambar 5.18 Halaman laporan pembelian



Gambar 5.19 Halaman tambah pembelian



Gambar 5.20 Halaman tabel pembelian

Halaman pengguna

A screenshot of a computer

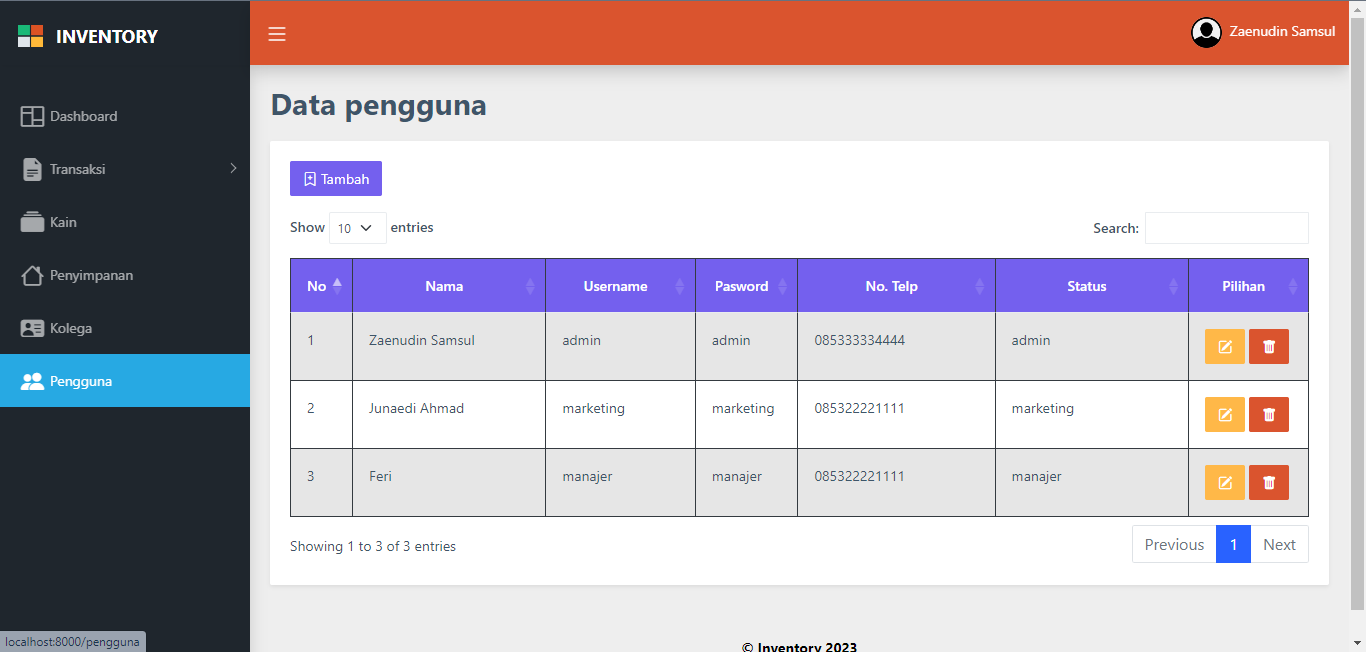
Description automatically generated

Gambar 5.21 Halaman edit pengguna

A screenshot of a computer

Description automatically generated

Gambar 5.22 Halaman tambah pengguna



Gambar 5.23 Halaman tabel pengguna

## Pengujian

Sebelum mengimplementasikan aplikasi di lapangan, proses pengujian sangat penting untuk mengidentifikasi potensi kesalahan. Dalam tahap pengujian, penulis memilih metode *black box*, yang memfokuskan pada hasil output dari sistem atau komponen tanpa memperhatikan mekanisme internalnya, hanya berdasarkan input yang diberikan dan kondisi eksekusi. Berikut adalah hasil dari pengujian yang telah dilakukan :

Tabel 5.1 Hasil pengujian

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Item Uji** | **Skenario Uji** | **Hasil yang diharapkan** | **Hasil** |
|  | Form *login* | Menginput *username* dan *password* | Aktor dapat masuk ke aplikasi inventori | Berhasil |
|  | Halaman *dashboard* | Klik *login* lalu masuk pada halaman *dashboard* | Aktor dapat mengakses halaman *dashboard* | Berhasil |
|  | Halaman tabel data kain | Klik menu kain | Aktor dapat mengakses halaman data kain | Berhasil |
|  | Tambah data kain | Klik tombol tambah pada menu kain | Aktor dapat menambah data kain | Berhasil |
|  | Edit data kain | Klik ikon edit pada menu kain | Aktor dapat mengedit data kain yang dipilih | Berhasil |
|  | Hapus data kain | Klik ikon hapus pada menu kain | Aktor dapat menghapus data kain yang dipilih | Berhasil |
|  | Halaman tabel data kolega | Klik menu kolega | Aktor dapat mengakses halaman data kolega | Berhasil |
|  | Tambah data kolega | Klik tombol tambah pada menu kolega | Aktor dapat menambah data kolega | Berhasil |
|  | Edit data kolega | Klik ikon edit pada menu kolega | Aktor dapat mengedit data kolega yang dipilih | Berhasil |
|  | Hapus data kolega | Klik ikon hapus pada menu kolega | Aktor dapat menghapus data kolega yang dipilih | Berhasil |
|  | Halaman tabel data penyimpanan | Klik menu penyimpanan | Aktor dapat mengakses halaman data penyimpanan | Berhasil |
|  | Tambah data penyimpanan | Klik tombol tambah pada menu penyimpanan | Aktor dapat menambah data penyimpanan | Berhasil |
|  | Edit data penyimpanan | Klik ikon edit pada menu penyimpanan | Aktor dapat mengedit data penyimpanan yang dipilih | Berhasil |
|  | Hapus data penyimpanan | Klik ikon hapus pada menu penyimpanan | Aktor dapat menghapus data penyimpanan yang dipilih | Berhasil |
|  | Halaman tabel data penjualan | Klik menu penjualan | Aktor dapat mengakses halaman data penjualan | Berhasil |
|  | Tambah data penjualan | Klik tombol tambah pada menu penjualan | Aktor dapat menambah data penjualan | Berhasil |
|  | Edit data penjualan | Klik ikon edit pada menu penjualan | Aktor dapat mengedit data penjualan yang dipilih | Berhasil |
|  | Hapus data penjualan | Klik ikon hapus pada menu penjualan | Aktor dapat menghapus data penjualan yang dipilih | Berhasil |
|  | Cetak data penjualan | Klik tombol print pada menu penjualan | Aktor dapat mencetak data penjualan sesuai dengan keinginan | Berhasil |
|  | Cetak surat jalan | Klik ikon surat jalan pada menu penjualan | Aktor dapat mencetak surat jalan penjualan | Berhasil |
|  | Halaman tabel data pembelian | Klik menu pembelian | Aktor dapat mengakses halaman data pembelian | Berhasil |
|  | Tambah data pembelian | Klik tombol tambah pada menu pembelian | Aktor dapat menambah data pembelian | Berhasil |
|  | Edit data pembelian | Klik ikon edit pada menu pembelian | Aktor dapat mengedit data pembelian yang dipilih | Berhasil |
|  | Hapus data pembelian | Klik ikon hapus pada menu pembelian | Aktor dapat menghapus data pembelian yang dipilih | Berhasil |
|  | Cetak data pembelian | Klik tombol print pada menu pembelian | Aktor dapat mencetak data pembelian sesuai dengan keinginan | Berhasil |
|  | Halaman tabel data pengguna | Klik menu pengguna | Aktor dapat mengakses halaman data pengguna | Berhasil |
|  | Tambah data pengguna | Klik tombol tambah pada menu pengguna | Aktor dapat menambah data pengguna | Berhasil |
|  | Edit data pengguna | Klik ikon edit pada menu pengguna | Aktor dapat mengedit data pengguna yang dipilih | Berhasil |
|  | Hapus data pengguna | Klik ikon hapus pada menu pengguna | Aktor dapat menghapus data pengguna yang dipilih | Berhasil |

# KESIMPULAN DAN SARAN



## Kesimpulan

Berdasarkan penelitian yang telah dilakukan penulis dengan judul Rancang Bangun Sistem Informasi Inventori Menggunakan Laravel 8 di PT. Cemara Kwangjin Tekstil. Kesimpulan yang penulis dapatkan dari penelitian ini yaitu :

Aplikasi ini memudahkan dalam sistem inventori yang sesuai dengan transaksi yang sebenarnya. Sesuai pula stok dan tempat penyimpanan kain sehingga hal tersebut membantu akan waktu dan kinerja untuk pekerja dalam mengetahui stok dan mengetahui tempat penyimpanan kain ketika dibutuhkan dan mudah pengambilan kain.

Aplikasi ini mencegah kesalahan dalam inputan dan stok kain yang sebenarnya karena akan terlihat pada perubahan data atau update yang telah diinput sehingga terlihat janggal. Dengan cara tersebut mencegah kesalahan inputan serta data kain tersebut bisa di konversi pada kertas atau pdf sesuai dengan kebutuhan.

Aplikasi ini memiliki fitur yang bisa mencetak laporan penjualan dan pembelian kain berdasarkan kebutuhan pengguna. Tentunya datanya yang akan dicetak sesuai dengan yang diperlukan, akurat perhitungan dan praktis.

## Saran

Setelah dijabarkan hasil penelitian ini, dapat disimpulkan bahwa aplikasi sistem informasi inventori yang berfungsi sebagai alat yang digunakan untuk mengelola inventori kain di PT. Cemara Kwangjin Tekstil menggunakan framework Laravel 8 dan Bootstrap 5, masih memiliki beberapa kekurangan yang perlu diperbaiki. Oleh karena itu, penulis memberikan beberapa saran sebagai pertimbangan untuk pengembangan sistem informasi di masa yang akan datang. Saran-sarannya yaitu :

Menjaga kelancaran dan kinerja optimal sistem tetap menjadi hal yang esensial. Proses pemeliharaan menjadi langkah krusial dalam mengidentifikasi dan memperbaiki potensi kesalahan atau masalah yang mungkin muncul pada *platform* aplikasi sistem informasi inventori ini.

Direkomendasikan untuk mempertimbangkan penambahan pilihan menu lain seperti menu supir dan kendaraan pengantar barang yang dapat memberikan kontribusi dalam pengelolaan inventori secara lebih holistik, terutama apabila digunakan dalam skenario yang lebih kompleks dari versi sebelumnya.

Disarankan untuk memperkaya fungsionalitas sistem dengan menambahkan fitur tambahan yang mampu memberikan nilai tambah, seperti opsi untuk mengonversi data menjadi format Excel atau fitur lain yang relevan. Dengan langkah ini, sistem akan semakin handal dan dapat diadaptasi dengan lebih baik sesuai dengan keperluan pengguna.

# DAFTAR PUSTAKA

Bintara, W. S. (2020a, Oktober 17). Pengertian *Microsoft Visio* beserta Fungsi, Manfaat, Fitur, Kelebihan. *Dianisa.com*. https://dianisa.com/pengertian-microsoft-visio/

Bintara, W. S. (2020b, November 8). Pengertian Google Chrome | Fitur, Kelebihan, Kekurangan. *Dianisa.com*. https://dianisa.com/pengertian-google-chrome/

Darmawan, D., & Hendra Permana, D. (2016). *DESAIN dan PEMROGRAMAN WEBSITE*. PT REMAJA ROSDAKARYA.

Gosal, R., & Rustam, A. (2022). Perancangan Sistem Informasi Inventori Berbasis Web Pada Gudang di PT. Spin Warriors. *Aisyah Journal Of Informatics and Electrical Engineering (A.J.I.E.E)*, *4*(1), Article 1. https://doi.org/10.30604/jti.v4i1.99

Intern, D. (2021a, Maret 9). *Apa itu Activity Diagram? Beserta Pengertian, Tujuan, Komponen*. Dicoding Blog. https://www.dicoding.com/blog/apa-itu-activity-diagram/

Intern, D. (2021b, Mei 18). *Contoh Use Case Diagram Lengkap dengan Penjelasannya*. Dicoding Blog. https://www.dicoding.com/blog/contoh-use-case-diagram/

K, Y. (2022, April 24). Pengertian MySQL, Fungsi, dan Cara Kerjanya (Lengkap). *Niagahoster Blog*. https://www.niagahoster.co.id/blog/mysql-adalah/

Kaban, R. (2019). *Bootstrap CSS Framework*. CV. ANDI Offset.

Maksum, M. A. (2022a, Mei 23). Pengertian XAMPP, Fungsi, dan Cara Menggunakannya. *Blog Dewaweb*. https://www.dewaweb.com/blog/apa-itu-xampp/

Maksum, M. A. (2022b, Juni 10). Apa itu Laravel? Pengertian, Fitur dan Kelebihannya. *Blog Dewaweb*. https://www.dewaweb.com/blog/apa-itu-laravel/

marketing. (2021, September 14). BELAJAR MENGENAL RANCANG BANGUN. *Alfabank Yogyakarta*. https://alfabankjogja.com/2021/09/14/belajar-mengenal-rancan-bangun/

Marshel Christian, Suparni, & Utami, L. A. (2021). Sistem Informasi Inventori Menggunakan Framework Laravel Pada CV. Grace Bhakti Utama Bogor. *JURNAL LARIK*, *1*. http://jurnal.bsi.ac.id/index.php/larik

Mubarok, A., Noor, C. M., & Sukajie, B. (2019). Sistem Informasi Inventori Dalam Mengolah Bahan Kayu Menjadi Produk Kusen di PB. Mekar Laksana. *Jurnal Informatika*, *6*(2), 305–309. https://doi.org/10.31311/ji.v6i2.5167

Mufida, E., Rahmawati, E., & Hertiana, H. (2019). Rancang Bangun Sistem Informasi Inventori Pada Salon Kecantikan. *Jurnal Mantik Penusa*, *3*(3). https://e-jurnal.pelitanusantara.ac.id/index.php/mantik/article/view/671

Rahmalia, N. (2020, Desember 18). *Balsamiq*, Tool Merancang Wireframe yang Ramah bagi Designer Pemula. *Glints Blog*. https://glints.com/id/lowongan/*Balsamiq*-adalah/

Rosa, A. S., & Shalahuddin, M. (2018). *Rekayasa Perangkat Lunak Terstruktur & Berorientasi Objek (edisi revisi)*. Informatika Bandung.

Setiawan, R. (2021, Oktober 12). *Memahami Class Diagram Lebih Baik*. Dicoding Blog. https://www.dicoding.com/blog/memahami-class-diagram-lebih-baik/

Sidik, B. (2017). *Pemrograman Web Dengan PHP7*. Informatika.

Sidiq, B., & Pohan, H. I. (2014). *Pemrograman Web Dengan HTML (Revisi Kelima)*. Informatika.

Tasari, G. (2021, April 25). *Mengenal Visual Studio Code*. Gamelab.ID. https://www.gamelab.id/news/468-mengena-visual-studio-code

# LAMPIRAN

Lampiran 1 : Hasil Penelitian Terdahulu

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Judul Artikel | Masalah | Metode | Solusi |
| 1 | Sistem Informasi Inventory Dalam Mengolah Bahan Kayu Menjadi Produk Kusen di PB. Mekar Laksana | Permasalahan perusahaan adalah permasalahan inventory, dimana ketidaksamaan stok bahan baku yang tersedia dengan data perusahaan. Kemudian tidak efisiennya waktu dalam pencatatan transaksi. | Metode *waterfall* dengan langkah : analisa kebutuhan perangkat lunak, desain, pengkodean, pengujian dan pemeliharaan. | Pembuatan aplikasi berbasis web dimana untuk data inventory dapat diakses dengan mudah bukan hanya secara offline saja melainkan juga secara online. |
| 2 | Sistem Informasi Inventory Menggunakan Framework Laravel Pada CV. Grace Bhakti Utama Bogor | Kesalahan saat penginputan secara manual akan menghasilkan laporan yang tidak akurat, hal ini merupakan permasalahan yang harus diselesaikan dalam perusahaan agar sistem berjalan secara efektif dan efisien. | Metode *waterfall* dengan tahap : analisis kebutuhan perangkat lunak, desain, pembuatan kode program, pengujian dan pendukung(support) | Sistem informasi dapat mempercepat waktu pencatatan penginputan data serta dapat meminimalisir kesalahan yang terjadi saat pencatatan data. Stok bisa dilihat dalam sistem, proses pencarian data barang atau stok barang juga menjadi lebih cepat, penyimpanan dan laporan barang masuk dan keluar juga menjadi lebih tertata dan efesien. |
| 3 | Sistem Informasi Inventory P.O.S (Point Of Sales) Berbasis Web Pada Counter Cellular | Proses input, output dan pelaporan barang hanya secara manual (masih menggunakan kertas), Seringkali terjadi adanya barang keluar ataupun barang masuk yang tidak tercatat datanya. | Metode extreme programming dengan langkah perencanaan, perancangan, coding atau implementasi dan pengujian | Perancangan dan pembangunan Sistem Informasi Inventory P.O.S (Point of Sales) berupa Diagram Konteks dan UML Diagram (Use Case Diagram, Activity Diagram, Class Diagram, dan Sequence Diagram), dengan hasil pengujian black box yang dilakukan telah beroperasi dengan baik sesuai kebutuhan owner retail. |
| 4 | Sistem Informasi Inventory (SITORY) Berbasis Web Dengan Metode Framwork For The Application System Thinking (FAST ) | JATISI (Jurnal Teknik Informatika dan Sistem Informasi) | Pengelolaan persediaan barang masih dilakukan secara manual dengan melakukan pencatatan di dalam buku besar. Hal ini menyebabkan beberapa permasalahan seperti stok barang tidak sesuai, pengecekan dan pencarian barang memerlukan waktu yang lama, sering terjadi stok yang kadaluarsa. | Metode Framework For The Application System Thinking (FAST dimulai tahap mendefinisikan lingkup, menganalisis masalah, analisis kebutuhan, desain logis, menganalisis keputusan, desain fisik & integrasi, kontruksi & pengujian, dan instalasi & pengiriman | Sistem Informasi Inventory (SITORY) berbasis web ini dapat membatu Owner, Bagian Gudang dan Kasir dalam mengelola data barang, data supplier, data karyawan, PO, Permintaan barang dan laporan secara terkomputerisaasi agar lebih tertata dan sistematis sehingga bisa menghindari kesalahan hitung Stok barang yang tidak sesuai, pengecekan serta pencarian barang memerlukan waktu yang lama dan stok barang yang kadaluarsa |
| 5 | Rancang Bangun Sistem Informasi Inventory Pada Salon Kecantikan | Penanganan data yang masih dilakukan dengan menggunakan sistem manual mengakibatkan sering terjadinya kesalahan yang ada. | Model Waterfall yaitu analisa kebutuhan software, design, code generation, testing, dan support, | Aplikasi perancangan sistem informasi inventory yang dapat menghasilkan sebuah laporan stok barang dan akan mempermudah proses monitoring stok barang masuk dan barang keluar. |
| 6 | Sistem Informasi Persediaan Jamu Madura Menggunakan Metode Waterfall | Pelaku usaha dagang jamu masih menggunakan cara manual untuk mengelola data persediaan jamu Madura dan butuh banyak waktu, hasilnya juga kurang akurat. | Metode waterfall yaitu Analisa, Desain, Implementasi, Pengujian, dan Pemeliharaan | Untuk membuat hal tersebut menjadi lebih cepat dan mudah maka perlu membangun sebuah Sistem Informasi Persediaan Jamu Madura |
| 7 | Sistem Informasi Inventori Barang Habis Pakai di Akademi Komunitas Negeri Pacitan Berbasis Web | Pengelolaan barang dilakukan dengan manual yang dapat mengakibatkan terjadinya kesalahan dengan perbedaan pencatatan laporan dengan stok fisik. | Metode waterfall yaitu Analisis dan Definisi Kebutuhan, Desain Sistem, Rancangan, Implementasi dan unit testing, Integrasi dan system testing dan Operation and maintenance | Sistem informasi yang dapat menghilangkan kesalahan pada saat input data serta pembuatan laporan sesuai informasi yang ada dan sesuai kebutuhan. |
| 8 | Sistem Informasi Stok Barang Berbasis Website Pada Koperasi Sekolah Terpadu Darul Hikam Bandung | Ketidak sesuaiannya jumlah stok yang ada di gudang dan jumlah stok yang tercatat di buku stok sehingga mengakibatkan permasalahan pada saat melakukan Stok Opname. | Metode waterfall dengan langkah langkah Requirements Analysis, System Design, Implementation dan Testing | Sistem informasi stok barang berbasis website. Dengan menggunakan metode pendekatan kualitatif dan untuk pengembangan sistemnya akan menggunakan metode waterfall, aplikasi akan dibuat meggunakan framework laravel dan untuk database nya akan menggunakan MySQL. |
| 9 | Sistem Informasi Inventory Gudang Dalam Ketersediaan Stok Barang Menggunakan Metode Buffer Stok | Belum ada sistem yang mengelola data tentang stok persediaan barang untuk itu perusahaan. | Masalah, pendekatan, perancangan, implementasi dan pengujian | Sistem informasi inventory gudang dalam ketersediaan stok barang menggunakan metode buffer stock berbasis web |
| 10 | Perancangan Sistem Informasi Inventory Safety Tools pada PT. Kumpul Mas Abadi | Pengelolaan inventory saat ini mengalami kendala karena masih dilakukan secara manual | Waterfall terdiri dari lima tahap, yaitu communication, planning, modelling, contruction dan deployment | Sistem informasi inventori berbasis website agar pengelolaan persediaan menjadi lebih efektif dan efisien yang mendukung peningkatan produktivitas |

Lampiran 2 : Hasil Wawancara

Narasumber :

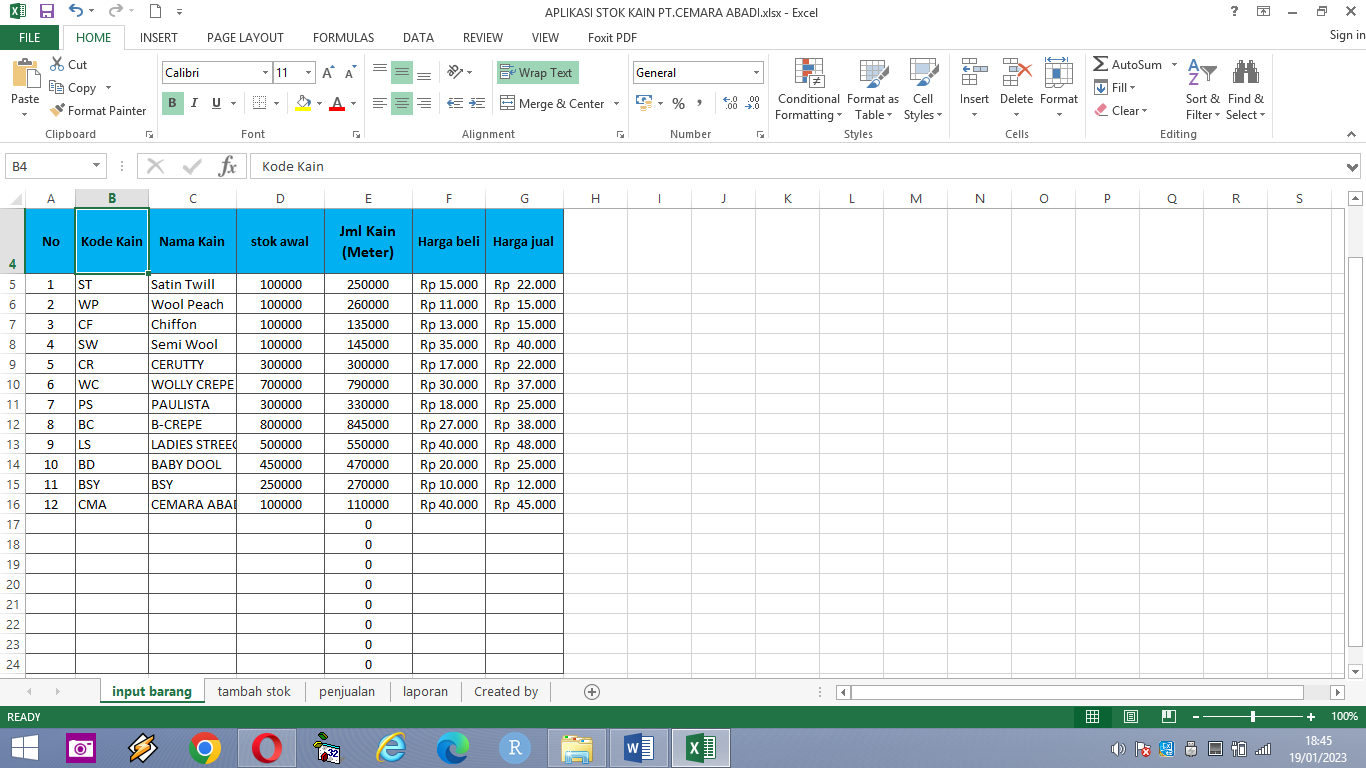
|  |  |  |
| --- | --- | --- |
| Nama  Jabatan  Tanggal  Alamat  Tema wawancara | :  :  :  :  : | Muhammad Iqbal  Kepala Divisi Gudang PT. Cemara Abadi Tekstil  25 Mei 2023  Kantor PT. Cemara Abadi Tekstil  Sistem Inventori PT. Cemara Abadi Textile |

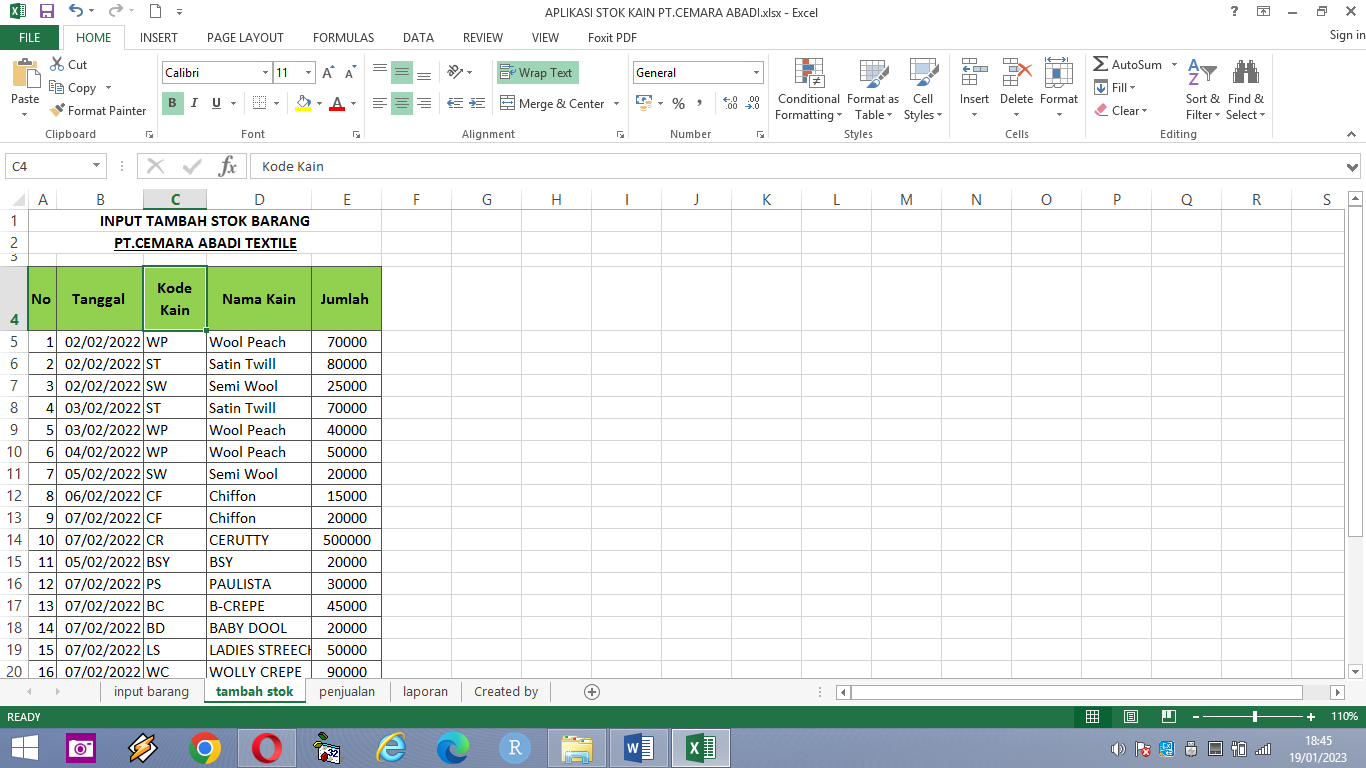
|  |  |
| --- | --- |
| **Pertanyaan** | **Jawaban** |
| Apakah sistem Inventori di perusahaan ini masih terdapat kendala mengenai pendataan barang yang ada ? | Masih terdapat beberapa kekurangan dalam sistem inventori barang dan pendataannya |
| Kekurangan atau kendala seperti apa yang bapak hadapi dalam menangani sistem Inventori di perusahaan ini? | Seperti dalam pendataan kain atau stok yang ada di lapangan yang diminta oleh bagian marketing penjualan, terkadang data kain atau barang yang tersedia tidak sesuai dengan yang ada di lapangan atau yang tersimpan di gudang |
| Bisa dijelaskan lebih spesifik atau lebih jelas pak, kendala Sistem Inventori yang terjadi seperti apa yang bapak tadi maksudkan? | Contoh data kain Wool Peach (WP) yang diminta oleh marketing semisal minta 100 yard, ternyata data yang di gudang kurang, atau tidak sesuai dengan data yang ada di laporan, meskipun kurangnya tidak banyak namun hal itu mengganggu proses penjualan atau pelayanan ke konsumen, karena hal itu mengkibatkan konsumen harus menunggu kain yang diminta tersedia atau sesuai dengan apa yang diminta oleh konsumen |
| Mengapa pak hal tersebut bisa terjadi ? | Kemungkinan hal tersebut terjadi karena adanya double data, atau data yang tertumpuk saat proses input barang, sehingga mengakibatkan data tersebut tidak sesuai dengan yang ada di lapangan |
| masalah apalagi pak yang bapak hadapi selama mengurus dan mengelola Sistem Inventori atau pergudangan di perusahaan ? | Misalnya seperti tata letak, atau denah penyimpanan kain yang terkadang tidak sesuai dan tidak semestinya dalam peletakannya |
| Contohnya pak, bisa bapak jelaskan pak ? | Misalnya di laporan kain jenis A harusnya disimpan di kolom atau di jajaran kain A, namun kadang di lapangan kain tersebut sering tercampur atau tercecer karena alasan letak penyimpanan yang penuh atau keterbatasan dalam penyimpanan |
| Sistem seperti apa yang digunakan selama ini di perusahaan ini ? | kita pakai atau masih kita gunakan dalam Sistem Inventori ini masih menggunakan sistem operasi berbasis Excel, yang saat dalam proses pencatatannya masih menggunakan metode semi manual |
| Apakah kemungkinan data tumpuk atau double data tersebut bisa terjadi saat proses pencatatan manual seperti itu? | jika admin melakukan kesalahan saat pencatatan atau input manual ke dalam excel tadi jika data tersebut kembali di inputkan, maka secara otomatis data tersebut akan menjadi data double atau data tumpuk |
| Bagaimana bapak menanganinya pada saat ini ? | sementara ini kami hanya melakukan pengecekan rutin saja untuk mengatasi masalah tersebut, dan melakukan koordinasi dengan team marketing atau penjualan agar data yang ada selalu sesuai dengan permintaan yang ada |
| Izin saya ingin melakukan atau sedikit membantu pekerjaan bapak yang ada saat ini dengan membuat prototype dari rancang bangun Sistem Inventori sebagai bahan dasar penelitian dari skripsi saya, dan semoga berguna juga untuk perusahaan dalam menanggulangi masalah yang ada di perusahaan ini | Saya izinkan anda untuk melakukan penelitian dari masalah yang ada di Sistem Inventori di perusahaan ini sebagai bahan dasar penelitian skripsi saudara berdasarkan dari sedikit macam masalah yang saya sudah sebutkan tadi |

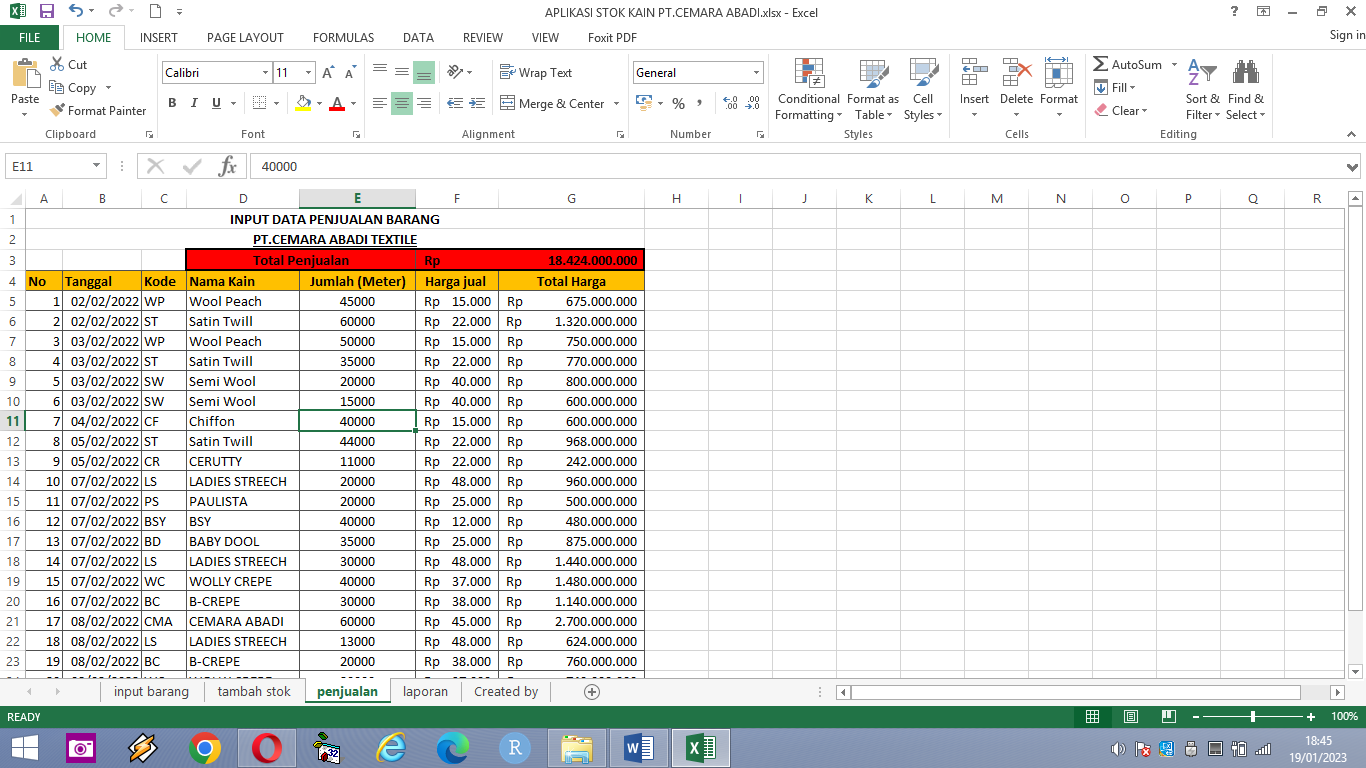
|  |
| --- |
| Kepala Divisi Gudang PT. Cemara Abadi Tekstil  Muhammad Iqbal |

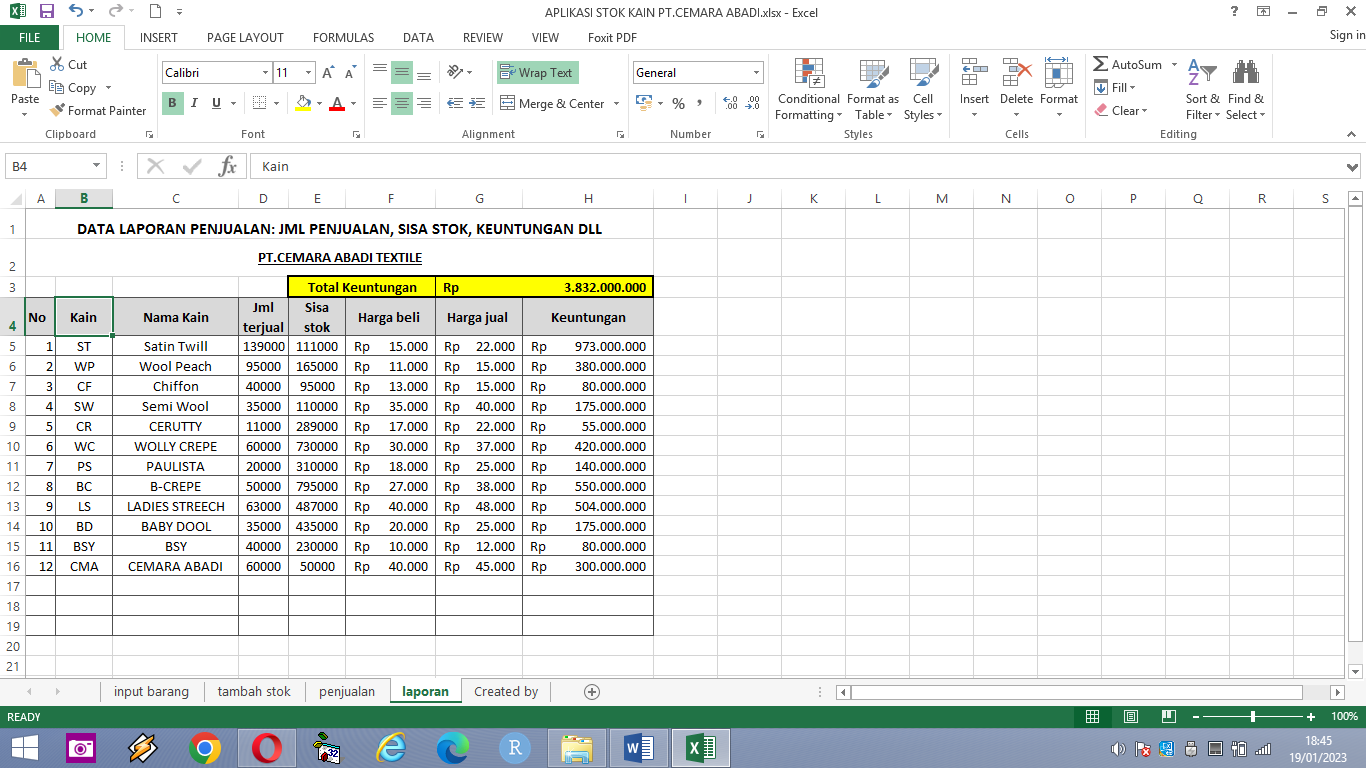
Lampiran 3 : Observasi

1. Data yang sedang digunakan









1. Pegawai diperusahaan PT. Cemara Kwangjin Tekstil

A group of young men standing together

Description automatically generated with medium confidence





1. Kain diperusahaan PT. Cemara Kwangjin Tekstil

A picture containing bag, accessory, art, sculpture

Description automatically generated

A picture containing scene, retail, convenience store, supermarket

Description automatically generated

A picture containing text, pipe, outdoor, stack

Description automatically generated

A picture containing text, funeral, letter, book

Description automatically generated

A picture containing plastic bag, plastic wrap, plastic, bag

Description automatically generated

A picture containing funeral, snow, covered, indoor

Description automatically generated