**SISTEM INFORMASI KEUANGAN MENGGUNAKAN METODE ANALISIS RASIO KEUANGAN BERBASIS WEB PADA CV CANDRA FOOD**

**PROPOSAL LAPORAN AKHIR**

**Oleh:**

**SEPTIAN WIJAYA AMINULLOH NIM. 1931733056**

****

**PROGRAM STUDI D-III MANAJEMEN INFORMATIKA**

**PSDKU KEDIRI**

**POLITEKNIK NEGERI MALANG**

**OKTOBER 2020**

# **BAB I PENDAHULUAN**

## **Latar Belakang**

Teknologi informasi saat ini sangat penting karena dengan adanya teknologi informasi dapat mendukung kegiatan yang ada di suatu organisasi. Teknologi informasi digunakan untuk mengelola informasi, seperti manajemen kompleksitas dan waktu respons yang lebih cepat terhadap masalah atau keputusan bisnis. Pada pengelolaan data keuangan, teknologi informasi menjadi hal penting perusahaan karena telah dipercaya dapat membantu bagian keuangan atau akuntansi dalam menghasilkan laporan keuangan yang lebih akurat, dan efektif dalam penyusunannya[1]. Oleh karena itu, kebutuhan akan sistem informasi analisis rasio keuangan dapat digunakan sebagai catatan dan sebagai alat pengambilan keputusan serta memudahkan investor untuk mengendalikan keuangan perusahaan

Analisis rasio merupakan salah satu alat analisis keuangan yang paling banyak digunakan, dimana dalam perhitungan rasio ini menggunakan perhitungan aritmatika sederhana yang dapat diintreprestasikan, Analisa rasio ini dapat menjelaskan atau memberi gambaran kepada penganalisa tentang baik atau buruknya keadaan atau posisi keuangan suatu perusahaan terutama apabila angka rasio tersebut dibandingkan dengan angka rasio pembanding yang digunakan sebagai standard dalam rasio keuangan. [2]

Seperti halnya pada CV Candra Food yang bergerak di bidang produksi bumbu masakan. Perusahan ini berkantor pusat di Kayen Kidul, Kabupaten Kediri dan di Jalan Gatot Subroto, Kota Kediri yang tentunya memiliki rasio keuangan yang harus selalu dianalisa secara berurutan dari satu periode ke periode berikutnya. Analisa rasio ini nantinya digunakan sebagai dasar informasi yang dapat berguna baik itu oleh pihak internal ataupun pihak eksternal perusahaan dan perusahaan dapat menggunakan analisa rasio keuangan.

CV Candra Food memiliki beberapa gudang sehingga mereka harus melaporkan setiap kegiatan dan anggaran, seperti pendapatan dan pengeluaran serta biaya setiap produksi. Berdasarkan wawancara dengan pemilik CV Candra Food, pengelolaan keuangan masih dilakukan menggunakan Excel yang dapat memiliki resiko kesalahan dalam memasukkan data angka keuangan atau rumus dari analisis keuangan yang berakibatkan tidak akuratnya dari hasil yang hendak dicapai oleh perusahaan sebagai penentu pihak manajemen didalam menentukan alokasi anggaran dan prediksi keuntungan di tahun berikutnya. Kemudian, hasil perhitungan tersebut akan dianalisis dan diinterpretasikan sehingga diketahui posisi keuangan yang sesungguhnya [1]. Semuanya ini harus dilakukan secara teliti, mendalam, dan jujur.

Kebutuhan akan sistem informasi keuangan perusahaan dapat digunakan sebagai catatan keuangan dan sebagai alat pengambilan keputusan. Dan sistem ini memudahkan investor untuk mengendalikan keuangan perusahaan. Teknologi informasi yang akan menjadi solusi permasalahan CV Candra Food adalah analisis rasio keuangan berbasis web. Sistem informasi ini dibangun dengan menggunakan bahasa pemrograman PHP, HTML dan menggunakan framework laravel. Framework laravel merupakan framework PHP ysng menekankan pada kesederhanaan dan fleksibilitas pada desainnya [3].

Penggunaan web ini dapat menjadi pilihan para pengguna untuk dapat mengakses informasi kapan dan dimana saja sehingga mempercepat aliran informasi. Penggunaan sistem dengan web ini memudahkan pemilik untuk mengetahui. menilai atau mengevaluasi keberhasilan perusahaan. Oleh karena itu, perlu dibangun sebuah Sistem Informasi Keuangan Menggunakan Metode Analisis Rasio Keuangan Berbasis Web Pada CV Candra Food. Sistem informasi keuangan ini diharapkan dapat membantu perusahaan untuk mengetahui keuangan perusahaan dalam satu periode, mengetahui keberhasilan dan kelemahan perusahaan serta dapat mengetahui langkah-langkah perbaikan kinerja manajemen sistem keuangan.

## **Rumusan Masalah**

Berdasasarkan latar belakang permasalahan tersebut, maka rumusan masalah yang akan diselesaikan adalah bagaimana merancang dan membangun Sistem Informasi Keuangan Menggunakan Metode Analisis Rasio Keuangan Berbasis Web Pada CV Candra Food.

## **Tujuan**

Tujuan yang ingin dicapai dalam pembuatan Sistem Informasi Keuangan Menggunakan Metode Analisis Rasio Keuangan Berbasis Web Pada CV Candra Food.

## **Batasan Masalah**

Batasan Masalah dalam pembuatan Sistem Informasi Keuangan Menggunakan Metode Analisis Rasio Keuangan Berbasis Web Pada CV Candra Food adalah :

1. Sistem yang dibuat hanya berbasis web menggunakan framework laravel.
2. Sistem ini dibuat dengan HTML dan PHP, untuk database menggunakan MYSQL
3. Pada sistem ini hanya fokus pada analisis rasio keuangan pada CV Candra Food yang beralamat di Jalan Joyoboyo No. 8 Desa Padangan, Kayen Kidul, Kabupaten Kediri
4. Pada sistem ini tidak terdapat fitur untuk perhtungan transaksi (kasir)

## **Manfaat**

Manfaat dalam pembuatan Sistem Informasi Keuangan Menggunakan Metode Analisis Rasio Keuangan Berbasis Web Pada CV Candra Food adalah :

1. Bagi Admin, Sistem Informasi Keuangan Menggunakan Metode Analisis Rasio Keuangan menghasilkan suatu informasi yang dapat dikelola dengan baik dan efisien sehinggan mempermudah pengguna lain dalam memperoleh informasi tersebut.
2. Bagi Bagian Keuangan, Sistem Informasi Keuangan Menggunakan Metode Analisis Rasio Keuangan menghasilkan catatan penjualan yang lebih terstruktur serta dapat mengetahui perencanaan yang efektif dan dasar evaluasi selanjutnya

# **BAB II TINJAUAN PUSTAKA**

## **Penelitian yang Relevan**

Dari penelitian sebelumnya untuk referensi dan tolak ukur dalam penelitian terdahulu dalam perancangan dan pembangunan sistem ini yaitu penelitian yang dilakukan oleh Isti Aning Ika Haryanti pada tahun 2019 yang berjudul “Rancang Bangun Sistem Informasi Keuangan Menggunakan Metode Analisis Rasio Keuangan Berbasis Website” [4] Pada penelitian ini ditemukan suatu permasalahan yaitu selama ini proses keuangan CV Global Media Kreatif masih dilakukan secara manual dan membutuhkan lebih banyak waktu dan upaya menggunakan Microsoft Excel. Sementara resume CV Global Media Kreatif adalah empat unit bisnis dengan proses bisnis yang berbeda, mereka harus melaporkan setiap kegiatan dan anggaran, seperti pendapatan dan pengeluaran, dan biaya setiap proyek. Dari permasalahan tersebut dibuat Sistem Informasi Keuangan Menggunakan Metode Analisis Rasio Keuangan Berbasis Website. Sistem ini menggunakan metode penelitian sistem *prototype* menggunakan *framework Laravel* dan database MySQL. Hasil dari penelitian ini adalah Sistem Informasi Keuangan Menggunakan Metode Analisis Rasio Keuangan Berbasis Website yang bisa digunakan oleh CV Global Media Kreatif untuk mengetahui posisi keuangan perusahaan dalam satu periode, mengetahui keberhasilan dan kelemahan-kelamahan perusahaan serta dapat mengetahui langkah-langkah perbaikan kinerja manajemen perusahaan.

Penelitian yang kedua dilakukan oleh Firza, Magdalena dan Fitri pada tahun 2018 yang berjudul “Sistem Informasi Akuntansi Untuk Menilai Kinerja Perusahaan Menggunakan Rasio Profitabilitas Runtun Waktu” [5]. Pada penelitian ini terdapat pemasalahan yaitu informasi tentang posisi laporan keuangan terkendala karena proses bisnis berbasis kertas tradisional, mengakibatkan besarnya konsumsi waktu untuk mencatat transaksi hingga membuat laporan keuangan. Dari permasalahan tersebut dibuat Sistem Informasi Akuntansi untuk Menilai Kinerja Perusahaan Menggunakan Rasio Profitabilitas Runtun Waktu untuk memudahkan dalam membuat laporan keuangan pada perusahaan. Sistem ini menggunakan metode penelitian iterasi, menggunakan bahasa pemrograman PHP Framework Codeigniter. Hasil dari penelitian ini adalah Sistem Informasi Akuntansi Untuk Menilai Kinerja Perusahaan Menggunakan Rasio Profitabilitas Runtun Waktu dengan standar akuntansi yang membantu pengguna dengan kemungkinan kesalahan yang rendah

Penelitian yang ketiga dilakukan oleh Tri dan Rio pada tahun 2018 yang berjudul “Rancang Bangun Aplikasi Pengukuran Kinerja Keuangan Perusahaan Menggunakan Metode Analisis Rasio Keuangan (Studi Kasus: Mitra Jaya Group Cabang Bandung” [6]. Pada penelitian ini terdapat pemasalahan yaitu Mitra Jaya Group masih menggunakan cara sederhana dalam menilai kinerja keuangannya, yaitu dengan melihat neraca dan laporan laba rugi sedangkan dengan hanya menggunakan laporan keuangan saja memiliki keterbatasan. Sistem ini menggunakan metode *Prototype* dengan bahasa pemrograman VB.Net. Hasil dari penelitian ini adalah Aplikasi Pengukuran Kinerja Keuangan Perusahaan Menggunakan Metode Analisis Rasio Keuangan telah terintegrasi langsung antara catatan akuntansi, laporan keuangan, dan laporan analisis keuangan. Disamping itu juga pembagian akses menu dalam aplikasi agar meminimalisir manipulasi data.

Pada penelitian yang telah dilakukan sebelumnya, terlihat bahwa sistem yang ada masih dilakukan secara manual, sehingga dalam penelitian ini akan dibuat Sistem Informasi Analisis Rasio Keuangan Berbasis Web pada CV Candra Food. Berikut ini adalah tabel 2.1 Resume Referensi

Tabel 2. 1 Resume Referensi

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Penulis | Tahun | Judul | Metode | Kesimpulan |
| 1 | Isti Aning Ika Haryanti | 2019 | Rancang Bangun Sistem Informasi Keuangan Menggunakan Metode Analisis Rasio Keuangan Berbasis Website | Framework Laravel dan Database MySQL | Pada CV Global Media Kreatif masih menggunakan Microsoft Excel sehingga perlu dibangunnya Sistem Informasi Keuangan Menggunakan Metode Analisis Rasio Keuangan Berbasis Website untuk mengetahui posisi keuangan perusahaan, mengetahui keberhasilan dan kelemahan perusahaan serta perbaikan kinerja manajemen perusahaan |
| 2 | Firza, Magdalena dan Fitri | 2018 | Sistem Informasi Akuntansi Untuk Menilai Kinerja Perusahaan Menggunakan Rasio Profitabilitas Runtun Waktu | PHP Framework Codeigniter | Pada penelitian ini informasi tentang laporan keuangan masih berbasis kertas tradisional, mengakibatkan besarnya konsumsi waktu untuk mencatat transaksi hingga membuat laporan keuangan sehingga perlu dibangunnya Sistem Informasi Akuntansi Untuk Menilai Kinerja Perusahaan |
| 3 | Tri dan Rio | 2018 | Rancang Bangun Aplikasi Pengukuran Kinerja Keuangan Perusahaan Menggunakan Metode Analisis Rasio Keuangan (Studi Kasus: Mitra Jaya Group Cabang Bandung) | VB.net | Pada Mitra Jaya Group masih menggunakan cara sederhana dalam menilai kinerja keuangannya, yaitu terbatasanya melihat neraca dan laporan laba rugi sehingga perlu dibangunnya Aplikasi Pengukuran Kinerja Keuangan Perusahaan Menggunakan Metode Analisis Rasio Keuangan yang terintegrasi langsung antara catatan akuntansi, laporan keuangan, dan laporan analisis keuangan. |
| 4 | Septian Wijaya Aminulloh | 2020 | Sistem Informasi Keuangan Menggunakan Metode Analisis Rasio Keuangan Berbasis Web Pada CV Candra Food | Pada CV Candra Food, pengelolaan keuangan masih menggunakan Excel yang memiliki resiko kesalahan dalam memasukkan data keuangan atau rumus dari yang berakibat tidak akuratnya hasil yang hendak dicapai oleh perusahaan sehingga perlu dibangunnya Sistem Informasi Keuangan Menggunakan Metode Analisis Rasio Keuangan Berbasis Web Pada CV Candra Food. Sistem Informasi ini menggunakan bahasa pemroraman Framework Laravel yang dilengkapi dengan fitur cetak laporan untuk mengetahui untung rugi pada CV Candra Food dalam waktu harian, bulanan, maupun tahunan. | |

## **Analisis Rasio Keuangan**

Analisis Rasio keuangan (*Financial Ratio Analysis*) merupakan kegiatan  
membandingkan angka-angka yang ada dalam laporan keuangan dengan cara  
membagi satu angka dengan angka lainnya, maka perbandingannya dapat  
dilakukan antara satu komponen dengan komponen lainnya dalam satu laporan  
keuangan atau antar komponen yang ada diantara laporan keuangan tersebut dalam bentuk angka-angka pada suatu periode tertentu. Hasil dari Analisa Rasio Keuangan (*Financial Ratio Analysis*) ini dapat digunakan oleh pihak manajemen perusahaan untuk menilai kinerjanya dalam suatu periode, apakah pihak manajemen perusahaan telah mencapai target yang telah ditetapkan atau sebaliknya, dan selain itu juga hasil dari rasio keuangan ini dapat dijadikan sebagai suatu penilaian terhadap kemampuan manajemen di dalam memberdayakan semua sumber daya perusahaan yang dimiliki secara lebih efektif.

### **Analisis Rasio**

Analisis rasio merupakan salah satu alat analisis keuangan yang paling banyak digunakan, dimana dalam perhitungan rasio ini menggunakan perhitungan aritmatika sederhana yang dapat diintreprestasikan, dimana setiap perhitungan rasio akan jauh lebih bermanfaat jika dibandingkan dengan hasil perhitungan rasio tahun sebelumnya. Analisa Rasio Keuangan merupakan proses analisis dan penilaian yang membantu dalam menjawab pertanyaan yang sudah sewajarnya diajukan, jadi itu merupakan alat untuk mencapai tujuan [4].

### **Metode Analisis Rasio Keuangan**

Pada dasarnya jumlah metode analisis rasio banyak sekali, karena rasio dapat dibuat menurut kebutuhan penganalisa. Namun dapat dibagi kedalam beberapa bentuk umum yang sering dipergunakan yaitu: rasio likuiditas, rasio solvabilitas, rasio aktivitas dan rasio profitabilitas. [8] Analisis rasio keuangan ini dapat  
mengungkapkan hubungan penting antara perkiraan dalam laporan keuangan dan dapat digunakan untuk mengevaluasi kondisi keuangan dan kinerja perusahaan [4].

Rasio profitabilitas merupakan rasio untuk menilai kemampuan perusahaah dalam mencari keuntungan. Rasio ini juga memberikan ukuran tingkat efektivitas manajemen suatu perusahaan [8]. Hal ini ditunjukkan oleh laba yang dihasilkan dari penjualan dan pendapatan investasi.

1. *Profit margin on sales*

Rasio ini menghitung sejauh mana kemampuan perusahaan menghasilkan laba bersih pada tingkat penjualan tertentu. Rasio ini bisa dilihat langsung pada analisis *common size* untuk laporan rugi laba (baris paling akhir). Rasio ini bisa diintepretasikan juga sebagai kemampuan perusahaan menekan biaya-biaya (ukuran efisiensi) di perusahaan pada periode tertentu rumus untuk mencari profit margin, seperti pada rumus 1 sebagai berikut.

Apabila *Profit margin on sales* diatas 1:1 atau 100% berarti *Profit margin on sales* baik dan semakin tinggi semakin baik karena dianggap kemampuan perusahaan dalam mendapatkan laba cukup tinggi [8].

1. Hasil Pengembalian Investasi (ROI)

Hasil pengembalian investasi atau lebih dikenal dengan nama *Return on Investment* (ROI) merupakan rasio yang menunjukkan hasil (*return*) atas jumlah aktitiva yang digunakan dalam perusahaan. ROI juga merupakan suatu ukuran tentang efektivitas manajemen dalam mengelolah investasinya. Disamping itu, hasil pengembalian investasi menunjukkan produktivitas dari seluruh dana perusahaan, baik modal pinjaman maupun modal sendiri. Semakin kecil (rendah) rasio ini, semakin kurang baik, demikian pula sebaliknya. Rumus untuk mencari *return on investment* seperti pada rumus 2 sebagai berikut.

Apabila *ROI* diatas 1:1 atau 100% berarti *ROI* baik dan semakin tinggi semakin baik [8].

1. Hasil Pengembalian Ekuitas Return on equity (ROE)

Hasil pengembalian ekuitas atau *return on equity* merupakan rasio untuk mengukur laba bersih sesudah pajak dengan modal sendiri. Rasio ini menunjukkan efisiensi penggunaan modal sendiri. Semakin tinggi rasio ini, semakin baik. Artinya posisi pemilik perusahaan semakin kuat, demikian pula sebaliknya. Rumus untuk mencari hasil pengembalian ekuitas seperti pada rumus 3 sebagai berikut.

Apabila ROE diatas 1:1 atau 100% berarti *ROE* baik dan semakin tinggi semakin baik [8].

## **Sistem Informasi**

Sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen atau variabel-variabel yang terorganisasi, saling berinteraksi, saling tergantung satu sama lain dan terpadu untuk mencapai suatu tujuan. Unsur-unsur yang mewakili suatu sistem secara umum adalah masukan (*input*), pengolahan (proses) dan keluaran (*output*). Suatu sistem dapat terdiri dari sistem-sistem bagian (*subsystem*). Misalnya, sistem komputer terdiri dari subsistem perangkat keras dan subsistem perangkat lunak. Masing-masing subsistem dapat terdiri dari subsistem-subsistem yang lebih kecil lagi atau terdiri dari komponen-komponen. Subsistem perangkat keras (*hardware*) dapat terdiri dari alat masukan, alat pemroses, alat keluaran dan simpanan luar [7].

Subsistem saling berinteraksi dan saling berhubungan membentuk satu kesatuan sehingga tujuan atau sasaran sistem tersebut dapat tercapai. Interaksi dari subsistem-subsistem sedemikian rupa, sehingga dicapai suatu kesatuanyang terpadu atau terintegrasi (*integrated*). Apabila suatu sistem yang mana masing-masing komponen berkerja secara individu, maka tujuan dari sistem komputer tidak akan tercapai [8].

Informasi adalah data yang sudah disusun sedemikian rupa sehingga bermakna dan bermanfaat karena dapat dikomunikasikan kepada seseorang yang akan menggunakannya dalam proses pengambilan keputusan. Ada tiga pilar utama yang menentukan kualitas informasi yakni akurasi, ketepatan waktu dan relevansi, lebih lanjut diungkapkan secara lengkap tentang syarat-syarat informasi yang baik yakni:

1. Ketersediaan (*availability*)

Salah satu syarat yang mendasar adalah tersedianya informasi itu sendiri. Informasi harus dapat diperoleh bagi orang yang hendak memanfaatkannya.

1. Mudah dipahami (*comprehensibility*)

Informasi harus mudah dipahami oleh pembuat keputusan, baik informasi tersebut diperuntukkan dalam pembuatan keputusan yang sifatnya rutin maupun strategis. Informasi yang rumit dan berbelit-belit hanya akan membuat kurang efektifnya keputusan manajemen.

1. Relevansi

Informasi yang diperlukan adalah yang benar-benar relevan dengan permasalahan, misi dan tujuan organisasi.

1. Bermanfaat

Informasi harus tersaji kedalam bentuk-bentuk yang memungkinkan pemanfaatannya oleh organisasi yang bersangkutan.

1. Tepat waktu

Informasi harus tersedia tepat pada waktunya, syarat ini utamanya sangat penting pada saat organisasi membutuhkan informasi ketika manajer hendak membuat keputusan yang krusial.

1. Keandalan

Informasi harus diperleh dari sumber-sumber yang dapat diandalkan kebenarannya. Pengolah data atau pemberi informasi harus dapat menjamin tingkat kepercayaan yang tinggi atas informasi yang disajikan.

1. Akurat

Syarat ini mengharuskan informasi harus bersih dari kesalahan dan kekeliruan. Ini berarti juga bahwa informasi harus jelas dan secara akurat mencerminkan makna yang terkandung dari data pendukungnya.

1. Konsisten

Informasi tidak boleh mengandung kontradiksi di dalam penyajiannya, karena konsistensi merupakan syarat penting bagi dasar pengambilan keputusan.

## **Teknologi Web**

Teknologi web adalah teknologi yang berhubungan dengan antarmuka untuk menjembatani server web dan klien mereka [10]. Teknologi web mencakup bahasa *markup*, bahasa pemrograman, dan standar untuk mengidentifikasi dokumen serta tampilan.

1. PHP

PHP merupakan bahasa pemrograman *script server-side* yang didesain untuk pengembangan *web*. Selain itu, PHP juga digunakan sebagai bahasa pemrograman umum. PHP di kembangkan pada tahun 1995 oleh Rasmus Lerdorf, dan sekarang dikelola oleh The PHP Group. Situs resmi PHP beralamat di [http://www.php.net](http://www.php.net/). Fungsi lain PHP adalah untuk mengkoneksikan halaman *website* dengan basis data yang digunakan.

Dalam *website* dinamis dan interaktif, bahasa pemrograman PHP digunakan sebagai media untuk mempersingkat tatanan bahasa pemrograman HTML dan CSS. Dengan menggunakan bahasa pemrograman HTML dan CSS, maka dibutuhkan baris kode yang sangat panjang. Sedangkan dengan menggunakan bahasa pemrograman PHP, baris kode yang dibutuhkan dapat dipersingkat hingga menjadi beberapa baris saja [13].

Dalam penulisan PHP menggunakan *syntax* <?php .. ?> dan seluruh *script* yang digunakan ditempatkan dalam *syntax* tersebut. Dalam penulisan PHP menggunakan dua cara yaitu internal dan eksternal. Pada cara penulisan secara internal *syntax* PHP ditempatkan di antara *syntax* <head> dan <body> sedangkan untuk penulisan secara eksternal *syntax* PHP ditempatkan pada berkas khusus kemudian dipanggil menggunakan tautan pada halaman *website.*

1. CSS

CSS (*Cascading Style Sheet)* adalah salah satu bahasa desain web (Style sheet language) yang mengontrol format tampilan sebuah halaman web yang ditulis dengan menggunakan penanda markup. Biasanya CSS digunakan untuk mendesain sebuah halaman HTML dan XHTML, tetapi sekarang CSS bisa diaplikasikan segala dokumen XML, termasuk SVG dan XUL, bahkan android [7]. Berikut ini adalah contoh kode CSS untuk mengatur font dan warna pada sebuah web

body {

overflow-x: hidden;

font-family: 'Roboto Slab', 'Helvetica Neue', Helvetica, Arial, sans-serif;

}

p {

line-height: 1.75;

}

a {

color: #fed136;

}

a:hover {

color: #fec503;

}

.text-primary {

color: #fed136 !important;

}

Hasil dari kode diatas adalah pada Gambar 2.1 Implementasi CSS



Gambar 2. 1 Implementasi CSS

1. HTML

HTML (*Hypertext Markup Language*) adalah bahasa pemrograman dasar yang digunakan untuk membangun suatu halaman *web*. HTML adalah bahasa *markup* yang umum digunakan oleh para *developer web* [14]. Karena kemudahan dalam menggunakannya, HTML menjadi bahasa pemrograman *web* yang populer di kalangan para *developer.*

HTML mempunyai fungsi di antaranya dapat menentukan format suatu teks, membuat *list*, membuat *link* ke dokumen lain, menyisipkan gambar, serta dapat menampilkan informasi dalam bentuk tabel. Selain itu, HTML juga berfungsi untuk mengatur tampilan dari halaman *web* dan isinya, menambahkan objek seperti gambar, *video,* suara, dan lain-lain.

Bahasa Pemrograman HTML merupakan bahasa pemrograman dasar yang wajib dikuasai oleh para *web programmer* untuk dapat mendesain *web* sesuai dengan keinginan pemesan *website*. Pada dasarnya *syntax* HTML terdiri dari beberapa *syntax* dasar yaitu : <html>, <head>, <title>, <body>. *Syntax*<body> berfungsi untuk meletakan konten yang dimuat oleh halaman *web* mulai dari *link*, *heading*, gambar, dan lain-lain, untuk meletakan tautan ke dokumen CSS, PHP, dan JavaScript adalah di *syntax* <head>. Sedangkan *syntax* <title> berfungsi sebagai tempat diberikannya judul suatu halaman *website*

1. JavaScript

JavaScript adalah bahasa pemrograman web yang bersifat *Client Side Programming Language*yaitu tipe bahasa pemrograman yang pemrosesannya dilakukan oleh *client*. Aplikasi *client* yang dimaksudkan adalah merujuk pada web *browser* pada Opera, *Google* *Chrome* dan *Mozilla* *Firefox*. Bahasa Pemrograman *Client* *Side* berbeda dengan bahasa pemrograman *Server* *Side* seperti PHP, dimana untuk *server* *side* seluruh kode program dijalankan di sisi *server*.

Dengan adanya JavaScript, maka teknik penulisan HTML dapat dilakukan dengan dua cara, yaitu dengan membuat program JavaScript untuk menghasilkan dokumen HTML atau dengan membuat dokumen HTML seperti biasa, apabila terdapat program JavaScript, maka *web designer* dapat menambahkan program JavaScript tersebut sebagai sisipan saja, dengan demikan JavaScript dapat membuat halaman *website* yang digunakan akan semakin interaktif dan dinamis [15].

Berikut ini adalah *script* sederhana JavaScript untuk konfirmasi pembelian barang yang berupa data stok barang dan data pembelian barang berdasarkan kode pembelian.

$('.btn-confirm').click(function(){

            var kode\_pembelian = $(this).attr('data-name');

            swal({

            title: "Konfirmasi Barang dengan Kode " + kode\_pembelian +" ?",

            text: "Data Stok dan Data Pembelian Akan Dikonfirmasi!",

            icon: "warning",

            buttons: true,

            dangerMode: true,

            })

        .then((willDelete) => {

            if (willDelete) {

                window.location = "Pembelian/Confirm/"+kode\_pembelian;

            } else {

                swal({

                    title: "Canceled!",

                    icon: "success",

                    })

            }

            });

        });

## **MySQL**

MySQL merupakan database yang pertama kali didukung oleh bahasa  
pemrograman script untuk internet (PHP dan Perl). MySQL dan PHP dianggap  
sebagai pasangan software pembangun aplikasi web yang ideal [16]. MySQL lebih  
sering digunakan untuk membangun aplikasi berbasis web, umumnya  
pengembangan aplikasinya menggunakan bahasa pemrograman script PHP

MySQL berfungsi untuk membuat, mengelola dan mengakses basis data secara terstruktur dan otomatis. Kemampuan MySQL yang paling menonjol terletak pada kemampuanya dalam mengelola suatu data secara *text based* yang bisa diakses pribadi maupun umum dalam suatu web [16]. Hampir semua penyedia *server web* atau *host* memberikan fasilitas MySQL bagi para *developer web* untuk pengelolaan database di websitenya. Di dunia *web*, tampilan muka My SQL disebut dengan phpMyAdmin. Beberapa contoh penggunaan MySQL dapat ditemukan pada *e-commerce*, Blog atau CMS.

Pada MySQL fitur dasar yang digunakan yaitu *Create, Read, Update dan Delete*. Berikut ini adalah perintah-perintah pada MySQL.

1. Perintah *Create* digunakan dalam pembuatan database ataupun pembuatan tabel. Beriku ini adalah contoh *script* untuk membuat tabel pada database :

CREATE TABEL nama\_tabel (  
nama\_kolom1 tipe\_kolom1(panjang\_data1),  
nama\_kolom2 tipe\_kolom2(panjang\_data2),  
nama\_kolom3 tipe\_kolom3(panjang\_data3),  
);

1. Perintah *Insert* digunakan untuk menambah data dalam tabel. Berikut ini adalah contoh *script* untuk menambah data dalam tabel :

INSERT INTO nama\_tabel (nama\_kolom1, nama\_kolom2,  
nama\_kolom3) VALUES (data1, data2, data3)

1. Perintah *Select* digunakan untuk menampilkan data dalam tabel. Berikut ini adalah contoh *script* untuk menampilkan data dalam tabel :

SELECT \* FROM nama\_tabel;

1. Perintah *Update* digunakan untuk menambah data dalam tabel. Berikut ini adalah contoh *script* untuk menambah data dalam tabel :

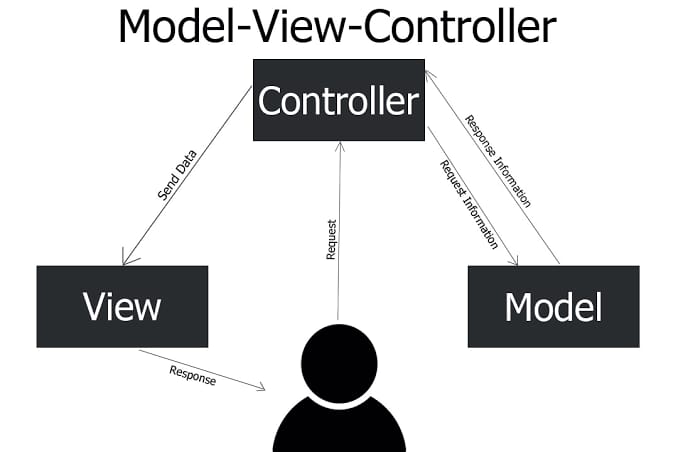
UPDATE nama\_tabel SET  
nama\_kolom1 = data1,  
nama\_kolom2 = data2,  
nama\_kolom3 = data3  
WHERE kolom\_id = data\_id

1. Perintah *Delete* digunakan untuk menghapus data dalam tabel. Berikut ini adalah contoh *script* untuk menghapus data dalam tabel :

DELETE FROM nama\_tabel WHERE kolom\_id=data\_id;

## **FrameworkLaravel**

*Laravel* merupakan *framework* PHP yang menekankan pada kesederhanaan  
dan fleksibilitas pada desainnya [9]. *Laravel* dirilis dibawah lisensi MIT dengan  
sumber kode yang disediakan di Github. Sama seperti *framework* PHP lainnya,  
*Laravel* dibangun dengan basis MVC (*Model-View Controller*). *Laravel* dilengkapi  
*command line tool* yang bernama “Artisan” yang bisa digunakan untuk *packaging*  
*bundle* dan instalasi *bundle*. Berikut ini adalah gambar 2.2 Arsitektur MVC



Gambar 2. 2 Arsitektur MVC

Berikut ini adalah kelebihan MVC (*Model-View-Controller*) [16]

1. Bagian model memberikan penempatan detil data yang terpisah dan tidak disebar di dalam keseluruhan aplikasi sehingga meningkatkan kecepatan dan fleksibilitas dalam proses pemeliharaan aplikasi.
2. Pemisahan model juga membuat objek model dapat digunakan oleh aplikasi lain dengan kebutuhan yang sama (*reuse*).
3. Pemisahan view memudahkan perakitan/integrasi dengan komponen aplikasi lainnya tanpa harus memperhatikan detil proses. Desainer hanya berkonsentrasi pada bentuk dan tampilan antarmuka pemakai (*user interface*).
4. Penggunaan Controller memungkinkan untuk perubahan proses tanpa harus mengganggu antarmuka pemakai. Detil proses disembunyikan oleh Controller sehingga tidak mengganggu presentasi ke user maupun pengelolaan data/informasi (manajemen database).

*Framework Laravel* dibuat oleh Taylor Otwell, proyek *Laravel* dimulai pada April 2011. Awal mula proyek ini dibuat karena Otwell sendiri tidak menemukan *framework* yang up-to-date dengan versi PHP. Mengembangkan *framework* yang sudah ada juga bukan merupakan ide yang bagus karena keterbatasan sumber daya. Dikarenakan beberapa keterbatasan tersebut, Otwell membuat sendiri *framework* dengan nama *Laravel*. Oleh karena itu *Laravel* membutuhkan PHP versi 5.3 keatas [9].

Beberapa fitur pada laravel :

1. Bundles, yaitu sebuah fitur dengan sistem pengemasan modular dan tersedia beragam di aplikasi.
2. Eloquent ORM, merupakan penerapan PHP lanjutan menyediakan metode internal dari pola “*active* *record*” yang menagatasi masalah pada hubungan objek database.
3. *Application* *Logic*, merupakan bagian dari aplikasi, menggunakan *controller* atau bagian *Route.*
4. *Reverse* *Routing*, mendefinisikan relasi atau hubungan antara Link dan Route.
5. *Restful* *controllers*, memisahkan logika dalam melayani HTTP *GET* *and* *POST.*
6. *Class* *Auto* *Loading*, menyediakan loading otomatis untuk *class* PHP.
7. *View* *Composer*, adalah kode unit *logikal* yang dapat dieksekusi ketika *view* sedang *loading.*
8. *IoC* *Container*, memungkin obyek baru dihasilkan dengan pembalikan *controller.*
9. *Migration*, menyediakan sistem kontrol untuk skema database.
10. *Unit* *Testing*, banyak tes untuk mendeteksi dan mencegah regresi.
11. *Automatic* *Pagination*, menyederhanakan tugas dari penerapan halaman.

## **Bootstrap**

Bootstrap adalah *front-end framework* yang mengedapankan tampilan untuk *mobile device*. Guna mempercepat dan mempermudah pengembangan website bootsrap menyediakan HTML, CSS dan *Javascript* siap pakai dan mudah untuk dikembangkan. Bootsrap merupakan *framework* untuk membangun desain web secara *responsif* artinya tampilan web yang dibuat oleh bootstrap akan menyesuaikan ukuran laya dari *browser* yang kita gunakan baik di *dekstop*, tablet ataupun *mobile* *device*[10]. Fitur ini bisa diaktifkan ataupun dinonaktifkan sesuai dengan keinginan kita sendiri sehingga, kita bisa membuat web untuk tampilan dekstop saja dan apabila dirender oleh *mobile* *browser* maka tampilan dari web yang kita buat tidak beradaptasi sesuai layar. Dengan boostrap kita bisa membanfun web dinamis ataupun statis.

Kelebihan dari menggunakan Bootstrap adalah kerangka ini dibangun menggunakan *Less*, sebuah teknologi CSS yang sederhana dan mudahuntuk digunakan. *Less* juga menawarkan lebih banyak kekuatan dan fleksibelitas dari CSS pada umumnya. Dengan *Less*, pengembang dapat mengakses dengan mudah informasi dan fungsi warna, variabel dan operasi penggunaan [11]. Berikut ini adalah cara instalasi Bootstrap dengan mudah :

1. Download file bootstrap melalui website <https://getbootstrap.com>
2. Pilih tombol download
3. Setelah terdownload, ekstrak file tersebut
4. Buat file .html pada folder tersebut
5. ketik *script* berikut ini kemudian simpan

<!Doctype html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>Belajar Bootstrap</title>

<link href="bootstrap/css/bootstrap.min.css" rel="stylesheet">

<script

src="bootstrap/js/jquery-3.2.1.js"></script>

<script src="bootstrap/js/bootstrap.min.js">

</script>

</head>

<body>

<h1> Instal Bootstrap Berhasil</h1>

</html>

## **Website**

*Website* adalah kumpulan halaman web yang saling terhubung dan file-filenya saling terkait. Web terdiri dari *page* atau halaman, dan kumpulan halaman yang dinamakan homepage. *Homepage* berada pada posisi teratas, dengan halaman-halaman terkait berada di bawahnya. Biasanya setiap halaman di bawah homepage disebut child page, yang berisi *hyperlink* ke halaman lain dalam web. Web mengijinkan pemberian *highlight* (penyorotan atau penggaris bawahan) pada kata-kata atau gambar dalam sebuah dokumen untuk menghubungkan atau menunjuk ke media lain seperti dokumen, frase, *movie* clip, atau file suara [12].

Secara umum *website* dibagi menjadi 3 jenis yaitu : *website* statis, *website* dinamis, dan *website* interaktif. *Website* Statis merupakan *web* yang memiliki halaman yang tidak berubah. Artinya adalah untuk melakukan perubahan pada suatu halaman dilakukan secara manual dengan mengedit *code* yang menjadi struktur dari situs tersebut. *Website* dinamis merupakan *website* yang secara struktur diperuntukan agar pemilik dapat melakukan perubahan konten (*edit, update, delete*). Sedangkan *website* interaktif merupakan *web* yang digunakan oleh para pemilik suatu *domain* untuk berinteraksi dengan orang lain dengan cara menyampaikan suatu informasi atau melakukan kegiatan lain pada *website* yang mereka buat.

## **Unified Modelling Language (UML)**

Menurut Nugroho UML (*Unified Modeling Language*) adalah ‘bahasa’  
pemodelan untuk sistem atau perangkat lunak yang berparadigma ‘berorientasi  
objek”. Pemodelan (*modeling*) sesungguhnya digunakan untuk penyederhanaan  
permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih  
mudah dipelajari dan dipahami. UML (*Unified Modeling Language*) adalah Metodologi kolaborasi antara metoda-metoda Booch, OMT (*Object Modeling Technique*),serta OOSE (*Object Oriented Software Enggineering*) dan beberapa metoda lainnya dengan mengadaptasi maraknya penggunaan bahasa “pemrograman berorientasi objek” (OOP) [14].

Pada *view* digunakan untuk melihat sistem yang dimodelkan dari beberapa aspek yang berbeda. Beberapa jenis *view* dalam UML antara lain : *use* *case* *view*, *logical* *view*, *component* *view*, *concurrency* *view*, dan *deploymeint* *view*. Diagram berbentuk grafik yang menunjukkan simbol elemen model yang disusun untuk mengilustrasikan bagian atau aspek tertentu dari sistem, adapun jenis diagram antara lain *use* *case* diagram, *class* diagram, *sequence* diagram dll.

1. **Use Case**

Menggambarkan sejumlah *external actor* dan hubungannya ke *use case* yang diberikan oleh *system*. *Use Case* atau diagram *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu *use case* digambarkan hanya yang dilihat dari luar oleh *actor* (keadaan lingkungan system yang dilihat user) dan bukan bagaimana fungsi yang ada didalam *system* [18]*.* Berikut ini adalah beberapa simbol dari *Use Case*.pada tabel 2.2 Simbol Use Case

Tabel 2. 2 Simbol Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **NO** | **GAMBAR** | **NAMA** | **KETERANGAN** |
| 1 |  | Actor | Menspesifikasikan himpuan peran yang pengguna mainkan ketika berinteraksi dengan usecase |
| 2 | <<include>> | Include | Menspesifikasikan bahwa use case sumber secara eksplisit. |
| 3 | <<extend>> | Extend | Menspesifikasikan bahwa use case target memperluas perilaku dari use case sumber pada satu titik yang diberikan. |
| 4 |  | Associaton | Apa yang menghubungkan antara objek satu dengan objek lainnya. |
| 5 |  | System | Menspesifikasikan paket yang menampilkan sistem secara terbatas. |
| 6 |  | Use Case | Deskripsi dan urutan aksi-aksi yang ditampilkan sistem yang dihasilkan suatu hasil yang terukur bagi suatu aktor |

1. **Activity Diagram**

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Perlu diperhatikan bahwa diagram aktivitas menggambarkan aktivitas system bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh system [18]. Berikut adalah simbol-simbol yang ada pada *aktivity diagram* pada tabel 2.3 Simbol Activity Diagram

Tabel 2. 3 Simbol Activity Diagram

|  |  |
| --- | --- |
| **Simbol** | **Deskripsi** |
| Satus awal | Status awal aktivitas sitem, sebuah diagram aktivitas memiliki sebuah status awal |
| Percabangan /decision | Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu |
| Penggabungan / join | Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu |
| Status akhir | Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir |
| Swimlane | Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi |

1. **Class Diagram**

Diagram kelas atau *class* diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Atribut merupakan variable-variabel yang dimiliki oleh suatu kelas sedangkan operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas [19]. Simbol *class* diagram dan keteranggannya dapat dilihat pada Tabel 2.4 Simbol Class Diagram

Tabel 2. 4 Simbol ClassDiagram

|  |  |  |
| --- | --- | --- |
| **No** | **Simbol** | **Deskripsi** |
| 1 | Kelas   |  | | --- | | **nama\_kelas** | | +atribut | | +operasi() | | Kelas pada struktur sistem. |
| 2 | Antarmuka / Interface  **nama\_interface** | Sama dengan konsep *interface* dalam pemrograman berorientasi objek. |
| 3 | Asosiasi / *Association* | Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan *multiplicity*. |
| 4 | Asosiasi berarah / *Directed* *association* | Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan *multiplicity*. |
| 5 | Generalisasi | Relasi antarkelas dengan makna generalisasi- spesialisasi (umum khusus). |
| 6 | Kebergantungan / *Dependency* | Relasi antarkelas dengan makna kebergantungan antarkelas. |
| 7 |  | Relasi antarkelas dengan makna semua-bagian (*whole*-*part*). |

# **BAB III ANALISIS DAN PERANCANGAN**

## **Analisis Permasalahan**

Pengelolaan keuangan yang masih dilakukan menggunakan Excel dapat memiliki resiko kesalahan dalam memasukkan data angka keuangan atau rumus dari analisis keuangan yang berakibatkan tidak akuratnya dari hasil yang hendak dicapai oleh perusahaan sebagai penentu pihak manajemen didalam menentukan alokasi anggaran dan prediksi keuntungan di tahun berikutnya. Kemudian, hasil perhitungan tersebut akan dianalisis dan diinterpretasikan sehingga diketahui posisi keuangan yang sesungguhnya.

## **Analisis Pemecahan Masalah**

Dalam Pemecahan masalah yang terjadi, dibutuhkan ketepatan dalam  
mengambil keputusan. Sehingga dalam pengamatan untuk menyelesaikan  
dibutuhkan beberapa tahap. Tahap-tahap tersebut seperti skema dibawah ini.:

Gambar 3. 1 Skema Pemecahan Masalah

### **Metode Pengumpulan Data**

Metode pengumpulan data digunakan untuk memperoleh informasi yang dibutuhkan dalam rangka mencapai tujuan penelitian. Dalam pengambilan data ini dilakukan dengan cara melakukan penelitian lapangan ke CV Candra Food. Metode pengambilan data dilakukan sebagai berikut :

1. Wawancara (*interview*)

Metode pengumpulan data dengan cara mendatangi secara langsung pihak  
CV Candra Food yang mempunyai wewenang untuk memberikan informasi.  
Sehingga informasi yang didapat lebih lengkap, benar dan akurat

1. Studi Literatur

Teknik ini digunakan untuk mengumpulkan data dengan bahan rujukan dari  
buku–buku, dokumen, yang berhubungan dengan masalah yang sedang dibahas  
berasal dari internet. Referensi yang digunakan sebagai sumber informasi dapat dilihat pada daftar pustaka di akhir laporan.

### **Metode Pengembangan Sistem**

Langkah-langkah untuk merancang Sistem Informasi Keuangan Menggunakan Metode Analisis Rasio Keuangan Berbasis Web Pada CV Candra Food menggunakan metode pengembangan sistem waterfall. Model waterfall adalah model yang berurutan dalam membangun software, dimana tahap demi tahap harus dilalui untuk ketahap selanjutnya. Metode *Waterfall* mengusulkan sebuah pendekatan kepada perkembangan perangkat lunak yang sistematik dimulai pada tingkat kemajuan sistem pada seluruh analis, desain, kode dan pengujian [17].

Gambar 3. 2 Metode *Waterfall*

Adapun aktivitas yang dilakukan dalam penelitian skripsi ini sebagai berikut:

1. *System* / *Information* *Engineering*

Bagian dari sistem yang terbesar dalam pengerjaan suatu proyek, dimulai dengan menetapkan berbagai kebutuhan dari semua elemen yang diperlukan sistem.

1. Analisis

Tahap menganalisis hal-hal yang diperlukan dalam pelaksanaan proyek pembuatan perangkat lunak.

1. *Design*

Tahap penerjemahan dari data yang dianalisis kedalam bentuk yang mudah dimengerti oleh *user.*

1. *Coding* dan *testing*

*Code* merupakan bahasa yang dapat dimengerti oleh mesin, dalam hal ini adalah komputer, maka desain tadi harus diubah bentuknya menjadi bentuk yang dapat dimengerti oleh mesin, yaitu ke dalam bahasa pemrograman melalui proses *coding*. *Testing* merupakan sesuatu yang dibuat dan diuji coba. Semua fungsi-fungsi software harus diujicobakan, agar software bebas dari error, dan hasilnya harus benar-benar sesuai dengan kebutuhan yang sudah didefinisikan sebelumnya.

1. Implementasi

Implementasi, dimana keseluruhan desain sistem yang telah disusun sebelumnya akan diubah menjadi kode-kode program dan modul-modul yang nantinya akan diintegrasikan menjadi sebuah sistem yang lengkap sesuai dengan kontrak kerja.

1. Pemeliharaan

Pemeliharaan yang termasuk diantaranya instalasi dan proses perbaikan sistem apabila ditemukan adanya kesalahan/bug yang tidak ditemukan pada tahap testing.

### **Analisis Kebutuhan Sistem**

Sebelum melakukan perancangan sistem, terlebih dahulu dilakukan analisis  
kebutuhan sistem yaitu dengan cara melakukan wawancara terhadap responden.  
Hal ini dimaksudkan agar dapat mengatasi ketidaksesuaian antara aplikasi yang  
dirancang dengan kebutuhan pengguna. Adapun kebutuhan system yang  
diperlukan antara lain :

1. Kebutuhan Fungsional

Kebutuhan fungsional meliputi pengguna sistem yaitu Staff Keuangandan Admin, adapun deskripsi dari beberapa kebutuhan fungsional antara lain :

1. *Staff* Keuangan

*Staff* Keuangan dapat melakukan berbagai aktifitas diantaranya sebagai berikut:

* Melakuan *login*
* Mengelola profil
* Mengelola harga
* Mengelola pre order
* Mengelola penjualan
* Mengelola terhutang
* Melakukan cetak laporan penjualan

1. Admin

Admin dapat melakukan berbagai aktivitas diantaranya sebagai berikut:

* Melakukan *login*
* Mengelola profil
* Mengelola sales
* Mengelola *staff*
* Mengelola pembelian
* Mengelola gaji
* Mengelola mengelola kebutuhan perusahaan
* Melakukan cetak laporan pembelian
* Melihat grafik barang

1. Kebutuhan non fungsional

Analisa kebutuhan *non fungsional* dilakukan untuk mengetahui spesifikasi kebutuhan yang melibatkan perangkat keras (*hardware),* perangkat lunak (*software*) dan pengguna (*user*). Adapun deskripsi dari beberapa kebutuhan fungsional antara lain :

1. Kebutuhan Perangkat Keras(*Hardware*)

Spesifikasi perangkat keras komputer yang digunakan untuk pembuatan program ini adalah:

* Personal Komputer dengan *processor* Intel Dual Core.
* Memori RAM minimum 4 GB Kingstone DDR4-2133
* Monitor min 11” *resolution* 1200x800.
* Harddisk 320 GB sebagai media penyimpanan.

1. Kebutuhan Perangkat Lunak (Software)

Spesifikasi perangkat lunak yang diperlukan untuk menjalankan aplikasi adalah:

* Windows 7/8/10.
* MySQL versi 8.0.22
* PHP versi 7.4.2
* JavaScript versi ES6
* NodeJS versi 14.15.3 LTS
* Laravel versi 6.0.0
* Visual Studio Code

## **Perancangan Sistem**

### **Use Case Diagram**

*Use case* digunakan untuk menemukan kebutuhan fungsional dari sistem yang dibangun, *Use case* menjelaskan interaksi antara aktor dan inisiator. Pada penelitian ini terdapat 2 aktor yaitu Staff Keuangandan Admin. Adapun rancangan *use case* yang dibuat dalam sistem ini adalah seperti pada Gambar 3.3



Gambar 3. 3 Use Case Diagram

Dalam perancangan *use case* pada sistem yang telah dirancang, beberapa penjelasan tentang masing-masing komponennya adalah sebagai berikut :

1. Definisi Aktor

Dalam *use case* pada Gambar 3.3 setiap aktor memiliki hak akses dan peran masing masing yang dijelaskan pada Tabel 3.1.

Tabel 3. 1 Definisi Aktor

|  |  |  |
| --- | --- | --- |
| **No** | **Aktor** | **Deskripsi** |
| 1. | Staff Keuangan | Staff keuanganadalah orang yang memiliki hak akses untuk dapat mengelola profil, mengelola harga, mengelola *pre order*, mengelola penjualan, mengelola terhutang dan melakukan cetak laporan penjualan |
| 2. | Admin | Admin adalah orang yang memiliki hak akses untuk dapat mengelola profil, mengelola *Staff,* mengelola Sales, mengelola pembelian, mengelola gaji, mengelola kebutuhan perusahaan, melakukan cetak laporan pembelian, melihat grafik barang |

1. Definisi *Use Case*

Dalam use case pada Gambar 3.3. setiap aktor memiliki *use case* masing -masing. *Use case* dari masing masing aktor dijelaskan di dalam Tabel 3.2 berikut ini :

Tabel 3. 2 Definisi Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Aktor** | **Use Case** | **Deskripsi** |
| 1. | *Staff* Keuangan | Mengelola Profil | *Staff* keuangan dapat melihat, menambah, mengubah dan menghapus data profil |
| 2. | *Staff* Keuangan | Mengelola Harga | *Staff* keuangan dapat melihat, menambah, mengubah dan menghapus data profil*.* |
| 3. | *Staff* Keuangan | Mengelola Pre Order | *Staff* keuangan dapat melihat, menambah, mengubah dan menghapus data profil*.* |
| 4. | *Staff* Keuangan | Mengelola Penjualan | *Staff* keuangan dapat melihat, menambah, mengubah dan menghapus data profil*.* |
| 5. | *Staff* Keuangan | Mengelola Terhutang | *Staff* keuangan dapat melihat, menambah, mengubah dan menghapus data profil*.* |
| 6. | *Staff* Keuangan | Melakukan Cetak Laporan Penjualan | *Staff* keuangan dapat mencetak laporan penjualan harian, mingguan, bulanan dan tahunan |
| 7. | Admin | Mengelola Profil | Admin dapat melihat, menambah, mengubah dan menghapus data profil |
| 8. | Admin | Mengelola Staff | Admin dapat melihat, menambah, mengubah dan menghapus data *Staff* |
| 9. | Admin | Mengelola Sales | Admin dapat melihat, menambah, mengubah dan menghapus data *Sales* |
| 10. | Admin | Mengelola Pembelian | Admin dapat melihat, menambah, mengubah dan menghapus data pembelian |
| 11. | Admin | Mengelola Gaji | Admin dapat melihat, menambah, mengubah dan menghapus data gaji |
| 12 | Admin | Mengelola Kebutuhan Perusahaan | Admin dapat melihat, menambah, mengubah dan menghapus data kebutuhan perusahaan |
| 13. | Admin | Melakukan Cetak Laporan Pembelian | Admin dapat mencetak laporan pembelian harian, mingguan, bulanan dan tahunan |
| 14 | Admin | Melihat Grafik Barang | *Staff* dapat melihat grafik barang secara harian, mingguan, bulanan dan tahunan |

### **Activity Diagram**

*Activity* diagram dalam sistem ini menjelaskan skema atu proses dari masing - masing fungsi sistem. Adapun beberapa *activity* diagram tersebut dijabarkan dalam beberapa bagian di bawah ini :

1. *Activity Diagram* Staff Keuangan
2. *Login*

*Activity* diagram ini menjelaskan tentang alur proses untuk *staff* melakukan *Login. Staff* keuanganyang telah memiliki hak akses dapat melakukan *Login* menggunakan *email* dan *password* yang terdatabase. Alur dari *activity* diagram proses *login* adalah *staff* keuanganmembuka Aplikasi tersebut, kemudian *staff* keuangan dapat memasukkan *email* dan *password* yang dimiliki, setelah berhasil *login* maka *staff* keuanganakan melihat halaman *dashboard.* Adapun *activity* diagram dari *login staff* keuangan ditunjukkan pada Gambar 3.4.

**

Gambar 3. 4 Activity Diagram *Login Staff* Keuangan

1. Profil *Staff* Keuangan

*Activity* diagram ini menjelaskan tentang alur proses *staff* keuangan untuk mengelola data profil. Alur dari *activity* diagram proses mengelola profil adalah *staff* keuangan memilih menu profil. Setelah masuk menu, *staff* keuangan dapat melihat, menambah, mengubah dan menghapus data profil. Adapun *activity* diagram dari profil *staff* keuangan adalah seperti pada Gambar 3.5.

**

Gambar 3. 5 Activity Diagram Profil *Staff* Keuangan

1. Menu Harga Barang

*Activity* diagram ini menjelaskan tentang alur proses untuk *Staff* keuangan melihat menu daftar harga. Setelah masuk menu, *staff* keuangan dapat melihat data harga barang. Adapun *activity* diagram dari melihat menu harga barang adalah seperti pada Gambar 3.6.



Gambar 3. 6 Activity Diagram Menu Harga Barang *Staff* Keuangan

1. Menu Terhutang Barang

*Activity* diagram ini menjelaskan tentang alur proses *staff* keuangan untuk mengelola data terhutang barang. Pada menu ini *staff* keuangan dapat mengelola data terhutang barang. Alur dari *activity* diagram proses mengelola terhutang barang adalah *staff* keuangan memilih menu terhutang barang. Setelah masuk menu, *staff* keuangan dapat melihat, menambah, mengubah dan menghapus data terhutang barang. Adapun *activity* diagram dari menu terhutang barang adalah seperti pada Gambar 3.7.



Gambar 3. 7 Activity Diagram Menu Terhutang Barang *Staff* Keuangan

1. Menu Pre Order Barang

*Activity* diagram ini menjelaskan tentang alur proses *staff* keuangan untuk mengelola data *pre order* barang. Pada menu ini *staff* keuangan dapat mengelola data *pre order* barang Alur dari *activity* diagram proses mengelola *pre order* barang adalah *staff* keuangan memilih menu *pre order* barang. Setelah masuk menu, *staff* keuangan dapat melihat, menambah, mengubah dan menghapus data *pre order* barang. Adapun *activity* diagram dari menu *pre order* barang adalah seperti pada Gambar 3.8.

**

Gambar 3. 8 Activity Diagram Menu *Pre Order* Barang *Staff* Keuangan

1. Menu Penjualan

*Activity* diagram ini menjelaskan tentang alur proses *staff* keuangan untuk mengelola data penjualan. Pada menu ini *staff* keuangan dapat mengelola data penjualan. Alur dari *activity* diagram proses mengelola penjualan adalah *staff* keuangan memilih menu penjualan. Setelah masuk menu, *staff* keuangan dapat melihat, menambah, mengubah dan menghapus data penjualan. Adapun *activity* diagram dari menu penjualan adalah seperti pada Gambar 3.9.



Gambar 3. 9 Activity Diagram Menu Penjualan *Staff* Keuangan

1. Laporan Penjualan

* Harian

*Activity* diagram ini menjelaskan tentang alur proses untuk *staff* keuangan melihat menu laporan penjualan harian. Alur dari *activity* diagram proses melihat menu laporan penjualan harian adalah *staff* keuangan memilih menu laporan penjualan harian. Setelah masuk menu, *staff* keuangan dapat melihat data penjualan dan melakukan cetak laporan penjulan harian. Adapun *activity* diagram dari melihat menu laporan penjulan dan proses cetak laporan penjualan adalah seperti pada Gambar 3.10.

**

Gambar 3. 10 Activity Diagram Menu Laporan Penjualan Harian Staff Keuangan

* Mingguan

*Activity* diagram ini menjelaskan tentang alur proses untuk *staff* keuangan melihat menu laporan penjualan mingguan. Alur dari *activity* diagram proses melihat menu laporan penjualan mingguan adalah *staff* keuangan memilih menu laporan penjualan mingguan. Setelah masuk menu, *staff* keuangan dapat melihat data penjualan dan melakukan cetak laporan penjulan mingguan. Adapun *activity* diagram dari melihat menu laporan penjulan dan proses cetak laporan penjualan adalah seperti pada Gambar 3.11.

.

**

Gambar 3. 11 Activity Diagram Menu Laporan Penjualan Mingguan *Staff* Keuangan

* Bulanan

*Activity* diagram ini menjelaskan tentang alur proses untuk *staff* keuangan melihat menu laporan penjualan bulaan. Alur dari *activity* diagram proses melihat menu laporan penjualan bulanan adalah *staff* keuangan memilih menu laporan penjualan bulanan. Setelah masuk menu, *staff* keuangan dapat melihat data penjualan dan melakukan cetak laporan penjulan bulanan. Adapun *activity* diagram dari melihat menu laporan penjulan dan proses cetak laporan penjualan adalah seperti pada Gambar 3.12.

**

Gambar 3. 12 Activiy Diagram Menu Laporan Penjualan Bulanan *Staff* Keuangan

* Tahunan

*Activity* diagram ini menjelaskan tentang alur proses untuk *staff* keuangan melihat menu laporan penjualan tahunan. Alur dari *activity* diagram proses melihat menu laporan penjualan tahunan adalah *staff* keuangan memilih menu laporan penjualan tahunan. Setelah masuk menu, *staff* keuangan dapat melihat data penjualan dan melakukan cetak laporan penjulan tahunan. Adapun *activity* diagram dari melihat menu laporan penjulan dan proses cetak laporan penjualan adalah seperti pada Gambar 3.13.

**

Gambar 3. 13 Activity Diagram Cetak Laporan Penjualan Tahunan *Staff* Keuangan

1. *Activity Diagram* Admin
2. Login

*Activity* diagram ini menjelaskan tentang alur proses untuk admin melakukan *Login.* Adminyang telah memiliki hak akses dapat melakukan *Login* menggunakan *email* dan *password* yang terdatabase. Alur dari *activity* diagram proses *login* adalah adminmembuka Aplikasi tersebut, kemudian admindapat memasukkan *email* dan *password* yang dimiliki, setelah berhasil *login* maka adminakan melihat halaman *dashboard.* Adapun *activity* diagram dari *login* admin ditunjukkan pada Gambar 3.14.

**

Gambar 3. 14 Activity Diagram *Login* Admin

1. Profil Saya

*Activity* diagram ini menjelaskan tentang alur proses admin untuk mengelola data profil. Alur dari *activity* diagram proses mengelola profil adalah adminmemilih menu profil. Setelah masuk menu, admin dapat melihat, menambah, mengubah dan menghapus data profil. Adapun *activity* diagram dari profil admin adalah seperti pada Gambar 3.15.

**

Gambar 3. 15 Activity Diagram Profil Admin

1. Menu *Staff* Keuangan

*Activity* diagram ini menjelaskan tentang alur proses admin untuk mengelola data *staff* keuangan. Pada menu ini admin dapat mengelola data *staff* keuangan. Alur dari *activity* diagram proses mengelola *staff* keuangan adalah admin memilih menu *staff* keuangan. Setelah masuk menu, admin dapat melihat, menambah, mengubah dan menghapus data *staff* keuangan. Adapun *activity* diagram dari menu *staff* keuangan adalah seperti pada Gambar 3.16.

**

Gambar 3. 16 Activity Diagram Menu Staff Keuangan Admin

1. Menu Salesman

*Activity* diagram ini menjelaskan tentang alur proses admin untuk mengelola data salesman. Pada menu ini admin dapat mengelola data salesman. Alur dari *activity* diagram proses mengelola salesman adalah admin memilih menu salesman. Setelah masuk menu, admin dapat melihat, menambah, mengubah dan menghapus data salesman. Adapun *activity* diagram dari menu tips adalah seperti pada Gambar 3.17.

**

Gambar 3. 17 Activity Diagram Menu Salesman Admin

1. Menu Data Barang

*Activity* diagram ini menjelaskan tentang alur proses admin untuk mengelola data barang. Pada menu ini admin dapat mengelola data barang. Alur dari *activity* diagram proses mengelola barang adalah admin memilih menu data barang. Setelah masuk menu, admin dapat melihat, menambah, mengubah dan menghapus data barang. Adapun *activity* diagram dari menu data barang adalah seperti pada Gambar 3.18.

**

Gambar 3. 18 Activity Diagram Menu Data Barang Admin

1. Menu Grafik Barang

* Harian

*Activity* diagram ini menjelaskan tentang alur proses untuk adminmelihat menu grafik barang harian. Setelah masuk menu admindapat melihat grafik barang harian. Adapun *activity* diagram dari melihat menu grafik barang harian adalah seperti pada Gambar 3.19.

**

Gambar 3. 19 Activity Diagram Menu Grafik Barang Harian Admin

* Mingguan

*Activity* diagram ini menjelaskan tentang alur proses untuk adminmelihat menu grafik barang mingguan Setelah masuk menu admindapat melihat grafik barang mingguan. Adapun *activity* diagram dari melihat menu grafik barang mingguan adalah seperti pada Gambar 3.20

**

Gambar 3. 20 Activity Diagram Menu Grafik Barang Mingguan Admin

* Bulanan

*Activity* diagram ini menjelaskan tentang alur proses untuk adminmelihat menu grafik barang bulanan. Setelah masuk menu admindapat melihat grafik barang bulanan. Adapun *activity* diagram dari melihat menu grafik barang bulanan adalah seperti pada Gambar 3.21.

**

Gambar 3. 21 Activity Diagram Menu Grafik Barang Bulanan Admin

* Tahunan

*Activity* diagram ini menjelaskan tentang alur proses untuk adminmelihat menu grafik barang tahunan. Setelah masuk menu admindapat melihat grafik barang tahunan. Adapun *activity* diagram dari melihat menu grafik barang tahunan adalah seperti pada Gambar 3.22.

**

Gambar 3. 22 Activity Diagram Menu Grafik Barang Tahunan Admin

1. Menu Pembelian

*Activity* diagram ini menjelaskan tentang alur proses admin untuk mengelola data pembelian. Pada menu ini admin dapat mengelola data pembelian. Alur dari *activity* diagram proses mengelola pembelian adalah admin memilih menu pembelian. Setelah masuk menu, admin dapat melihat, menambah, mengubah dan menghapus data pembelian. Adapun *activity* diagram dari menu pembelian adalah seperti pada Gambar 3.23.

**

Gambar 3. 23 Activity Diagram Menu Pembelian Admin

1. Menu Gaji

*Activity* diagram ini menjelaskan tentang alur proses admin untuk mengelola data gaji . Pada menu ini admin dapat mengelola data gaji. Alur dari *activity* diagram proses mengelola gaji adalah admin memilih menu gaji. Setelah masuk menu, admin dapat melihat, menambah, mengubah dan menghapus data gaji. Adapun *activity* diagram dari menu gaji adalah seperti pada Gambar 3.24.

**

Gambar 3. 24 Activity Diagram Menu Gaji Admin

1. Menu Kebutuhan Lainnya

* Data Kebutuhan Lain

*Activity* diagram ini menjelaskan tentang alur proses admin untuk mengelola data kebutuhan lain. Pada menu ini admin dapat mengelola data kebutuhan lain yang perusahaan butuhkan. Alur dari *activity* diagram proses mengelola kebutuhan lain adalah admin memilih menu kebutuhan lain. Setelah masuk menu, admin dapat melihat, menambah, mengubah dan menghapus data kebutuhan lain. Adapun *activity* diagram dari menu kebutuhan lain adalah seperti pada Gambar 3.25.

**

Gambar 3. 25 Activity Diagram Menu Kebutuhan Lain Admin

* Form Kebutuhan Lain

*Activity* diagram ini menjelaskan tentang alur proses admin untuk mengelola form kebutuhan lain. Pada menu ini admin dapat mengelola form kebutuhan lain. Alur dari *activity* diagram proses mengelola form kebutuhan lain adalah admin memilih menu form kebutuhan lain. Setelah masuk menu, admin dapat melihat, menambah, mengubah dan menghapus data form kebutuhan lain. Adapun *activity* diagram dari menu form kebutuhan lain adalah seperti pada Gambar 3.26.

**

Gambar 3. 26 Activity Diagram Form Kebutuhan Lain Admin

1. Cetak Laporan Pembelian

* Harian

*Activity* diagram ini menjelaskan tentang alur proses untuk admin melihat menu Laporan Pembelian. Alur dari *activity* diagram proses melihat menu laporan Pembelian adalah *customer* memilih menu transaksi. Setelah masuk menu, *customers* dapat melihat data transaksi dan melakukan cetak data transaksi. Adapun *activity* diagram dari melihat menu transaksi dan proses cetak transaksi adalah seperti pada Gambar 3.27.

**

Gambar 3. 27 Activity Diagram Cetak Laporan Pembelian Harian Admin

* Mingguan

*Activity* diagram ini menjelaskan tentang alur proses untuk admin melihat menu transaksi. Alur dari *activity* diagram proses melihat menu transaksi adalah *customer* memilih menu transaksi. Setelah masuk menu, *customers* dapat melihat data transaksi dan melakukan cetak data transaksi. Adapun *activity* diagram dari melihat menu transaksi dan proses cetak transaksi adalah seperti pada Gambar 3.28.

**

Gambar 3. 28 Activity Diagram Cetak Laporan Pembelian Mingguan Admin

* Bulanan

*Activity* diagram ini menjelaskan tentang alur proses untuk admin melihat menu transaksi. Alur dari *activity* diagram proses melihat menu transaksi adalah *customer* memilih menu transaksi. Setelah masuk menu, *customers* dapat melihat data transaksi dan melakukan cetak data transaksi. Adapun *activity* diagram dari melihat menu transaksi dan proses cetak transaksi adalah seperti pada Gambar 3.29.

**

Gambar 3. 29 Activity Diagram Cetak Laporan Pembelian Bulanan Admin

* Tahunan

*Activity* diagram ini menjelaskan tentang alur proses untuk admin melihat menu transaksi. Alur dari *activity* diagram proses melihat menu transaksi adalah *customer* memilih menu transaksi. Setelah masuk menu, *customers* dapat melihat data transaksi dan melakukan cetak data transaksi. Adapun *activity* diagram dari melihat menu transaksi dan proses cetak transaksi adalah seperti pada Gambar 3.30.

**

Gambar 3. 30 Activity Diagram Cetak Laporan Pembelian Tahunan Admin

### **Class Diagram**

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Atribut merupakan variable-variabel yang dimiliki oleh suatu kelas sedangkan operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas. Adapun class diagram yang dideskripsikan pada Gambar 3.31.



Gambar 3. 31 Class Diagram

### **Relasi Antar Tabel**

Dalam database sistem ini terdapat 14 tabel, yaitu tabel admins, log activities, staffs, roles, users, sales, out\_transaksies, in\_transaksies, incomes, costs, barangs, hargais, gajis, assets. Adapun relasi dari masing - masing tabel di deskripsikan pada Gambar 3.32.



Gambar 3. 32 Relasi Antar Tabel

### **Arsitektur Diagram**

Arsitektur dalam proses kerja Sistem Informasi Keuangan Menggunakan Metode Analisis Rasio Keuangan Berbasis Web Pada CV Candra Food pada Gambar 3. menunjukkan tentang arsitektur yang dimiliki oleh sistem. Sistem Informasi Keuangan Menggunakan Metode Analisis Rasio Keuangan Berbasis Web Pada CV Candra Food terdapat *staff* keuangan dan admin. Server menyediakan data yang ditujukan untuk bagian keuangan dan admin dapat mengelola data

### **Antarmuka Sistem**

Antarmuka Sistem diperlukan di dalam proses pembuatan aplikasi halaman situs web karena bagian ini langsung berhubungan dengan *user*. Rancangan interface halaman Staff dan Admin adalah sebagai berikut

1. Antarmuka Sistem Staff

Pada Tabel 3.3 merupakan antarmuka sistem staff yang dilengkapi dengan penjelasan fitur-fitur yang dapat di akses oleh staff

Tabel 3. 3 Antarmuka Sistem Staff

|  |  |  |
| --- | --- | --- |
| No | Keterangan | Penjelasan |
| 1 |  | Pada halaman login ini *staff* keuangan memasukkan *email* dan *password* untuk masuk ke halaman utama *staff* keuangan |
| 2 |  | Pada halaman dashboard ini *staff* keuangan dapat melihat data ringkas penjualan, konfirmasi diterima, konfirmasi ditolak dan data staff |
| 3 |  | Pada halaman harga barang ini *staff* keuangan dapat melihat data harga barang secara keseluruhan, mengedit harga barang dan menghapus harga barang. *Staff* keuangan juga dapat mencari data harga barang berdasarkan nama sales |
| 4 |  | Pada halaman pilih sales, *staff* keuangan memilih nama sales sebagai syarat untuk masuk ke dalam halaman from harga barang |
| 5 |  | Pada halaman form tambah harga barang, *staff* keuangan dapat melakukan tambah harga barang berdasarkan kode barang. |
| 6 |  | Pada halaman terhutang barang ini *staff* keuangan dapat melihat data harga barang secara keseluruhan, mengedit terhutang barang dan menghapus terhutang barang. *Staff* keuangan juga dapat mencari data terhutang barang berdasarkan nama sales |
| 7 |  | Pada halaman pilih sales, *staff* keuangan memilih nama sales sebagai syarat untuk masuk ke dalam halaman from terhutang barang |
| 8 |  | Pada halaman form tambah terhutang barang, *staff* keuangandapat melakukan tambah terhutang berdasakan kode terhutang. |
| 9 |  | Pada halaman pre order *staff* keuangan dapat melihat data pre order secara keseluruhan, melihat detail pre order dan mengonfirmasi pre order sales. *Staff* keuangan juga dapat mencari data pre order berdasarkan nama sales |
| 10 |  | Pada halaman detail pre order, *staff* keuangan melihat pre order secara detail dan dapat melakukan cetak *delivery order*. |
| 11 |  | Pada halaman penjualan ini *staff* keuangan dapat melihat data penjualan secara keseluruhan dan melihat detai penjualan. *Staff* keuangan juga dapat mencari data penjualan berdasarkan nama sales |

1. Antarmuka Sistem Admin

Pada Tabel 3.3 merupakan antarmuka sistem admin yang dilengkapi dengan penjelasan fitur-fitur yang dapat di akses oleh admin

Tabel 3. 4 Antarmuka Sistem Admin

|  |  |  |
| --- | --- | --- |
| No | Keterangan | Penjelasan |
| 1 |  | Pada halaman login ini admin memasukkan *email* dan *password* untuk masuk ke halaman utama *admin* |
| 2 |  | Pada halaman dashboard ini *admin* dapat melihat data ringkas penjualan, konfirmasi diterima, konfirmasi ditolak dan data sales terbaru |
| 3 |  | Pada halaman profil saya *admin* dapat melihat data profil *admin* dan dapat mengubah data diri profil *admin* |
| 4 |  | Pada halaman ubah password *admin* dapat mengubah password lama menjadi password baru |
| 5 |  | Pada halaman staff, *admin* dapat melihat data staff, menambah data staff dengan memilih button tambah, mengubah data staff dengan memilih button ubah dan menghapus data staff dengan memilih button hapus |
| 6 |  | Pada halaman tambah data staff, *admin* dapat menambahkan data staff dengan mengisi form yang telah disediakan |
| 7 |  | Pada halaman detail staff, *admin* dapat melihat data detail staff, mengubah data staff dan mengatur ulang password secara default |
| 8 |  | Pada halaman sales, *admin* dapat melihat data sales, menambah data sales dengan memilih button tambah, mengubah data sales dengan memilih button ubah dan menghapus data sales dengan memilih button hapus |
| 9 |  | Pada halaman tambah data sales, *admin* dapat menambahkan data sales dengan mengisi form yang telah disediakan |
| 10 |  | Pada halaman detail sales, *admin* dapat melihat data detail sales, mengubah data sales dan mengatur ulang password secara default |
| 11 |  | Pada halaman data barang, *admin* dapat melihat data barang, menambah data barang dengan memilih button tambah, mengubah data barang dengan memilih button ubah dan menghapus data barang dengan memilih button hapus |
| 12 |  | Pada halaman tambah data barang, *admin* dapat menambahkan data barang dengan mengisi form yang telah disediakan |
| 13 |  | Pada halaman *chart* barang harian, *admin* dapat melihat data grafik barang masuk dan barang keluar serta melihat persentase antara barang masuk dan barang keluar berdasarkan data harian. *Admin* juga dapat mencari *chart* barang berdasarkan tanggal dan kode barang |
| 14 |  | Pada halaman *chart* barang mingguan, *admin* dapat melihat data grafik barang masuk dan barang keluar serta melihat persentase antara barang masuk dan barang keluar berdasarkan data mingguan. *Admin* juga dapat mencari *chart* barang berdasarkan minggu, bulan dan kode barang |
|  |  | Pada halaman *chart* barang bulanan, *admin* dapat melihat data grafik barang masuk dan barang keluar serta melihat persentase antara barang masuk dan barang keluar berdasarkan data bulanan. *Admin* juga dapat mencari *chart* barang berdasarkan bulan dan kode barang |
|  |  | Pada halaman *chart* barang tahunan, *admin* dapat melihat data grafik barang masuk dan barang keluar serta melihat persentase antara barang masuk dan barang keluar berdasarkan data tahunan. *Admin* juga dapat mencari *chart* barang berdasarkan tahunan dan kode barang |
|  |  | Pada halaman data pembelian, *admin* dapat melihat data pembelian, menambah data pembelian dengan memilih button tambah, mengubah data pembelian dengan memilih button ubah dan menghapus data barang dengan memilih button hapus |
|  |  | Pada halaman pilih pembelian, *admin* memilih kode pembelian untuk masuk ke form pembelian. |
|  |  | Pada halaman tambah pembelian, *admin* dapat menambahkan data pembelian dengan mengisi form yang telah disediakan |
|  |  | Pada halaman data gaji, *admin* dapat melihat data gaji, menambah data gaji dengan memilih button tambah, mengubah data gaji dengan memilih button ubah dan menghapus data gaji dengan memilih button hapus |
|  |  | Pada halaman tambah gaji, *admin* dapat melakukan tambah gaji pegawai dengan mengisi form yang telah disediakan |
|  |  | Pada halaman data kebutuhan perusahaan, *admin* dapat melihat data gaji, menambah data kebutuhan perusahaan dengan memilih button tambah, melihat detail data kebutuhan perusahaan dengan memilih button ubah dan menghapus data gaji dengan memilih button hapus |
|  |  | Pada halaman tambah kebutuhan perusahaan, *admin* dapat menambahkan data kebutuhan perusahaan dengan mengisi form yang telah disediakan |
|  |  | Pada halaman laporan keuangan harian, *admin* dapat melihat laporan keuangan harian dan mencetak laporan dengan memilih button cetak laporan |
|  |  | Pada halaman laporan keuangan mingguan, *admin* dapat melihat laporan keuangan mingguan dan mencetak laporan dengan memilih button cetak laporan |
|  |  | Pada halaman laporan keuangan bulanan, *admin* dapat melihat laporan keuangan bulanan dan mencetak laporan dengan memilih button cetak laporan |
|  |  | Pada halaman laporan keuangan tahunan, *admin* dapat melihat laporan keuangan tahunan dan mencetak laporan dengan memilih button cetak laporan |

## **Skenario Pengujian Sistem**

Pengujian sistem diperlukan dalam pembuatan sistem untuk mengetahui apakah sistem berjalan dengan baik atau tidak. Pengujian Sistem dilakukan dengan menguji tampilan *interface* dari beberapa hak akses *user* yang meliputi : *owner* dan *sales*

Pengujian metode *Black* *Box* adalah pengujian terhadap fungsionalitas *input* atau *output* dari suatu perangkat lunak. Penguji mendefinisikan sekumpulan kondisi *input* kemudian melakukan sejumlah pengujuan terhadap program sehingga menghasilkan suatu *output* yang nilainya dapat dievaluasi. Adapun skenario dari pengujian sistem yang dibut adalah seperti Tabel

Tabel 3. 5 Pengujian Sistem *Staff*

|  |  |  |
| --- | --- | --- |
| Pengujian Sistem *Staff* | | |
| Pengujian | Yang diharapkan | Hasil |
| Login | Sistem bisa melakukan perbedaan hak akses login | Dikatakan berhasil jika bisa melakukan perbedaan hak akses login. |
| Dashboard | * Sistem bisa menampilkan jumlah data pembelian, jumlah konfirmasi diterima dan konfirmasi ditolak. * Sistem bisa menampilkan data sales dan transaksi masuk. | * Dikatakan berhasil jika sistem Bisa menampilkan jumlah data pembelian, jumlah konfirmasi diterima dan konfirmasi ditolak. * Dikatakan berhasil jika sistem bisa menampilkan data sales dan transaksi masuk. |
| Client | Sistem bisa menampilkan, menambah dan merubah data client | Dikatakan berhasil jika sistem bisa menampilkan,menambah dan merubah data client |
| Pesanan | Sistem bisa menampilkan, menambah dan merubah data pesanan | Dikatakan berhasil jika sistem bisa menampilkan, menambah dan merubah data pesanan |
| Data Transaksi | Sistem bisa menampilkan data transaksi dan mencetak data transaksi | Dikatakan berhasil jika sistem bisa menampilkan data transaksi dan mencetak data transaksi |
| Tips | Sistem bisa menampilkan, menambah, merubah dan menghapus data tips | Dikatakan berhasil jika sistem bias menampilkan, menambah, merubah dan menghapus data tips |
| Profile | Sistem bisa menampilkan, menambah, merubah dan menghapus data profile | Dikatakan berhasil jika sistem bisa menampilkan, menambah, mengubah dan menghapus data profile |
| Treatment | Sistem bisa menampilkan, menambah, merubah dan menghapus data treatment | Dikatakan berhasil jika sistem bisa menampilkan, menambah, mengubah dan menghapus data treatment |

Tabel 3. 6 Pengujian Sistem *Sales*

|  |  |  |
| --- | --- | --- |
| Pengujian Sistem *Sales* | | |
| Pengujian | Yang diHarapkan | Hasil |
| Login | Sistem bisa melakukan perbedaan hak akses login | Dikatakan berhasil jika  bisa melakukan perbedaan hak akses login. |
| Home | Sistem bisa menampilkan menu home | Dikatakan berhasil jika sistem bisa menampilkan menu home . |
| Profile | Sistem bisa menampilkan data profile | Dikatakan berhasil jika sistem menampilkan data profile |
| Tips | Sistem bisa menampilkan data tips | Dikatakan berhasil jika sistem bisa menampilkan data tips |
| Cek Shoes | Sistem bisa menampilkan data transaksi sepatu yang di laundry berdasarkan nomer nota | Dikatakan berhasil jika sistem bisa menampilkan data transaksi sepatu yang di laundry berdasarkan nomer nota |
| Contact | Sistem bisa menampilkan menu contact | Dikatakan berhasil jika sistem bisa bisa menampilkan menu contact |
| Pesan | Setelah melakukan login sistem bisa menampilkan, menambah, merubah dan menghapus data pesanan | Dikatakan berhasil jika sistem bisa Setelah melakukan login sistem bisa menampilkan, menambah, merubah dan menghapus data pesanan |
| Checkout | Setelah login sistem bisa menampilkan data pesan dan melakukan proses pembayaran | Dikatakan berhasil Setelah login sistem bisa menampilkan data pesan dan melakukan proses pembayaran |

# **BAB IV JADWAL KEGIATAN**

Waktu pelaksanaan penelitian akan dilaksanakan mulai bulan Oktober sampai April. Berikut jadwal penelitian yang telah di uraikan pada tabel 4.1 Jadwal  
Penelitan.

# **DAFTAR PUSTAKA**