**IMPLEMENTASI SISTEM INFORMASI KEUANGAN DENGAN METODE PERPETUAL DAN PERIODIK   
PADA CV PANEN REJO SEMARANG**

# **TUGAS AKHIR**



**DISUSUN:**

**MUHAMMAD AGUS NADHIF**

**G.211.13.0019**

**PROGRAM STUDI SI – TEKNIK INFORMATIKA**

**JURUSAN TEKNOLOGI INFORMASI**

**FAKULTAS TEKNOLOGI INFORMASI DAN KOMUNIKASI**

**UNIVERSITAS SEMARANG**

**2017**

# **PERNYATAAN PENULIS TUGAS AKHIR**

**DENGAN JUDUL**

**IMPLEMENTASI SISTEM INFORMASI KEUANGAN DENGAN METODE PERPETUAL DAN PERIODIK**

**PADA CV PANEN REJO SEMARANG**

Dengan ini saya:

NAMA : MUHAMMAD AGUS NADHIF

NIM : G.211.13.0019

PROGRAM STUDI : TEKNIK INFORMATIKA

“Saya menyatakan dan bertanggungjawab dengan sebenarnya bahwa Tugas Akhir (TA) ini adalah hasil karya saya sendiri kecuali cuplikan dan ringkasan yang masing-masing telah saya jelaskan sumbernya. Jika pada waktu selanjutnya ada pihak lain yang mengklaim bahwa Tugas Akhir (TA) ini sebagai karyanya, yang disertai dengan bukti-bukti yang cukup, maka saya bersedia untuk dibatalkan gelar Sarjana Komputer saya beserta segala hak dan kewajiban yang melekat pada gelar tersebut”.

Semarang, Mei 2018

Penulis

Muhammad Agus Nadhif

NIM.G.211.13.0019

# **PERSETUJUAN TUGAS AKHIR**

**DENGAN JUDUL**

**IMPLEMENTASI SISTEM INFORMASI KEUANGAN DENGAN METODE PERPETUAL DAN PERIODIK**

**PADA CV PANEN REJO SEMARANG**

OLEH

NAMA : MUHAMMAD AGUS NADHIF

NIM : G.211.13.0019

DISUSUN DALAM RANGKA MEMENUHI SYARAT GUNA

MEMPEROLEH GELAR SARJANA KOMPUTER

PROGRAM STUDI S1 – TEKNIK INFORMATIKA

JURUSAN TEKNOLOGI INFORMASI

FAKULTAS TEKNOLOGI INGORMASI DAN KOMUNIKASI

UNIVERSITAS SEMARANG

TELAH DIPERIKSA DAN DISETUJUI

SEMARANG,

KETUA PROGRAM STUDI PEMBIMBING TUGAS AKHIR   
 S1-TEKNIK INFORMATIKA

April Firman Daru, S.Kom, M.Kom Whisnumurti Adhiwibowo,S.T.,M.Kom   
 NIS. 06557003102133 NIS. 06557003102137

# **PENGESAHAN TUGAS AKHIR**

**DENGAN JUDUL**

**IMPLEMENTASI SISTEM INFORMASI KEUANGAN DENGAN METODE PERPETUAL DAN PERIODIK**

**PADA CV PANEN REJO SEMARANG**

OLEH

NAMA : MUHAMMAD AGUS NADHIF

NIM : G.211.13.0019

Telah diujikan dan dipertahankan dihadapan Dewan Penguji pada Sidang Tugas Akhir (TA) Hari ………… tanggal …………………… pandangan kami, Tugas Akhir ini memadai dari segi kualitas maupun kuantitas untuk tujuan penganugerahan gelar Sarjana Komputer (S.Kom)

Semarang, ……………………

Dewan penguji,

KETUA TIM PENGUJI

……………………………………

NIS.

PENGUJI PENDAMPING I PENGUJI PENDAMPING II

……………………………….. ………………………………

NIS. NIS.

# **ABSTRACT**

CV Panen Redjo Semarang is a retail store that sells various agricultural production facilities and agricultural supporting infrastructure as well as for plant lovers. Where all the financial statements and store records are still manuals written in books, and often trouble when looking for data perpetual and periodic reports and must re-analyze to see the details of the report. This happens because less use of computer applications or information systems to manage reports.

Writing methodology used include primary and secondary data, with data collection method in the form of literature study, observation and interview on the owner of CV Panen Redjo, in system implementation by using web apps based.

Based on these descriptions, the need for an efficient system or application to manage financial statements between perpetual and periodic methods, so that in report management easier and efficient.

*Keywords: Financial Information System, Perpetual and Periodic Report,*

# **ABSTRAK**

CV Panen Redjo Semarang merupakan toko retail yang menjual berbagai sarana produksi pertanian dan prasarana penunjang pertanian maupun untuk pencinta tanaman. Dimana semua laporan keuangan dan catatan toko yang dilakukan masih bersifat manual ditulis dalam buku-buku, dan sering terjadi kesusahan ketika mencari data laporan perpetual maupun periodik dan harus menganalisa ulang untuk melihat detail laporan. Ini terjadi karena kurang memanfaatkan aplikasi komputer atau sistem informasi untuk mengelola laporan.

Metodologi penulisan yang digunakan meliputi data primer dan sekunder, dengan metode pengumpulan data berupa studi kepustakaan, observasi dan wawancara pada pemilik CV Panen Redjo, dalam implementasi sistem dengan menggunakan berbasis *web apps*.

Berdasarkan uraian tersebut maka dibutuhkannya sebuah sistem atau aplikasi yang efisen untuk mengelola laporan keuangan antara metode perpetual dan periodik, sehingga dalam pengelolaan laporan lebih mudah dan efisien.

*Kata Kunci: Sistem Informasi Keuangan, Laporan Perpetual dan Periodik,*

Pembimbing Tugas Akhir

Whisnumurti Adhiwibowo,S.T.,M.Kom  
 NIS. 06557003102137

# **KATA PENGANTAR**

Puji syukur penulis panjatkan kepadaTuhan Yang Maha Esa Allah SWT yang telah melimpahkan Rahmat, Taufik serta Hidayah-Nya, sehingga penulis dapat melaksanakan Kerja Praktek serta dapat menyelesaikan laporannya tepat waktu dan tanpa adanya halangan apapun. Dalam pelaksanaan kerja praktek dan penyusunan laporan, penulis mendapat banyak bantuan, dukungan dan bimbingan dari berbagai pihak. Oleh Karena itu dalam kesempatan ini penulis ingin menyampaikan ucapan terimakasih kepada:

1. Bapak Andy Kridasusila, SE, MM selaku Rektor Universitas Semarang.
2. Bapak Susanto, S.Kom, M.Kom. selaku Wali Dosen dan Dekan Fakultas Teknologi Informasi Dan Komunikasi Universitas Semarang.
3. Bapak April Firman Daru, S.Kom, M.Kom. selaku Ketua Prgram Studi S1 – Teknik Informatika Fakultas Teknologi Informasi Dan Komunikasi Universitas Semarang.
4. Bapak Whisnumurti Adhiwibowo,S.T.,M.Kom selaku dosen pembimbing yang telah membantu dan memberikan bimbingan kepada penulis sehingga penulis dapat menyesesaikan laporan ini.
5. Keluarga, Teman dan pihak CV. Panen Rejo yang selalu memberikan dukungan, arahan dan bimbingan.

Akhir kata penulis menyadari bahwa pelaksanaan Tugas Akhir dan penyusunan laporan ini masih belum sempurna. Oleh karena itu, kritik dan saran sangat diharapkan, mohon maaf atas kesalahan penulisan kata dan kalimat. Semoga penyusunan laporan ini bermanfaat.

Semarang, Mei 2018

Penulis

# **DAFTAR ISI**

Halaman Judul ………………………………………………………………. i

Halaman Pengesahan ………………………………………………………... ii

Kata Pengantar ………………………………………………………………. iii

Daftar Isi …………………………………………………………………….. iv

Daftar Gambar ………………………………………………………………. vi

Daftar Tabel …………………………………………………………………. viii

BAB I PENDAHULUAN ……………………………………………… 1

1. Latar Belakang …………………………………………………….. 1
2. Batasan Masalah …………………………………………………... 1
3. Tujuan Kerja Praktek ……………………………………………… 2
4. Metodologi Pelaksanaan …………………………………………... 2
5. Sistematika Penulisan ……………………………………………... 3

BAB II TINJAUAN UMUM …………………………………………… 5

1. Sejarah CV Panen Rejo ……………………………………... 5
2. Visi, Misi, dan Value CV Panen Rejo ………………………. 5
3. Tentang CV Panen Rejo …………………………………….. 6
4. Struktur Organisasi CV Panen Rejo ………………………… 9
5. Tugas dan Wewenang Masing-Masing Jabatan …………………... 10

BAB III LANDASAN TEORI ………………………………………….. 12

1. Pengertian Aplikasi ……………………………………………….. 12
2. Inventory atau Persediaan ………………………………………… 12
3. Penjualan ………………………………………………………….. 13
4. United Modeling Language (UML) ………………………………. 15
5. Waterfall …………………………………………………………... 23

BAB IV ANALISA DAN PERANCANGAN …………………………... 25

1. Analisa Sistem yang Berjalan ……………………………………... 25
2. Analisa Kebutuhan ………………………………………………... 25
3. Analisa Desain …………………………………………………….. 26
4. Perancangan Database …………………………………………….. 39
5. Perancangan Antar Muka …………………………………………. 41

BAB V PENUTUP ……………………………………………………... 52

1. Kesimpulan ………………………………………………………... 52
2. Saran ………………………………………………………………. 52

DAFTAR PUSTAKA ………………………………………………………. 53

# **DAFTAR GAMBAR**

Gambar 2.1 Peta Lokasi CV Panen Rejo ……………………………… 6

Gambar 2.2 Gerbang Toko CV Panen Rejo …………………………... 7

Gambar 2.3 Halaman Toko CV Panen Rejo ………………………….. 7

Gambar 2.4 Toko CV Panen Rejo …………………………………….. 8

Gambar 2.5 Sarana Tanaman Perkebunan CV Panen Rejo …………... 8

Gambar 2.6 Prasarana Pertanian dan Perkebunan CV Panen Rejo …... 9

Gambar 2.7 Struktur Oraganisasi CV Panen Rejo ……………………. 9

Gambar 3.1 Metode *Waterfall* ………………………………………………. 23

Gambar 4.1 *Use Case Diagram* ……………………………………………... 27

Gambar 4.2 *Activity diagram* mengelola penjualan ………………………… 30

Gambar 4.3 *Activity diagram* mengelola data produk ………………………. 31

Gambar 4.4 *Activity diagram* mengelola laporan …………………………… 32

Gambar 4.5 *Activity diagram* cetak laporan ………………………………… 33

Gambar 4.6 Activity diagram analisa pendapatan ………………………….. 33

Gambar 4.7 *Activity diagram* mengelola pengguna ………………………… 34

Gambar 4.8 *Activity diagram* *edit password* ………………………………... 35

Gambar 4.9 *sequence diagram* penjualan ………………………………….. 36

Gambar 4.10 *sequence diagram* produk …………………………………….. 36

Gambar 4.11 *sequence diagram* laporan ……………………………………. 37

Gambar 4.12 *sequence diagram* analisa pendapatan ……………………….. 37

Gambar 4.13 *sequence diagram* pengguna …………………………………. 38

Gambar 4.14 *class diagram* ………………………………………………… 38

Gambar 4.15 Gambar Rancangan Masuk Aplikasi …………………………. 42

Gambar 4.16 Gambar Rancangan Dashbard ………………………………... 42

Gambar 4.17 Gambar Rancangan Data Kategori …………………………… 43

Gambar 4.18 Gambar Rancangan Tambah Data Kategori ………………….. 43

Gambar 4.19 Gambar Rancangan Detail Data Kategori ……………………. 44

Gambar 4.20 Gambar Rancangan Edit Data Kategori ……………………… 44

Gambar 4.21 Gambar Rancangan Data Produk …………………………….. 45

Gambar 4.22 Gambar Rancangan Tambah Data Produk …………………… 45

Gambar 4.23 Gambar Rancangan Detail Data Produk ……………………... 46

Gambar 4.24 Gambar Rancangan Edit Data Produk ……………………….. 46

Gambar 4.25 Gambar Rancangan Data *Inventory* ………………………….. 47

Gambar 4.26 Gambar Rancangan Update Data *Inventory* ………………….. 47

Gambar 4.27 Gambar Rancangan Data Laporan …………………………… 48

Gambar 4.28 Gambar Rancangan Analisa dan Laporan ……………………. 48

Gambar 4.29 Gambar Rancangan Cetak Data Laporan …………………….. 49

Gambar 4.30 Gambar Rancangan Data Pengguna ………………………….. 49

Gambar 4.31 Gambar Rancangan Tambah Data Pengguna ………………… 50

Gambar 4.32 Gambar Rancangan Detail Data Pengguna …………………... 50

Gambar 4.33 Gambar Rancangan Edit Data Pengguna …………………….. 51

# **DAFTAR TABEL**

Tabel 3.1 Notasi-Notasi *Use Case Diagram* ………………………………... 16

Tabel 3.2 Notasi-Notasi *Class Diagram* ……………………………………. 17

Tabel 3.3 Notasi-notasi *Activity Diagram* …………………………………... 19

Tabel 3.4 Notasi-notasi *Sequence Diagram* ………………………………… 21

Tabel 4.1 Keterangan aktor dalam perancangan aplikasi …………………...

*inventory* dan penjualan ………………………………………..... 27

Tabel 4.2 Sekenario *Use Case* Mengelola Penjualan ……………………….. 27

Tabel 4.3 Sekenario *Use Case* Mengelola Data Produk ……………………. 28

Tabel 4.4 Sekenario *Use Case* Mengelola Laporan ………………………… 28

Tabel 4.5 Sekenario *Use Case* Analisa Perkembangan ……………………... 30

Tabel 4.6 Sekenario *Use Case* Mengelola Pengguna ……………………….. 30

Tabel 4.7 Perancangan Tabel Produk ……………………………………… 39

Tabel 4.8 Perancangan Tabel Kategori ……………………………………. 39

Tabel 4.11 Perancangan Tabel Penjualan …………………………………… 40

Tabel 4.12 Perancangan Tabel Pembayaran ………………………………… 40

Tabel 4.13 Perancangan Tabel Laporan …………………………………….. 41

Tabel 4.14 Perancangan Tabel Pengguna …………………………………… 41

# **BAB I**

# **PENDAHULUAN**

1. **Latar Belakang**

Sistem informasi merupakan salah satu hal terpenting dalam suatu perusahaan. Dengan adanya sistem informasi maka organisasi atau perusahaan dapat menjamin kualitas informasi yang disajikan dan dapat mengambil keputusan berdasarkan informasi tersebut. Seiring perkembangan teknologi maka kebutuhan akan informasi yang cepat, tepat dan akurat sangat diperlukan. Karena itu, keberadaan sistem informasi sudah menjadi kebutuhan mutlak bagi perusahaan dalam menjalankan proses bisnisnya.

Salah satu siklus pada sistem informasi keuangan adalah siklus pendapatan yang terdiri dari penjualan, piutang dagang, dan penerimaan kas yang akan memberikan pendapatan bagi perusahaan atas penjualannya. Penjualan merupakan salah satu kegiatan utama yang dilakukan perusahaan sehari-hari untuk memperoleh laba yang akan digunakan untuk pembiayaan aktifitas perusahaan dan mengembangkan perusahaan.

Penjualan yang dilakukan oleh perusahaan akan berkaitan dengan jumlah persediaan di gudang karena ketika melakukan penjualan maka jumlah persediaan di gudang pun akan berkurang. Oleh karena itu perusahaan harus memesan kembali barang yang dijual sebelum kehabisan stock. Kegiatan penjualan dapat terjadi dengan dua cara yaitu tunai dan kredit. Penjualan tunai secara umum dilakukan melalui cara yang lebih mudah dibandingkan penjualan kredit karena pembayaran pada penjualan tunai dapat diterima langsung oleh perusahaan. Penjualan kredit secara umum menimbulkan piutang usaha perusahaan sehingga memerlukan proses penagihan untuk mendapatkan kas. Oleh karena itu sistem informasi akuntansi sangat diperlukan dalam mengendalikan siklus pendapatan dan persediaan agar berjalan dengan lancar dan semestinya.

Salah satu cara untuk meningkatkan strategi perkembangan perusahan adalah dengan menggunakan fasilitas Sistem Informasi berguna untuk mengumpulkan, mencatat, menyimpan, memelihara, dan mengolah data-data dalam proses transaksi akuntansi yang rutin sehingga menghasilkan informasi akuntansi dan keuangan. Informasi yang dihasilkan tersebut dibutuhkan oleh perusahaan dalam pengambilan keputusan, membuat laporan internal dan eksternal, merencanakan strategi untuk dapat bersaing dengan perusahaan lain, dan membuat pengendalian internal. Jika tidak ada system informasi akuntansi maka perusahaan akan mengalami kesulitan dalam mengendalikan aktivitas-aktivitas yang terjadi sehingga mempengaruhi kondisi perusahaan.

CV. Panen Redjo adalah toko retail yang bergerak dibidang penjualan produk sarana produksi pertanian dan bertujuan untuk memenuhi kebutuhan pertanian, perkebunan, dan hobiis tanaman, serta berbagai obat – obatan dan hasil olahan.

Masalah yang terkait pada toko adalah dimana semua laporan penjualan yang dicatat hanya total penjualan dari setiap pembelian konsumen tanpa mengetahui produk apasaja yang terjual tiap harinya dan tidak ada pencatatan laporan stok barang, serta laporan penjualan harian sudah menggunakan Microsoft excel dalam pendataannya, dimana proses tersebut kurang efektif karena dibutuhkan ketelitian dalam pendataan dengan jumlah penjualan barang yang bisa mengakibatkan terjadinya redundancy data (data ganda) dalam pembuatan laporan.

Dengan permasalahan yang didapat, untuk memenuhi kebutuhan yaitu dengan menggunakan aplikasi pencatatan keuangan, berdasarkan uraian tersebut penulis mengambil pembahasan mengenai “Implementasi Sistem Informasi Keuangan dengan Metode Sistem Perpetual dan Periodik pada CV Panen Redjo Semarang” yang di harapkan menjadi solusi untuk dapat membantu CV Panen Redjo dalam pengelolaan data keuangan.

1. **Rumusan Masalah**

Berdasarkan latar belakang masalah diatas maka dapat dirumuskan yaitu bagaimana menghasilkan Sistem Informasi Keuangan untuk memcatat secara perpetual dan periodik, agar dapat membantu dalam laporan keuangan perusahaan, dan mempermudah pemilik untuk menganalisa perkembangan perusahaan.

1. **Batasan Masalah**

Adapun Batasan masalah dari Implementasi Sistem Keuangan dengan Metode Sistem Perpetual dan Periodik pada CV Panen Redjo ini meliputi :

1. Alat bantu yang digunakan dalam pembangunan, perancangan dan analisa sistem adalah UML (*Unified Modeling Language*).
2. Implementasi Sistem Informasi Keuangan dengan Metode Sistem Perpetual dan Periodik hanya fokus untuk mencatat sistem perpetual dan periodik.
3. Perancangan aplikasi ini menggunakan model pengembangan *waterfall.*
4. Sistem informasi ini meliputi penginputan data pembelian barang, penjualan barang, dan pencatatan keuangan.
5. **Tujuan Tugas Akhir**

Tujuan dibuatnya tugas akhir ini adalah untuk menghasilkan sebuah Aplikasi Sistem Informasi Keuangan yang mampu membantu mempermudah pekerjaan pencatatan pengelolaan data keuangan dengan metode sistem perpetual dan periodik yang ada pada CV Panen Redjo Semarang.

1. **Manfaat Tugas Akhir**
   1. **Bagi Penulis**
      1. Sebagai sarana untuk menerapkan ilmu yang didapat dalam mengimplementasikan sistem informasi keuangan.
      2. Menambah wawasan, pengetahuan dan pengalaman dalam sebuah sistem akuntansi.
   2. **Bagi Universitas Semarang**
      1. Menambah ragam hasil penelitian atau tugas akhir sebagai pembendaharan pustaka akademik.
      2. Sebagai dorongan bagi akademik untuk menjadi tolak ukur keberhasilan dalam memberikan ilmu kepada mahasiswa.
   3. **Bagi Pembaca**

Dapat digunakan sebagai referensi dan bahan acuan apabila hendak melakukan penelitian dan perkembangan lebih lanjut.

1. **Metodologi Penelitian**

Pengumpulan Data adalah sumber atau bahan mentah yang sangat berharga bagi suatu proses yang akan menghasilkan informasi, oleh karena itu dalam pengumpulan sumber data perlu penanganan secara cermat, sehingga menghasilkan data yang akurat dan berkualitas.

* + 1. **Metode Pengumpulan Data.**
       - 1. ***Wawancara***

Metode pengumpulan sumber data ini diperoleh secara langsung dengan melakukan wawancara langsung dengan Pemilik CV. Panen Redjo yaitu Bapak Tan Kiem Poo dan salah satu bagian kasir Bapak Achmad Nur Yadin. Dengan mengajukan pertanyaan tentang proses pengelolaan data dan kendala yang terjadi tentang pencatatan keuangan dan pengelolaan barang.

* + - * 1. **Observasi**

Metode pengumpulan sumber data ini diperoleh dengan pengamatan secara langsung proses barang masuk dan keluar, dan proses penjualan agar didapat data yang sesuai kebutuhan yaitu cara pencatatan keuangan sebagai bahan dasar penelitian . Peneliti tidak ikut serta dalam kegiatan, hanya berperan mengamati kegiatan.

* + - * 1. **Studi pustaka**

Studi pustaka dilakukan dengan cara mempelajai teori-teori *literature* dan buku-buku yang berhubungan dengan system pencatatan keuangan dan sistem informasi sebagai dasar dalam pelakasaan penelitian.

* + 1. **Jenis Data**

Dengan sumber data yang di peroleh meliputi data primer dan data sekunder sebagai berikut:

1. **Data primer**

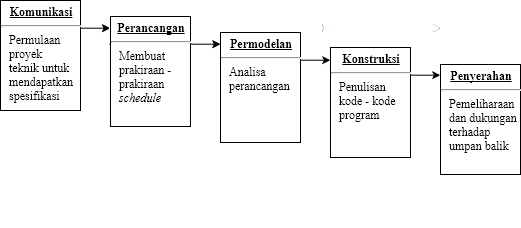
Merupakan data uang didapat secara langsung dari narasumber yang bersangkutan dengan penelitian melalui wawancara dengan pihak toko dan dari pengamatan penulis.

1. **Data sekunder**

Merupakan data yang diperoleh secara tidak langsung dari berbagai literature yang berhubungan dengan implementasi sistem informasi keuangan dengan metode sistem perpetual dan periodik, data ini berupa bukti, catatan dalam pembuatan laporan.

* + 1. **Metode Pengembangan Sistem**

Metodologi pengembangan sistem yang digunakan pada permodelan UML (*United Modeling Language*) dan penerapan pengembangan adalah model *waterfall*. Metode pengembangan *waterfall system* kadang dinamakan siklus hidup klasik, dimana hal ini menyiratkan pendekatan sistematis dan berurutan pada pengembangan perangkat lunak, yang dimulai dengan spesifikasi kebutuhan pengguna dan berlanjut melalui tahapan-tahapan perencanaan, permodelan, kontruksi, serta penyerahan sistem ke pengguna yang diakhiri dengan dukungan berkelanjutan pada perangkat lunak lengkap yang dihasilkan (Pressman, 2012). Metode ini meliputi:



Gambar 1.1 Metode *Waterfall* (Pressman, 2012)

1. **Tahapan dalam Komunikasi**

Langkah ini merupakan analisis terhadap kebutuhan software, dan tahap untuk mengadakan pengumpulan data dengan melakukan pertemuan di Toko Panen Redjo. Kegiatan penelitihan, guna mengumpulkan data-data tambahan baik yang ada di jurnal, artikel, maupun dari internet.

1. **Tahapan dalam Perencanaan**

Proses perencanaan merupakan lanjutan dari proses komunikasi. Membuat perkiraan-perkiraan dan schedule untuk sistem. Tahapan ini menentukan tahapan – tahapan jadwal dalam perancangan sistem dan jadwal kegiatan pembangunan system yang menghasilkan waktu dalam pembuatan sistem yang dibuat dengan menggunakan *software Microsoft Project* sebagai acuan dari kegiatan-kegiatan dalam perancangan hingga terselesaikan dalam pembuatan aplikasi dan selesainya dalam laporan penelitian.

1. **Tahapan dalam Permodelan**

Menganalisa dan melakukan perancangan untuk sistem sesuai kebutuhan dan data yang ada sebelum dibuat coding. Proses ini berfokus pada rancangan struktur data, arsitektur software, representasi interface.

Tahap ini menggambarkan alur sistem perangkat lunak yang mendasar dengan menggunakan alat bantu perancangan sistem, yaitu UML (*Unified Modeling Language*) dan *mockup* untuk sebagai dasar desain antarmuka / *interface* *user*.

1. **Tahapan dalam Kontruksi**

Desain program diterjemahkan ke dalam kode-kode dengan menggunakan *framework Laravel* dengan menggunakan *Visual Studio Code* dan database menggunakan *MariaDB*.

Tahap ini menvisualisasikan desain atau aplikasi yang memenuhi kebutuhan pengguna dari identifikasi. Kebutuhan pengguna biasanya melibatkan pemecahan masalah, perencanaan, pengujian perangkat lunak.

Program yang dibangun langsung diuji untuk menyesuaian proses-proses, fungsi perhitungan, fungsi pencatatan, dan pengujian untuk kesalahan pengguna (*human error*) guna mendapatkan aplikasi yang sesuai dengan hasil penelitian.

1. **Tahapan dalam Penyerahan *Software***

Tahap ini bisa dikatakan final dalam pembuatan sebuah *software* atau sistem. Setelah melakukan analisis, desain dan pengkodean maka sistem yang sudah jadi akan diserahkan dan digunakan oleh *user*. Kemudian *software* yang telah dibuat harus dilakukan pemeliharaan secara berkala, dalam hal ini peneliti hanya akan melakukan pemantauan dan pemeliharaan untuk memastikan aplikasi berjalan tanpa adanya permasalah pengkodean maupun *error logic*.

1. **Sistematika Penulisan**

Penyusunan laporan untuk Tugas Akhir memaparkan pembahasan tentang yang dilakukan penulis, dalam penyusunan laporan dengan sistematika sebagai berikut:

BAB I : PENDAHULUAN

Memaparkan latar belakang masalah, perumusan masalah yang akan dihadapi, Batasan masalah yang diambil, tujuan tugas akhir, metode pengumpulan data yang digunakan, dan sistematia penulisan laporan perancangan.

BAB II : TINJAUAN UMUM PERUSAHAAN

Menguraikan penjelasan tentang sejarah perusahaan, struktur organisasi, pengertian, tugas dan fungsi masing – masing struktur di CV Panen Rejo.

BAB III : LANDASAN TEORI

Memuat tentang landasan teori yang relevan terhadap perancangan, pembuatan sistem mengenai pencatatan keuangan metode perpetual dan periodik yang diambil dari kutipan buku, dan internet sebagai landasan dalam pembuatan aplikasi dan penjelasan yang berhubungan dengan penelitian yang dibahas.

BAB IV : PERENCANAAN DAN ANALISA PERANCANGAN SISTEM

Menguraikan tentang perencanaan pelaksanaan pembuatan aplikasi dan desain aplikasi yang di dasarkan pada analisa perancangan masukan penjualan dan pembelian, perancangan proses pencatatan perhitungan secara perpetual dan periodik, dan perancangan keluaran.

BAB V : IMPLEMENTASI SISTEM

Menguraikan dan membahas tentang implimentasi program aplikasi pencatatan keuangan dengan metode perpetual dan periodik dan cara-cara penggunaan dalam melakukan proses penginputan, pemprosesan data, serta dilakukan pengujian untuk kelayakan program aplikasi.

BAB VI : PENUTUP

Memuat mengenai penguraian terhadap kesimpulan hasil implementasi sistem dan saran guna mendukung sistem yang dibuat untuk perusahaan yang bersangkutan dengan tujuan melancarkan dan mempermudah dalam pengelolaan.

DAFTAR PUSTAKA

LAMPIRAN

# **BAB II**

# **TINJAUAN UMUM CV PANEN REJO**

1. **Sejarah**

CV Panen Rejo berdiri pada tanggal 11 Agustus 2009, yang beralamat di Jalan MH Tamrin. No. 77 Semarang dengan bidang usaha toko ritail pertanian yang terbagi menjadi beberapa kelompok: saprotan, obat – obatan, dan pupuk. Saat awal berdirinya CV Panen Rejo merupakan toko pribadi yang di kelola oleh keluarga dan hanya menjual produk - produk pertanian, seperti insektisida, zpt, pupuk dan alat – alat penunjang pertanian seperti spayer pegas, cangkul, dan sabit.

CV Panen Rejo melakukan perluasan dalam memenuhi kebutuhan para petani dan para hobiis pertanian, untuk mulai di tambah beberapa produk untuk para hobiis tanaman seperti rockwull, rak pot, pupuk untuk tanaman hidroponik. Dengan tujuan untuk mengakomodasi meningkatnya para hobiis tanaman dan pertanian di daerah perkotaam semarang.

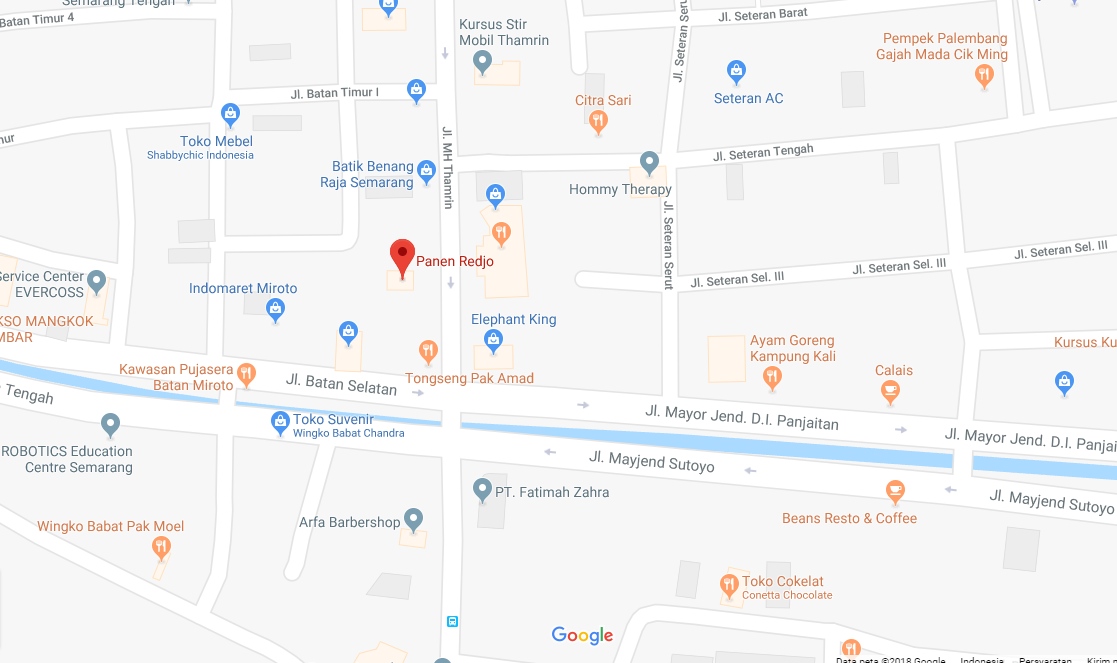
Dengan pertumbuhan toko yang semakin maju karna banyaknya peminat hobiis tanaman dan hobiis pertanian, CV Panen Rejo sekarang memiliki karyawan sebanyak 6 orang untuk menghandel customer yang datang dan di tambahkan devisi teknisi untuk perbaikan alat penunjang pertanian dan pembuatan tempat media penanaman hidroponik atau aquaponik.

1. **Visi dan Misi**
2. **Visi**

Menjadi toko ritail pertanian terlengkap yang tersebar diseluruh kota besar di indonesia

1. **Misi**
2. Menyediakan sarana pertanian, media informasi, produksi olahan herbal di bidang pertanian kepada masyarakat umum
3. Memberikan informasi atau pengetahuan mengenai pertanian kepada masyarakat pada umumnya
4. Membangun jaringan agribisnis
5. **Tentang CV Panen Rejo**
6. **Alamat**

CV Panen Rejo beralamat di Jalan MH Tamrin. No. 77 Semarang



Gambar 2.1 Peta Lokasi

1. **Kontak** **Informasi :**

Telp : 024-3557557

Fax : 024-3511637



Gambar 2.2 Halaman Toko



Gambar 2.3 Ruangan Toko

1. **Sarana dan Prasarana**



Gambar 2.4 Sarana dan Prasarana

* 1. Sarana

CV Panen Rejo adalah tempat pembelian sarana pertanian, sarana perkebunan, dan sarana obat tanaman.

* 1. Prasarana

Prasarana penunjang pertubuhan dan pembuahan tanaman

1. **Struktur Organisasi**



Gambar 2.5 Struktur Oraganisasi

1. **Tugas dan Wewenang Masing – Masing Jabatan**
2. Pemilik Toko

Pemilik sebagai wewenang yang bertugas dalam hal perancangan, pengelolaan dan pengambilan keputusan terhadap perusahaan dan membina tenaga kerja.

1. Kasir dan Keuangan

Kasir & Keuangan bertugas untuk mengatur dan mengelola keuangan perusahaan. Proses keluar dan masuknya uang perusahaan semua harus diketahui dan disetujui oleh pihak keuangan.

1. Pengadaan dan Produksi

Tugas dari pengadaan produksi tersebut melakukan pengawasan terhadap pergerakan barang mulai dari pembelian bahan, permintaan bahan baku, siklus pembuatan secara keseluruhan, sampai pengiriman barang jadi.

1. Staff Toko

Staff Toko bertugas dalam pelayanan terhadap pelanggan dengan menawarkan produk – produk yang diinginkan.

# **BAB III**

# **LANDASAN TEORI**

1. **Pengertian Sistem**

Sistem didefinisikan sebagai sekumpulan prosedur yang saling berkaitan dan saling terhubung untuk melakukan suatu tugas bersama – sama. Secara garis besar, sebuah sistem informasi terdiri dari tiga komponen utama. Ketiga komponen tersebut mencakup software, hardware, dan brainware. Ketiga komponen ini saling berkaitan satu sama lain. (Pratama, 2014).

Suatu sistem yang baik harus memiliki tujuan dan sasaran yang tepat karena hal ini akan sangat menentukan dalam mendefinisikan masukan yang dibutuhkan sistem dan keluaran yang dihasilkan.

1. **Persediaan**

Dalam perusahaan dagang maupun manufaktur, persediaan sangat dan termasuk bagian aktivasi lancar yang aktif.

Suatu aktifa yang meliputi barang-barang milik perusahaan dengan maksud untuk dijual dalam periode usaha tertentu atau persediaan barang – barang yang masih dalam pengerjaan atau proses produksi ataupun persediaan bahan baku yang menunggu penggunaannya dalam proses produksi. (Alexandri 2009).

Sedangkan menurut Ristono (2009) persediaan merupakan teknik untuk manajemen material yang berkaitan dengan perusahaan.

1. **Penjualan**

Didalam sebuah perusahaan khususnya dalam perusahaan yang bergerak dibidang usaha perdagangan, penghasilan utama yang dihasilkannya ialah berasal dari penjualan.

1. Pengertian penjualan menurut beberapa ahli
   1. Menurut Mulyadi (2013) menyatakan bahwa “Penjualan adalah penjualan yang dilaksanakan oleh perusahaan dengan cara mewajibkan pembeli melakukan pembayaran harga barang terlebih dahulu sebelum barang diserahkan oleh perusahaan kepada pembeli”.
   2. Menurut Swastha (2010) menyatakan bahwa “Menjual adalah ilmu dan seni mempengaruhi pribadi yang dilakukan oleh penjual untuk mengajak orang lain agar bersedia membeli barang atau jasa yang ditawarkan”.
   3. Menurut Sulistiyowati (2010) manyatakan bahwa "Penjualan adalah pendapatan yang berasal dari penjualan produk perusahaan, disajikan setelah dikurangi potongan penjualan dan retur penjualan".
   4. Berdasarkan pengertian diatas mengenai penjualan, maka disimpulkan bahwa penjualan adalah salah satu prosedur yang digunakan untuk memperjualbelikan barang dan jasanya kepada konsumen, yang tujuannya untuk memperoleh keuntungan atau laba.
2. Jenis – jenis penjualan

Menurut Susanto (2013) berdasarkan dari jenis transaksi membagi kedalam:

* + 1. Penjualan secara tunai, yaitu penjualan yang bersifat cash and carry, pada umumnya terjadi secara spontan. Pembayaran dalam jangka waktu satu bulan yang masih sebagai penjualan tunai.
    2. Penjualan secara kredit, penjualan yang periode pembayarannya rata-rata diatas satu bulan.
    3. Penjualan secara tender, yaitu penjualan yang dilaksanakan melaui proses tertentu untuk memenuhi permintaan pihak pembeli yang membuka tender tersebut, untuk memenangkan tender selain harus memenuhi berbagai prosedur yaitu pemenuhan dokumen tender yang berupa jaminan tender, jugaharus dapat bersaing dengan pihak lain.
    4. Penjualan *export*, yaitu penjualan yang dilakukan dengan pembelian luar negeri yang mengimport barang.
    5. Penjualan secara konsinyasi yaitu, menjual barang-barang secara titipan kepada pembeli atau penjual. Apabila barang tersebut tidak laku dijual, maka barang tersebut akan dikembalikan kepada si penjual.
    6. Penjual melaui grosir, yaitu penjualan yang tidak langsung kepada konsumen, tetapi melalui pedagang pertama (grosir). Grosir berfungsi sebagai perantara antara pabrik atau import dengan pelanggan atau toko eceran.

1. **Pengertian Pencatatan**

Pada suatu perusahaan tentunya diperlukan untuk adanya pencatatan persediaan, karena akan membantu kegiatan operasional perusahaan, pencatatan persediaan sangat membantu dalam mengontrol serta mengelola masuk keluarnya persediaan, dan pencatatan adalah proses analisis atas suatu transaksi atau peristiwa keuangan yang terjadi dalam entitas dengan cara menempatkan transaksi di sisi debet dan sisi kredit menurut Pura (2013).

1. **Sistem Pencatatan Periodik**

Menurut Santoso (2010) sistem pencatatan periodik adalah suatu sistem pengelolaan persediaan dimana dalam penentuan persediaan dilakukan melakukan melalui perhitungan secara fisik (physical counting) yang lazim dilakukan pada setiap akhir periode akuntansi dalam rangka penyiapan laporan keuangan. Melaui perhitungan fisik ini, jumlah kuantitas porsediaan (inventory quantity) akan diketahui (misalnya dalam berat, meter, kilogram dan sebagainya) sehingga nilai persediaan (inventory value) dapat dihitung dengan mengalikan jumlah kuantitas persediaan dengan suatu harga”.

Sedangkan menurut Martani (2012) sistem pencatatan periodik adalah sistem periodik merupakan sistem pencatatan persediaan dimana kuantitas persediaan ditentukan secara periodik yaitu hanya pada saat perhitungan fisik yang biasanya dilakukan secara stock opname.

1. **Sistem Pencatatan Perpetual**

Menurut Santoso (2010) sistem pencatatan perpetual adalah persediaan terus-menerus (perpetual inventory system) Merupakan suatu sistem pengelolaan persediaan dimana pencatatan mutasi persediaan dilakukan secara terus menerus dan berkesinambungan sehingga mutasi persediaan selama satu periode termonitor dan setiap saat jumlah maupun nilai persediaan selama satu periode termonitor dan setiap saat jumlah maupun nilai persediaan dapat diketahui tanpa melakukan secara fisik.

Sedangkan Menurut Martani (2012) sistem pencatatan perpetual adalah merupakan sistem pencatatan persediaan dimana pencatatan yang up-to-date terhadap barang persediaan selalu dilakukan setiap terjadi perubahan nilai persediaan.

1. **Laporan Keuangan**

Menurut Baringin (2014) mengemukakan pengertian laporan keuangan adalah Laporan keuangan merupakan hasil dari kegiatan operasi normal perusahaan akan memberikan informasi keuangan yang berguna bagi entitas di dalam perusahaan itu sendiri maupun entitas lain di luar perusahaan.

Menurut Hanafi dan dkk (2012) secara umum ada tiga bentuk laporan keuangan yang pokok dihasilkan oleh suatu perusahaan. Neraca, Laporan Rugi laba, dan Laporan Aliran Kas. Laporan-laporan keuangan tersebut pada dasarnya ingin melaporkan kegiatan operasional, sekaligus mengevaluasi keberhasilan strategi perusahaan untuk mencapai tujuan yang ingin dicapai.

1. ***United Modeling Language* (*UML*)**

Menurut Nugroho (2010), UML (*Unified Modeling Language*) adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma berorientasi objek. Pemodelan (*modeling*) sesungguhnya digunakan untuk menyederhanakan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipahami dan dipelajari. *UML* juga menyediakan standar notasi ataupun diagram yang digunakan untuk pemodelan sistem,

1. ***Use* *Case* *Diagram***

Menurut Nugroho (2010), *Use Case Diagram* digunakan untuk memodelkan fungsionalitas - fungsionalitas sistem/perangkat lunak dilihat dari pengguna yang ada di luar sistem (yang sering dinamakan sebagai aktor). *Use Case* pada dasarnya merupakan unit fungsionalitas koheren yang diekspresikan sebagai transaksi-transaksi yang terjadi antara aktor dan sistem. Kegunaan dari *use case diagram* adalah untuk mendaftarkan aktor-aktor dan *use case – use case* dan memperlihatkan aktor-aktor mana yang berpartisipasi dalam masing-masing *use case*. Berikut notasi-notasi yang digunakan dalam *use case diagram*:

Tabel 3.1 Notasi-Notasi *Use Case Diagram*

|  |  |  |  |
| --- | --- | --- | --- |
| **NOTASI** | **KEGUNAAN** | | **SIMBOL** |
| *Actor* | Menggambarkan semua objek diluar sistem (bukan hanya pengguna sistem/ perangkat lunak) yang berinteraksi dengan sistem yang dikembangkan. | |  |
| *Use Case* | Menggambarkan fungsionalitas yang dimiliki sistem. | |  |
| *Collaboration* | Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi). | |  |
| *Assosiation* | Lintasan komunikasi antara *actor* dengan *use case.* | |  |
| **NOTASI** | **KEGUNAAN** | | **SIMBOL** |
| *Include* | Penambahan perilaku ke suatu *use case* dasar yang secara eksplisit mendeskripsikan penambahan tersebut. | | <<include>> |
| *Extend* | Penambahan perilaku ke suatu *use case* dasar. | | <<extend>> |
| *Generalization* | Relasi antara pengklasifikasi yang memiliki deskripsi yang bersifat labih umum dengan berbagai pengklasifikasi yang lebih spesifik, digunakan dalam struktur pewarisan. | |  |
| *Dependency* | | Relasi antar dua elemen model. |  |

1. ***Class* *Diagram***

Menurut Nugroho (2010), dalam notasi *UML*, himpunan kelas-kelas beserta hubungan / relasi / asosiasi antar kelas biasanya digambarkan menggunakan sebuah *diagram UML* yang dinamakan diagram kelas (*class diagram*). Jika kita perhatikan lebih jauh, sesungguhnya diagram kelas memiliki dua kegunaan / fungsi yang sangat penting, yaitu:

1. Mempresentasikan keadaan statis kelas-kelas yang terlibat dalam sistem. Kelas-kelas ini bisa saja merupakan kelas-kelas dalam bahasa pemrograman dan kelas-kelas persisten yang hadir dalam bentuk tabel-tabel yang ada di sistem basis data relasional.
2. Hubungan antar kelas dalam sistem atau perangkat lunak yang sedang kita kembangkan dapat terlihat dengan mudah.

Berikut notasi-notasi yang digunakan dalam *class diagram UML*:

Tabel 3.2 Notasi-Notasi *Class* Diagram

|  |  |  |
| --- | --- | --- |
| **NOTASI** | **KETERANGAN** | **SIMBOL** |
| *Class* | *Class* adalah balok-balok pembangun pada pemrograman berorientasi objek. Sebuah *class* digambarkan sebagai sebuah kotak yang terbagi menjadi 3 bagian. Bagian atas adalah bagian nama dari *class*. Bagian tengah mendefinisikan atribut *class*. Bagian bawah mendefinisikan *method* dari sebuah *class*. |  |
| *Assosiation* | Sebuah asosiasi merupakan sebuah *relationship* paling umum antara 2 *class*, dan dilambangkan oleh sebuah garis yang menghubungkan antar 2 *class*. Garis ini dapat melambangkan tipe - tipe *relationship* dan juga dapat menampilkan hukum-hukum multiplisitas pada sebuah *relationship*. | 1..n  owned  by 1 |
| *Composition* | Jika sebuah *class* tidak bisa berdiri sendiri dan harus menjadi bagian dari *class* yang lain, maka *class* tersebut memiliki relasi *composition* terhadap *class* tempatnya bergantung tersebut. |  |
| *Dependency* | Kadang kala class menggunakan *class* yang lain. Hal ini disebut *dependecy*. Umumnya *dependency* digunakan untuk menunjukkan operasi pada suatu *class* yang menggunakan *class* yang lain. |  |
| **NOTASI** | **KETERANGAN** | **SIMBOL** |
| *Aggregation* | *Aggregation* mengindikasikan keseluruhan bagian *relationship* dan biasanya disebut relasi “bagian dari”. Sebuah *aggregation* digambarkan sebagai sebuah garis dengan sebuah jajaran genjang yang tidak berisi / tidak solid. |  |
| *Generalization* | Sebuah *generalization* dilambangkan dengan sebuah panah dengan kepala panah yang tidak solid yang mengarah ke arah “*parent*”-nya / induknya. |  |

1. ***Activity* *Diagram***

Menurut Nugroho (2010), diagram aktivitas (*activity diagram*) sesungguhnya merupakan bentuk khusus dari state machine yang bertujuan untuk memodelkan komputasi-komputasi dan aliran - aliran kerja yang terjadi dalam sistem / perangkat lunak yang sedang dikembangkan. State pada diagram aktivitas merepresentasikan state dari komputasi yang dieksekusi, bukan state dari suatu objek biasa.

Biasanya, suatu *diagram* aktivitas mengasumsikan komputasi - komputasi dilaksanakan tanpa adanya interupsi-interupsi eksternal berbasis *event* terjadi padanya.

Suatu *diagram* aktivitas memuat di dalamnya *activity state* dimana suatu *activity state* merepresentasikan eksekusi pernyataan dalam suatu prosedur atau kinerja suatu aktivitas dalam suatu aliran kerja. Alih-alih menunggu selesainya atau event seperti yang terjadi pada state tunggu, *activity state* menunggu selesainya komputasi.

Saat suatu aktivitas selesai maka akan berlanjut ke *activity state* berikutnya yang terlihat pada diagram aktivitas. Penyelesaian transisi dalam suatu diagram aktivitas biasanya akan terpicu saat aktivitas sebelumnya selesai.

*Activity state* biasanya tidak memiliki transisi-transisi yang diakibatkan oleh *event-event* eksplisit, tapi mungkin dibatalkan oleh transisi-transisi pada state yang melingkupinya. Berikut adalah notasi - notasi / simbol-simbol yang digunakan pada *activity diagram*:

Tabel 3.3 Notasi-notasi *Activity Diagram*

|  |  |  |
| --- | --- | --- |
| **NOTASI** | **KETERANGAN** | **SIMBOL** |
| *Initial* | Titik awal untuk memulai suatu aktivitas. |  |
| *Final* | Titik akhir untuk mengakhiri aktivitas. |  |
| *Activity* | Menandakan sebuah aktivitas. |  |
| *Decision* | Pilihan untuk menggambil keputusan. |  |
| *Fork* | Menunjukkan kegiatan yang dilakukan secara paralel. |  |
| *Join* | Untuk menggabungkan beberapa kegiatan secara paralel menjadi satu. |  |

1. ***Sequence* *Diagram***

Menurut Nugroho (2010), *Sequence* *Diagram* memperlihatkan interaksi sebagai *diagram* dua matra (dimensi). Matra vertikal adalah sumbu waktu, waktu bertambah dari atas ke bawah. Matra horizontal memperlihatkan peran pengklasifikasian yang merepresentasikan objek - objek mandiri yang terlibat dalam kolaborasi.

Masing - masing pengklasifikasian direpresentasikan sebagai kolom-kolom vertikal dalam *sequence diagram* yang sering disebut sebagai garis waktu (*life line*). Selama objek ada, peran digambarkan menggunakan garis tegas.

Selama aktivitas prosedur pada objek aktif, garis waktu digambarkan sebagai garis ganda. Pesan-pesan digambarkan sebagai suatu tanda panah dari garis waktu suatu objek ke garis waktu objek lainnya.

Panah-panah menggambarkan aliran pesan antar peran pengklasifikasian digambarkan dalam urutan waktu kejadiannya dari atas ke bawah. Berikut selengkapnya notasi-notasi yang digunakan dalam *sequence* *diagram*:

Tabel 3.4 Notasi-notasi Sequence *Diagram*

|  |  |  |
| --- | --- | --- |
| **NOTASI** | **KETERANGAN** | **SIMBOL** |
| *Object*  (Partisipan) | Objek atau biasa disebut partisipan merupakan *intance* dari sebuah *class* dan dituliskan tersusun secara horizontal. Digambarkan sebagai sebuah *class* (kotak) dengan nama objek di dalamnya yang diawali dengan titik koma. |  |
| *Actor* | *Actor* juga dapat bekomunikasi dengan objek, maka *actor* juga dapat diurutkan sebagai kolom. |  |
| **NOTASI** | **KETERANGAN** | **SIMBOL** |
| *Life Line* | *Life line* mengindikasikan keberadaan sebuah *object* dalam baris waktu. Notasi untuk *life line* adalah garis putus-putus vertikal yang ditarik dari sebuah *object*. |  |
| *Activation* | Activation dinotasikan sebagai sebuah kotak persegi empat yang digambarkan pada sebuah *life line*. *Activation* mengindikasikan sebuah *object* yang akan melakukan sebuah aktivasi. |  |
| *Boundary* | *Boundary* terletak antara sebuah sistem dengan sekelilingnya. Semua *form*, laporan-laporan, antar muka diperangkat keras seperti *printer* dan *scanner* dan antar muka ke sistem lainnya adalah termasuk ke dalam kategori. |  |
| *Control* | *Control* berhubungan dengan fungsionalitas seperti pemanfaatan sumber daya, pemrosesan terdistribusi, atau penanganan kesalahan. |  |
| *Entity* | *Entity* digunakan untuk menangani informasi yang mungkin akan disimpan secara permanen. *Entity* bisa juga merupakan sebuah tabel pada struktur basis data. |  |
| *Message* | *Message*, digambarkan dengan anak panah horizontal antar *activation*. *Message* mengindikasikan komunikasi antara objek-objek. |  |
| **NOTASI** | **KETERANGAN** | **SIMBOL** |
| *Self Message* | *Self message* mengindikasikan komunikasi kembali ke dalam sebuah objek itu sendiri. |  |

1. **Balsamiq Mockups**

Menurut website resmi Balsamiq https://balsamiq.com/. Balsamiq Mockups adalah alat wireframing cepat yang membantu bekerja lebih cepat & lebih 42 pintar. Balsamiq Mockups menciptakan pengalaman sketsa di papan tulis, tetapi menggunakan komputer, membuat mockups menjadi cepat. Akan menghasilkan lebih banyak ide, sehingga dapat membuang yang buruk dan menemukan solusi terbaik.

1. **Pengujian Sistem**

Pengujian bertujuan untuk mencari kesalahan. Pengujian yang baik adalah pengujian yang memiliki kemungkinan besar dalam menemukan kesalahan. Karena itu, harus merancang dan mengimplementasikan sistem berbasis komputer atau produk dengan “kemampuan untuk diuji”. Pada saat yang sama, tes itu sendiri harus menunjukan serangkaian karakterristik yang bertujuan untuk menemukan sebanyak mungkin kesalahan dengan usaha sekecil mungkin (Pressman, 2012).

1. Pengujian Kotak Hitam

Pengujian kotak hitam, juga disebut pengujian perilaku, berfokus pada persyaratan fungsional perangkat lunak. Artinya, teknik pengujian kotak hitam memungkinkan untuk membuat beberapa kumpulan kondisi masukan yang sepenuhnya akan melakukan semua kebutuhan fungsional untuk program. Pengujian kotak hitam bukan teknik alternatif untuk kotak putih. Sebaliknya, ini merupakan pendekatan pelengkap yang mungkin dilakukan untuk mengungkap kelas kesalahan yang berbeda dari yang diungkap oleh metode kotak putih. Pengujian kotak hitam berupaya untuk menemukan kesalahan dalam kategori berikut:

1. Fungsi yang salah atau hilang
2. Kesalahan antar muka
3. Kesalahan dalam struktur data atau akses basis data eksternal
4. Kesalahan perilaku atau kinerja
5. Kesalahan inisialisasi dan penghentian
6. Pengujian Kotak Putih

Pengujian kotakk putih, terkadang disebut juga pengujian kotak kaca *(glass-box testing),* merupakan sebuah filosofi perancangan test case yang menggunakan struktur kontrol yang dijelaskan sebagai bagian dari perancangan peringkat komponen untuk menghasilkan test case.

Untuk melakukan pengujian kotak putih*,* perlu diketahui tahap-tahap sebagai berikut:

1. Menganalisa *soucre code* untuk membuat *flow graph.*
2. Mengidentifikasi jalur tes untuk mencapai pemenuhan tes berdasarkan pada *flow graph.*
3. Mengevaluasi kondisi tes yang akan dicapai dalam tiap tes.
4. Memberikan nilai masukan dan keluaran berdasarkan pada kondisi

Perhitungan *white –box* :

V(G) = E­ – N + 2

Keterangan :

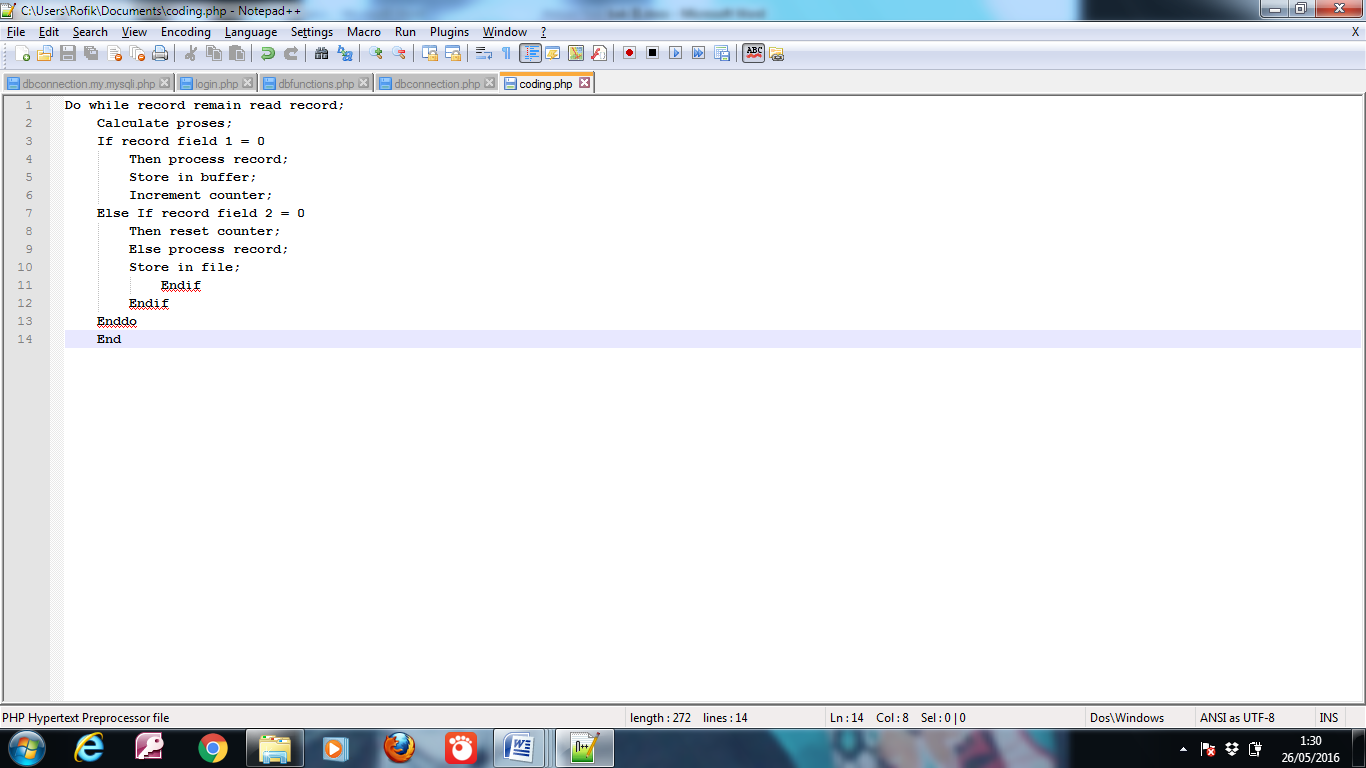
V(G) : *Cyclomatic Complexity flow graph*

E : Jumlah *Edge* (panah)

N : Jumlah *Node* (lingkaran)

Berikut ini merupakan contoh dalam melakukan pengujian kotak putih :

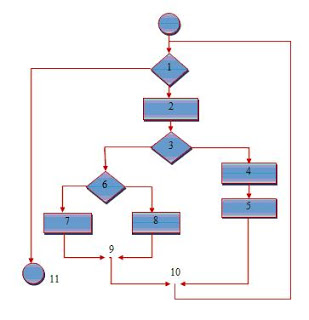
1. Tahap ke-1 tentukan *soucre code* yang akan di uji.



Gambar 3.2 *Soucre code*

Sumber: www.learningame.org

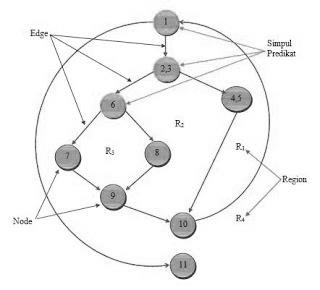
1. Tahap ke-2 membuat Flow chart



Gambar 3.3 *Flow chart*

Sumber: www.learningame.org

1. Tahap ke-3 membuat Flow graph



Gambar 3.4 *Flow graph*

Sumber: www.learningame.org

1. Tahap ke-4 menentukan jalur independen

Jalur program independen adalah setiap jalur yang melalui program yang memperkenalkan setidaknya satu kumpulan pernyataan-pernyataan pemrosesan atau kondisi baru. Bila dinyatakan dalam grafik alir, jalur independen harus bergerak sepanjang setidaknya satu *edge* yang belum dilintasi sebelum jalur tersebut didefinisikan. misalnya pada gambar 3.6 *flowgraph* adalah:

V(G) = 11 – 9 + 2 = 4

Berdasarkan urutan alirnya, didapat suatu kelompok basis *flow graph*

1. Jalur 1 : 1 – 11
2. Jalur 2 : 1-2-3-4-5-10-1-11
3. Jalur 3 : 1-2-3-6-7-9-10-1-11
4. Jalur 4 : 1-2-3-6-8-9-10-1-11
5. ***Flowchart***

*Flowchart* adalah diagram yang menggambarkan proses, sistem atau algoritma komputer. *Flowchart* banyak digunakan dalam berbagai bidang untuk mendokumentasikan, mempelajari, merencanakan, meningkatkan, dan mengomunikasikan proses yang sering rumit kedalam diagram yang jelas dan mudah dipahami.

*Flowchart* kadang-kadang dieja sebagai diagram alur, menggunakan persegi panjang, oval, jajargenjang dan banyak bentuk lain untuk menentukan jenis langkah, bersama dengan menghubungkan panah untuk menentukan aliran dan urutan. Jika kita mempertimbangkan semua bentuk diagram alur, mereka adalah salah satu diagram yang paling umum, yang digunakan oleh orang-orang teknis dan non-teknis di berbagai bidang. *Flowchart* kadang-kadang disebut dengan nama yang lebih khusus seperti *Process Flowchart*, *Process Map*, *Functional Flowchart*, *Business Process Mapping*, *Business Process Modeling and Notation* (BPMN), atau *Process Flow Diagram* (PFD). Mereka terkait dengan diagram populer lainnya, seperti *Data Flow Diagrams* (DFD) dan *Unified Modeling Language* (UML) *Activity Diagrams*.

# **BAB IV**

# **PERENCANAAN DAN ANALISA PERANCANGAN SISTEM**

1. **Perencanaan Sistem**

Perencanaan sistem adalah proses membuat sebuah laporan perencanaan sistem yang menggunakan sumber sistem informasi yang berhubungan dan mendukung tujuan bisnis dan operasi organisasi. Hal-hal yang diperlukan yaitu membuat studi kelayakan untuk sistem yang akan dibuat, seperti membuat kajian bagaimana proses sistem yang akan berjalan dengan sistem baru dan bagaimana pengaruhnya, sehingga dapat berfungsi secara maksimal.

Batasan ruang lingkup yang akan dirumuskan sebelum membuat perencanaan adalah mengumpulkan data, menentukan masalah dan mendiskusikan kebutuhan sistem dalam pengolahan data barang penjualan dan laporan keuangan pada CV Panen Rejo.

Dengan permasalahan tidak adanya pencatatan produk, penjualan barang yang detail, maka dibutuhkan perancangan sistem yang detail dengan pengelolaan produk dan pencatatan penjualan serta laporan keuangan yang baik guna mempermudah dalam pekerjaan.

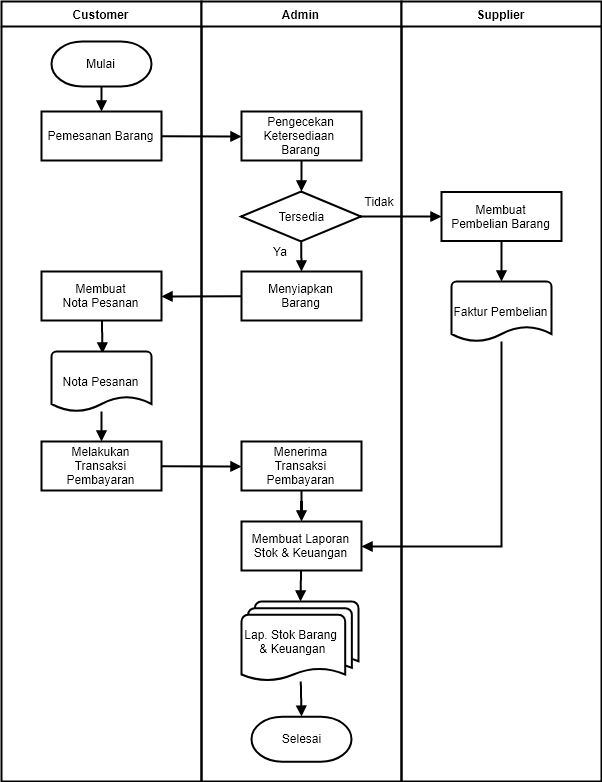
1. **Analisa Sistem**

Tahap analisa sistem merupakan kegiatan untuk melihat sistem yang sudah berjalan, melihat bagian mana yang bagus dan tidak bagus, dan kemudian mendokumentasikan kebutuhan yang akan dipenuhi dalam sistem yang baru.

Saat ini proses yang berjalan pada CV Panen Rejo sudah menggunakan *Microsoft Excel* dalam melakukan pencatatan transaksi keuangan, akan tetapi sistem persediaan belum ada pencatatan dan penjualan hanya mencatat nilai transaksi menggunakan *Microsoft Excel* tanpa mengetahui produk yang terjual apa saja dan laporan keuangan masih dari hasil pencatatan penjualan saja setiap harinya yang di lakukan oleh kasir maupun pemilik toko dan melakukan penghitungan kembali setiap hari jika barang menipis atau kosong baru melakukan pembelian ulang. Dengan masalah tersebut maka diperlukan sistem pencatatan laporan perusahaan yang lebih baik untuk CV Panen Rejo.

* 1. Diagram Flowchart Proses Bisnis

Proses bisnis yang dilakukan oleh CV Panen Rejo.



Gambar 4.1 Diagram Flowchart Proses Bisnis pada CV Panen Rejo

1. **Analisa Kebutuhan**

Analisa kebutuhan merupakan hal yang penting untuk mengetahui kebutuhan-kebutuhan yang nantinya akan digunakan untuk mendukung proses pembuatan sistem baru.

1. Analisa Kebutuhan Pengguna

Dalam sistem ini penulis menganalisa siapa saja yang akan menggunakan sistem dan apa yang dilakukan dalam sistem ini. Sistem ini akan digunakan oleh:

*Administrator* atau admin yang memiliki hak akses dalam penggunaan sistem, dan mampu mengoperasikan computer.

1. Analisa Kebutuhan *Hardware*

Kebutuhan perangkat keras yang digunakan untuk membangun aplikasi ini minimal memiliki spesifikasi sebagai berikut:

1. *Processor Dual core*
2. *RAM* minimum 2GB
3. *Keyboard dan Mouse*
4. *Harddisk* 160GB
5. *Monitor* dan *Printer*
6. Analisa Kebutuhan *Software*

Kebutuhan software (perangkat lunak) yang dibutuhkan untuk membangun sistem penudukung ini, antara lain:

* 1. Sistem Operasi Windows 7 / Sistem Operasi Linux
  2. Web Server (Apache, MYSQL *Database*, PHP *Script* *Language*) berfungsi sebagai server untuk mengelola pemprosesan dari aplikasi pencatatan keuangan.
  3. Web Browser Google Chrome di tujukan untuk memjalankan aplikasi pencatatan keuangan dengan berbasis web

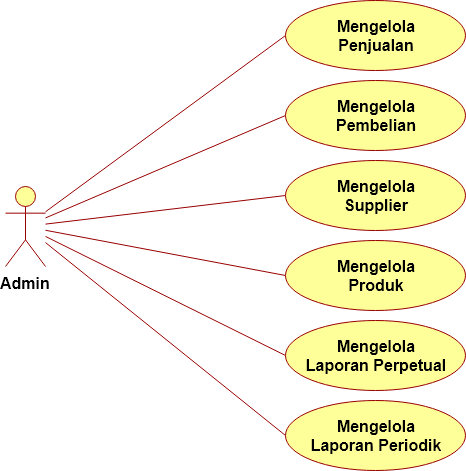
Dalam menyusun sistem baru, software dan pemprograman yang di gunakan adalah berbasis web meliputi perpaduan antara PHP, HTML. CSS, JS, dan MYSQL.

1. **Perancangan Sistem**

Perancangan sistem merupakan proses lanjutan dari hasil analisa sehingga dapat dihasilkan suatu perancangan sistem yang diperlukan dalam pembuatan perangkat lunak yaitu Implementasi Sistem Informasi Keuangan dengan Metode Sistem Perpetual dan Periodik pada CV Panen Redjo Semarang.

* 1. *Use Case Diagram*

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



Gambar 4.2 *Use Case Diagram*

Tabel 4.1 Sekenario *Use Case* Mengelola Penjualan

|  |  |  |
| --- | --- | --- |
| Nama Use Case | Mengelola Penjualan | |
| Pelaku Sistem Utama | Admin | |
| Diskripsi | Use case ini merupakan proses untuk melakukan pengelolaan data barang keluar dan pencatatan penjualan. Dalam proses dapat dilakukan proses tambah, edit penjualan | |
| Prakondisi | Admin telah masuk ke dalam menu utama | |
| Bidang khas event: | Kegiatan Aktor | Respon Sistem |
|  | Langkah 1 :  Admin memilih menu transaksi penjualan | Langkah 2 :  Sistem menampilkan transaksi penjualan |
|  | Langkah 3 :  Admin menginputkan data penjualan | Langkah 4 :  Sistem menyimpan data tersebut |
| Bidang Alternatif 1 | Langkah 3:  Admin memilih data penjualan | Langkah 4 :  Sistem menampilkan data tersebut |
|  | Langkah 5 :  Admin merubah data penjualan | Langkah 6 :  Sistem mengupdate data tersebut |
| Bidang Alternatif 2 | Langkah 3:  Admin memilih data penjualan | Langkah 4 :  Sistem menampilkan data tersebut |
|  | Langkah 5 :  Admin menghapus data penjualan | Langkah 6 :  Sistem menghapus data tersebut |
| Kesimpulan | Admin mempunyai peran untuk mengubah data yang ada dalam *master* data penjualan. | |
| Post Kondisi | Perubahan data yang ada di transaksi penjualan. | |

Tabel 4.2 Sekenario *Use Case* Mengelola Pembelian

|  |  |  |
| --- | --- | --- |
| Nama Use Case | Mengelola Pembelian | |
| Pelaku Sistem Utama | Admin | |
| Diskripsi | *Use case* ini merupakan proses untuk melakukan pengelolaan data barang masuk dan pencatatan pembelian. Dalam proses dapat dilakukan proses tambah, edit pembelian | |
| Prakondisi | Admin telah masuk ke dalam menu utama | |
| Bidang khas event: | Kegiatan Aktor | Respon Sistem |
|  | Langkah 1 :  Admin memilih menu transaksi pembelian | Langkah 2 :  Sistem menampilkan transaksi pembelian |
|  | Langkah 3 :  Admin menginputkan data pembelian | Langkah 4 :  Sistem menyimpan data tersebut |
| Bidang Alternatif 1 | Langkah 3:  Admin memilih data pembelian | Langkah 4 :  Sistem menampilkan data tersebut |
|  | Langkah 5 :  Admin merubah data pembelian | Langkah 6 :  Sistem mengupdate data tersebut |
| Bidang Alternatif 2 | Langkah 3:  Admin memilih data pembelian | Langkah 4 :  Sistem menampilkan data tersebut |
|  | Langkah 5 :  Admin menghapus data pembelian | Langkah 6 :  Sistem menghapus data tersebut |
| Kesimpulan | Admin mempunyai peran untuk mengubah data yang ada dalam data pembelian. | |
| Post Kondisi | Perubahan data yang ada di transaksi pembelian. | |

Tabel 4.3 Sekenario *Use Case* Mengelola Supplier

|  |  |  |
| --- | --- | --- |
| Nama Use Case | Mengelola Supplier | |
| Pelaku Sistem Utama | Admin | |
| Diskripsi | *Use case* ini merupakan proses untuk melakukan pengelolaan data barang masuk dan pencatatan supplier. Dalam proses dapat dilakukan proses tambah, edit supplier | |
| Prakondisi | Admin telah masuk ke dalam menu utama | |
| Bidang khas event: | Kegiatan Aktor | Respon Sistem |
|  | Langkah 1 :  Admin memilih menu data supplier | Langkah 2 :  Sistem menampilkan data supplier |
|  | Langkah 3 :  Admin menginputkan data supplier | Langkah 4 :  Sistem menyimpan data tersebut |
| Bidang Alternatif 1 | Langkah 3:  Admin memilih data supplier | Langkah 4 :  Sistem menampilkan data tersebut |
|  | Langkah 5 :  Admin merubah data supplier | Langkah 6 :  Sistem mengupdate data tersebut |
| Bidang Alternatif 2 | Langkah 3:  Admin memilih data supplier | Langkah 4 :  Sistem menampilkan data tersebut |
|  | Langkah 5 :  Admin menghapus data supplier | Langkah 6 :  Sistem menghapus data tersebut |
| Kesimpulan | Admin mempunyai peran untuk mengubah data yang ada dalam data supplier. | |
| Post Kondisi | Perubahan data yang ada di data supplier. | |

Tabel 4.4 Sekenario *Use Case* Mengelola Produk

|  |  |  |
| --- | --- | --- |
| Nama Use Case | Mengelola Produk | |
| Pelaku Sistem Utama | Admin | |
| Diskripsi | *Use case* ini merupakan proses untuk melakukan pengelolaan data barang. Dalam proses dapat dilakukan proses tambah, edit barang | |
| Prakondisi | Admin telah masuk ke dalam menu utama | |
| Bidang khas event: | Kegiatan Aktor | Respon Sistem |
|  | Langkah 1 :  Admin memilih menu data barang | Langkah 2 :  Sistem menampilkan data barang |
|  | Langkah 3 :  Admin menginputkan data barang | Langkah 4 :  Sistem menyimpan data tersebut |
| Bidang Alternatif 1 | Langkah 3:  Admin memilih data barang | Langkah 4 :  Sistem menampilkan data tersebut |
|  | Langkah 5 :  Admin merubah data barang | Langkah 6 :  Sistem mengupdate data tersebut |
| Bidang Alternatif 2 | Langkah 3:  Admin memilih data barang | Langkah 4 :  Sistem menampilkan data tersebut |
|  | Langkah 5 :  Admin menghapus data barang | Langkah 6 :  Sistem menghapus data tersebut |
| Kesimpulan | Admin mempunyai peran untuk mengubah data yang ada dalam data barang. | |
| Post Kondisi | Perubahan data yang ada di data barang. | |

Tabel 4.5 Sekenario *Use Case* Mengelola Laporan Perpetual

|  |  |  |
| --- | --- | --- |
| Nama Use Case | Mengelola Laporan Perpetual | |
| Pelaku Sistem Utama | Admin | |
| Diskripsi | Use case ini merupakan pencatatan pelaporan data penjualan dan pembelian dalam metode perpetual | |
| Prakondisi | User telah masuk kedalam menu utama | |
| Bidang khas event: | Kegiatan Aktor | Respon Sistem |
|  | Langkah 1 :  Admin memilih menu laporan | Langkah 2 :  Sistem merespon dengan menampilkan menu laporan |
|  | Langkah 3 :  Admin memilih laporan data | Langkah 4 :  Sistem menampilkan data tersebut |
|  | Langkah 5 :  Admin mencetak laporan data pencatatan perpetual | Langkah 6 :  Laporan data pelanggan tercetak |
| Bidang alternatif 1 | Langkah 3 :  Admin memilih menu laporan perpetual | Langkah 4 :  Sistem menampilkan form tersebut |
|  | Langkah 5 :  Admin memilih laporan perpetual dalam range tanggal | Langkah 6 :  Sistem menampilkan data tersebut |
|  | Langkah 7 :  Admin mencetak laporan perpetual | Langkah 9 :  Laporan perpetual tercetak |
| Kesimpulan | Dalam laporan ini untuk melihat data keuangan pembelian, penjualan, dan perhitungan laba | |
| Post Kondisi | Menampilkan laporan perpetual | |

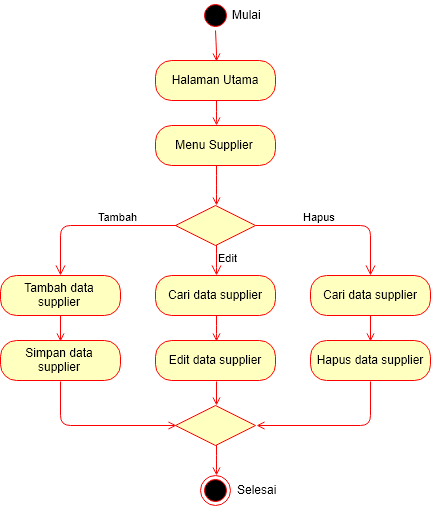
Tabel 4.6 Sekenario *Use Case* Mengelola Laporan Periodik

|  |  |  |
| --- | --- | --- |
| Nama Use Case | Mengelola Laporan Periodik | |
| Pelaku Sistem Utama | Admin | |
| Diskripsi | Use case ini merupakan pencatatan pelaporan data penjualan dan pembelian dalam metode periodik | |
| Prakondisi | User telah masuk kedalam menu utama | |
| Bidang khas event: | Kegiatan Aktor | Respon Sistem |
|  | Langkah 1 :  Admin memilih menu laporan | Langkah 2 :  Sistem merespon dengan menampilkan menu laporan |
|  | Langkah 3 :  Admin memilih laporan data | Langkah 4 :  Sistem menampilkan data tersebut |
|  | Langkah 5 :  Admin mencetak laporan data pencatatan periodik | Langkah 6 :  Laporan data pelanggan tercetak |
| Bidang alternatif 1 | Langkah 3 :  Admin memilih menu laporan periodik | Langkah 4 :  Sistem menampilkan form tersebut |
|  | Langkah 5 :  Admin memilih laporan perpetual dalam range tanggal | Langkah 6 :  Sistem menampilkan data tersebut |
|  | Langkah 7 :  Admin mencetak laporan periodik | Langkah 9 :  Laporan periodik tercetak |
| Kesimpulan | Dalam laporan ini untuk melihat data keuangan pembelian, penjualan, dan perhitungan laba | |
| Post Kondisi | Menampilkan laporan periodik | |

* 1. *Activity* *Diagram*

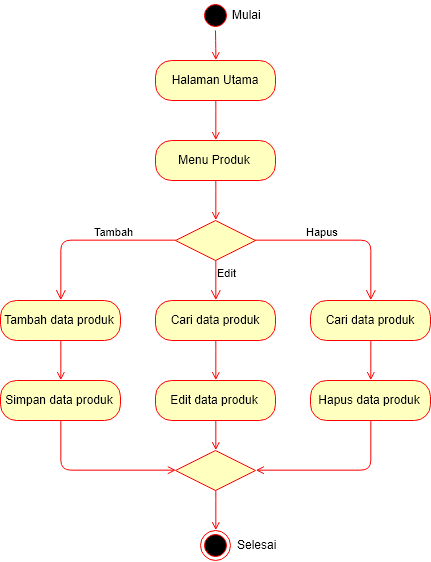
*Activity diagram* memberikan suatu gambaran ilustrasi alur dari setiap fungsi yang ada pada sistem.

1. *Activity diagram* mengelola supplier



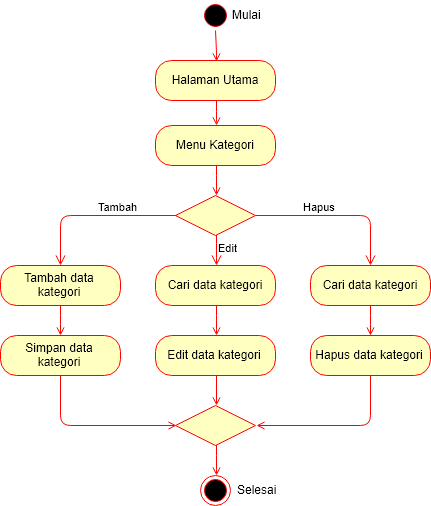
Gambar 4.3 *Activity diagram* mengelola supplier

1. *Activity diagram* mengelola data produk



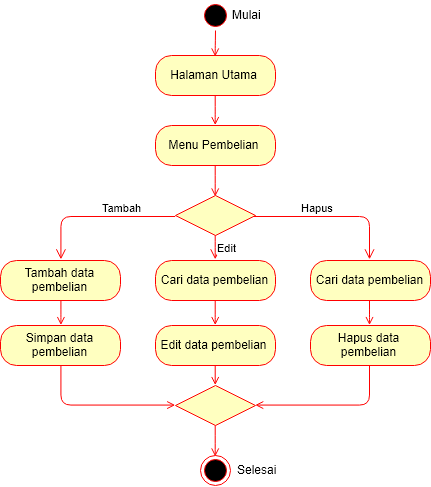
Gambar 4.4 *Activity diagram* mengelola data produk

1. *Activity diagram* mengelola kategori



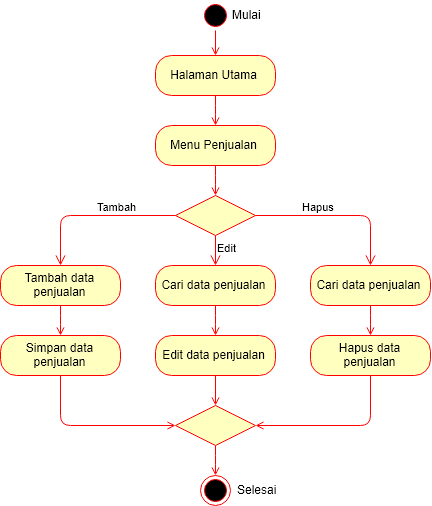
Gambar 4.5 *Activity diagram* mengelola kategori

1. *Activity diagram* mengelola pembelian



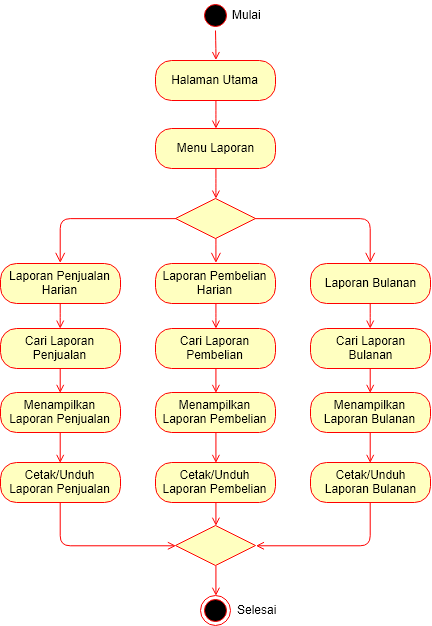
Gambar 4.6 *Activity diagram* mengelola pembelian

1. *Activity diagram* mengelola penjualan



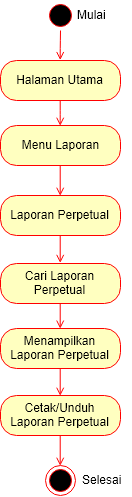
Gambar 4.7 *Activity diagram* mengelola penjualan

1. *Activity diagram* mengelola laporan



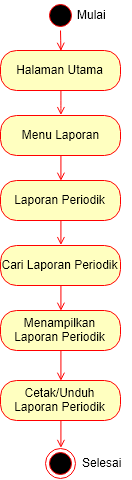
Gambar 4.8 *Activity diagram* mengelola laporan

1. *Activity diagram* mengelola laporan perpetual



Gambar 4.9 *Activity diagram* mengelola laporan perpetual

1. *Activity diagram* mengelola laporan periodik

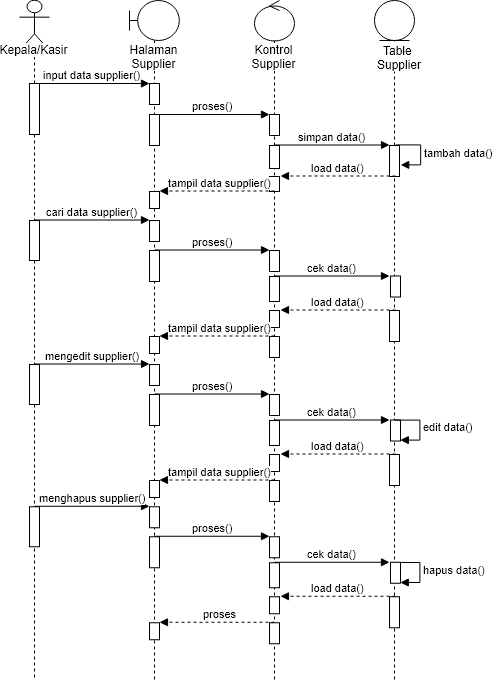


Gambar 4.10 *Activity diagram* mengelola laporan periodik

* 1. *Sequence Diagram*

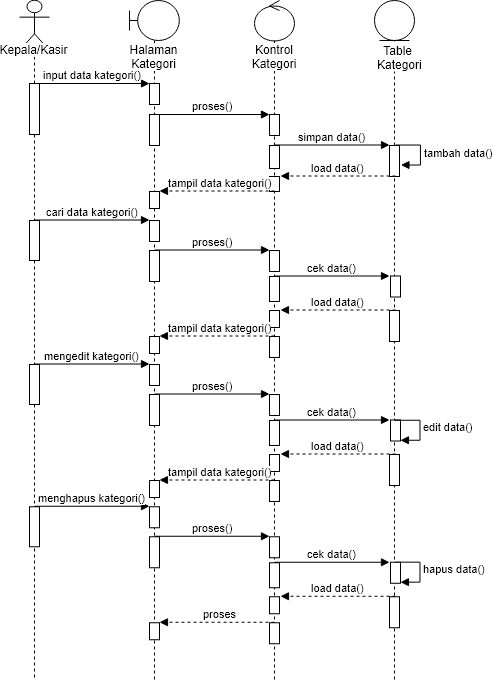
*Sequence diagram* menjelaskan interaksi antar objek-objek yang disusun dalam urutan waktu. Dalam sistem ini terdapat *sequence diagram* yang merupakan penjabaran dari masing-masing *usecase* yang ada pada *usecase diagram*.

1. *sequence diagram* supplier



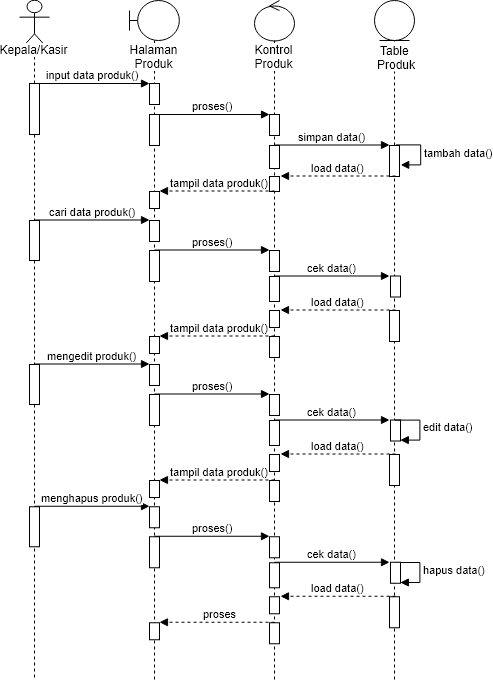
Gambar 4.11 *sequence diagram* supplier

1. *sequence diagram* kategori



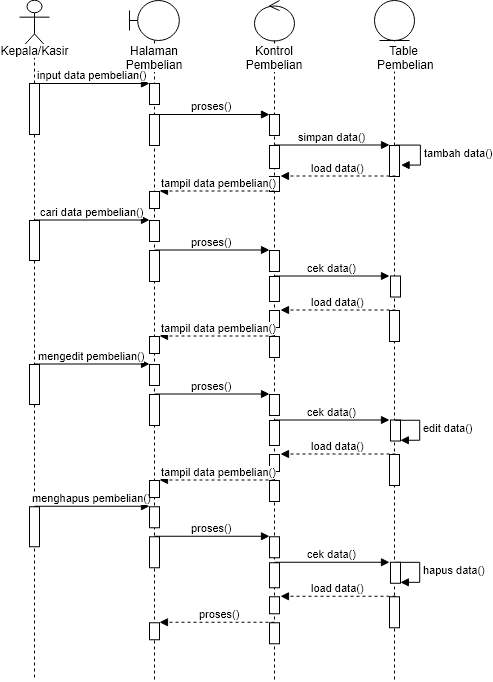
Gambar 4.12 *sequence diagram* kategori

1. *sequence diagram* produk



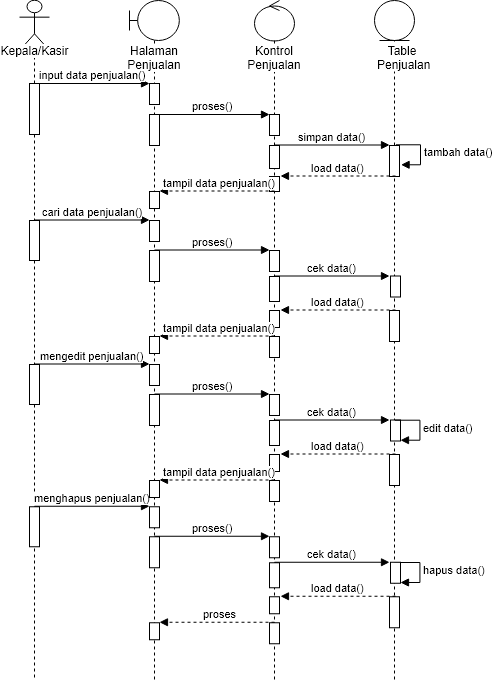
Gambar 4.13 *sequence diagram* produk

1. *sequence diagram* pembelian



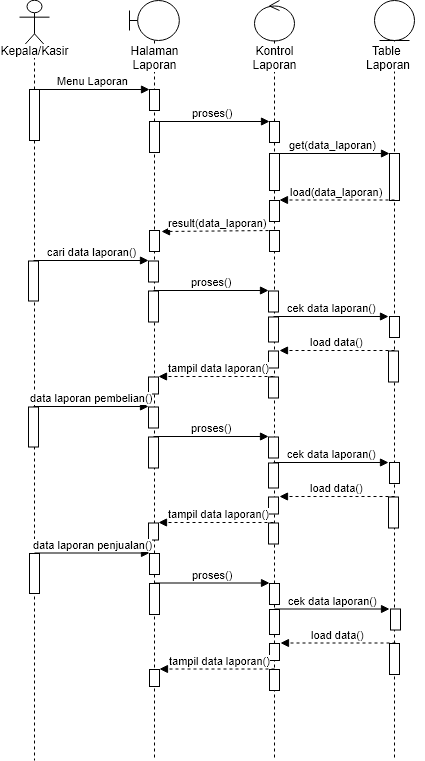
Gambar 4.14 *sequence diagram* pembelian

1. *sequence diagram* penjualan



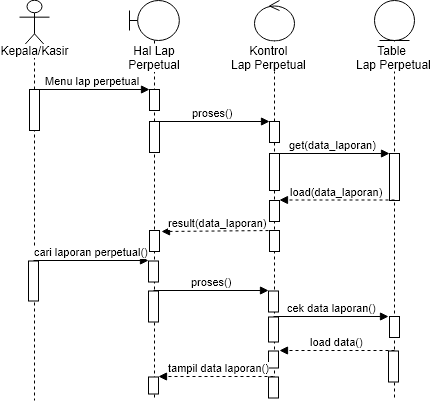
Gambar 4.15 *sequence diagram* penjualan

1. *sequence diagram* laporan



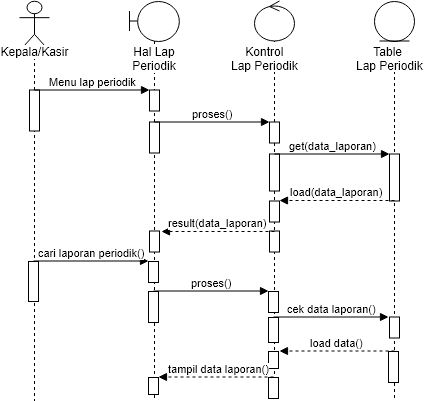
Gambar 4.16 *sequence diagram* laporan

1. *sequence diagram* laporan perpetual



Gambar 4.17 *sequence diagram* laporan perpetual

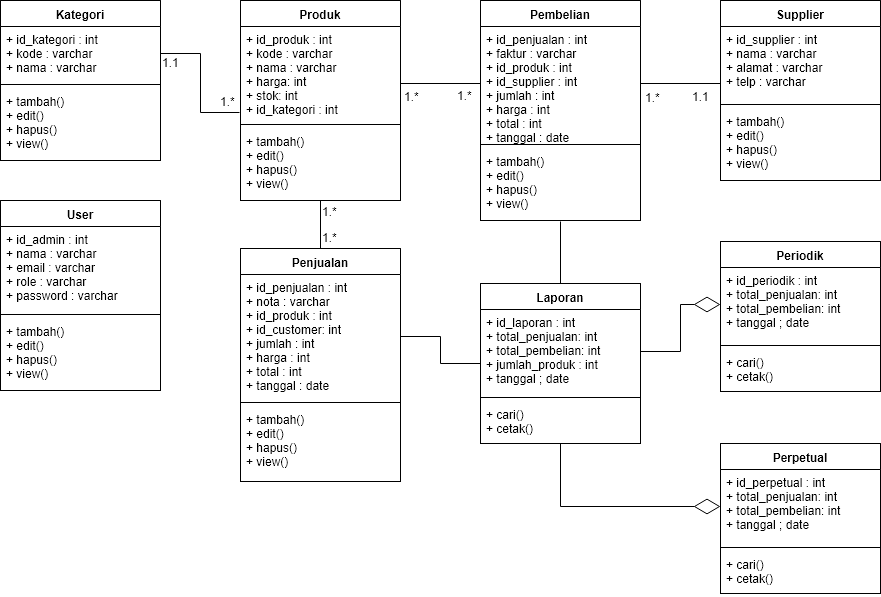
1. *sequence diagram* laporan periodik



Gambar 4.18 *sequence diagram* laporan periodik

* 1. *Class Diagram*

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



Gambar 4.19 *class diagram*

1. **Perancangan *Database***

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

1. Rancangan Tabel Produk

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

Tabel 4.7 Perancangan Tabel Produk

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

1. Rancangan Tabel Kategori

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

Tabel 4.8 Perancangan Tabel Kategori

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

1. Rancangan Tabel Supplier

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

Tabel 4.8 Perancangan Tabel Supplier

|  |  |  |
| --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** |
| id\_supplier | Int | Primary key |
| nama\_ supplier | Varchar |  |
| alamat\_supplier | Varchar |  |
| telp\_supplier | Varchar |  |

1. Rancangan Tabel Pembelian

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

Tabel 4.8 Perancangan Tabel Pembelian

|  |  |  |
| --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** |
| id\_pembelian | Int | Primary key |
| faktur\_ pembelian | Varchar | Index key |
| id\_produk | Int | Foreign key |
| id\_supplier | Int | Foreign key |
| jumlah\_ pembelian | Int |  |
| harga\_ pembelian | Int |  |
| total\_ pembelian | Int |  |
| tanggal\_ pembelian | Date |  |

1. Rancangan Tabel Penjualan

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

Tabel 4.9 Perancangan Tabel Penjualan

|  |  |  |
| --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** |
| id\_pembelian | Int | Primary key |
| nota\_ penjualan | Varchar | Index key |
| id\_produk | Int | Foreign key |
| jumlah\_ penjualan | Int |  |
| harga\_ penjualan | Int |  |
| total\_ penjualan | Int |  |
| tanggal\_ penjualan | Date |  |

1. Rancangan Tabel Laporan

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

Tabel 4.11 Perancangan Tabel Laporan

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

1. Rancangan Tabel Perpetual

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

Tabel 4.11 Perancangan Tabel Laporan

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

1. Rancangan Tabel Laporan Periodik

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

Tabel 4.11 Perancangan Tabel Laporan

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

1. Rancangan Tabel Pengguna

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

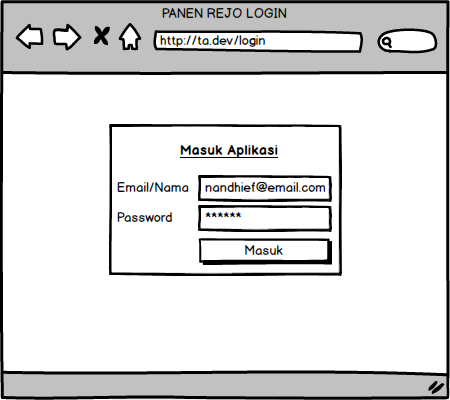
Tabel 4.12 Perancangan Tabel Pengguna

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

1. **Perancangan Antar Muka**

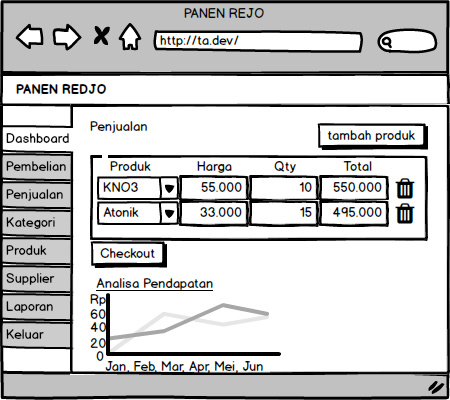
Berdasarkan perancangan aplikasi yang dibuat, maka akan di design perancangan aplikasi. Perancangan ini akan menjadi acuan bagi pihak user yang akan menggunakan aplikasi.

1. Rancangan Halaman Login



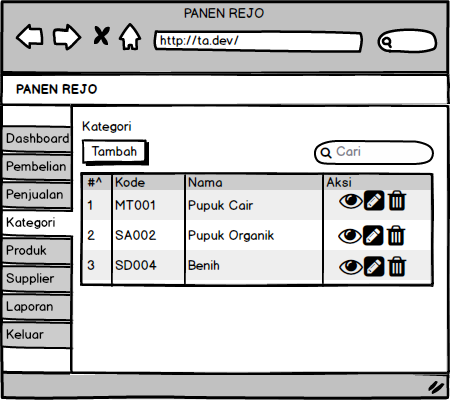
Gambar 4.20 Gambar Rancangan Masuk Aplikasi

1. Rancangan Halaman Dashboard

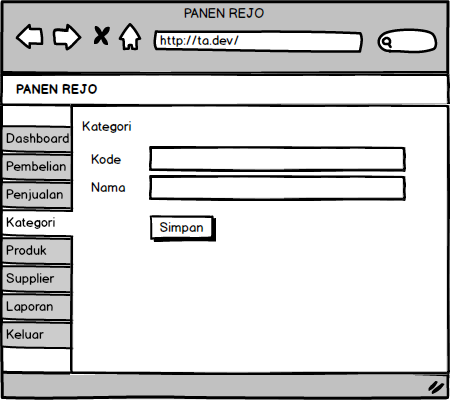


Gambar 4.21 Gambar Rancangan Dashbard

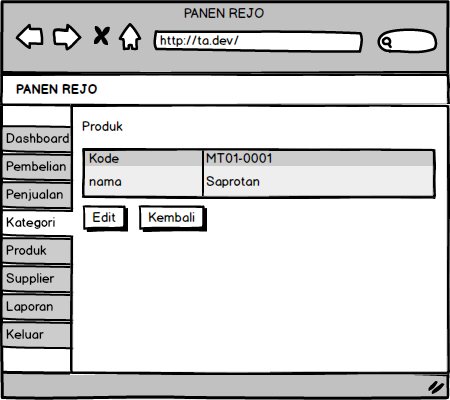
1. Rancangan Halaman Kategori



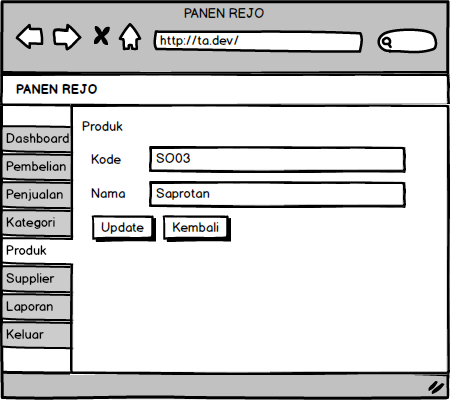
Gambar 4.22 Gambar Rancangan Data Kategori



Gambar 4.23 Gambar Rancangan Tambah Data Kategori

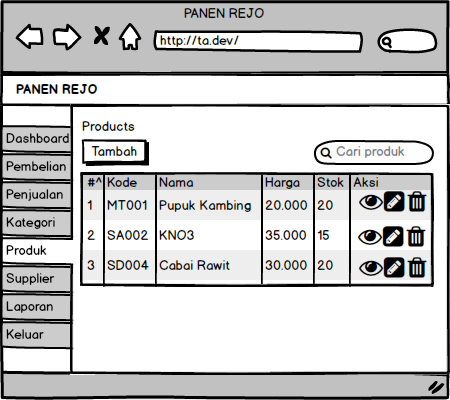


Gambar 4.24 Gambar Rancangan Detail Data Kategori

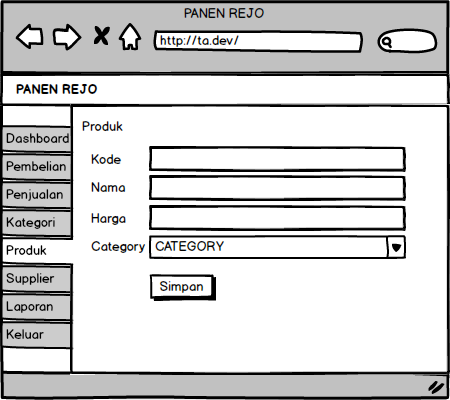


Gambar 4.25 Gambar Rancangan Edit Data Kategori

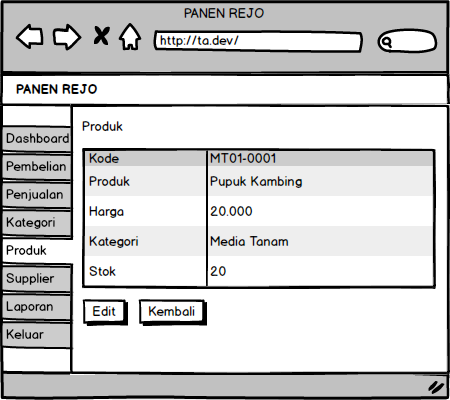
1. Rancangan Halaman Produk



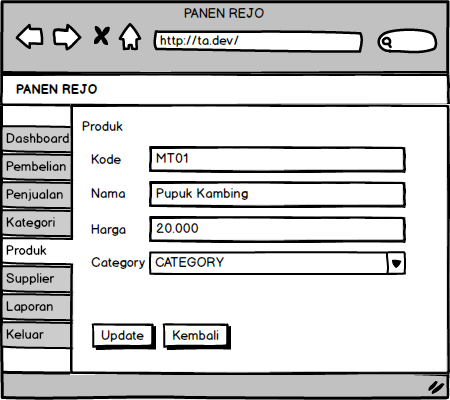
Gambar 4.26 Gambar Rancangan Data Produk



Gambar 4.27 Gambar Rancangan Tambah Data Produk

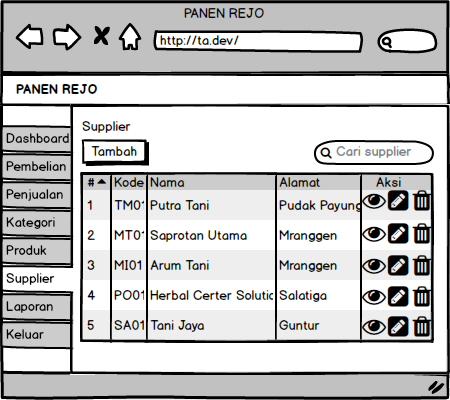


Gambar 4.28 Gambar Rancangan Detail Data Produk

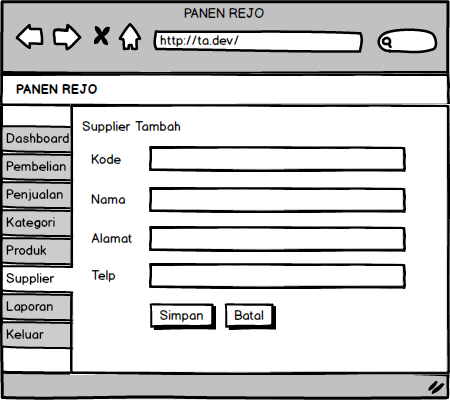


Gambar 4.29 Gambar Rancangan Edit Data Produk

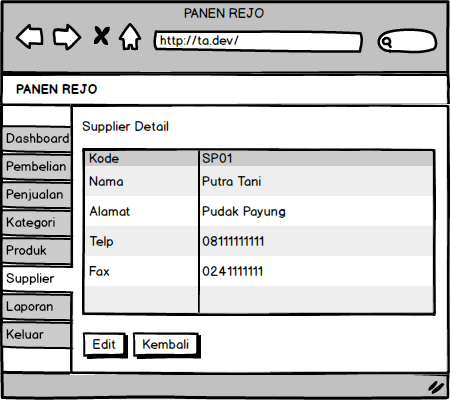
1. Rancangan Halaman Supplier



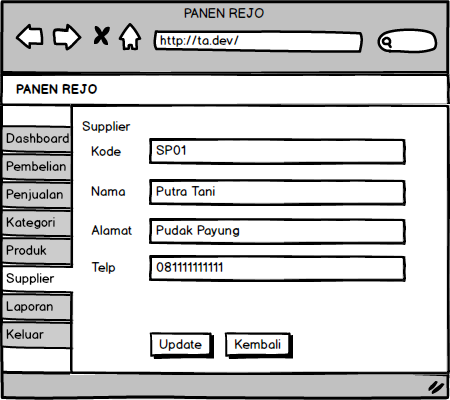
Gambar 4.30 Gambar Rancangan Data Supplier



Gambar 4.31 Gambar Rancangan Tambah Data Supplier

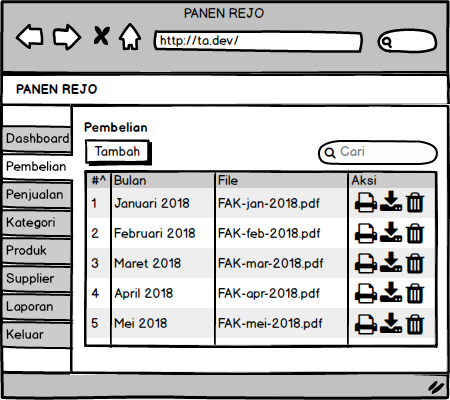


Gambar 4.32 Gambar Rancangan Detail Data Supplier

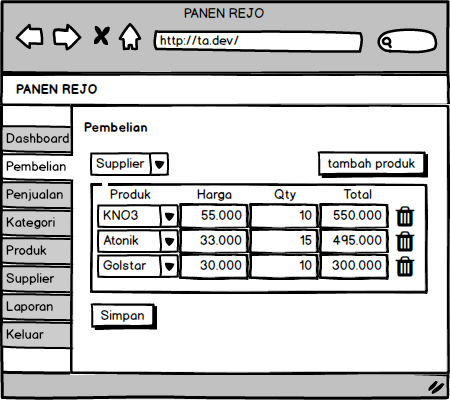


Gambar 4.33 Gambar Rancangan Edit Data Supplier

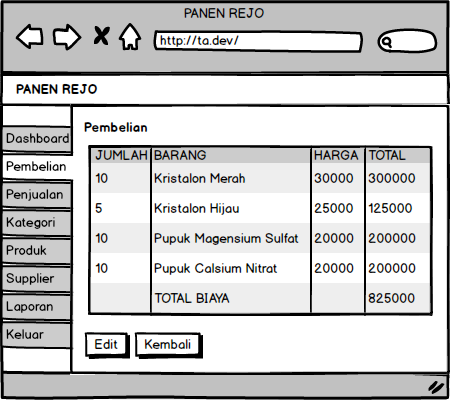
1. Rancangan Halaman Pembelian



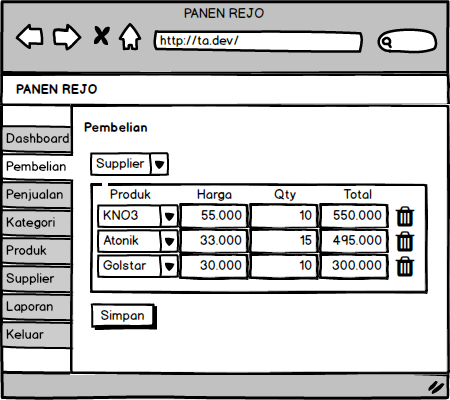
Gambar 4.34 Gambar Rancangan Data Pembelian



Gambar 4.35 Gambar Rancangan Tambah Data Pembelian

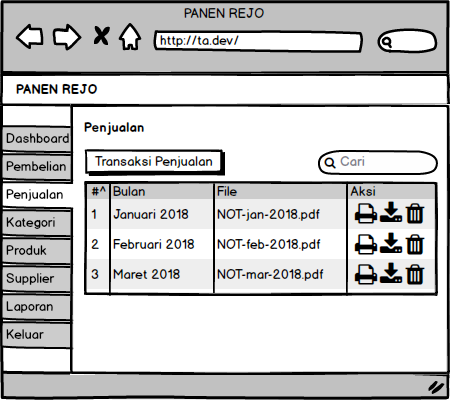


Gambar 4.36 Gambar Rancangan Detail Data Pembelian

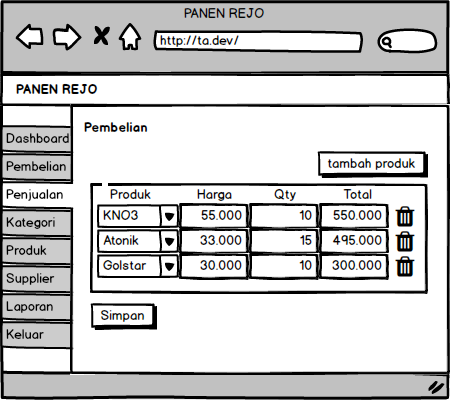


Gambar 4.37 Gambar Rancangan Edit Data Pembelian

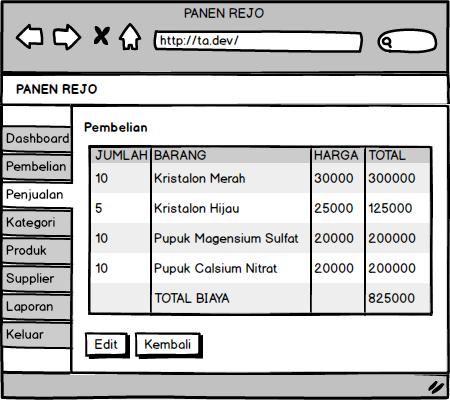
1. Rancangan Halaman Penjualan



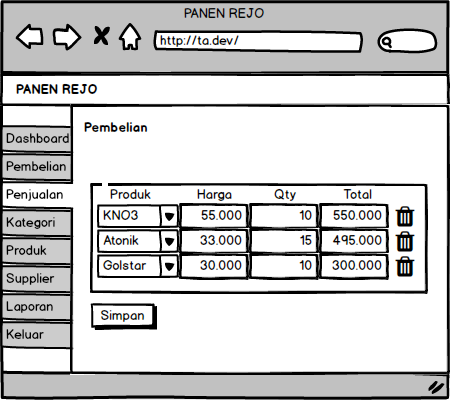
Gambar 4.38 Gambar Rancangan Data Penjualan



Gambar 4.39 Gambar Rancangan Tambah Data Penjualan

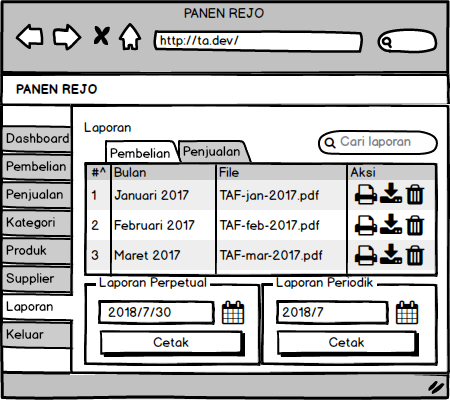


Gambar 4.40 Gambar Rancangan Detail Data Penjualan



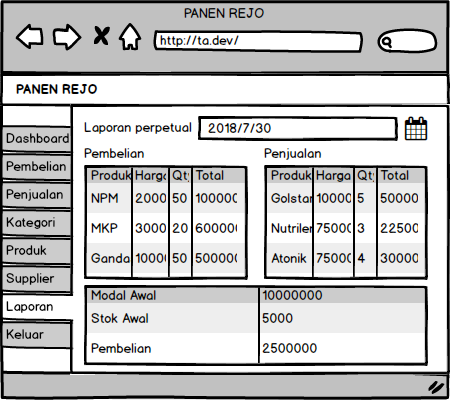
Gambar 4.41 Gambar Rancangan Edit Data Penjualan

1. Rancangan Halaman Laporan



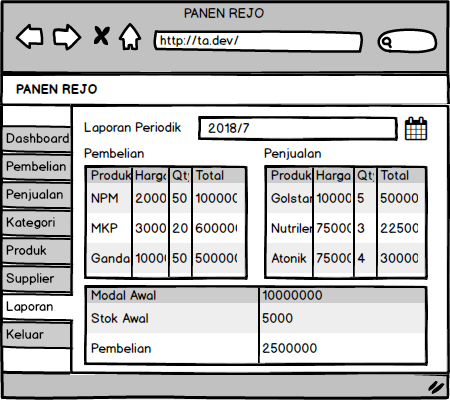
Gambar 4.42 Gambar Rancangan Data Laporan

1. Rancangan Halaman Laporan Perpetual



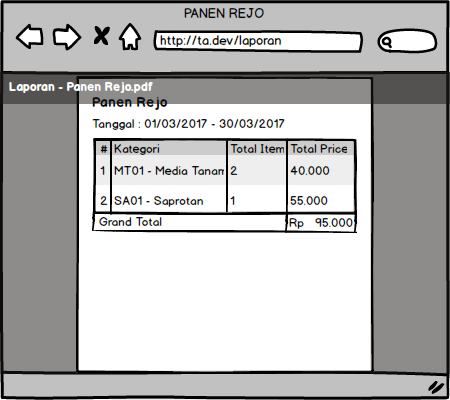
Gambar 4.43 Gambar Rancangan Data Laporan Perpetual

1. Rancangan Halaman Laporan Periodik



Gambar 4.44 Gambar Rancangan Data Laporan Periodik

1. Rancangan Halaman Cetak Laporan



Gambar 4.45 Gambar Rancangan Cetak Laporan

# **BAB V**

# **IMPLEMENTASI SISTEM**

1. **Implementasi Sistem**

Tahap implementasi merupakan tahap kelanjutan dari kegiatan perancangan sistem. Tahapan ini juga merupakan tahap meletakkan sistem supaya siap untuk dioperasikan dan dapat dipandang sebagai usaha untuk mewujudkan sistem yang telah dirancang.

Pada bab ini akan diuraikan mengenai implementasi dari program Implementasi Sistem Informasi Keuangan dengan Metode Sistem Perpetual dan Periodik pada CV Panen Redjo Semarang yang dibuat menggunakan *Framework Laravel* dan *MariaDb* sebagai software aplikasi pengolah database. Pada sistem ini yang dapat mengakses, yaitu Admin/Kasir/Pemilik. Admin dapat mengolah semua menu yang dibutuhkan oleh sistem seperti data produk, data stok, data supplier, data penjualan, data pembelian, dan laporan.

1. Hirarki Input Output Sistem

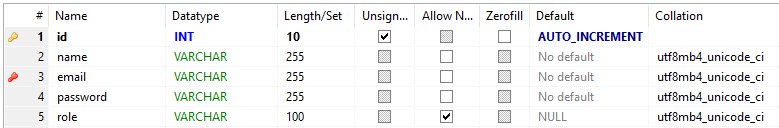


Gambar 5.1 Hirarki Sistem

1. Implementasi Database

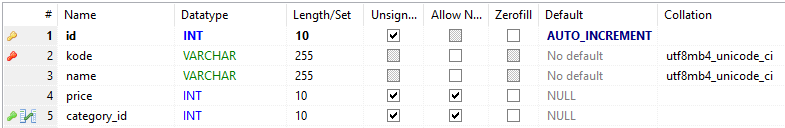
Penulis menggunakan MariaDb untuk menyimpan data, Database diberi nama “suparjo” yang berisi beberapa table, diantaranya table users, products, incomes, outgos, suppliers, reports, perpetuals, dan periodics.

* + 1. Tabel users



Gambar 5.2 Gambar tabel users

* + 1. Tabel products



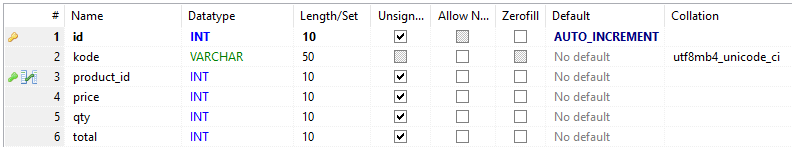
Gambar 5.3 Gambar tabel products

* + 1. Tabel incomes



Gambar 5.4 Gambar tabel incomes

* + 1. Tabel outgos



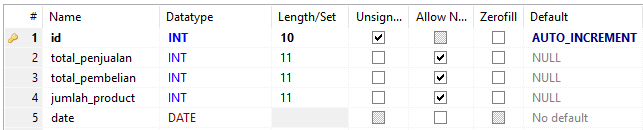
Gambar 5.5 Gambar tabel outgos

* + 1. Tabel suppliers



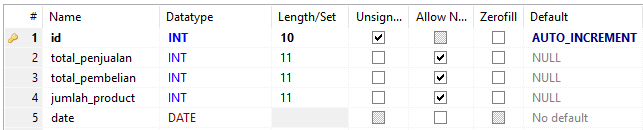
Gambar 5.6 Gambar tabel suppliers

* + 1. Tabel reports



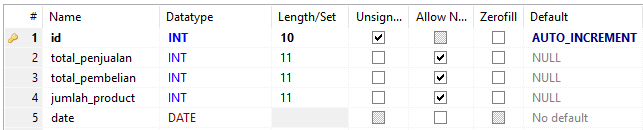
Gambar 5.7 Gambar tabel reports

* + 1. Tabel perpetuals



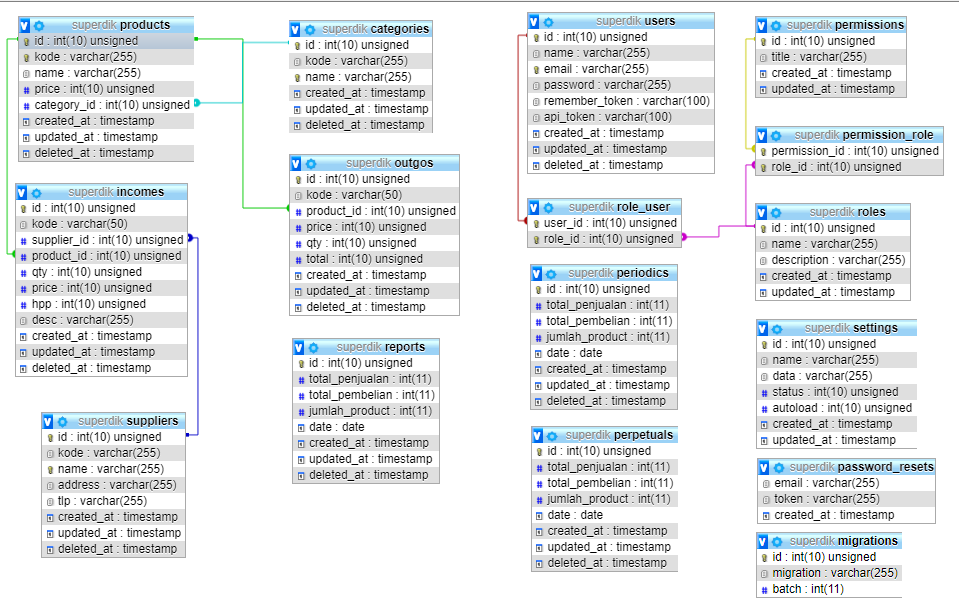
Gambar 5.8 Gambar tabel perpetuals

* + 1. Tabel periodics



Gambar 5.9 Gambar tabel periodics

1. Hasil Gambar Relasi Tabel



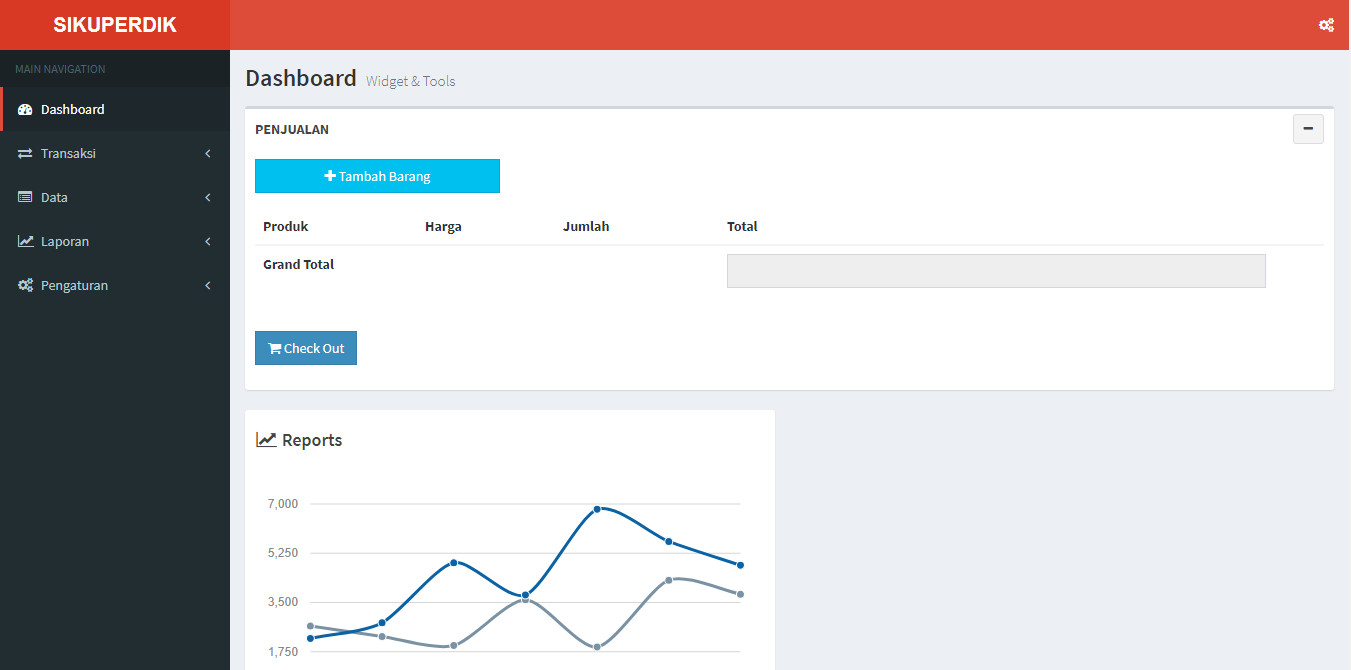
Gambar 5.10 Gambar relasi tabel

1. **Implementasi Tampilan Sistem**

Tampilan aplikasi terdiri dari login, menu utama admin, menu data produk, menu data stok, menu data supplier, menu data penjualan, menu data pembelian, menu dan laporan.

Untuk lebih jelasnya dapat dilihat pada tampilan sebagai berikut:

1. Tampilan Halaman Dashboard

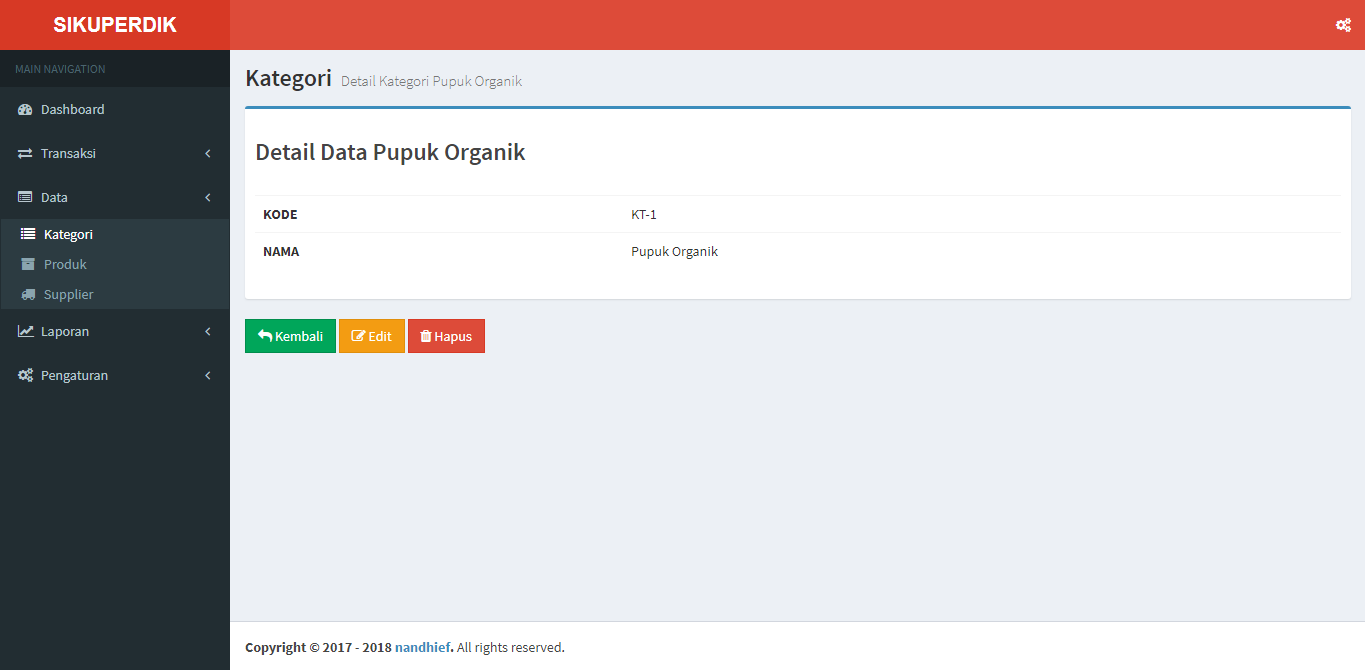


Gambar 5.11 Gambar halaman dashboard

Gambar 5.12 Gambar halaman kategori



Gambar 5.13 Gambar halaman tambah kategori

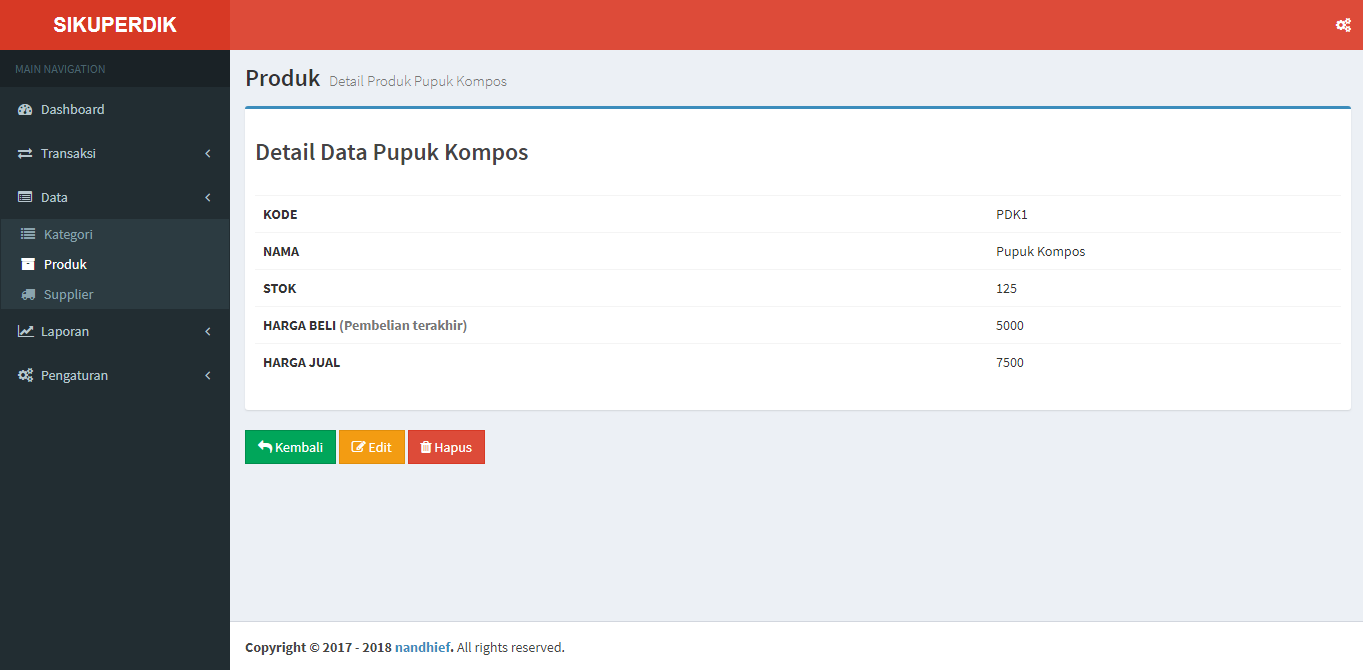


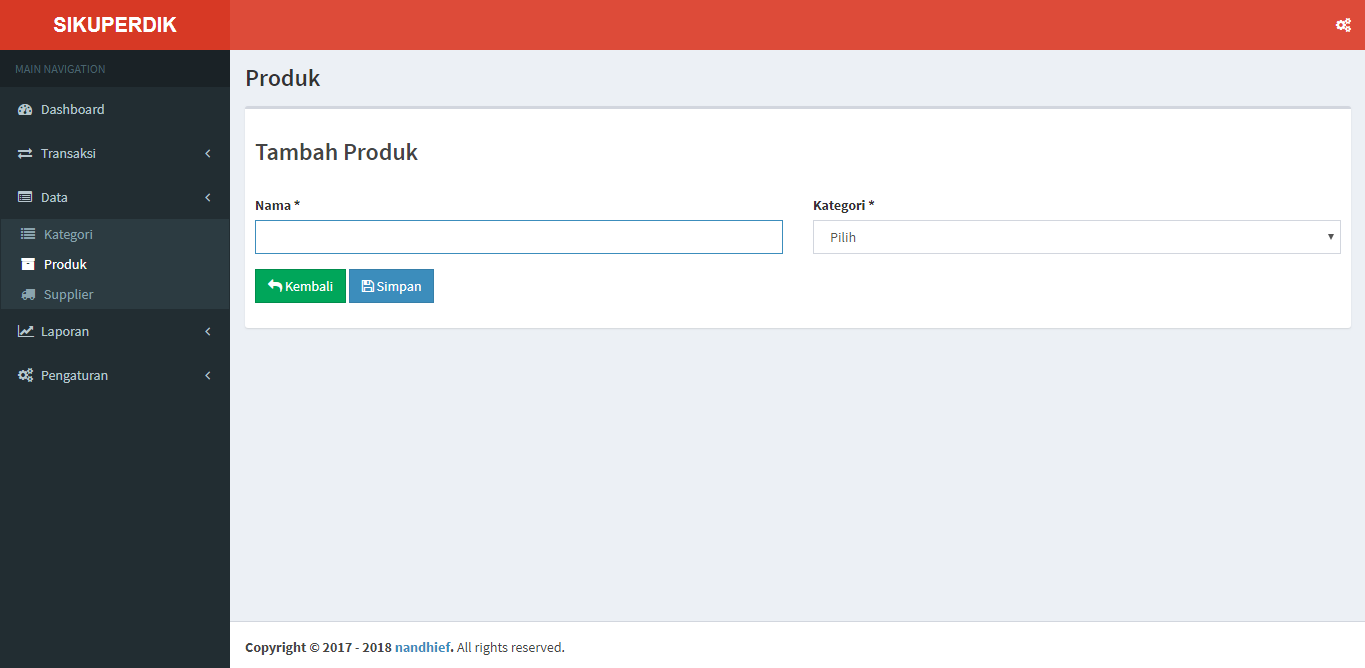
Gambar 5.14 Gambar halaman detail kategori

1. Tampilan Halaman Produk

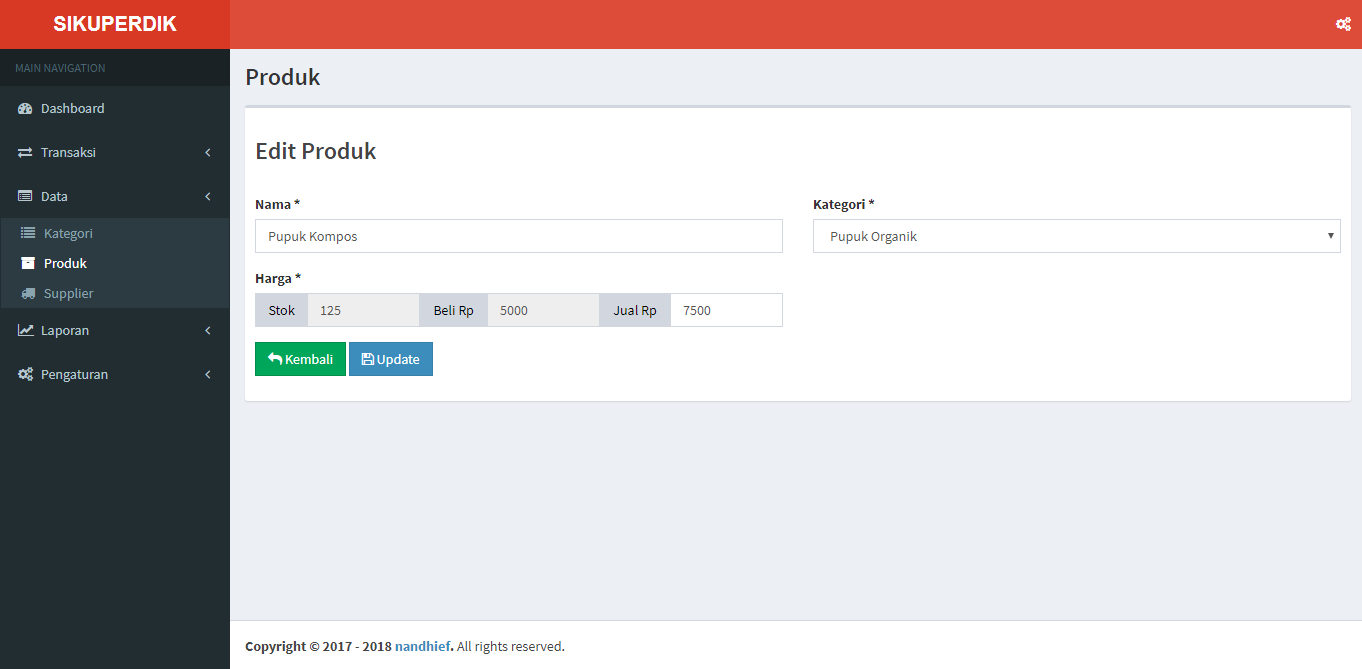


Gambar 5.15 Gambar halaman produk

 Gambar 5.16 Gambar halaman detail produk

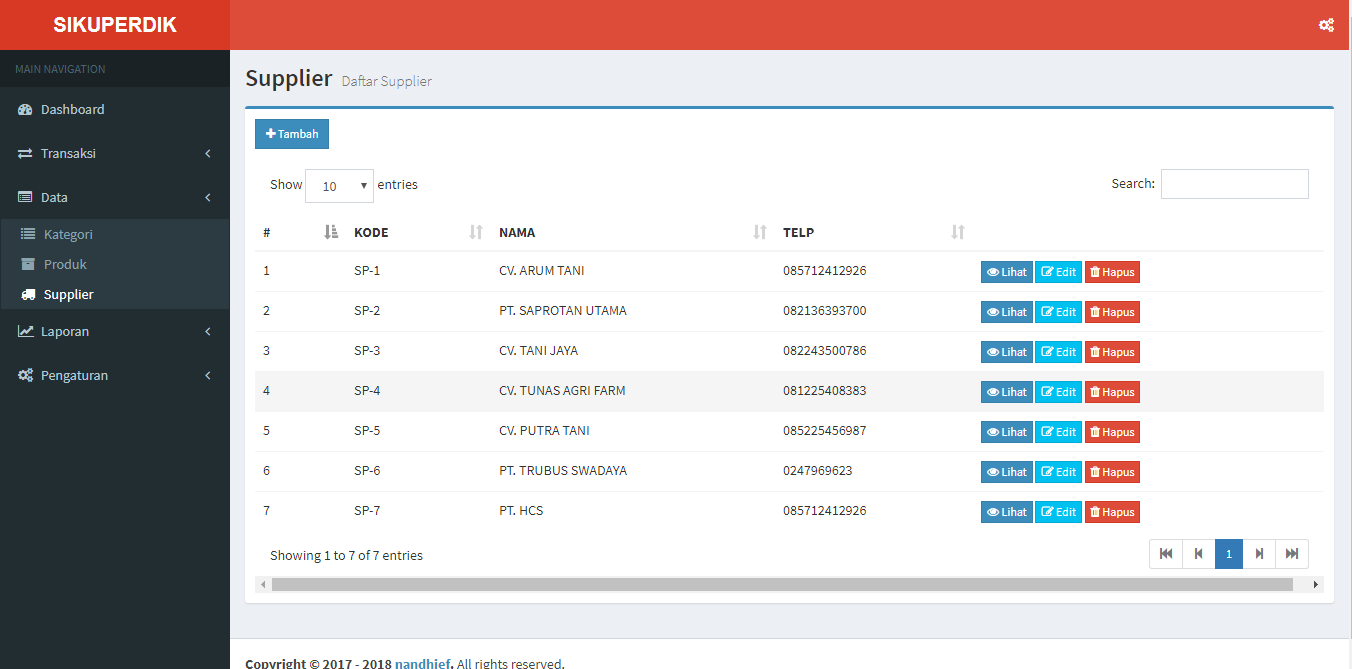


Gambar 5.17 Gambar halaman tambah produk

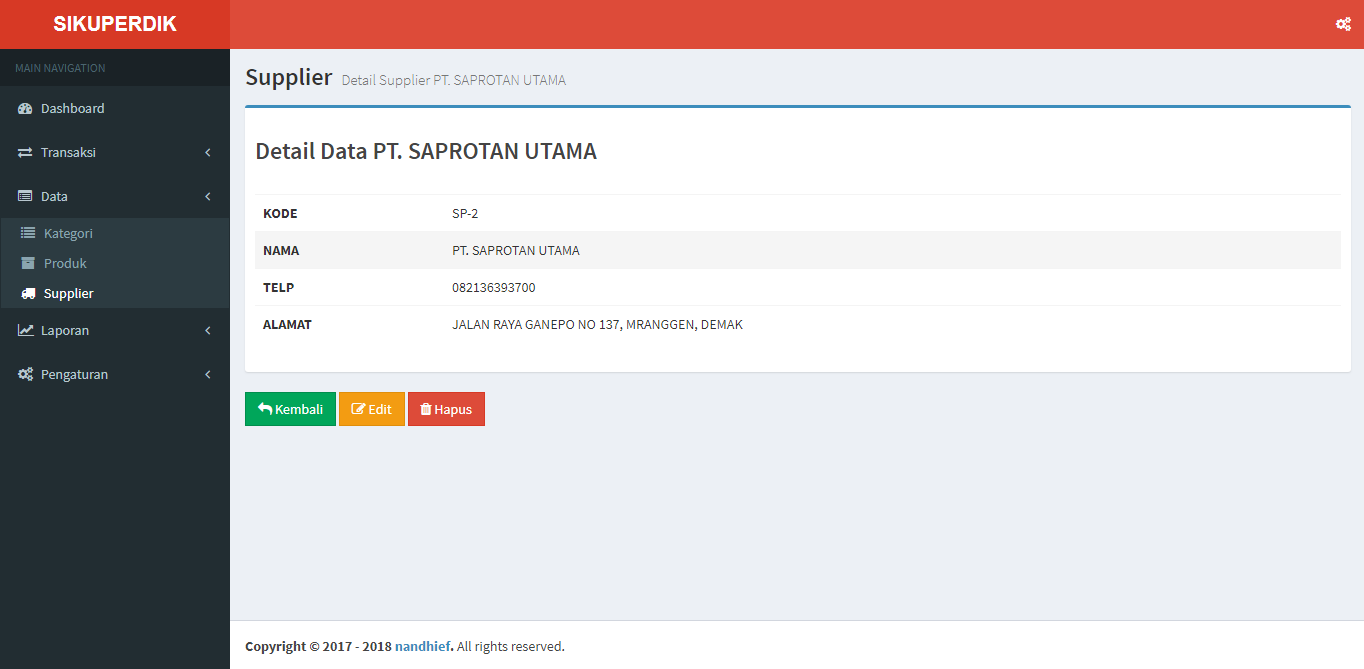


Gambar 5.18 Gambar halaman update produk

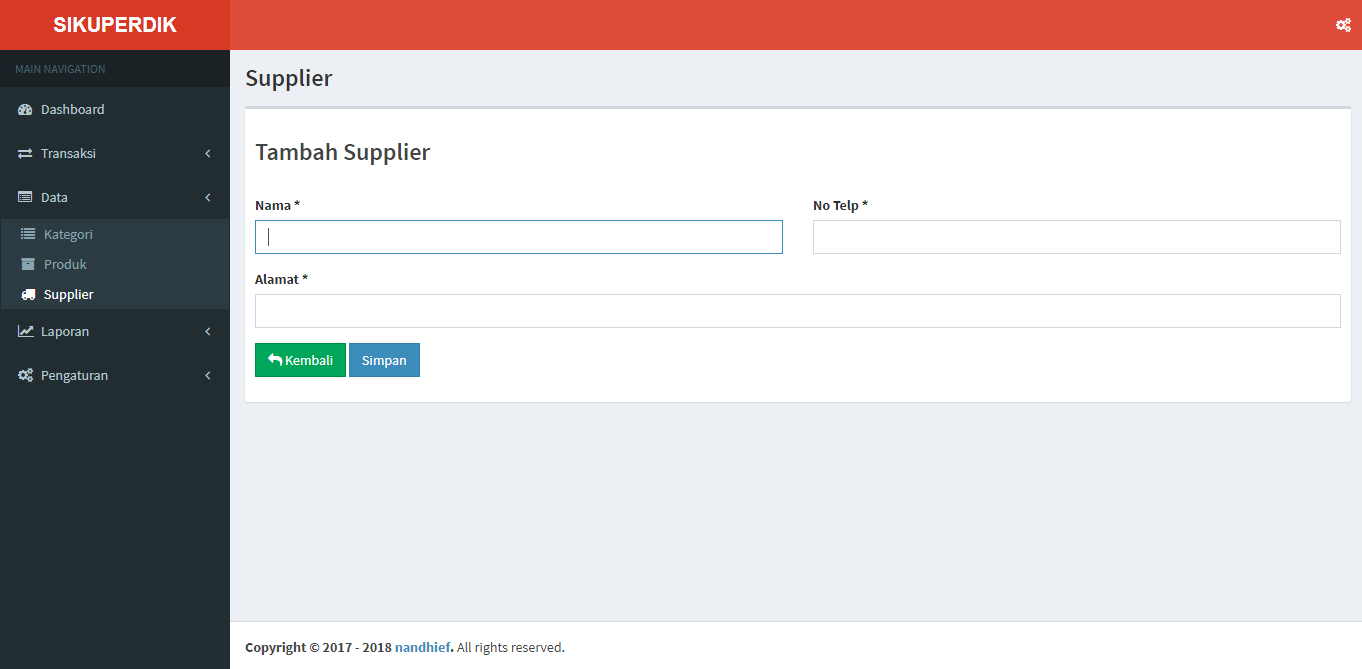
1. Tampilan Halaman Supplier



Gambar 5.19 Gambar halaman supplier

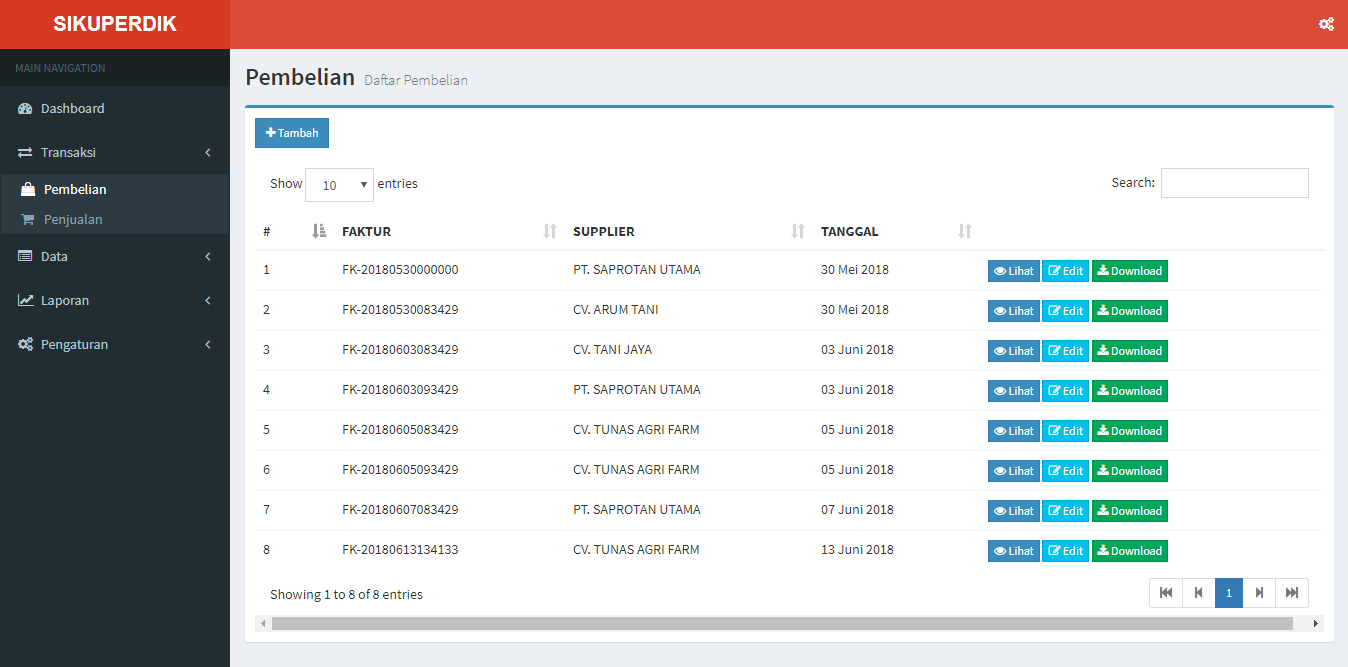


Gambar 5.20 Gambar halaman detail supplier

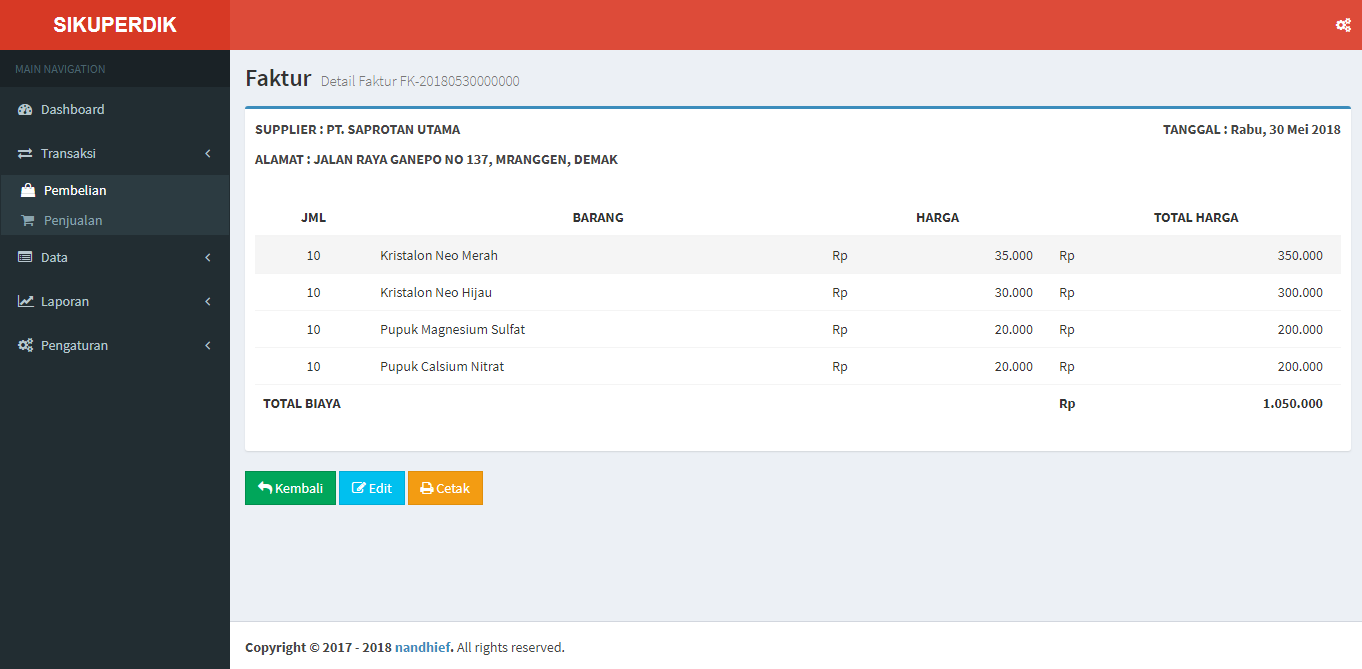


Gambar 5.21 Gambar halaman tambah supplier

1. Tampilan Halaman Pembelian



Gambar 5.22 Gambar halaman pembelian

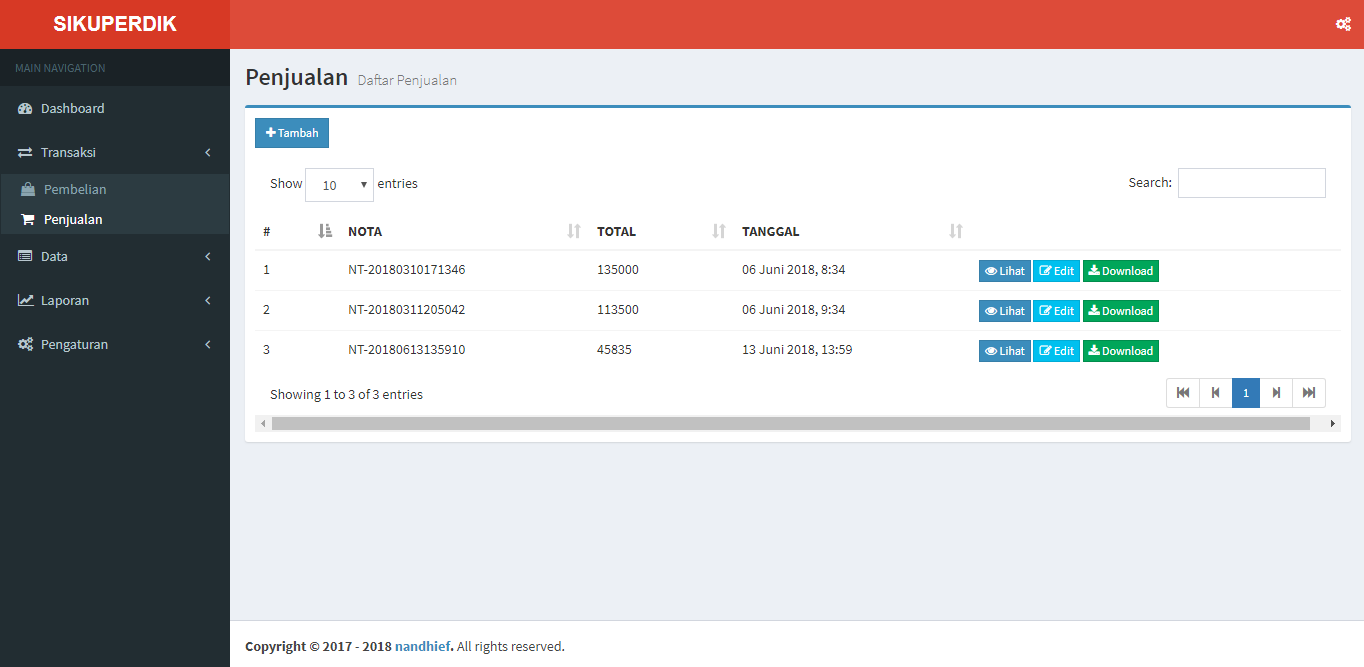


Gambar 5.23 Gambar halaman detail pembelian

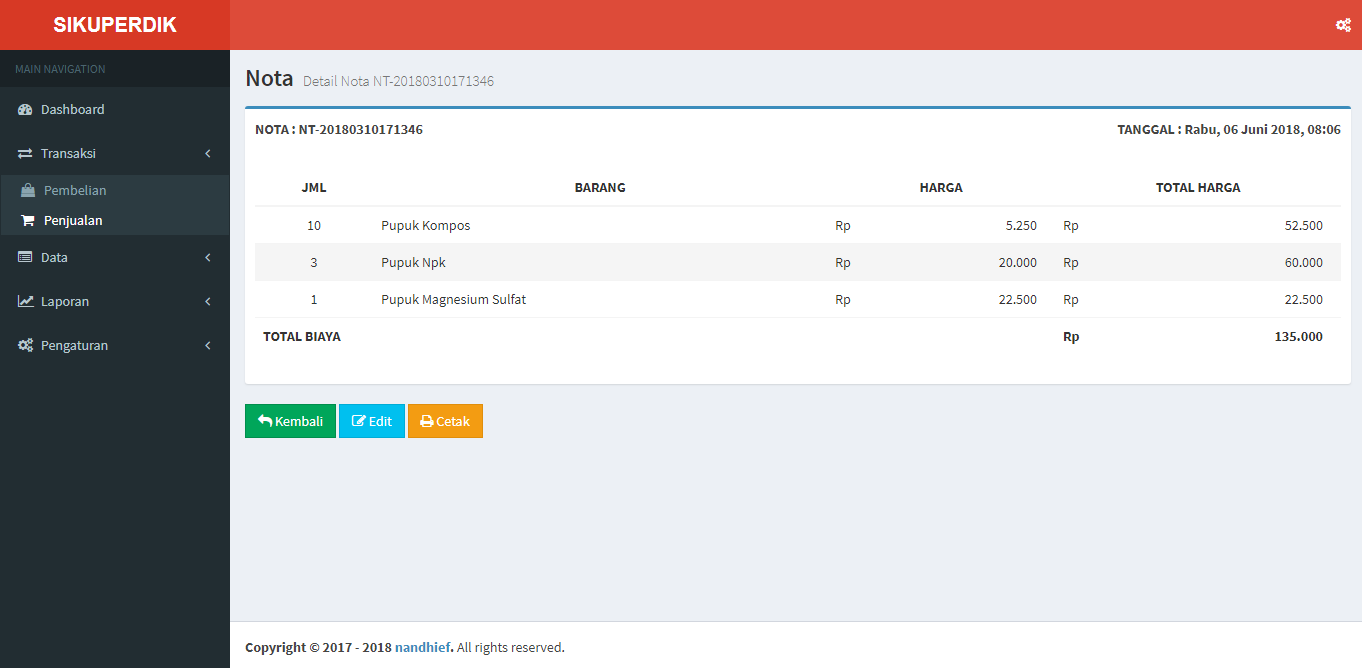


Gambar 5.24 Gambar halaman tambah pembelian

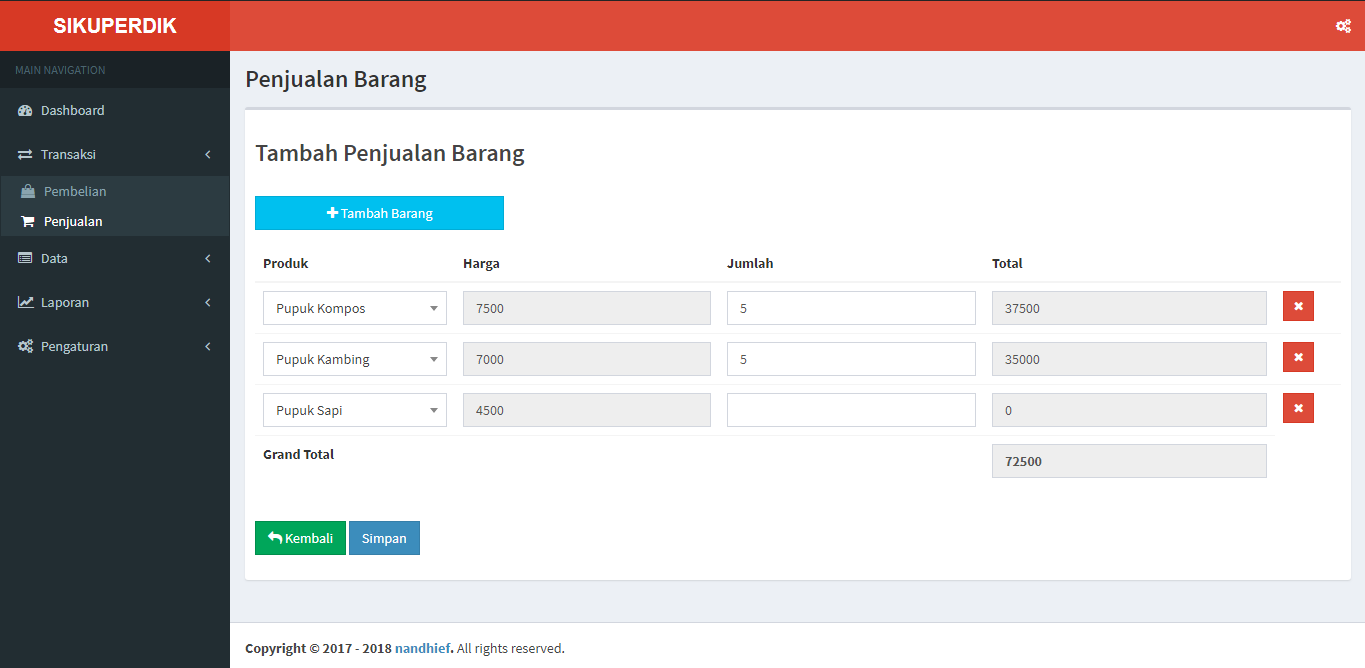
1. Tampilan Halaman Penjualan



Gambar 5.25 Gambar halaman penjualan

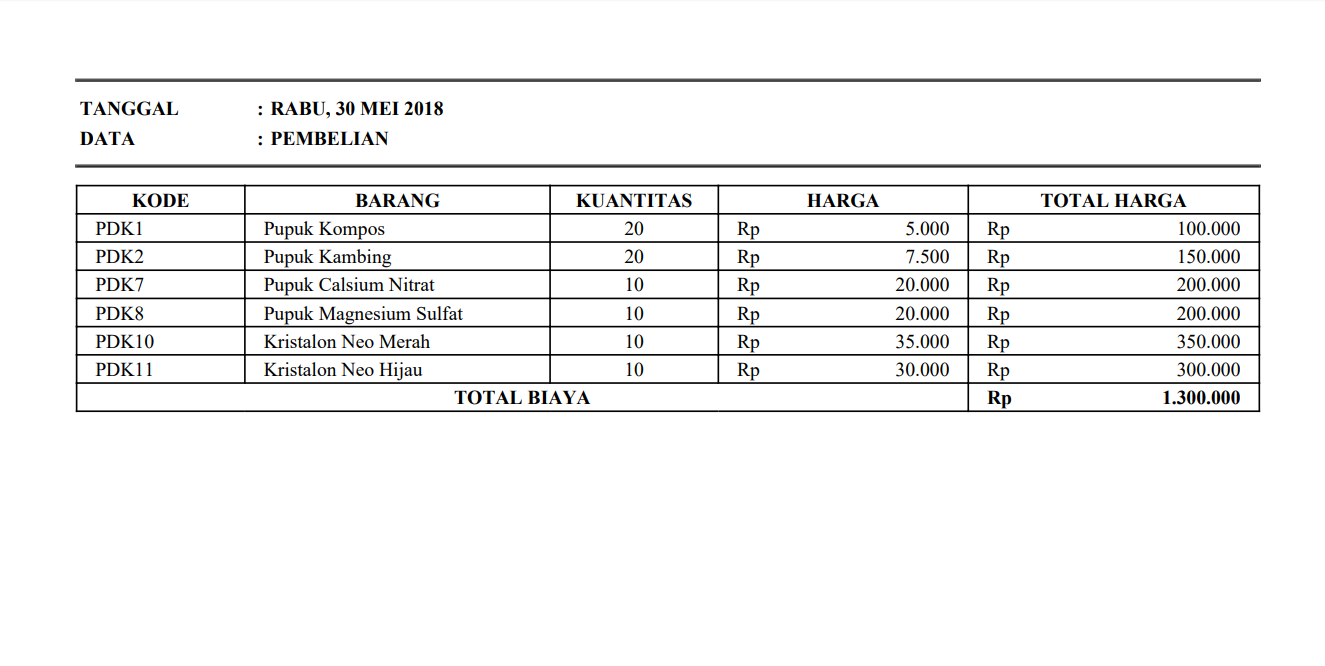


Gambar 5.26 Gambar halaman detail penjualan

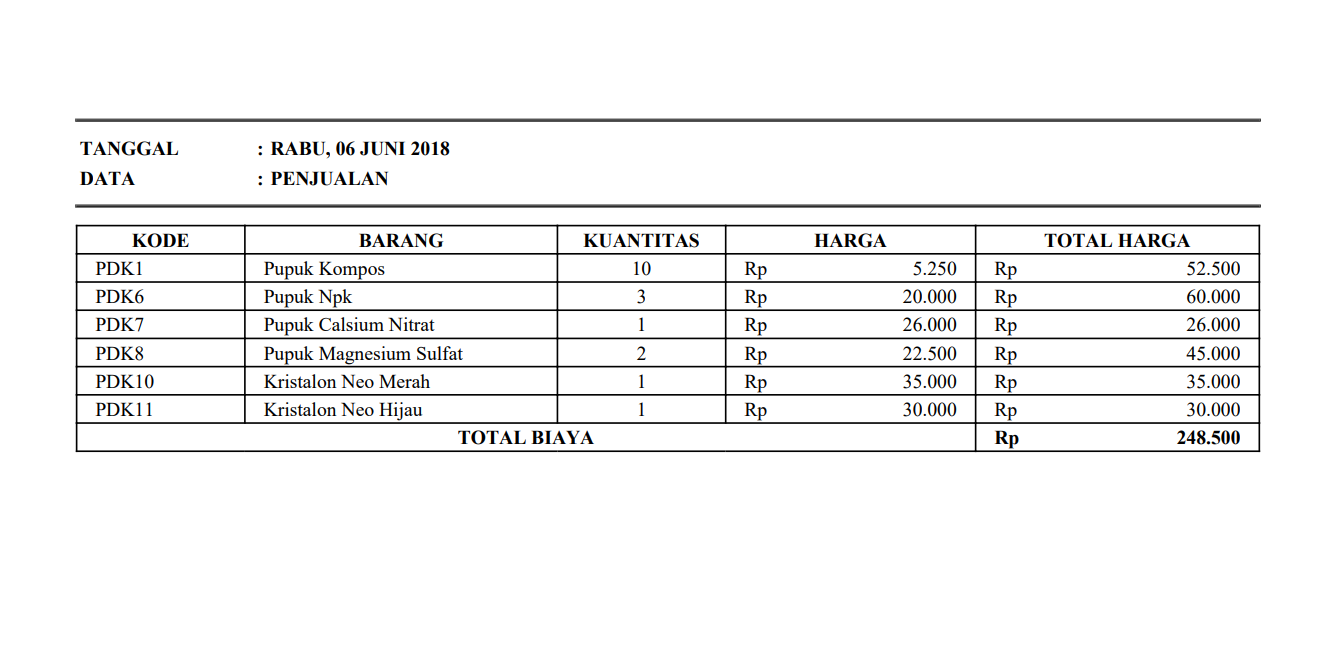
 Gambar 5.27 Gambar halaman tambah penjualan

1. Tampilan Halaman Laporan

 Gambar 5.28 Gambar halaman laporan



Gambar 5.29 Gambar halaman cetak laporan pembelian



Gambar 5.30 Gambar halaman cetak laporan penjualan

1. **Pengujia Sistem**
2. Pengujian White Box
   * + 1. Kode Program Controller Pembelian Produk :

*class IncomeController extends Controller {*

*public function index() {*

*$incomes = Income::with(['supplier'])->groupBy('kode')->get();*

*if (request()->ajax()) {*

*$no = 1;*

*foreach ($incomes as $r) {*

*$data[] = collect($r)->prepend($no++, 'no');*

*}*

*return response()->json(compact('data'), 200);*

*}*

*return view('incomes.index', ['incomes' => $incomes->count()]);*

*}*

*public function create() {*

*$suppliers = Supplier::pluck('name', 'id')*

*->prepend('-- Pilih --', '');*

*$products = Product::select('id', 'name')->get();*

*return view('incomes.create', compact('suppliers', 'products'));*

*}*

*public function store(Request $request) {*

*$this->validate($request, [*

*'supplier\_id' => 'required',*

*'rows' => 'required',*

*]);*

*foreach ($request->rows as $row) {*

*$this->validate($request, [*

*'product\_id' . $row => 'required',*

*'price' . $row => 'required',*

*'qty' . $row => 'required',*

*]);*

*}*

*$faktur = 'FK-' . str\_replace(' ', '', str\_replace(':', '',*

*str\_replace('-', '', \Carbon\Carbon::now())));*

*foreach ($request->rows as $row) {*

*Income::create([*

*'kode' => $faktur,*

*'supplier\_id' => $request->supplier\_id,*

*'product\_id' => $request->{'product\_id' . $row},*

*'price' => $request->{'price' . $row},*

*'qty' => $request->{'qty' . $row},*

*]);*

*}*

*return redirect()->route('incomes.index')->withSuccess('Data Berhasil Disimpan');*

*}*

*public function show($kode) {*

*$incomes = Income::whereKode($kode)->get();*

*return view('incomes.show', compact('incomes'));*

*}*

*public function edit($kode) {*

*$incomes = Income::whereKode($kode)->get();*

*$products = Product::select('id', 'name')->get();*

*if (request()->ajax()) {*

*return $products;*

*}*

*return view('incomes.edit', compact('incomes', 'products'));*

*}*

*public function update(Request $request, $id) {*

*foreach ($request->id as $id) {*

*$this->validate($request, [*

*'price' . $id => 'required',*

*'qty' . $id => 'required',*

*]);*

*$income = Income::find($id);*

*$income->update([*

*'price' => $request->{'price' . $id},*

*'qty' => $request->{'qty' . $id},*

*]);*

*}*

*return redirect()->route('incomes.show', $income->kode)->withSuccess("Berhasil Diupdate");*

*}*

*public function destroy($id) {*

*$income = Income::find($id);*

*$income->delete();*

10

*return redirect()->route('incomes.index')*

*->withSuccess("Berhasil Dihapus");*

*}*

*public function print($kode) {*

*$incomes = Income::whereKode($kode)->get();*

11

*$pdf = \PDF::loadView('incomes.print', compact('incomes'));*

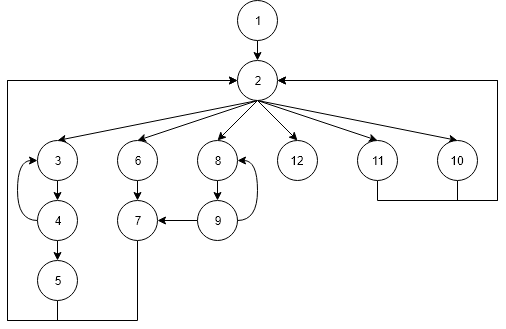
*return $pdf->stream();*

*}*

12

*}*

1. Flowgraph pembelian barang



Gambar 5.31 Gambar flowgraph pembelian produk

Keterangan :

Node 1 : mulai

Node 2 : index pembelian

Node 3 : Form tambah pembelian

Node 4 : Validasi form data input

Node 5 : Menyimpan data

Node 6 : Menampilkan data

Node 7 : Form update pembelian

Node 8 : Validasi form data input

Node 9 : Ubah pembelian

Node 10 : Hapus pembelian

Node 11 : Selesai

1. Cyclomatic Complexity

Cyclomatic Complexity merupakan metode yang digunakan untuk pencarian jalur independen (R) yang memungkinkan. Perhitungan Cyclomatic Complexity sebagai berikut :

V(G) : Cyclomatic Complexity

E : Jumlah edge grafik alir = 19

N : Jumlah node grafik alir = 12

V(G) = E-N+2

= 19 – 12 + 2 = 7

Berdasarkan perhitungan tersebut, maka didapat jumlah jalur independe / *path* (R) sebanyak 10 jalur dengan rincian sebagai berikut :

R1 : 1-2-3-4-5-2-12

R2 : 1-2-3-4-3-4-5-2-12

R3 : 1-2-6-7-2-12

R4 : 1-2-6-7-2-11-12

R5 : 1-2-8-9-7-2-12

R6 : 1-2-8-9-8-9-7-2-12

R7 : 1-2-10-2-12

1. Pengujian Black Box

Pengujian Black Box merupakan pengujian perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan dan keluaran dari perangkat lunak sesuai dengan apa yang dibutuhkan. Dalam kasus sistem pengolahan data nilai akan dilakukan pengujian black box dapat dilihat sebagai berikut:

Tabel 5.1 pengujian black box pembelian

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Fungsi yang diuji | Cara pengujian | Hasil yang diharapkan | Hasil pengujian |
| 1. | Tampilan halaman login admin | Jalankan aplikasi dengan membuka alamat web di browser | Menampilkan halaman login | Berhasil |
| 2. | Halaman utama | Melakukan login kedalam sistem | Menampilkan halaman menu utama | Berhasil |
| 3. | Halaman pembelian | Memilih menu pembelian | Menampilkan data pembelian, Muncul form Tambah data. Jika inputan sesuai, data akan disimpan dan tampil daftar klaim. Apabila inputan tidak lengkap, muncul peri- ngatan untuk melengkapi data | Berhasil |
| 4. | Halaman penjualan | Memilih menu penjualan | Menampilkan data penjualan, Muncul form Tambah data. Jika inputan sesuai, data akan disimpan dan tampil daftar klaim. Apabila inputan tidak lengkap, muncul peri- ngatan untuk melengkapi data | Berhasil |
| 5. | Halaman supplier | Memilih menu supplier | Menampilkan data supplier, Muncul form Tambah data. Jika inputan sesuai, data akan disimpan dan tampil daftar klaim. Apabila inputan tidak lengkap, muncul peri- ngatan untuk melengkapi data | Berhasil |
| 6. | Halaman produk | Memilih menu produk | Menampilkan data produk, Muncul form Tambah data. Jika inputan sesuai, data akan disimpan dan tampil daftar klaim. Apabila inputan tidak lengkap, muncul peri- ngatan untuk melengkapi data | Berhasil |
| 7. | Halaman laporan | Memilih menu laporan | Menampilkan data laporan, muncul form untuk memilih periode, lalu tampil laporan klaim barang berupa file pdf | Berhasil |

1. Pemeliharaan Sistem

Pemeliharaan sistem dilakukan pada saat sistem mulai diimplementasikan atau mulai digunakan oleh user. Pemeliharaan ini dimaksudkan adalah sebagai berikut:

1. Penjagaan sistem

Melakukan pemantauan dan pemeriksaan rutin pada aplikasi yang sedang digunakan sehingga sistem tetap beroperasi dengan baik.

1. Perbaikan sistem

Melakukan perbaikan pada aplikasi apabila terjadi kesalahan (bugs) pada saat aplikasi dioperasionalkan.

1. Peningkatan sistem

Melakukan modifikasi dan inovasi terhadap aplikasi ketika terdapat potensi peningkatan aplikasi setelah aplikasi berjalan.

1. Backup

Melakukan salinan data secara berkala guna menghindari kehilangan dan kerusakan data.

1. Penggunaan sistem

Menggunakan aplikasi sesuai fungsi dan kegunaan masing-masing menu untuk memastikan sistem tetap berjalan dengan baik.

# **BAB VI**

# **PENUTUP**

1. **Kesimpulan**

Berdasarkan hasil dari pembahasan sistem, pengujian sistem dan analisa permasalahan yang dibutuhkan CV Panen Rejo dan merancang aplikasi *interface* pada bab sebelumnya, maka penulis mengambil kesimpulan sebagai berikut :

1. Sistem aplikasi implementasi sistem keuangan dengan metode perpetual dan periodik pada CV Panen Rejo dapat mempermudah proses optomisasi pengelolaan keuangan dan barang, sehingga kegiatan penentuan harga jual, stok barang dan data keuangan tercatat efektif dan efisien,
2. Dengan adanya sistem ini, semua kegiatan operasional dapat terkomputasi dengan rapih, sehingga memudahkan pengambilan data sewaktu-waktu diperlukan.
3. Sistem ini dijalankan pada local area yang terpusat di komputer pemilik toko.
4. Sistem aplikasi ini dikembangkan dengan framework Laravel yang memiliki fitur-fitur yang dapat memudahkan penggunanya serta memudahkan bagi pengembang yang akan melanjutkan pengembangan sistem dengan fitur fitur baru.
5. Sistem aplikasi ini memiliki batasan hak akses untuk mengelola dengan masuk menggunakan email dan password yg sudah di daftarkan oleh kepala toko.
6. **Saran**

Berdasarkan kesimpulan diatas, saran penulis yang berhubungan dengan Implementasi Sistem Keuangan Dengan Metode Perpetual dan Periodik pada CV Panen Rejo adalah sebagai berikut :

1. Sistem aplikasi ini datap di kembangkan kedalam aplikasi toko online untuk mempermudah transaksi dan menginformasikan produk-produk kepada pelanggan.
2. Untuk inventory harus di kembangkan lagi dengan penambahan pengadaan barang dibagian *stock opname*,
3. Perluadanya pengembangan sistem dengan melibatkan manajemen akuntansi dalan menghitung neraca laporan keuangan.
4. Metode penjualan dalam pencarian produk dengan menggunakan QR-CODE atau BARCODE *scanner*, untuk mempercepat saat transaksi,

# **DAFTAR PUSTAKA**

Adi Nugroho. 2010. Rekayasa Perangkat Lunak Berbasis Objek dengan Metode USDP. Andi. Yogyakarta.

Agus Ristono.2009. Manajemen Persediaan Edisi 1. Graham Ilmu: Yogyakarta.

Azhar Susanto. 2013. Sistem Informasi Akuntansi. Bandung: Lingga Jaya.

Jogiyanto, 2010. Analisis dan Desain Sistem Informasi, Edisi IV, Andi Offset, Yogyakarta.

Mulyadi. 2013.Sistem Akuntansi, Edisi Ketiga, Cetakan Keempat, Salemba Empat, Jakarta.

Simarmata. Janner (2010), Rekayasa Perangkat Lunak, Andi Offset, Yogyakarta.

Sulistiyowati, Leny. 2010. Panduan Praktis Memahami Analisis Laporan Keuangan, Jakarta: PT Elex Media Komputindo.

Swastha, Basu. 2010. Manajemen Penjualan: Pelaksanaan Penjualan, BPFE Yogyakarta.