

## Penerapan Metode SDLC Waterfall Dalam Perancangan Sistem Informasi Administrasi Keuangan Berbasis Desktop

Weiskhy Steven Dharmawan<sup>[1]</sup>, Deasy Purwaningias<sup>[2]</sup>, Deni Risdiansyah<sup>[3]</sup>

Program Studi Sistem Informasi, UBSI PSDKU Pontianak<sup>[1, 2, 3]</sup>

Jl. Abdurrahman Saleh No. 18A, Pontianak, Kalimantan Barat

Email : deasy.dwg@bsi.ac.id<sup>[1]</sup>, weisky.wvn@bsi.ac.id<sup>[2]</sup>, deni.dr@bsi.ac.id<sup>[3]</sup>

### ABSTRAKS

Berdasarkan penelitian yang telah dilakukan pada Kantor Kecamatan Pengkadan diperoleh bahwa suatu sistem yang diterapkan belum menggunakan teknologi komputerisasi dalam proses kegiatannya sehari. Berdasarkan penelitian tersebut penulis mencoba membuat sebuah rancangan aplikasi guna menunjang aktivitas kegiatan perusahaan dengan menggunakan bahasa pemrograman VB.NET. Untuk metode pengembangan software dilakukan dengan menggunakan metode SDLC Waterfall. Dengan penggunaan aplikasi ini diharapkan dapat mempermudah dan mengolah data keuangan, serta mempermudah pembuatan laporan kas masuk dan kas keluar untuk menghasilkan laporan yang jauh lebih baik. Hasil pengamatan dari penelitian yang telah dilakukan pada Kantor Kecamatan Pengkadan, diketahui bahwa sistem pencatatan Media kertas merupakan alat yang digunakan untuk mencatat setiap data pada kantor tersebut. Akibatnya laporan yang dihasilkan sering terjadi kesalahan karena hilang dan rusaknya bukti transaksi. Berdasarkan permasalahan yang terjadi dibutuhkan sebuah rancangan system yang terkomputer agar dapat mengurangi humans error pada proses pembuatan laporan maupun proses pencatatan transaksi. Peneliti mengembangkan sebuah rancangan perangkat lunak yang dapat mengatasi permasalahan yang terjadi pada Kantor Kecamatan Pengkadan. Rancangan yang dibuat menggunakan bahasa pemrograman Visual Studio 2010 dan menerapkan metode SDLC Waterfall sebagai metod pengembangan perangkat lunak. Perangkat lunak ini diharapkan dapat mempermudah proses pencatatan pengolahan data keuangan, mengurangi kemungkinan terjadinya humans error, menghasilkan laporan yang lebih baik dengan waktu yang singkat, dan mempermudah management dalam mengambil keputusan.

**Kata Kunci: Sistem Informasi, Visual Studio 2010, Metode SDLC Waterfall.**

### ABSTRACT

*Based on the research conducted at the Pengkadan District Office, it was found that the implemented system had not used computerized technology in the daily process. Based on this research the author tries to make an application design to support the activity activities of the company using the VB.NET programming language. The software development method is done using the Waterfall SDLC method. By using this application, it is expected to simplify and process financial data, and make it easier to make cash and cash reports to produce far better reports. The results of observations from the research that has been done at the Sub-District Office of Pengkadan, it is known that the paper recording system is a tool used to record every data in the office. As a result, reports generated often occur because of evidence of lost and damaged transactions. Based on the problems that occur, a computerized system design is needed in order to reduce human error in the process of making reports and the process of recording transactions. Researchers develop software designs that can overcome problems that occur in the Pengkadan District Office. The design was made using the Visual Studio 2010 programming language and applied the Waterfall SDLC method as a method of software development. This software is expected to simplify the process of recording financial data processing, reduce the possibility of human error, produce better reports with a short time, and facilitate management in making decisions.*

**Keyword: Information System, Visual Studio 2010, SDLC Waterfall Method.**

### 1. PENDAHULUAN

Perkembangan teknologi yang sangat pesat seperti sekarang ini, baik di instansi pemerintah

maupun perusahaan dituntut untuk dapat mengikuti serta mengetahui perkembangan informasi dalam segala bidang khususnya dalam

bidang komputer. Teknologi informasi berbasis komputer mampu meningkatkan efisiensi dan efektivitas kinerja. Kasemin (2015) teknologi ini memungkinkan kita untuk membuat pengolahan suatu dokumen secara digital. pengolahan dan penyimpanan data secara digital dapat meningkatkan efektifitas kinerja dan efisiensi waktu. perusahaan maupun instansi pemerintahan pun semakin mengandalkan teknologi sistem informasi akuntansi diantaranya untuk administrasi laporan keuangan. sehingga dapat mempermudah bagian keuangan dalam membuat dokumen dan laporan keuangannya yang diperlukan dalam waktu singkat, serta meminimalisir kesalahan dalam menginput data keuangan. Kantor Kecamatan Pengkadan adalah sebuah instansi pemerintah yang berada di Kabupaten Kapuas Hulu.

Kantor Kecamatan Pengkadan saat ini dalam proses pengolahan data administrasi keuangan belum menggunakan sistem yang terkomputerisasi, bagian keuangan untuk mencatat dan pembuatan laporan keuangan masih menggunakan Microsoft Excel. Oleh karena itu bagian keuangan memerlukan adanya sistem informasi administrasi pengolahan data keuangan yang terkomputerisasi guna untuk mendukung dan memudahkan dalam pengolahan data keuangan dan penyimpanannya.

Setelah melakukan pengamatan dan melihat ruang lingkup pada Kantor Kecamatan Pengkadan. Penulis memberikan saran bahwa sebaiknya Kantor Kecamatan Pengkadan menggunakan sebuah sistem yang telah terkomputerisasi untuk pengolahan data administrasi keuangan. karena selain mempermudah pencarian dokumen sistem yang sudah terkomputerisasi juga dapat menghindari faktor seperti kehilangan dokumen dan pemakaian tempat yang berlebihan untuk menyimpan dokumen dalam bentuk *hard copy*.

## 2. TINJAUAN PUSTAKA

Tinjauan pustaka merupakan kumpulan referensi-referensi yang dipelajari dalam pembuatan penelitian ini.

### 2.1. Metode Pengembangan Software

Pengembangan perangkat lunak menggunakan Metode *Waterfall*, yaitu sebuah model pengembangan perangkat lunak yang menyediakan

pendekatan alur hidup perangkat lunak secara *sekuensial* atau terurut dimulai dari analisis, pengkodean, pengujian dan tahap pendukung/support. Arizona (2017). Adapun tahapan metode *Waterfall* diuraikan sebagai berikut :

1. *Analisa Kebutuhan Software*  
 Dalam analisa kebutuhan *software* ini bertujuan menganalisa semua kebutuhan termasuk dokumen dan *interface* yang diperlukan guna menentukan solusi *software* yang akan digunakan sebagai proses komputerisasi sistem.
2. *Desain*  
 Dalam tahap ini, desain sesuai kebutuhan sistem akan dibuat terkait rancangan *database*, *software architecture* dan *User Interface* yang akan dibuat. Penggunaan *Unified Modeling Language* (UML) dimaksud menjelaskan lebih terperinci dalam rancangan pembuatan program dan rancangan *database*. UML yang akan digunakan adalah *Activity Diagram*.
3. *Code Generation*  
 Pada tahap ini, implementasi desain dibuat kedalam program perangkat lunak. Pada tahap ini dibuat sistem baru dengan menggunakan *Visual Studio 2010* sebagai bahasa pemrograman, *MSQL* untuk membuat *database*. *ODBC Connector* sebagai koneksi *database*, *XAMMP* sebagai penghubung server *database* dan *Cristal Report 8.5* sebagai *tools* pembuatan laporan.
4. *Testing*  
 Pengujian Program menggunakan *Black box* dengan menghasilkan sesuai dengan harapan yang telah dirancang sebelumnya. Penggunaan *Black box* dalam pengujian aplikasi akan memberikan penjelasan tentang kesesuaian harapan dalam pembuatan program.
5. *Support*  
 Proses atau pemeliharaan ini penulis mengupayakan pengembangan sistem yang sudah dibuat terkait *software* dan *hardware* yang akan digunakan.

### 2.2. Konsep Dasar Sistem

(Romney, 2015) mengatakan sistem adalah kumpulan dari dua atau lebih komponen yang saling bekerja dan berhubungan untuk mencapai tujuan tertentu. Romney juga berpendapat bahwa

perusahaan adalah sebuah sistem yang terdiri dari beberapa departemen yang bertindak sebagai subsistem yang membentuk sistem perusahaan tersebut.

Penulis menyimpulkan sistem bisa diartikan sebagai sekumpulan subsistem, komponen ataupun elemen yang saling bekerjasama dengan tujuan yang sama untuk menghasilkan *output* yang sudah ditentukan sebelumnya.

### 2.2.1 Karakteristik Sistem

Berdasarkan Hutahean (2014) pengertian Supaya sistem itu dikatakan sistem yang baik maka sistem tersebut harus memiliki karakteristik atau sifat-sifat antara lain sebagai berikut:

- a. Komponen  
Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen sistem terdiri dari komponen yang berupa subsistem atau bagian-bagian dari sistem.
- b. Batasan sistem (*boundary*)  
Batasan sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lain atau dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan. Batasan suatu sistem menunjukkan ruang lingkup (*scope*) dari sistem tersebut.
- c. Lingkungan luar sistem (*environment*)  
Lingkungan luar sistem (*environment*) adalah diluar batas dari mempengaruhi operasi sistem. Lingkungan dapat bersifat menguntungkan yang harus tetap di jaga dan dikendalikan, karena kalau tidak akan mengganggu kelangsungan hidup dari sistem.
- d. Penghubung sistem (*interface*)  
Penghubung sistem merupakan media penghubung antara satu subsistem lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari subsistem ke subsistem lain. Keluaran (*output*) dari subsistem akan menjadi masukan (*input*) untuk subsistem lain melalui penghubung.
- e. Masukan sistem (*input*)  
Masukan adalah energi yang dimasukkan kedalam sistem, yang dapat berupa perawatan (*maintenance input*), dan masukan sinyal (*signal input*). *maintenance input* adalah energi yang dimasukkan agar sistem dapat beroperasi. *signal input* adalah energi yang diproses untuk didapatkan keluaran. Contoh dalam sistem komputer program adalah *maintenance input* sedangkan data adalah *signal input* untuk diolah menjadi informasi.
- f. Keluaran sistem (*output*)  
Keluaran sistem adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Contoh komputer menghasilkan panas yang merupakan sisa pembuangan, sedangkan informasi adalah keluaran yang dibutuhkan.
- g. Pengolah sistem  
Suatu sistem menjadi bagian pengolah yang akan merubah masukan menjadi keluaran. Sistem produksi akan mengolah bahan baku menjadi barang jadi, sistem akuntansi akan mengolah data menjadi laporan-laporan keuangan.
- h. Sasaran sistem  
Suatu sistem pasti mempunyai tujuan (*goal*) atau sasaran (*objective*). Sasaran dari sistem sangat menentukan *input* yang dibutuhkan oleh sistem dan keluaran yang akan dihasilkan sistem.

### 2.2.2 Klasifikasi Sistem

Berdasarkan Hutahean (2014) pengertian sistem dapat diklasifikasikan dalam beberapa sudut pandang:

1. Klasifikasi sistem sebagai:
  - a. Sistem abstrak (*abstract System*)  
Sistem abstrak adalah sistem yang berupa pemikiran-pemikiran atau ide-ide yang tidak tampak secara fisik.
  - b. Sistem fisik (*physical system*)  
Sistem fisik adalah sistem yang ada secara fisik.
2. Sistem diklasifikasikan sebagai:
  - a. Sistem alamiah (*natural system*)  
Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia. Misalnya sistem perputaran bumi.
  - b. Sistem buatan manusia (*human made system*)  
Sistem buatan manusia adalah sistem yang dibuat oleh manusia yang melibatkan

interaksi antara manusia dengan mesin (*human machine system*).

3. Sistem diklasifikasikan sebagai:
  - a. Sistem Tertentu (*deterministic system*)  
Sistem tertentu adalah sistem yang beroperasi dengan tingkah laku yang sudah dapat diprediksi, sebagai keluaran sistem yang dapat diramalkan.
  - b. Sistem tak tentu (*probabilistic system*)  
Sistem tak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur *probabilistic*.
4. Sistem diklasifikasikan sebagai:
  - a. Sistem tertutup (*close system*)  
Sistem tertutup adalah sistem yang tidak terpengaruhi dan tidak berhubungan dengan lingkungan luar, sistem bekerja otomatis tanpa ada turut campur lingkungan luar. Secara teoritis sistem tertutup ini ada, kenyataannya tidak ada sistem yang benar-benar tertutup, yang ada hanya *relatively closed system*.
  - b. Sistem terbuka (*open system*)  
Sistem terbuka adalah sistem yang berhubungan dan terpengaruhi dengan lingkungan luarnya. Sistem ini menerima *input* dan *ouput* dari lingkungan luar atau subsistem lainnya. Karena sistem terbuka terpengaruh lingkungan luar maka harus mempunyai pengendalian yang baik.

### 2.3. Pengertian Sistem Informasi

Sistem informasi merupakan gabungan dari empat bagian utama. Keempat bagian utama tersebut mencakup perangkat lunak (*software*), perangkat keras (*hardware*), infrastruktur, dan Sumber Daya Manusia (SDM) yang terlatih. Keempat bagian utama ini saling berkaitan untuk menciptakan sebuah sistem yang dapat mengelola data menjadi informasi yang bermanfaat Risdiansyah (2017).

### 2.4. Peralatan Pendukung (*Tools System*)

Suatu alat yang digunakan untuk menggambarkan bentuk logika model dari suatu sistem dengan menggunakan symbol-simbol, lambang-lambang, berupa diagram-diagram yang menunjukkan secara tepat arti dan fungsinya.

### 2.4.1. Unified Modeling Language (UML)

Salah satu alat untuk pemodelan visual suatu perangkat lunak adalah UML. Pemodelan dimaksudkan agar semua orang yang terlibat dalam pengembangan proyek perangkat lunak dapat saling memahami persoalan Sasongko, (2014).

Menurut Rosa *Unified Modeling Language* (UML) adalah salah satu standar bahasa yang banyak digunakan didunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman. Irmayani dan Susyatih (2017).

1. Diagram *Use-Case Diagram* ini memperlihatkan himpunan *use-case* dan actor-aktor (suatu jenis khusus dari kelas). Diagram ini sangat penting untuk mengorganisasi dan memodelkan perilaku suatu sistem yang dibutuhkan serta diharapkan oleh pengguna.
2. *Activity Diagram* Diagram aktivitas ini adalah tipe khusus dari diagram status yang memperlihatkan aliran dari suatu sistem aktivitas ke aktivitas lainnya dalam suatu sistem. Diagram ini penting terutama dalam pemodelan fungsi-fungsi suatu sistem dan memberikan tekanan pada aliran kendali antar objek.
3. *Sequence Diagram* menggambarkan bagaimana sistem merespon kegiatan *user*. *Sequence Diagram* yang dibuat yaitu yang berhubungan langsung dengan kegiatan utama dari sistem.
4. *Deployment Diagram* menggambarkan hubungan antara *software* dan *hardware* terhadap sistem dan apa saja *output* yang dihasilkan.

### 2.4.2 Visual Basic. NET

Ricyanto mendefinisikan bahwa Visual Basic.Net adalah generasi penerus Visual Basic 6 dari Microsoft. Dengan Visual Basic.Net, dapat membangun aplikasi *Windows*, *Web service* dan aplikasi web dengan ASP.NET secara cepat dan mudah. Aplikasi yang dibuat dengan Visual Basic.Net dibangun diatas *service common language runtime* sehingga memiliki keunggulan- keunggulan dari .Net Framework", Dharmawan (2017).

## 3. METODOLOGI

Metodologi merupakan cara atau teknis yang diterapkan pada penelitian ini.

### 3.1 Pengumpulan Data Premier

Pengumpulan data dilakukan dengan melakukan wawancara langsung kepada pimpinan dan beberapa pegawai Kantor Kecamatan Pengkadan.

### 3.2 Pengumpulan Data Sekunder

Mengumpulkan data dengan cara membaca dan mempelajari buku, makalah, jurnal, artikel dan bahan-bahan dari internet yang sesuai dengan topik yang sedang dibahas.

### 3.3 Pengembangan Perangkat Lunak

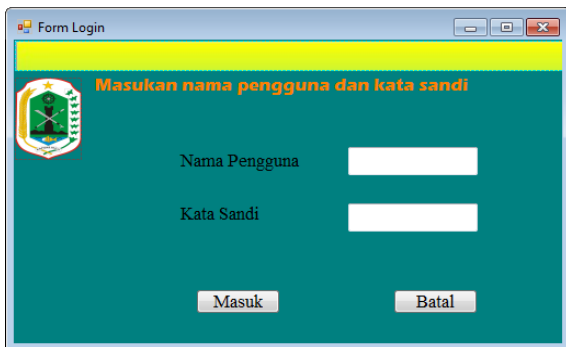
Metode pengembangan perangkat lunak yang digunakan adalah metode Waterfall. Sebuah pendekatan kepada perkembangan perangkat lunak yang sistematis dan sekuensial mulai dari tingkat dan kemajuan sistem pada analisis, desain, kode, pengujian, dan pemeliharaan. Terdiri dari beberapa tahapan, yaitu: tahap analisis, perancangan, pengkodean, dan pengujian

## 4. HASIL DAN PEMBAHASAN

Berikut akan dijelaskan implementasi dari program yang telah dibuat.

### 4.1. Tampilan Halaman Pembuka

Berikut ini merupakan tampilan *User Interface* hasil Perancangan Sistem Informasi Administrasi Keuangan Berbasis VB. NET :



Sumber: Hasil Penelitian (2018)

**Gambar 1. Implementasi Tampilan Halaman Pembuka (Login)**

Sebelum masuk ke dalam sebuah aplikasi maka pengguna memasukkan nama pengguna dan kata sandi, jika kata sandi yang dimasukkan salah maka

akan menampilkan pesan "Login gagal periksa kembali nama pengguna dan kata sandi anda" jika nama pengguna dan kata sandi dimasukkan dengan benar pilih tombol masuk maka akan menampilkan menu utama. Sedangkan tombol batal digunakan untuk membatalkan masuk ke menu utama dan membersihkan *textbox*.

### 4.2. Tampilan Halaman Utama

Tampilan yang terdapat dalam menu utama adalah, menu data, menu transaksi, menu laporan keuangan, menu keluar. Didalam menu data terdapat submenu berupa data pengguna, data pegawai, data akun. Didalam menu transaksi terdapat submenu transaksi pendapatan dan transaksi pengeluaran. didalam submenu laporan keuangan terdapat submenu kas masuk, kas keluar, jurnal umum, buku besar, neraca saldo dan laba rugi.

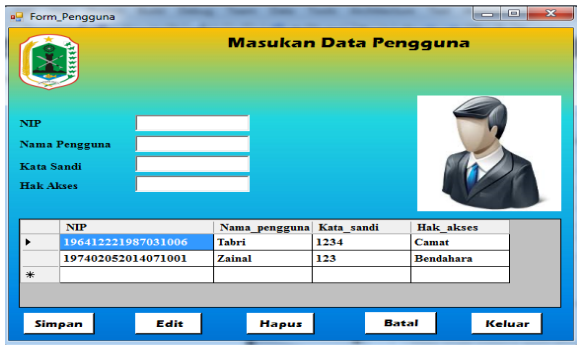


Sumber: Hasil Penelitian (2018)

**Gambar 2. Implementasi Tampilan Halaman Utama**

### 4.3. Tampilan Form Pengguna

*form* data pengguna ini dipanggil dari menu utama pilih menu data kemudian pilih submenu data pengguna, kemudian masukan data pengguna yaitu nip, nama pengguna, kata sandi, hak akses. Didalam *form* pengguna terdapat beberapa tombol yang digunakan diantaranya tombol simpan, edit, hapus, batal, dan tombol keluar.

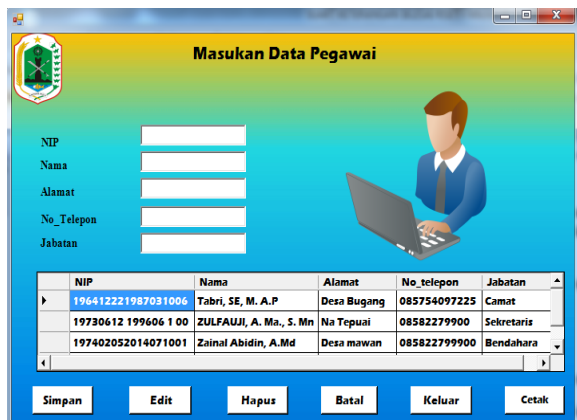


Sumber: Hasil Penelitian (2018)

**Gambar 3. Implementasi Tampilan Halaman Pengguna**

#### 4.4. Tampilan Form Pegawai

form data pegawai ini dipanggil dari menu utama pilih menu data kemudian pilih submenu data pegawai, kemudian masukan data pegawai yaitu Nip, nama, alamat, no telepon, jabatan. Didalam form data pegawai terdapat beberapa tombol yang digunakan diantaranya tombol simpan, edit, hapus, batal, dan tombol keluar. Tombol simpan digunakan untuk menyimpan data yang baru dimasukan. Tombol edit digunakan untuk mengedit data yang telah dimasukan. Tombol hapus untuk menghapus data yang telah dimasukan. Tombol batal digunakan untuk membatalkan data yang diisi pada *textbox* dan mengosongkan semua *textbox*. Sedangkan tombol keluar digunakan untuk kembali ke form menu utama.



Sumber: Hasil Penelitian (2018)

**Gambar 4. Implementasi Tampilan Penggawai**

#### 4.5. Tampilan Form Akun

Form data akun ini dipanggil dari menu utama, pilih menu data kemudian pilih submenu data

akun, kemudian masukan data akun yaitu kode akun, nama akun, jenis akun.

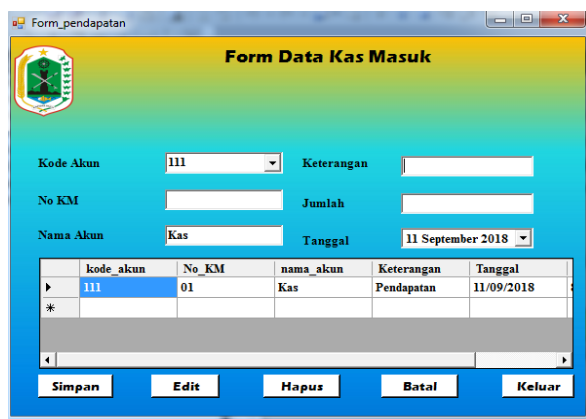


Sumber: Hasil Penelitian (2018)

**Gambar 5. Implementasi Tampilan Halaman Akun**

#### 4.6. Tampilan Kas Masuk

form data pendapatan ini dipanggil dari menu utama, pilih menu transaksi kemudian pilih submenu transaksi pendapatan, kemudian masukan data kas masuk yaitu kode akun, nama akun, tanggal, keterangan, jumlah. Didalam form data kas masuk terdapat beberapa tombol yang digunakan diantaranya tombol simpan, edit, hapus, batal, dan tombol keluar. Tombol simpan digunakan untuk menyimpan data yang baru dimasukan. Tombol edit digunakan untuk mengedit data yang telah dimasukan. Tombol hapus untuk menghapus data yang telah dimasukan. Tombol batal digunakan untuk membatalkan data yang diisi pada *textbox* dan mengosongkan semua *textbox*.



Sumber: Hasil Penelitian (2018)

**Gambar 6. Implementasi Tampilan Halaman Bantuan**

#### 4.7. Tampilan Kas Keluar

form data pengeluaran ini dipanggil dari menu utama, pilih menu transaksi kemudian pilih submenu transaksi pengeluaran, kemudian masukan data kas keluar yaitu kode akun, nama akun, keterangan, tanggal dan jumlah. Didalam form data kas keluar terdapat beberapa tombol yang digunakan di antaranya tombol simpan, edit, hapus, batal, dan tombol keluar. Tombol simpan digunakan untuk menyimpan data yang baru dimasukan. Tombol edit digunakan untuk mengedit data yang telah dimasukan. Tombol hapus untuk menghapus data yang telah dimasukan. Tombol batal digunakan untuk membatalkan data yang diisi pada *textbox* dan mengosongkan semua *textbox*.

kode_akun	No_KK	nama_akun	Keterangan	Tanggal	Jumlah
511	01	Beban Gaji	Gaji Pegawai	11/09/2018	400000
512	02	Beban Listrik dan Te	Pembayaran Listrik dan Telepon	12/09/2018	400000
513	03	Beban Pegawai	Biaya Operasional	13/09/2018	250000
514	04	Beban Perlengkapan	Pembelian ATK	14/09/2018	250000

Sumber: Hasil Penelitian (2018)

Gambar 7. Implementasi Tampilan Kas Keluar

#### 4.8. Tampilan Halaman Buku Besar

Pada halaman ini terdapat form data laporan buku besar ini dipanggil dari menu utama, pilih menu laporan keuangan kemudian pilih submenu buku besar, kemudian klik tombol cetak maka akan muncul tampilan laporan buku besar. Klik tombol keluar maka akan tampil form menu utama.

Tanggal	Kode akun	Nama akun	Keterangan	Debet
11/09/2018	511	Beban Gaji	Pembayaran Listrik da	400.000,00
13/09/2018	513	Beban Pegawai	Pembelian ATK	250.000,00

Sumber: Hasil Penelitian (2018)

Gambar 2. Implementasi Buku Besar dan laporannya

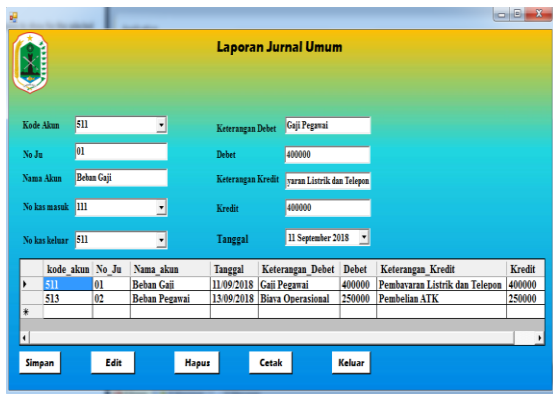
#### 4.9. Tampilan Laporan Laba Rugi, Jurnal Umum, dan Neraca Saldo

form laporan laba rugi, Jurnal Umum dan neraca saldo form data laporan dipanggil dari menu utama, pilih menu laporan keuangan kemudian pilih submenu laba rugi, maka akan tampil form laporan laba rugi.

Tanggal	No Km	Kode akun	Nama akun	Keterangan	Jumlah
11/09/2018	01	111	Kas	Pendapatan	8.500.000,00
Jumlah Pendapatan					8.500.000,00

Sumber: Hasil Penelitian (2018)

Gambar 2. Implementasi Laporan Laba rugi



Sumber: Hasil Penelitian (2018)  
**Gambar 2. Implementasi Laporan Jurnal Umum**



Sumber: Hasil Penelitian (2018)  
**Gambar 2. Implementasi Neraca Saldo**

#### 4.10. Implementasi Code Generation

Dalam hal ini penulis memberikan contoh sample yang digunakan dalam pembuatan aplikasi implementasi Code Generation dalam tahap metode Waterfall :

```
Imports System.Data.Odbc
Public Class login
    Private Sub menuutama_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        Timer1.Start()
    End Sub
```

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtlogin.Click
    Call koneksi()
    cmd = New OdbcCommand("select * from pengguna where nama_pengguna ='" & txtuser.Text & "' and kata_sandi='" & txtpass.Text & "'", conn)
    dr = cmd.ExecuteReader
    dr.Read()
    If Not dr.HasRows Then
        MsgBox("Login gagal periksa kembali nama pengguna dan kata sandi anda")
    Exit Sub
```

### 5. KESIMPULAN DAN SARAN

Berdasarkan pembahasan yang telah dikemukakan maka dapat diambil kesimpulan bahwa dengan menerapkan metode SDLC Waterfall untuk perancangan sistem informasi aplikasi yang dibuat diharapkan dapat memudahkan dalam pengolahan data administrasi.

Metode Waterfall dapat digunakan dalam pengembangan aplikasi ini karena sifatnya fokus terhadap masing-masing fase sehingga pengerjaan dilakukan secara maksimal karena tidak adanya pengerjaan secara paralel. Dengan adanya aplikasi sistem informasi administrasi ini diharapkan dapat membantu perusahaan dalam meningkatkan pelayanan dan pengolahan data keuangan di berbagai wilayah baik dalam kota maupun luar kota. Sedangkan pada pengolahan datanya antara lain menambah data, menyimpan data, mengubah data, menghapus data dan sampai pada pembuatan laporan.

### REFERENSI

Arizona, N. D. (2017). Aplikasi Pengolahan Data Anggaran Pendapatan dan Belanja Desa ( APBDES ) Pada Kantor Desa Bakau Kecamatan Jawai Berbasis Web, 1(2), 105–119.  
 Dharmawan, W. S. (n.d.). Game Logika Menara Hanoi dengan Bahasa Pemrograman Visual Basic . Net, 1–10.  
 Hutahean. (2014). Konsep Sistem Informasi.  
 Irmayani, W., & Susyati, E. (2017). Sistem Informasi



Anggaran Pendapatan dan Belanja Desa Berorientasi Objek. *Jurnal Khatulistiwa Informatika*, V(1), 58–63.

Kasemin. (2015). Agresi Perkembangan Teknologi Informasi.

Risdiansyah, D. (2017). Perancangan Sistem Informasi Bimbingan Konseling Berbasis Desktop pada SMA Kemala Bhayangkari 1 Kubu Raya. *Khatulistiwa Informatika*, V(2), 86–91.

Romney. (2015). *Accounting Informationm Systems*.

Sasongko, A. (2014). Perancangan Aplikasi Rekam Data Cuaca Hasil Pengamatan Observasi Stasiun Meteorologi BMKG Berbasis Website. *Jurnal Khatulistiwa Informatika*, 2(2), 115–126.