

Perancangan Dan Implementasi Sistem Informasi Laporan Keuangan Apotek Jati Jaya
Dengan Menggunakan Bahasa Pemrograman PHP Dan Database MySQL

Yandri

Fakultas Ilmu Komputer, UPI "YPTK" Padang

Email : hendriekaputra10@gmail.com

Corresponding Author : hendriekaputra10@gmail.com

Abstract

The development of technology at this time continues to grow rapidly. This also has an impact on the development of the business world including trade. Tight business competition encourages every company to be more careful in determining the policies taken, so that fast, accurate and relevant information is needed to facilitate human work and minimize errors in every transaction activity. One of the problems that often arises is the incompatibility of financial statements with transactions that occur. To prevent errors in the processing of reports required an information system that has been computerized to facilitate the creation of reports supported by a basic method of accounting. With the computer can facilitate employees in managing the Financial Statement Information System. With this financial reporting system, the data will be stored in a database, so that the data is more structured and neatly arranged. With the Financial Reporting System can facilitate store employees in knowing the profit or loss.

Keywords: Website, PHP, MySQL, E-library.

1. Pendahuluan

Perkembangan teknologi pada saat ini terus berkembang dengan pesat. Hal tersebut juga berdampak terhadap perkembangan dunia bisnis termasuk perdagangan. Persaingan bisnis yang ketat mendorong setiap perusahaan untuk lebih cermat dalam menentukan kebijakan yang diambil, sehingga dibutuhkan informasi yang cepat, akurat dan relevan untuk mempermudah pekerjaan manusia serta meminimalisir kesalahan setiap kegiatan transaksi. Salah satu masalah yang sering timbul adalah ketidakcocokan laporan keuangan dengan transaksi yang terjadi. Untuk mencegah kesalahan dalam pengolahan laporan diperlukan sebuah sistem informasi yang sudah terkomputerisasi untuk mempermudah pembuatan laporan didukung dengan sebuah metode dasar akuntansi.

2. Landasan Teori

Konsep Dasar Sistem Informasi

Sistem informasi bukan merupakan hal yang baru. Yang baru adalah komputerisasinya. Sebelum adanya komputer, teknik penyaluran informasi yang memungkinkan manajer merencanakan serta mengendalikan operasi yang ada. Komputer menambahkan satu atau lebih dimensi, seperti kecepatan, ketelitian, dan penyediaan data dengan volume yang lebih besar yang memberikan bahan pertimbangan yang lebih banyak untuk mengambil keputusan (Tata Sutabri, 2016).

Pengertian Sistem

Sistem (*system*) pada dasarnya adalah sekelompok unsur yang erat hubungannya satu dengan yang lain, yang berfungsi bersama-sama untuk mencapai tujuan tertentu. Suatu sistem dibuat untuk menangani sesuatu yang berulang kali atau secara rutin terjadi. Suatu sistem dapat dirumuskan sebagai setiap kumpulan komponen atau subsistem yang dirancang untuk mencapai suatu tujuan. Dengan pendekatan sistem kita berhubungan dengan komponen perseorangan, dan kita lebih menekankan perannya di dalam sistem dari pada perannya sebagai suatu keseluruhan individu

Karakteristik Sistem

Selain itu sebuah sistem juga memiliki karakteristik atau sifat-sifat tertentu, yang mencirikan bahwa hal tersebut bisa dikatakan sebagai sebuah sistem. Adapun karakteristik yang dimaksud adalah sebagai berikut :

1. **Komponen Sistem (*Components*)**
Suatu sistem terdiri dari komponen-komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan.
2. **Batasan sistem (*Boundary*)**
Batasan sistem merupakan daerah yang membatasi antara sistem dengan sistem lainnya atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisah-pisahkan.
3. **Lingkungan Luar Sistem (*Environment*)**
Bentuk apapun yang ada diluar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut dengan lingkungan luar sistem.
4. **Penghubung Sistem (*Interface*)**
Media yang menghubungkan sistem dengan subsistem yang lain yang disebut hubungan sistem atau *interface*. Penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lain.
5. **Masukan Sistem (*input*)**
Energi yang dimasukkan ke dalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*).
6. **Keluaran Sistem (*Output*)**
Hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain. Seperti contoh sistem informasi, keluaran yang dihasilkan adalah informasi, dimana informasi ini dapat digunakan sebagai masukan untuk pengambilan keputusan atau hal-hal yang merupakan input bagi subsistem.
7. **Pengolah Sistem (*Procces*)**
Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran. Sebagai contoh sistem akuntansi. Sistem ini akan mengolah data transaksi menjadi laporan-laporan yang dibutuhkan oleh pihak manajemen.
8. **Sasaran Sistem (*Objective*)**
Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat deterministik. Kalau suatu sistem tidak memiliki sasaran, maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang direncanakan.

Klasifikasi Sistem

Menurut Tata Sutabri (2016) Sistem dapat diklasifikasikan dari beberapa sudut pandang, antaranya adalah sebagai berikut :

1. Sistem Abstrak dan Sistem Fisik
Sistem Abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya sistem teologia, yaitu sistem yang berupa pemikiran hubungan antara manusia dengan tuhan, sedangkan sistem fisik merupakan sistem yang ada secara fisik, misalnya sistem komputer, sistem produksi, sistem penjualan, sistem administrasi personalia dan lain sebagainya.
2. Sistem alamiah dan sistem buatan manusia
Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem perputaran bumi, terjadinya siang malam, pergantian musim. Sedangkan sistem buatan manusia merupakan sistem yang melibatkan interaksi manusia dengan mesin yang disebut *human machine system*. Sistem informasi berbasis komputer merupakan contoh *human machine system* karena menyangkut penggunaan komputer yang berinteraksi dengan manusia.
3. Sistem Deterministik dan Sistem Probabilistik
Sistem yang beroperasi dengan tingkah laku yang dapat diprediksi disebut sistem *deterministic*. Sistem komputer adalah contoh dari sistem yang tingkah lakunya dapat dipastikan berdasarkan program-program komputer yang dijalankan, sedangkan sistem yang bersifat *probabilistic* adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur *probabilistic*.
4. Sistem Terbuka dan Sistem Tertutup
Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh oleh lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa campur tangan pihak luar, sedangkan sistem terbuka adalah sistem yang berhubungan dan dipengaruhi oleh lingkungan luarnya. Sistem ini menerima masukan dan menghasilkan keluaran untuk subsistem lainnya.

Pengertian PHP

Menurut Eka Praja Wiyata Mandala (2015) *Hypertext Preprocessor* (PHP) adalah bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML. PHP banyak dipakai untuk membuat situs web dinamis. PHP dapat digunakan untuk membangun sebuah CMS (*Content Management System*). PHP (*Hypertext Preprocessor*) merupakan bahasa utama *script server side* yang disisipkan pada HTML yang dijalankan di server, dan juga bisa digunakan untuk membuat aplikasi desktop (Betha Sidik, 2017).

Sejarah singkat PHP

PHP pertama kali dibuat oleh Rasmus Lerdoff pada tahun 1994. Versi pertama yang di release pada awal tahun 1995, dikenal sebagai Tool *Personal Home Page*, yang terdiri atas engine parser yang sangat sederhana yang hanya mengerti beberapa makro khusus dan sejumlah utilitas yang sering digunakan pada halaman-halaman web. Parser diprogram ulang pada tahun 1995 dan diberi nama PHP/FI versi 2.0 FI berasal dari paket Ramus lainnya.

Pada pertengahan 1997 terjadi perubahan pengembangan PHP. Pengembangan dilakukan oleh tim yang terorganisasi bukan oleh Rasmus sendiri saja. Parser

dikembangkan oleh Zeev Surasaki dan Andi Gutmans yang kemudian menjadi dasar untuk versi 3.

Versi terakhir (PHP4) menggunakan engine script Zend untuk lebih meningkatkan kinerja dan mempunyai dukungan yang banyak berupa ekstensi dan fungsi dari berbagai *library* pihak ketiga (*third party*), dan berjalan seolah modul asli dari berbagai server web yang populer.

Sejak Januari 2001 PHP3 dan PHP4 disertakan pada sejumlah produk server web komersial seperti server web StrongHold RedHat. PHP versi 4.2.0 di-*release* pada tanggal 22 April 2002. Perbaikan pada bug (kesalahan-kesalahan) terutama pada upload file melalui browser telah dibetulkan, dan banyak penambahan fungsi yang lebih memudahkan lagi pengembang aplikasi untuk membuat program yang lebih baik. Jumlah fungsi yang ada sampai dengan versi ini mencapai ribuan fungsi dan dikelompokkan menjadi 109 fungsi, banyak bertambah dari versi sebelumnya. Sampai dengan versi 4.3.7 tercatat ada 125 kelompok fungsi yang dimiliki oleh PHP.

Saat ini pengembangan PHP telah direncanakan untuk mulai memasuki versi 6, sampai dengan buku ini disusun, PHP versi 5 telah mencapai versi 5.3.8 yang stabil, dan telah dikeluarkan juga versi 5.4 yang baru dilepaskan sebagai versi kandidat ke-4, PHP versi 5.4.0 RC4 (Betha Sidik, 2017).

My Structure Query Language (MySQL)

Menurut Eka Praja Wiyata Mandala (2015) SQL merupakan singkatan dari *Structure Query Language* (SQL). SQL digunakan untuk melakukan akses dan memanipulasi *database*.

MySQL adalah Program *database* yang berbasis *open source*. MySQL mampu menangani *database* yang kompleks dan cukup besar. MySQL juga dapat menangani *database client server*.

Konsep Basis Data (*Database*)

Basis data (*database*) dapat dibayangkan sebagai sebuah lemari arsip. Jika memiliki lemari arsip dan bertugas mengelolanya, maka akan melakukan hal-hal seperti memberi sampul, memberi nomor, lalu menempatkan arsip-arsip tersebut dengan urutan tertentu dalam lemari.

Basis data dapat diartikan sebagai markas atau gudang, temoat bersarang atau berkumpul. Sedangkan data merupakan representasi fakta dunia nyata yang mewakili suatu objek seperti manusia, barang, hewan peristiwa, konsep, keadaan, dan sebagainya yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya (Fathansyah, 2015).

Konsep Dasar Keuangan

Pengertian Laporan Keuangan

Laporan keuangan akan menjadi lebih bermanfaat untuk pengambilan keputusan ekonomi, apabila dengan informasi laporan keuangan tersebut dapat diprediksi apa yang akan terjadi di masa mendatang. Dengan mengolah lebih lanjut laporan keuangan melalui proses perbandingan, evaluasi dan analisis trend, akan diperoleh prediksi tentang apa yang mungkin akan terjadi di masa mendatang. Di sinilah arti pentingnya suatu analisis terhadap laporan keuangan (Safriadi Pohan, 2017).

Tujuan Laporan Keuangan

Tujuan laporan keuangan adalah memberikan informasi yang berguna bagi investor dan kreditor dalam pengambilan keputusan investasi dan kredit. Jenis keputusan yang dibuat oleh pengambil keputusan sangatlah beragam, begitu juga dengan metode pengambilan keputusan yang mereka gunakan dan kemampuan mereka untuk memproses informasi.

Tujuan umum laporan keuangan adalah sebagai berikut :

1. Memberikan informasi yang terpercaya tentang sumber daya ekonomi dan kewajiban perusahaan.
 2. Memberikan informasi yang terpercaya tentang sumber kekayaan bersih yang berasal dari kegiatan usaha mencari laba.
 3. Memungkinkan untuk menaksir potensi perusahaan dalam menghasilkan laba.
- Memberikan informasi yang diperlukan lainnya tentang perubahan aset dan kewajiban (Hery, 2015).

3. Metode Penelitian

Software Development Life Cycle (SDLC)

SDLC atau *Software Development Life Cycle* atau sering disebut *System Development Life Cycle* adalah proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi dalam mengembangkan sistem-sistem perangkat lunak sebelumnya (Rosa A.S dan M Shalahuddin, 2018).

Menurut Rosa A.S dan M Shalahuddin (2018) tahap-tahap yang ada pada SDLC adalah sebagai berikut:

1. Perencanaan (*Planning*)
Mengembangkan rencana manajemen proyek dan dokumen perencanaan lainnya.
2. Analisa Kebutuhan (*requirements analysis*)
Menganalisa kebutuhan pemakai sistem perangkat lunak (*user*) dan mengembangkan kebutuhan *user*.
3. Desain Sistem (*Design System*)
Pada tahap ini sistem analisa sudah mendapatkan gambaran tentang sistem yang akan dibuat. Mentransformasikan kebutuhan detail menjadi kebutuhan yang sudah lengkap, dokumen desain sistem fokus pada bagaimana dapat memenuhi fungsi-fungsi yang dibutuhkan.
4. Tahap Seleksi Sistem (*Selection System*)
Mendemonstrasikan sistem perangkat lunak bahwa telah memenuhi kebutuhan yang dispesifikasikan.
5. Tahap Implementasi Sistem (*Implementation System*)
Termasuk pada persiapan implementasi, implementasi perangkat lunak pada lingkungan *user* dan menjalankan resolusi dari permasalahan yang teridentifikasi dari fase integrasi dan pengujian. Pada tahap ini sistem sudah siap untuk dioperasikan.
6. Tahap Perawatan Sistem (*Maintenance System*)
Mendeskrripsikan pekerjaan untuk mengoperasikan dan memelihara sistem informasi.

Alat Bantu Perancangan Sistem

Pada perkembangan teknologi perangkat lunak, diperlukan adanya bahasa yang digunakan untuk memodelkan perangkat lunak yang akan dibuat dan perlu adanya standarisasi agar orang di berbagai negara dapat mengerti permodelan perangkat lunak (Rosa A.S dan M.Shalahuddin, 2018).

Pada perkembangan teknik pemrograman berorientasi objek, muncullah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek, yaitu *Unified Modeling Language* (UML).

Unified Modeling Language (UML)

Dalam jurnal (Diah Puspitasari, 2016), “*Sistem Informasi Perpustakaan Sekolah Berbasis Web*”, UML adalah salah standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek (Sukamto dan Shalahuddin, 2013).

Use Case Diagram

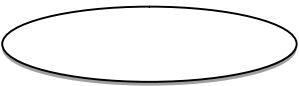

Use case atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use Case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan apa saja yang berhak menggunakan fungsi-fungsi itu (Rosa A. S dan M. Shalahuddin, 2018).

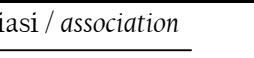
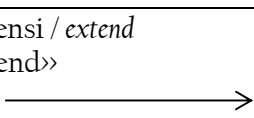
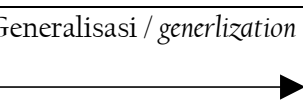
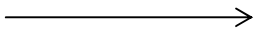
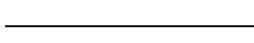
Ada dua hal utama pada *use case* yaitu pendefenisian apa yang disebut aktor dan *use case* (Rosa A. S dan M. Shalahuddin, 2018).

1. Aktor merupakan orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambaran orang, tapi aktor belum tentu merupakan orang.
2. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

Simbol-simbol yang digunakan pada *diagram use case* dapat dilihat pada Tabel 1 berikut ini :

Tabel 1. Simbol-Simbol yang digunakan pada *Use Case Diagram*

No	Simbol	Deskripsi
1		Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antara unit atau aktor.
2	Aktor / actor Nama actor 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri.

3	Asosiasi / <i>association</i> 	Komunikasi antara aktor dan <i>usecase</i> yang berpartisipasi pada <i>usecase</i> atau <i>usecase</i> memiliki interaksi dengan aktor.
4	Ekstensi / <i>extend</i> «extend» 	Relasi <i>usecase</i> tambahan ke sebuah <i>usecase</i> dimana <i>usecase</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.
5	Generalisasi / <i>generalization</i> 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
6	Menggunakan / <i>include/ uses</i> «include»  «uses» 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> .

Sumber : Buku Rekayasa Perangkat Lunak (Rosa A. S dan M. Shalahuddin, 2018)

Class Diagram

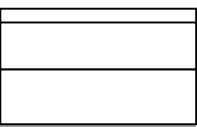
Diagram kelas atau class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi (Rosa A. S dan M. Shalahuddin, 2014).


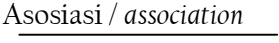
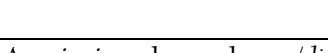

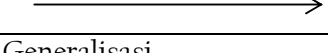
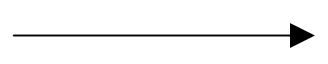
1. Atribut merupakan variabel-variabel yang dimiliki suatu kelas
2. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas

Diagram kelas dibuat gara pembuat program atau *programmer* membuat kelas-kelas sesuai dengan rancangan didalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron. Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sehingga *programmer* dapat membuat sesuai dengan rancangan diagram kelas (Rosa A. S dan M. Shalahuddin, 2014).

Simbol-simbol yang ada pada diagram kelas dapat dilihat pada tabel 2 berikut ini :

Tabel 2. Simbol-Simbol yang digunakan pada *Class Diagram*

No	Simbol	Deskripsi
1	2	3
1	Kelas 	Kelas pada struktur sistem.

2	Antarmuka / <i>interface</i> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
3	Asosiasi / <i>association</i> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
4	Asosiasi berarah / <i>directed association</i> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
5	Generalisasi 	Relasi antar kelas dengan makna generalisasi-spealisasi (umum khusus).
6	Kebergantungan / <i>dependency</i> 	Relasi antarkelas dengan makna kebergantungan antarkelas.
7	Agregasi / <i>aggregation</i> 	Relasi antar kelas dengan makna semua-bagian (<i>whole-part</i>).

Sumber : Buku Rekayasa Perangkat Lunak (Rosa A. S dan M. Shalahuddin, 2018)

4. Analisa Dan Perancangan Analisa Sistem Yang Sedang Berjalan

Analisa sistem merupakan penguraian suatu sistem informasi yang utuh kedalam komponen-komponen dengan maksud untuk mengidentifikasi permasalahan dan proses pemahaman dan penentuan rincian yang harus diselesaikan oleh sistem informasi. Sistem yang sedang berjalan diartikan sebagai sistem yang dipakai atau sistem diterapkan, sedangkan analisa sistem yang sedang berjalan diartikan sebagai cara untuk memahami terlebih dahulu masalah-masalah yang dihadapi oleh sistem.

Analisa Input dan Output Pada Sistem Lama

Berdasarkan analisa sistem yang sedang berjalan pada Apotek Jati Jaya ini didapatkan analisa input dan analisa output yang sedang berjalan.

Analisa Input

Analisa input diperlukan untuk mengetahui apa saja yang menjadi input dalam penyajian laporan pada Apotek Jati Jaya, sehingga mempermudah dalam penelitian.

Analisa Output

Informasi yang dihasilkan adalah Bukti Pembayaran, Bukti Pembayaran ini merupakan alat bantu tagihan atas transaksi pembelian pada Apotek Jati Jaya.

1. Bukti laporan

Bukti pembayaran ini merupakan alat bantu tagihan kepada pelanggan atas transaksi pembelian pada Apotek Jati Jaya yang dapat dilihat pada Gambar 1.

Tgl	Uraian	Debit	Kredit	Saldo
21/11/19	200.000		200.000	200.000
22/11/19	2.100.000		2.100.000	2.300.000
23/11/19	1.500.000		1.500.000	800.000
24/11/19	500.000		500.000	300.000
25/11/19	1.000.000		1.000.000	1.300.000
26/11/19	300.000		300.000	800.000
27/11/19	1.000.000		1.000.000	1.800.000
28/11/19	500.000		500.000	1.300.000
29/11/19	1.000.000		1.000.000	2.300.000
30/11/19	1.000.000		1.000.000	1.300.000
31/11/19	1.000.000		1.000.000	2.300.000

Gambar 1. Bukti dilakukan secara manual
Sumber: Apotek Jati Jaya, 2020

Evaluasi Kerja Sistem Lama

Setelah melakukan tahap analisa terhadap sistem yang sedang berjalan saat ini dapat ditemukan kelemahan-kelemahan pada sistem yang lama, terutama mengenai data manual. Dimana pada sistem yang lama transaksi dilakukan secara manual sehingga banyak terjadi kesalahan-kesalahan pada Apotek Jati Jaya. Maka dengan dirancangnya sistem informasi berbasis *web* dapat membantu dan mempermudah kinerja pada Apotek Jati Jaya.

Perancangan Sistem Dengan Menggunakan UML

Dengan menggunakan model UML membantu dalam memvisualisasikan, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan software berbasis OO (*Object-Oriented*). Perancangan ini menggunakan alat bantu berupa UML (*Unified Modelling Language*) agar mempermudah memindahkan konsep sistem yang dirancang kedalam bentuk program, dimana perancangannya digambarkan dalam bentuk diagram-diaagram berikut:

Use Case Diagram

Use Case Diagram yang ditampilkan akan digunakan untuk menjelaskan fitur-fitur yang dapat digunakan oleh pengguna. Diagram ini juga digunakan untuk verifikasi apakah seluruh fungsi yang dijelaskan didalam *use case* telah diimplementasikan ke dalam website tersebut.

Defenisi Aktor

Defenisi aktor adalah deskripsi aktor yang berhubungan dengan sistem yang dijelaskan pada Tabel 3.

Tabel 3. Defenisi aktor

No.	Aktor	Deskripsi
1	Admin	Aa admin adalah orang yang bertugas dan memiliki hak akses penuh untuk melakukan operasi terhadap keseluruhan sistem.

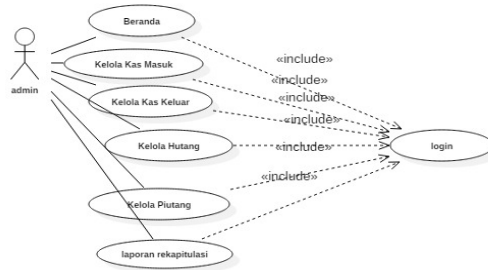
Defenisi *Use Case*

Defenisi dalam *Use case diagram* adalah kegiatan-kegiatan yang akan terjadi di dalam sistem antara para aktor dengan *use case*, dapat dijelaskan pada Tabel 4.

Tabel 4. Defenisi Use Case

No	Usecase	Deskripsi	Aktor
1	Beranda	Admin, dapat melihat beranda dari web.	Admin,
2	Login	Admin dan dapat melakukan login ke dalam sistem.	Admin
3	Produk	Pelanggan dan Pengunjung dapat melihat produk.	Pelanggan dan Pengunjung
4	Tentang Kami	Pengunjung dapat melihat informasi tentang toko.	Pengunjung
5	Pemesanan	Pelanggan dapat melihat pemesanan yang dilakukan.	Admin dan Pelanggan
6	Keranjang Belanja	Pelanggan dapat melihat isi keranjang belanja.	Pelanggan
7	Akun Saya	Pelanggan dapat melihat dan mengubah informasi akun.	Pelanggan
8	Mengelola Produk	Admin dapat mengelola data produk.	Admin
9	Mengelola Pemesanan	Admin dapat melihat dan memproses pemesanan dari pelanggan.	Admin
12	Mengelola Pelanggan	Admin dapat mengelola data pelanggan	Admin
13	Mengelola Laporan	Admin dapat melihat dan mencetak laporan-laporan.	Admin
14	Logout	Admin dan Pelanggan dapat logout dari sistem.	Admin dan Pelanggan

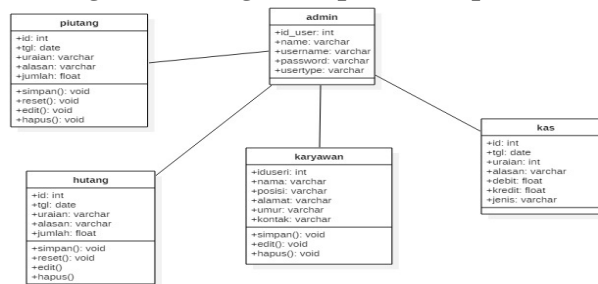
Untuk lebih jelasnya *Use Case Diagram* Laporan Keuangan pada Apotek Jati Jaya dapat digambarkan seperti Gambar 2.



Gambar 2. Use Case Diagram Apotek Jati Jaya

Class Diagram

Class diagram menggambarkan struktur dan deskripsi class, package dan objek beserta hubungan satu sama lain seperti containment, pewarisan, asosiasi, dan lain-lain. Pada website yang dirancang ini class diagram dapat dilihat pada Gambar 3.



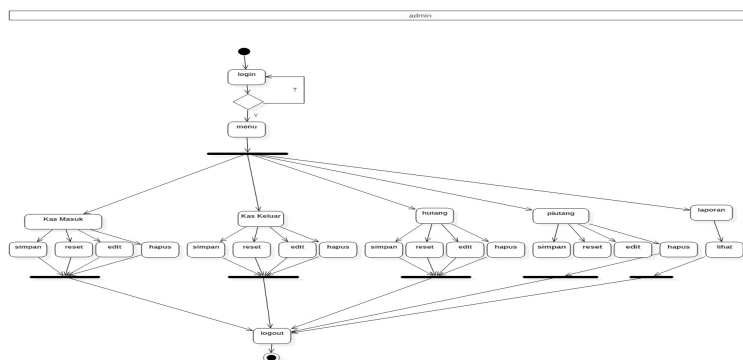
Gambar 3. Class Diagram Apotek Jati Jaya

Activity Diagram

Activity diagram menggambarkan aktivitas sistem bukan apa yang dilakukan oleh aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Sebuah aktivitas dapat direalisasikan oleh satu use case atau lebih. Aktivasi menggambarkan proses yang berjalan, sementara use case menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas.

Activity Diagram Admin

Secara global activity diagram pada admin dapat dilihat pada Gambar 4.



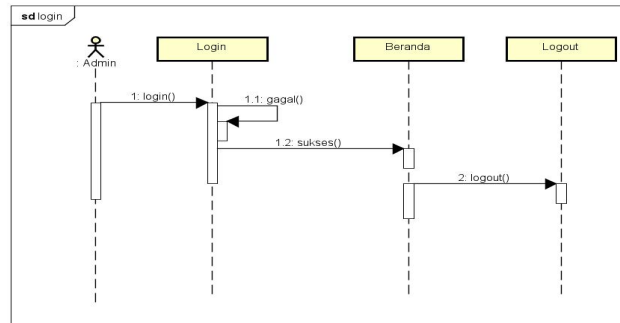
Gambar 4. Activity Diagram Admin

Sequence Diagram

Sequence diagram menggambarkan kelakuan objek pada use case dengan mendiskripsikan waktu hidup objek dan message yang dikirimkan admin diterima antar objek.

Sequence Diagram Login Admin

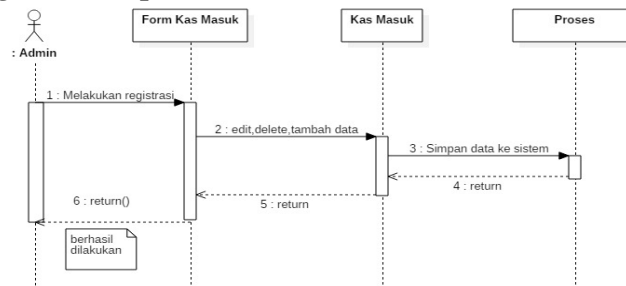
Pada bagian ini menjelaskan bagaimana cara admin masuk ke dalam sistem digambarkan pada Gambar 5.



Gambar 5. Sequence Diagram Login Admin

Sequence Diagram Kas Masuk

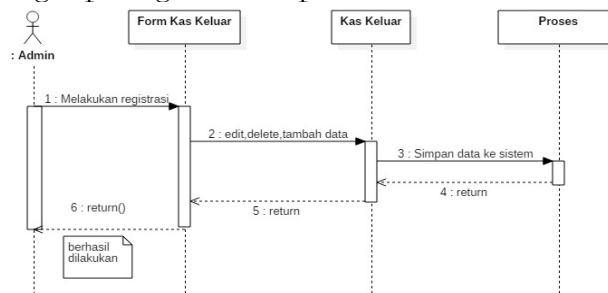
Pada bagian ini menjelaskan bagaimana cara admin mengelola atau memanipulasi kas, yang dapat digambarkan pada Gambar 6.



Gambar 6. Sequence Diagram Kas Masuk

Sequence Diagram Kas Keluar.

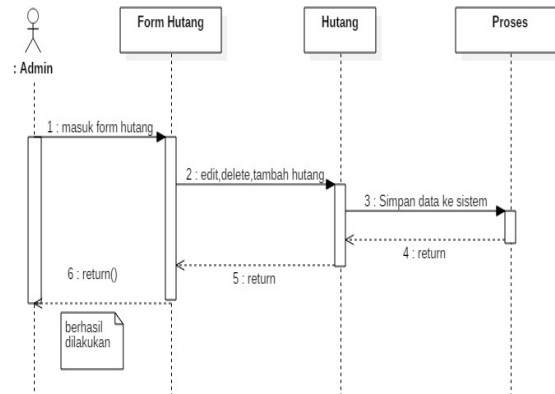
Sequence diagram pada bagian ini menjelaskan bagaimana cara admin mengelola atau memanipulasi kas, yang dapat digambarkan pada Gambar 7.



Gambar 7. Sequence Diagram Kas Keluar

Sequence Diagram Data Hutang

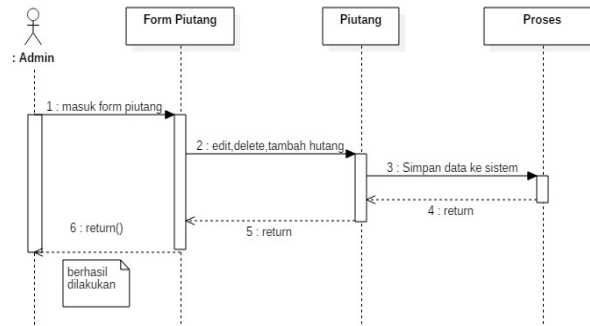
Pada bagian ini menjelaskan hutang admin yang dapat digambarkan pada Gambar 8.



Gambar 8. *Sequence Diagram* Hutang oleh Admin

Sequence Diagram Data Piutang

Pada bagian ini menjelaskan bagaimana cara piutang pada admin yang dapat digambarkan pada Gambar 9.



Gambar 9. *Sequence Diagram* Piutang oleh Admin

Desain Output

Tujuan utama dari desain *output* adalah menghasilkan suatu bentuk keluaran yang efektif, mudah dipahami, cepat dan tepat waktu, dengan kata lain hasil keluaran yang akan dihasilkan haruslah memudahkan bagi setiap unsur yang terlibat atau orang yang menggunakannya.

Keluaran pada umumnya merupakan hasil dari proses yang akan disajikan dalam bentuk *home page*. Adapun desain *home page* dalam perancangan sistem ini antara lain sebagai berikut:

Desain Tampilan Menu Utama

Pada tampilan awal program atau tampilan *interface* ini menampilkan menu yang akan dipilih admin, seperti dashboard, kas masuk, kas keluar, laporan, berikut tampilan desain menu utama dapat dilihat pada gambar 10.

ADMIN	
<input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/>	SLIDER
FOOTER	

Gambar 10. Desain Menu Utama

Desain Input

Desain *input* merupakan suatu alat masukan data, yang mana *input* yang dibutuhkan dalam proses mengakses sistem. Adapun bentuk rancangan *input* tersebut adalah sebagai berikut:

Desain Registrasi Karyawan

Desain *registrasi* karyawan ini merupakan *form* yang akan karyawan ketika melakukan *registrasi* dapat dilihat pada gambar 11.

ADMIN	
<input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/>	Tambah karyawan Nama Lengkap : Posisi : Alamat : Umur : Kontak :
FOOTER	

Gambar 11. Desain Registrasi

Desain *Login Admin*

Desain *login user* merupakan *form* yang akan digunakan oleh *admin* untuk masuk ke dalam sistem yang dapat dilihat pada gambar 12.

HOME	DAFTAR	ABOUT	LOGIN
Username : <input type="text"/>			
Password : <input type="password"/>			
FOOTER			

Gambar 12. Desain *Login Admin*

Desain Kas Masuk

Desain *input* ini merupakan form yang digunakan admin untuk menginputkan data pendapatan yang dapat dilihat pada gambar 13.

ADMIN	
Kas Masuk	
<input type="text"/>	<input type="button" value="Tambah"/>
<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>
FOOTER	

Gambar 13. Desain Input Produk

Desain File

Setelah melakukan perancangan *output* dan *input*, selanjutnya adalah merancang *file* yang digunakan untuk penyimpanan data dalam *database*. Bentuk dari desain *file* yang dirancang dalam sistem ini adalah sebagai berikut :

File Tabel Admin

Tabel user digunakan untuk menyimpan data-data admin yang ada pada toko apotek jati jaya. Desain tabel user dapat dilihat pada tabel 5.

Tabel 5. Desain File Tabel Admin

Nama Database: db_keuangan

Nama Tabel : user

Field Key : id

Field Name	Type	Width	Description
<i>Id_user</i>	<i>int</i>	<i>11</i>	<i>Id_user</i>
<i>Nama</i>	<i>varchar</i>	<i>25</i>	<i>nama</i>
<i>Username</i>	<i>varchar</i>	<i>25</i>	<i>Username</i>
<i>Password</i>	<i>varchar</i>	<i>25</i>	<i>password</i>
<i>Usertype</i>	<i>Varchar</i>	<i>50</i>	<i>usertype</i>
<i>Total</i>		<i>136 Bytes</i>	

File Tabel Kas

Tabel kas ini digunakan untuk menampung data-data yang berhubungan dengan transaksi yang terjadi pada sisitem ini, dengan bentuk rancangan seperti pada tabel 6.

Tabel 6. Desain File Tabel Kas

Nama Database: db_keuangan

Nama Tabel : kas

Field Key : id

Field Name	Type	Width	Description
<i>Id</i>	<i>Int</i>	<i>11</i>	<i>Id</i>
<i>Tgl</i>	<i>Date</i>	<i>-</i>	<i>Tgl</i>
<i>Uraian</i>	<i>Varchar</i>	<i>11</i>	<i>Uraian</i>
<i>Debit</i>	<i>Float</i>	<i>-</i>	<i>Debit</i>
<i>Kredit</i>	<i>Float</i>		<i>Kredit</i>
<i>Jenis</i>	<i>Varchar</i>	<i>20</i>	<i>Jenis</i>
<i>Total</i>		<i>42 Bytes</i>	

File Tabel Karyawan

Tabel *karyawan* digunakan untuk menampung data-data yang berhubungan dengan karyawan pada sistem ini, dengan bentuk rancangan seperti tabel 7.

Tabel 7. Desain File Tabel Karyawan

Nama Database: db_keuangan

Nama Tabel : karyawan

Field Key : id_user

Field Name	Type	Width	Description
<i>Id_User</i>	<i>Int</i>	<i>11</i>	<i>Id_User</i>
<i>Nama</i>	<i>Varchar</i>	<i>50</i>	<i>Nama</i>
<i>Posisi</i>	<i>Varchar</i>	<i>50</i>	<i>Posisi</i>

Alamat	Varchar	100	Alamat
Umur	Varchar	50	Umur
Kontak	Varchar	50	Kontak
Total		311 Bytes	

File Tabel Piutang

Tabel piutang digunakan untuk menampung data-data yang berhubungan dengan piutang pada sistem ini, dengan bentuk rancangan seperti tabel 8.

Tabel 8. Desain File Tabel Piutang

Nama Database: db_keuangan

Nama Tabel : piutang

Field Key : id

Field Name	Type	Width	Description
Id_piutang	Int	11	Id
Tgl	Date	-	Tgl
Uraian	Varchar	50	Uraian
Alasan	Varchar	50	Alasan
Jumlah	Float	-	Jumlah
Total		111 Bytes	

File Tabel Hutang

Tabel lastonline digunakan untuk menampung data-data yang berhubungan dengan data hutang dengan rancangan seperti tabel 9.

Tabel 9. Desain File Tabel Hutang

Nama Database: db_keuangan

Nama Tabel : hutang

Field Key : id

Field Name	Type	Width	Description
Id	Int	11	Id
Tgl	Date	-	Tgl
Uraian	Varchar	50	Uraian
Alasan	Varchar	50	Alasan
Jumlah	Float	-	Jumlah
Total		111 Bytes	

5. Implementasi Dan Pengujian Sistem

Implementasi Sistem

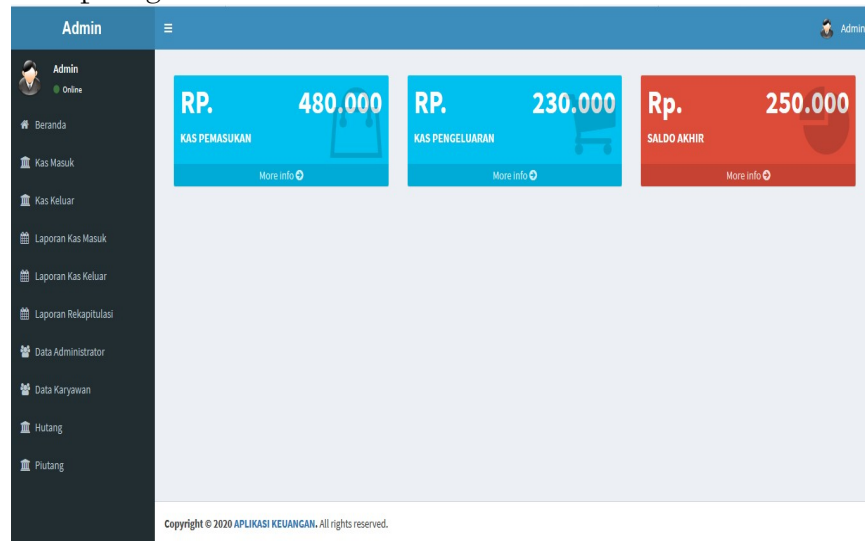
Tahap implementasi merupakan salah satu tahap dalam pengembangan sistem, dimana tahap ini meletakkan sistem informasi supaya siap untuk dipakai. Dalam melanjutkan kegiatan implementasi perlu dilakukan beberapa hal yaitu menerapkan rencana implementasi yang merupakan kegiatan awal dari tahap implementasi sistem.

Pengujian Sistem

Pada tahap pengujian sistem ini dilakukan bertujuan untuk mengetahui sejauh mana sistem yang dirancang dapat mengatasi masalah yang terdapat pada sistem informasi keuangan.

Tampilan Halaman Utama

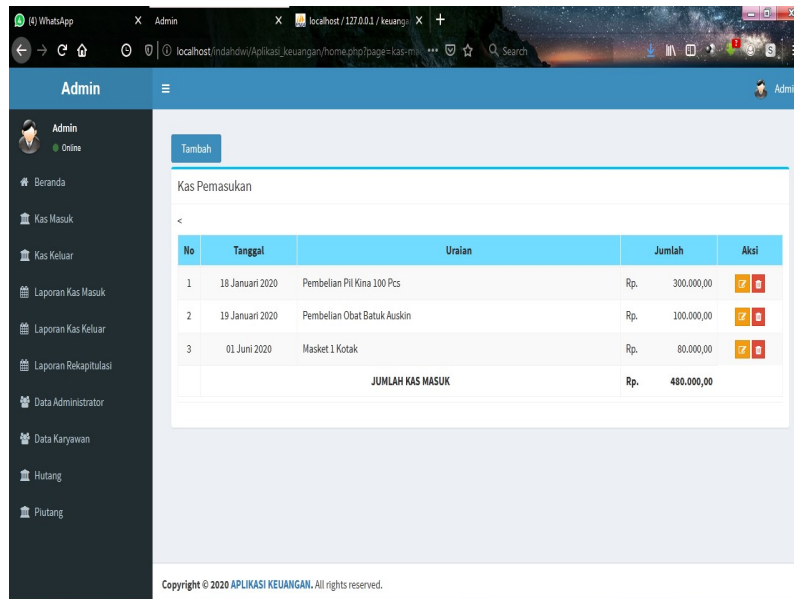
Halaman ini merupakan tampilan halaman utama pada sistem informasi keuangan dapat dilihat seperti gambar 14 berikut ini :



Gambar 14. Tampilan Halaman Utama

Tampilan Kas Masuk

Halaman ini merupakan tampilan kas masuk dapat dilihat seperti gambar 15 berikut ini :



Gambar 15. Tampilan Kas Masuk

Tampilan Kas Keluar

Halaman ini merupakan kas keluar dapat dilihat seperti pada gambar 16 berikut ini :

No	Tanggal	Uraian	Jumlah	Aksi
1	19 Februari 2020	Beli Perlengkapan Covid	Rp. 100.000,00	[Edit] [Hapus]
2	19 Maret 2020	pembayaran listrik air dan dll	Rp. 80.000,00	[Edit] [Hapus]
3	04 Juni 2020	beli bensin	Rp. 50.000,00	[Edit] [Hapus]
JUMLAH KAS KELUAR			Rp. 230.000,00	

Gambar 16. Tampilan Kas Keluar

Tampilan Laporan Kas Masuk

Halaman ini merupakan tampilan laporan kas masuk dapat dilihat seperti gambar 17 berikut ini :

No	Tanggal	Uraian	Jumlah
1	18 Januari 2020	Pembelian P3 Kasa 100 Pcs	Rp. 300.000,00
2	19 Januari 2020	Pembelian Obat Bismil Anakin	Rp. 100.000,00
3	01 Juni 2020	Membeli 1 Kotak	Rp. 80.000,00
JUMLAH			Rp. 480.000,00

PADANG, 04 Juli 2020
KARYAWAN
INDAH DWI PUTRI

Gambar 17. Tampilan Halaman About

Tampilan Laporan Kas Keluar

Pada halaman ini tampilan laporan kas keluar untuk lebih jelasnya dapat dilihat pada gambar 18 berikut ini :

No	Tanggal	Uraian	Jumlah
1	19 Februari 2020	Beli Perlengkapan Covid	Rp. 100.000,00
2	19 Maret 2020	pembayaran listrik air dan dll	Rp. 80.000,00
3	04 Juni 2020	beli bensin	Rp. 50.000,00
JUMLAH			Rp. 230.000,00

PADANG, 04 Juli 2020
KARYAWAN
INDAH DWI PUTRI

Gambar 4.12 Tampilan Laporan Kas Keluar

Tampilan Laporan Rekapitulasi

Pada halaman ini menampilkan laporan rekapitulasi seperti pada gambar 19 berikut ini :

No	Tanggal	Uraian	Debit	Kredit	Saldo
1	18 Januari 2020	Pembelian Pili Kina 100 Pcs	Rp. 300.000,00		Rp. 300.000,00
2	19 Januari 2020	Pembelian Obat Batuk Auskin	Rp. 100.000,00		Rp. 400.000,00
3	19 Februari 2020	Beli Perlengkapan Covid		Rp. 100.000,00	Rp. 300.000,00
4	19 Maret 2020	pembayaran listrik air dan dll		Rp. 80.000,00	Rp. 220.000,00
5	01 Juni 2020	Masket 1 Kotak	Rp. 80.000,00		Rp. 300.000,00
6	04 Juni 2020	beli bensin		Rp. 50.000,00	Rp. 250.000,00
JUMLAH			Rp. 480.000,00	Rp. 230.000,00	Rp. 250.000,00

Gambar 19. Tampilan Laporan Rekapitulasi

Tampilan Halaman Data Karyawan Pada Apotek Jati Jaya

Pada halaman ini menambah dan menampilkan daftar-daftar anggota yang sudah diinputkan sebelumnya, seperti pada gambar 20 berikut ini :

No	Nama	Posisi	Alamat	Umur	Kontak	Aksi
1	Yuhanna Nasution, S.Farm	Karyawan	Jl.Dr.H. Abdullah Ahmad No. 23 A, Padang	29	085264847852	[Edit] [Hapus]
2	Syifa Kharunnisa, S.Farm	Karyawan	Jl.Dr.H. Abdullah Ahmad No. 23 A, Padang	28	081363648795	[Edit] [Hapus]

Gambar 20. Tampilan Halaman Daftar Anggota Pada Admin

Tampilan Halaman Hutang

Pada halaman ini menampilkan halaman pada sistem seperti pada gambar 21 berikut ini :

No	Tanggal	Penghutang	Alasan	Jumlah	Aksi
1	12 Februari 2020	Jodi Saputra	Beli Bensin Motor	Rp. 30.000,00	[Edit] [Hapus]
2	19 Maret 2020	Riri Ramadani	Buat Sekolah Anak	Rp. 200.000,00	[Edit] [Hapus]
3	06 April 2020	Welsi	Pembelian Pampers	Rp. 125.000,00	[Edit] [Hapus]
4	06 Januari 2020	Angga Dinawan	Pembayaran uang makan siang	Rp. 175.000,00	[Edit] [Hapus]
JUMLAH HUTANG				Rp. 530.000,00	

Gambar 21. Tampilan Halaman Hutang

Tampilan Halaman Piutang

Pada halaman ini menampilkan halaman pada sistem seperti pada gambar 22 berikut ini :

No	Tanggal	Nama	Alasan	Jumlah	Aksi
1	18 November 2018	Donatur a/n Jodi Saputra		Rp. 500.000,00	[Edit] [Hapus]
2	19 November 2018	Donatur a/n Angga Suparno		Rp. 500.000,00	[Edit] [Hapus]
3	01 Juni 2020	Velsa Novia	Uang muka karyawan	Rp. 400.000,00	[Edit] [Hapus]
JUMLAH PIUTANG			Rp.	1.400.000,00	

Gambar 22. Tampilan Halaman Hutang

4. Kesimpulan Dan Saran

Kesimpulan

Berdasarkan uraian pada bab-bab sebelumnya, maka dapat ditarik kesimpulan sebagai berikut :

1. Dengan adanya komputer dapat mempermudah karyawan dalam mengelola sistem informasi laporan keuangan.
2. Dengan adanya sistem laporan keuangan ini data akan disimpan didalam sebuah database, sehingga data lebih terstruktur dan tersusun rapi.
3. Dengan adanya sistem laporan keuangan dapat mempermudah karyawan toko dalam mengetahui laba atau rugi.

Saran

Berdasarkan kesimpulan diatas, maka penulis mencoba memberikan saran sebagai berikut :

1. Agar aplikasi yang dirancang dapat bekerja secara efektif dan efisien maka diperlukan tenaga terampil dalam pengoperasian yang dibuat.
2. Dalam penerapan sistem komputersasi sebaiknya didukung oleh perangkat atau alat yang memadai, baik dari segi manusia (SDM) maupun segi peralatannya (*software* dan *Hardware*).

5. Daftar Pustaka

- Arief, M.Rudyanto. Pemrograman web Dinamis menggunakan PHP dan MySQL. ANDI:Yogyakarta.
- Kustiyahningsih, Yeni.(2011).Pemrograman Basis Data berbasis web menggunakan PHP dan MySQL. GrahaIlmu:Yogyakarta.
- Notoadmojo, S.(2002) penelitian kesehatan. Jakarta :RienikaCipta.
- Seram, Novan Putra.(2012-2016) Sistem Informasi UNIPDU Jombang.
- Sumanto. 1995. Metodologi Pnenlitian Sosial dan Pendidikan: Aplikasi Metode Kuantitatif dan Stattistika dalam penelitian. Yogyakarta:Andi Offset.
- Sunarfrihantono, Bimo. (2003). PHP dan MYSQL untuk Web. ANDIOFFSET:Yogyakarta.

Sutabri, Tata. Teknologi Informasi. ANDI:Yogyakarta.

Sutarman.(2007). Membangun Aplikasi Web dngan PHP dan MySQL.GrahaIlmu:Yogyakarta.

Wijayanti, Eva Kurnia.(2014). Rancangan Bangun Website penyewaan penjualan kamera.Universitas Muhammadiyah Ponorogo.

Yin, R.Y. 2004.Studi Kasus: Desain dan Metode. Jakarta:Raja GrafindoPersada.