



## PENGEMBANGAN APLIKASI AKUNTANSI SEDERHANA BERBASIS WEB DENGAN BAHASA PEMOGRAMAN PHP, DATABASE MYSQL

Gabiel Gesa Widi Purnomo. Totok Dewayanto<sup>1</sup>

Departemen Akuntansi Fakultas Ekonomika dan Bisnis Universitas Diponegoro  
Jl.Prof. Soedharto SH Tembalang, Semarang 50239, Phone: +6282135240978

### ABSTRACT

*This study aims to develop a simple accounting application that can be used by companies, organizations to manage their financial data or as educational material for the general public about accounting practices.*

*Currently, the entry of transaction data is usually done manually by financial staff. Financial transactions can only be done on local computers, so if the financial staff or the person concerned is out of the office, they cannot make transactions. This is an obstacle for managers, because this data is very important to know the financial condition of the company. Seeing the problems that occur, it is necessary to create an accounting application that can help the work of the finance department and managers in viewing financial information.*

*The system development method used in this study is incremental. In this application there are accounting logics that are used to calculate transactions and balances from the accounts used. The result of this application is a financial report addressed to the manager. Ultimately, this application can help the work of the finance department in entering transactions and managers in viewing financial reports*

**Keywords:** *Analysis, Design, Application, Accounting*

### PENDAHULUAN

Salah satu faktor kesuksesan organisasi sangat bergantung kepada kemampuan mengumpulkan, mencatat, dan mengorganisasi data mengenai operasi-operasi perusahaan secara akurat dan tepat, mampu mengelola data secara efektif, serta menggunakannya untuk analisis aktivitas-aktivitas perusahaan. Jumlah informasi yang telah tersedia terus berkembang sedemikian besar dan pandangan bahwa data sebagai aset organisasi telah mulai tumbuh. Namun tanpa kemampuan yang memadai dalam pengelolaan data untuk menjadi informasi yang relevan secara cepat terhadap suatu persoalan, maka jumlah data yang besar justru akan menjadi beban baru yang harus ditanggung perusahaan (Hariyanto, 2004: 3)

Pada era sekarang, pencatatan data-data transaksi keuangan yang berlangsung pada perusahaan dan organisasi dilakukan secara *manual* oleh staf keuangan. Pencatatan transaksi keuangan hanya bisa dilakukan pada komputer lokal yang ada pada kantor saja, sehingga bila staf keuangan atau orang yang bersangkutan sedang berada di luar, maka ia tidak bisa melakukan pencatatan transaksi secara *real-time*. Ini merupakan suatu hambatan bagi manajer, karena data-data ini sangat penting untuk mengetahui keadaan keuangan perusahaan.

Dengan demikian, penulis tertarik untuk membuat aplikasi baru yang diharapkan akan mempermudah memproses pencatatan transaksi keuangan serta mengolahnya menjadi informasi yang berguna. Para staf keuangan dapat memproses transaksi keuangan secara *real-time*, dengan begitu manajer dapat melihat keadaan keuangan secara *real-time* meskipun tidak berada pada kantor, dan staf informasi dapat melakukan pemeliharaan data dengan lebih mudah.

---

<sup>1</sup> *Corresponding author*

---

## TINJAUAN LITERATUR

### Aplikasi

Aplikasi adalah sebuah program/perangkat lunak komputer yang digunakan untuk tugas yang spesifik atau khusus, seperti akuntansi, analisis dalam bidang sains atau medis, atau pemrosesan kata. Bagian “aplikasi” mengacu kepada keseluruhan set dari program yang secara kolektif mengimplementasikan sebuah proses bisnis yang spesifik. Program-program individual yang mengimplementasikan bagian dari proses bisnis ini dikenal sebagai “program aplikasi” atau “perangkat lunak aplikasi” (Nader, 1992: 20).

Program aplikasi atau program siap pakai. Program yang di desain untuk melaksanakan suatu fungsi bagi pengguna atau aplikasi yang lain. Contoh-contoh aplikasi ialah program pemroses kata dan *Web Browser*. Aplikasi akan menggunakan sistem operasi (OS) komputer dan aplikasi lain yang mendukung. Istilah ini mulai perlahan masuk ke dalam istilah Teknologi Informasi semenjak tahun 1993, yang biasanya juga disingkat dengan app. Secara historis, aplikasi adalah *software* yang dikembangkan oleh sebuah perusahaan (Febrian, 2007:35).

### Akuntansi

Akuntansi adalah proses pengidentifikasian, pencatatan, dan pengkomunikasian kejadian – kejadian ekonomi suatu organisasi (perusahaan ataupun bukan perusahaan) kepada para pemakai informasi yang berkepentingan. Akuntansi memberikan jasa yang sangat vital dengan memasok informasi yang dibutuhkan oleh para pengambil keputusan untuk membuat “pilihan–pilihan yang masuk akal di antara alternatif – alternatif penggunaan sumber – sumber daya yang langka dalam menjalankan bisnis dan kegiatan - kegiatan ekonomi”(Simamora, 2000 : 4).

Pengertian langka disini adalah sumber daya tersebut istimewa. Sumber daya itu ada dalam perusahaan, namun penggunaannya untuk aktivitas ekonomi tidak dapat langsung dipakai oleh karyawan. Misalkan ketika ingin mengambil dana dari kas untuk pembelian komputer, maka karyawan harus melapor kepada bagian keuangan untuk mendapat izin. Ketika melakukan penjualan barang produksi, maka karyawan harus melapor ke bagian keuangan untuk dicatat pendapatannya.

Akuntansi dapat didefinisikan sebagai sistem informasi yang menghasilkan laporan kepada pihak – pihak yang berkepentingan mengenai aktivitas ekonomi dan kondisi perusahaan.(Warren, et al.2006 : 10). Aktivitas ekonomi disini adalah aktivitas perusahaan dalam menggunakan sumber daya perusahaan untuk kelangsungan hidup perusahaan tersebut. Misalkan pembelian peralatan, penggunaan kas, penjualan saham, dan lain – lain. Kondisi perusahaan yang dimaksud disini yaitu keadaan keuangan pada perusahaan tersebut. Dari pencatatan aktivitas – aktivitas ekonomi tersebut, maka dapat diketahui pendapatan, pengeluaran, laba, rugi, aktiva dan pasiva dalam perusahaan tersebut.

Menurut AAA (American Accounting Association) (dalam Kardiman, et al .2007 : 2), akuntansi adalah proses pengidentifikasian, pengukuran, dan penyampaian informasi ekonomi yang memungkinkan dilakukannya penilaian dan keputusan yang tepat bagi para pemakai informasi tersebut. Pengertian ini menandakan bahwa akuntansi adalah sebuah sistem informasi, yaitu sebuah sistem yang menghasilkan informasi keuangan yang berdasarkan informasi tersebut dapat dilakukan penilaian dan keputusan yang tepat bagi para pemakainya.

**Gambar 1 Proses Akuntansi (Simamora, 2000: 4)**



### **Entity Relationship Diagram (ERD)**

ERD adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak. ERD berbeda dengan DFD yang merupakan suatu model jaringan fungsi yang akan dilaksanakan oleh sistem, sedangkan ERD merupakan model jaringan data yang menekankan pada struktur – struktur dan relationship data (Ladjamuddin, 2005:142). ERD digunakan untuk membuat model data yang dipakai dalam aplikasi, serta relasi antar data tersebut.

### **Data Flow Diagram (DFD)**

Diagram alir data merupakan model dari sistem untuk menggambarkan pembagian sistem ke modul yang lebih kecil. Salah satu keuntungan menggunakan diagram alir data adalah memudahkan pemakai atau *user* yang kurang menguasai bidang komputer untuk mengerti sistem yang akan dikerjakan (Ladjamuddin, 2005 : 64). DFD dipakai untuk membuat model proses yang berjalan pada aplikasi, serta proses yang mengakses data.

Dalam memahami dalam membaca DFD, maka pengembangan DFD disusun berdasarkan tingkatan dari atas ke bawah yaitu Diagram Konteks, Diagram Nol, dan Diagram Rinci . Dalam penggambaran DFD, diperlukan Kamus data, Kamus data sering disebut juga dengan sistem *data dictionary* adalah katalog fakta tentang data dan kebutuhan – kebutuhan informasi dari suatu sistem informasi. Dengan menggunakan kamus data, analisis sistem dapat mendefinisikan data yang mengalir di sistem dengan lengkap. Pada tahap analisis, kamus data digunakan sebagai alat komunikasi antara analisis sistem dengan pemakai sistem tentang data yang mengalir di sistem, yaitu tentang data yang masuk ke sistem dan tentang informasi yang dibutuhkan oleh pemakai sistem.

### **Flowchart**

*Flowchart* adalah bagan – bagan yang mempunyai arus yang menggambarkan langkah – langkah suatu masalah. *Flowchart* merupakan cara penyajian dari suatu algoritma (Ladjamuddin, 2005:263). *Flowchart* menggambarkan logika – logika yang berjalan pada proses di dalam aplikasi. Ada dua macam *flowchart* yang menggambarkan proses dengan komputer yaitu Sistem *Flowchart* dan Program *Flowchart*

### **PHP**

PHP adalah suatu bahasa pemrograman *open source* yang digunakan secara luas terutama untuk pengembangan *web* dan dapat disimpan dalam bentuk HTML. Untuk menghasilkan sebuah HTML, script yang ditulis menggunakan PHP mempunyai perintah yang lebih singkat dibandingkan bahasa pemrograman lain seperti Perl atau C. *User* hanya perlu memasukkan kode untuk melakukan sesuatu (misalnya menulis suatu kalimat) di antara tag awal dan tag akhir PHP. (Komputer, 2006 : 11)

PHP singkatan dari PHP Hypertext Preprocessor yang digunakan sebagai bahasa *script server-side* dalam pengembangan *web* disisipkan pada dokumen HTML. Penggunaan PHP memungkinkan *web* dapat dibuat dinamis sehingga *maintenance* situs *web* tersebut menjadi lebih mudah dan efisien. PHP merupakan *software open-source* yang disebar dan dilisensikan secara gratis serta dapat di-*download* secara bebas dari situs resminya, <http://www.php.net>. PHP ditulis menggunakan bahasa C. (Peranginangin, 2006 : 2)

PHP dibuat dan diperkenalkan pertama kali oleh Rasmus Lerdorf pada tahun 1995 menggunakan nama PHP/FI. PHP/FI merupakan akronim dari Personal Home Page /Forms Interpreter. Pada tahun 1997, dikeluarkan PHP/FI versi 2.0. Pada versi inilah pemrogram dapat menempelkan kode terstruktur di dalam tag HTML. Yang menarik, kode PHP juga bisa berkomunikasi dengan *database* dan melakukan perhitungan-perhitungan yang kompleks.

### **MYSQL**

MySQL adalah sebuah program *database server* yang mampu menerima dan mengirimkan datanya dengan sangat cepat, *multi user* serta menggunakan perintah standar SQL (*Structured Query Language*) MySQL memiliki dua bentuk lisensi, yaitu FreeSoftware dan Shareware. MySQL yang biasa digunakan adalah MySQL FreeSoftware yang berada di bawah lisensi GNU/GPL (*General Public License*). Keterangan lengkap dapat dilihat pada <http://www.gnu.org/licenses/>.

Selain konsumen juga dapat memiliki produk MySQL yang sifatnya komersial, biasa disebut dengan MySQL AB. (Nugroho, 2005 : 1)

SQL adalah sebuah bahasa permintaan *database* yang terstruktur. Bahasa SQL dibuat sebagai bahasa yang dapat merelasikan beberapa tabel dalam *database* maupun merelasikan antar *database*. Bahasa SQL ditulis langsung dalam sebuah program *database server* yang melayani permintaan *user*. SQL adalah bahasa permintaan yang digunakan dalam MySQL (Nugroho, 2005 : 5)

MySQL merupakan *database* yang sangat terkenal di kalangan *programmer web* karena MySQL merupakan *database* yang stabil dalam penyimpanan data. Begitu banyak programmer, baik *programmer web* atau *programmer aplikasi database* menggunakan MySQL sebagai media penyimpanan data. (Ibrahim, 2008 : 67)

### Apache

Server HTTP Apache atau *Server Web/WWW* Apache adalah *web server* yang dapat dijalankan di banyak sistem operasi (Unix, BSD, Linux, Microsoft Windows dan Novell Netware serta *platform* lainnya) yang berguna untuk melayani dan menfungsikan situs *web*. Protokol yang digunakan untuk melayani fasilitas *web/www* ini menggunakan HTTP.

*Web server* adalah suatu program (dan juga mesin yang menjalankan program) yang mengerti protokol HTTP dan dapat menanggapi permintaan – permintaan dari *web browser* yang menggunakan protokol tersebut. (Febrian, 2007 : 450).

### XAMPP

XAMPP merupakan sebuah paket instalasi untuk PHP, Apache, dan MySQL. Dengan menggunakan XAMPP, *user* tidak perlu lagi repot menginstall ketiga *software* itu secara terpisah. XAMPP sama seperti PHP yang sudah sering digunakan oleh *programmer web* dalam membuat aplikasi *web*. (Ibrahim, 2008 : 3)

### METODE PENELITIAN

Bab tiga akan menjelaskan tentang metodologi penelitian, yaitu urutan – urutan dalam melakukan penelitian ini. Pertama, bab tiga akan membahas metode pengumpulan data. Metode terdiri dari tiga metode, yaitu observasi, studi pustaka, dan wawancara. Disini akan dijelaskan sasaran – sasaran apa yang harus dipenuhi dalam setiap tahapan.

Bagian kedua dari bab tiga yaitu metode pengembangan sistem. Metode yang dipakai disini adalah metode *incremental*. Dalam bagian kedua ini juga terdapat kerangka berpikir, yaitu urutan – urutan pekerjaan yang dilakukan oleh penulis. Disini akan dijelaskan target – target yang harus dipenuhi dalam setiap tahapan metode *incremental*.

#### Metode Pengumpulan Data

Penelitian ini menggunakan data primer, dan referensi melalui literatur, buku, artikel maupun secara online menggunakan media *internet* untuk mendapatkan referensi yang berhubungan dengan penulisan skripsi ini. Sasaran dari tahapan ini adalah :

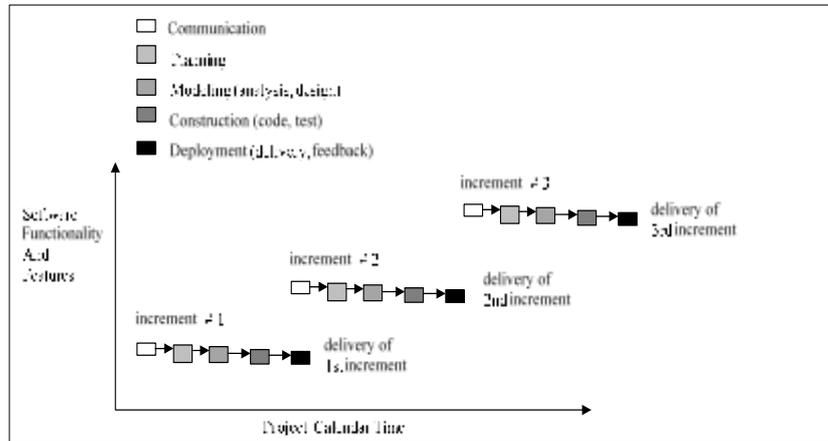
1. Mencari literatur – literatur yang berhubungan dengan akuntansi, PHP, MySQL, dan jQuery.
2. Mencari studi sejenis yang berhubungan dengan aplikasi akuntansi.

#### Metode Pengembangan Sistem

Dalam penelitian ini, penulis menggunakan metode penelitian *incremental*. Ada banyak situasi dimana syarat awal sebuah *software* telah didefinisikan secara jelas, namun lingkungan pengembangannya secara umum tidak selalu bisa linier. Sebagai tambahan, bisa saja ada kebutuhan mendesak untuk memberikan beberapa fungsi dari *software* kepada *user*, kemudian diperbaiki, dan dikembangkan dalam fungsi tersebut dalam pengembangan *software* selanjutnya. Artikel menggunakan Bahasa Indonesia.

Pemilihan metode ini karena masalah yang dihadapi sudah dimengerti dan ditetapkan dari awal, namun dalam prosesnya terdapat penambahan – penambahan dalam penyempurnaan aplikasi. Berikut adalah gambar paradigma *incremental*.

Gambar 2 Paradigma Incremental (Sumber : Pressman, 2010 : 39)



### Kerangka Berpikir

Berikut adalah penjelasan dari kerangka berpikir ini :

#### 1. *Communication*

Tahap pertama adalah komunikasi, dimana ini sangat penting untuk mendapatkan syarat – syarat kebutuhan dari *user*. Target dari tahapan ini adalah mendapatkan kebutuhan dari *user* yaitu aplikasi akuntansi berbasis *web*, dimana didalam aplikasi tersebut dapat membuat laporan keuangan dan penginformasian transaksi – transaksi yang ada.

#### 2. *Planning*

Disini dibuat jadwal – jadwal perencanaan untuk membuat aplikasi tersebut. Target dari tahapan ini adalah membuat penjadwalan dari tahap *planning* , konstruksi, dan sampai tahap *deployment*.

#### 3. *Modeling*

Tahap ini bertujuan untuk mendesain aplikasi yang akan dibuat. Setelah informasi keuangan diformulasikan secara lengkap, kemudian diimplementasikan dengan membuat perancangan aplikasi yang akan dibangun. Target dari tahapan ini, yakni:

1. Menganalisis kebutuhan aplikasi yang akan dibuat.
2. Menganalisis tahapan – tahapan proses akuntansi.
3. Merancang *normalization, entity relationship diagram, data flow diagram, kamus data, dan flowchart*.
4. Merancang *user interface*

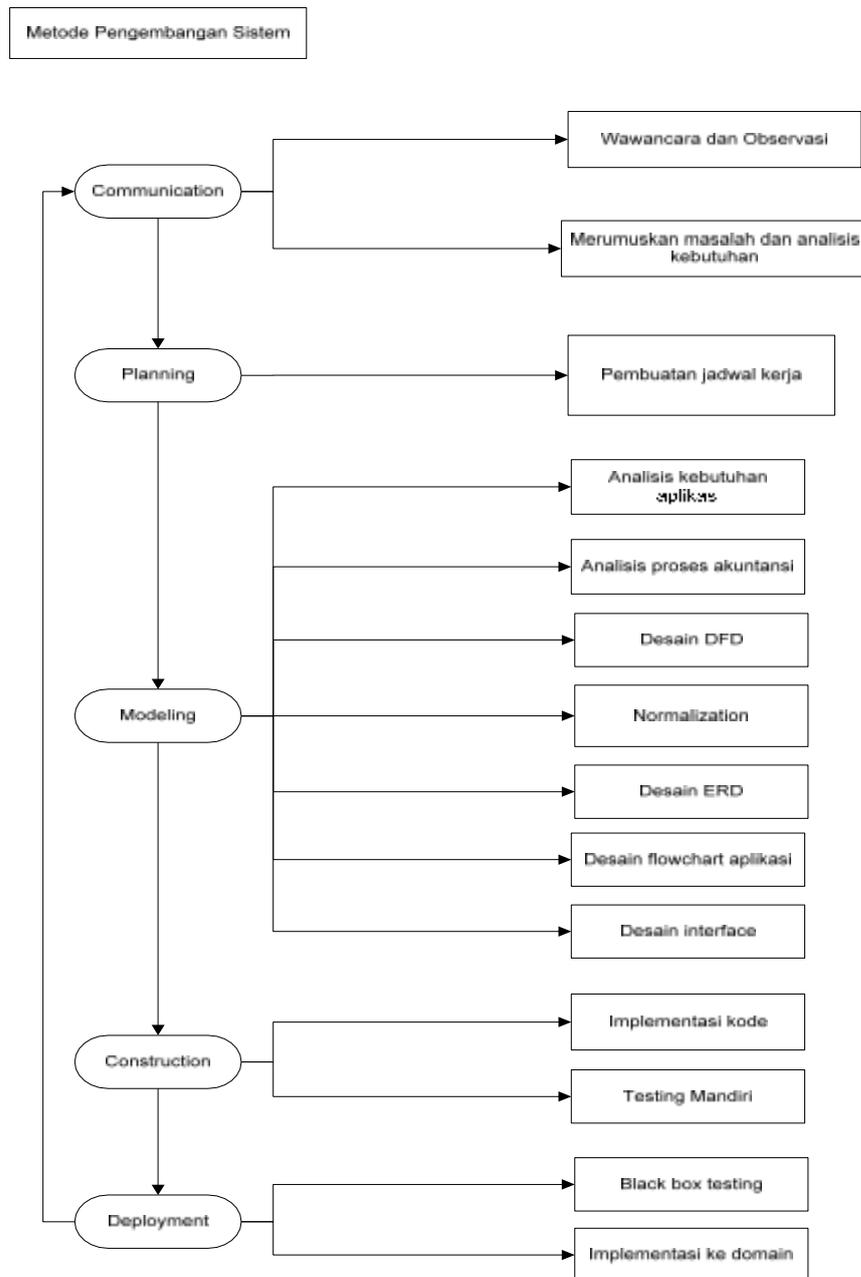
#### 4. *Construction*

Tahap ini dimana desain yang telah dibuat akan diimplementasikan melalui penulisan kode-kode komputer dalam membangun aplikasi. Serta dilakukan uji coba pada program tersebut agar aplikasi dapat berjalan seperti yang diinginkan.

#### 5. *Deployment*

Tahap terakhir dalam pembuatan aplikasi dengan melakukan uji coba penuh terhadap program yang telah dibangun. Setelah program berhasil di uji, langkah selanjutnya dengan memasukan program tersebut ke dalam domain untuk dapat di akses melalui internet.

Gambar 3 Kerangka Berpikir



## HASIL PENELITIAN DAN PEMBAHASAN

### Planning

Dalam tahap ini dibuat rancangan untuk membuat aplikasi. Beberapa tahapan kerja yang akan dilakukan antara lain :

1. Analisis kebutuhan aplikasi yang akan dibuat
2. Analisis tahapan proses akuntansi di
3. Perancangan DFD.
4. *Normalization*
5. Perancangan ERD.
6. Perancangan *flowchart*.
7. Perancangan *interface* serta pengkodean aplikasi.
8. *Testing* kepada *user* untuk mendapatkan *feedback*.

### Analisis Kebutuhan Aplikasi

Berdasarkan wawancara dan observasi, maka aplikasi yang akan dibuat adalah aplikasi akuntansi berbasis *web*. *Web server* yang digunakan adalah Apache, menggunakan bahasa PHP dan *database* MySQL. Semua ini dapat di-*install* menggunakan XAMPP. Kemudian untuk membuat *chart* menggunakan FusionChart Free dan menggunakan jQuery untuk membuat aplikasi lebih interaktif. Untuk memakai aplikasi ini, maka harus dibuat beberapa program untuk masalah berikut

1. *User* : berguna untuk mengklasifikasikan *user* yang menggunakan aplikasi ini. Terdapat dua *user*, yaitu *admin* dan *user*. *Admin* dapat mengganti data – data yang ada dan bisa memasukkan transaksi, sedangkan *user* hanya bisa melihat jurnal, buku besar, laporan keuangan, dan grafik.
2. Rekening : berguna untuk mengklasifikasikan transaksi – transaksi yang ada. Perhitungan transaksi yang ada berdasarkan kelompok rekening ini, yaitu harta, kewajiban, modal, pendapatan, dan beban.
3. Jurnal : berguna untuk memasukkan transaksi tersebut ke dalam *database*. Terdapat empat jurnal, yaitu jurnal umum, jurnal penyesuaian, jurnal penutup, dan jurnal pembalik.
4. Buku besar : berguna untuk mencatat semua transaksi yang berhubungan dengan rekening tertentu. Disini dapat terlihat keadaan perubahan rekening karena transaksi yang menggunakan rekening tersebut.
5. Laporan keuangan : berguna untuk melihat keadaan keuangan
6. Grafik : berguna untuk melihat saldo rekening tertentu pada rentang waktu tertentu. Program ini menggunakan diagram batang agar menarik dan interaktif

### Mekanisme Pencatatan Transaksi

Berikut akan dipaparkan proses pencatatan transaksi keuangan.

1. *User* harus terlebih dahulu membuat subklasifikasi rekening. *User* akan memasukkan kode subklasifikasi nama klasifikasi, dan kelompok rekening. Ada lima kelompok rekening yang digunakan, yaitu harta, kewajiban, modal, pendapatan, dan beban. Masing – masing kelompok rekening mempunyai perhitungan yang berbeda ketika ditempatkan di debet atau kredit. Setelah itu, *user* dapat membuat rekening, yaitu dengan memasukkan kode rekening, nama rekening, serta subklasifikasi rekening yang akan digunakan.
2. Selanjutnya, *user* akan memasukkan transaksi tersebut kedalam jurnal. Rekening yang sudah dibuat akan digunakan disini. *User* bisa memasukkan rekening lebih dari satu, tapi rekening tersebut tidak bisa muncul dua kali. Selain itu, *user* harus memasukkan jumlah uang yang digunakan dalam transaksi tersebut. Jumlah uang di kolom debet dan kolom kredit harus sama. Bila tidak, maka transaksi tidak bisa dimasukkan. Terdapat empat jurnal disini, yaitu :
  - a. Jurnal Umum  
Jurnal yang dipakai untuk pencatatan transaksi secara umum.
  - b. Jurnal Penyesuaian  
Jurnal yang dipakai untuk menyesuaikan rekening yang telah berubah nilainya. Contoh akunnya yaitu peralatan dan pendapatan.
  - c. Jurnal Penutup  
Jurnal yang dipakai untuk meng-nolkan rekening pendapatan dan beban, karena akan dimasukkan kedalam ikhtisar laba rugi.
  - d. Jurnal Koreksi  
Jurnal yang dipakai untuk mengoreksi kesalahan yang ada pada transaksi. Jurnal ini membalik posisi rekening yang digunakan pada saat transaksi. Jumlah uang yang digunakan tetap sama saat transaksi.
3. Jurnal  
Jurnal disini adalah data – data transaksi yang telah berlangsung. *User* bisa melihat rekening – rekening apa saja yang digunakan dalam transaksi tersebut.

#### 4. Buku Besar

*User* dapat melihat transaksi – transaksi apa saja yang menggunakan rekening yang dipilih. Disini, *user* juga bisa melihat saldo rekening tersebut selama transaksi berlangsung.

#### 5. Laporan Keuangan

Laporan keuangan yang dibuat ada tiga, yaitu laporan laba rugi, neraca saldo, dan laporan perubahan modal. Laporan laba rugi akan memperlihatkan laba atau rugi yang didapat, neraca saldo menginformasikan aktiva dan pasiva yang dimiliki oleh, dan laporan perubahan modal memperlihatkan perubahan modal yang terjadi di karena aktivitas – aktivitas ekonomi

### Proses Perhitungan Akuntansi

Setelah *user* memasukkan transaksi tersebut, maka aplikasi akan melakukan perhitungan untuk membuat buku besar, dan laporan keuangan. Berikut adalah penjelasannya :

#### 1. Jurnal Umum

Perhitungan transaksi – transaksi yang ada berdasarkan kelompok rekening dan jumlah uang yang dipakai. Karakteristik rekening dapat dilihat pada tabel 2.1 Bila rekening harta atau beban berada pada debit, maka saldonya akan ditambah dengan jumlah uang yang digunakan. Bila rekening harta atau beban berada pada kredit, maka saldonya akan dikurangi dengan jumlah uang yang digunakan. Bila rekening kewajiban, modal, dan pendapatan berada pada debit, maka saldonya akan dikurangi. Bila rekening kewajiban, modal, dan pendapatan berada pada kredit, maka saldonya akan ditambah.

#### 2. Jurnal Penyesuaian

Aturan perhitungan pada jurnal penyesuaian sama seperti jurnal umum. Namun, rekening yang digunakan disini adalah rekening yang bisa menyusut atau berkurang nilainya, seperti peralatan, perlengkapan, dan gedung. Berkurangnya nilai rekening tersebut akan ditampung pada rekening dengan klasifikasi beban.

#### 3. Jurnal Penutup

Jurnal penutup berfungsi untuk meng-nolkan rekening pendapatan dan beban. Pada jurnal ini, *user* tidak bisa memasukkan rekening baru, karena sudah ditentukan oleh aplikasi. Aplikasi akan mencari selisih antara jumlah pendapatan dan beban. Bila hasilnya positif, maka mendapatkan laba. Bila hasilnya negatif, maka menderita rugi.

#### 4. Jurnal Koreksi

Jurnal koreksi berfungsi untuk mengkoreksi jurnal yang telah dimasukkan. Yang dilakukan oleh jurnal koreksi yaitu membalik posisi debit atau kredit rekening pada saat transaksi. Jumlah uang yang digunakan di jurnal koreksi sama dengan jurnal yang akan dikoreksi. Dengan ini, maka saldo dari rekening – rekening yang ada pada jurnal tersebut akan kembali seperti sebelum proses penjurnalan

#### 5. Buku Besar

Buku besar menyimpan semua transaksi yang berhubungan dengan suatu rekening tertentu. Perhitungan dalam buku besar berdasarkan karakteristik dari akun tersebut. Di dalam buku besar akan terlihat jumlah saldo dari rekening tersebut ketika melakukan transaksi.

#### 6. Laporan Laba Rugi

Laporan ini adalah untuk menunjukkan laba atau rugi yang didapatkan. Untuk mendapatkan laba atau rugi adalah dengan mendapatkan selisih antara jumlah pendapatan dan jumlah beban. Didalam laporan ini akan terlihat saldo dari semua rekening pendapatan dan rekening beban

#### 7. Neraca Saldo

Neraca saldo akan memperlihatkan kondisi aktiva dan pasiva yang dimiliki. Aktiva adalah harta yang dimiliki, sedangkan pasiva adalah modal dan kewajiban. Semua rekening harta, kewajiban, dan modal akan terlihat disini beserta saldo masing – masing rekening dan jumlah saldo aktiva dan pasiva.

## 8. Laporan Perubahan Modal

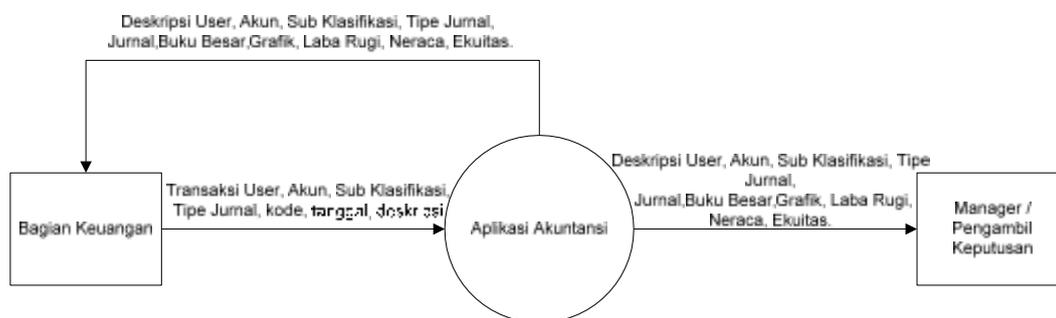
Laporan ini untuk menunjukkan perubahan modal karena aktivitas – aktivitas ekonomi yang berjalan. Disini akan dihitung laba atau rugi yang diterima dan pengambilan pribadi yang dilakukan. Laba akan menambah modal, sedangkan rugi dan pengambilan pribadi akan mengurangi modal.

### Target Akhir=

Target akhir yang akan dicapai adalah sebuah aplikasi akuntansi yang bisa membantu bagian keuangan untuk dapat bekerja lebih efisien dan para manajer dalam menerima informasi. Selain itu, para manajer juga dapat dengan mudah untuk mengakses informasi – informasi dalam aplikasi ini.

### Data Flow Diagram

Gambar 4 Diagram Konteks Aplikasi Akuntansi



Dalam diagram konteks diatas terdapat dua buah entitas, yaitu bagian keuangan dan *manager* atau pengambil keputusan. Dalam hal ini, yang bisa memasukkan transaksi hanyalah bagian keuangan saja, sedangkan *manager* tidak bisa memasukkan transaksi. Manager hanya bisa menampilkan data – data tentang keuangan serta melihat laporan keuangan.

### Construction.

Dalam fase konstruksi, bahasa yang digunakan adalah PHP dengan menggunakan *database* MySQL.

Dalam pengkodean ini, penulis menggunakan aturan – aturan akuntansi untuk memasukkan transaksi – transaksi ke dalam jurnal. Kemudian dari jurnal – jurnal tersebut akan dibuat laporan keuangan. Dari laporan keuangan tersebut dapat diketahui kondisi keuangan saat ini. Pada tahap ini penulis juga melakukan *testing* mandiri untuk mengetahui jalannya program yang telah dibuat.

### Deployment

Pada tahapan ini, dilakukannya *blackbox testing* untuk menguji aplikasi yang telah dibuat. Cara pengujian *blackbox testing* dilakukan dengan menjalankan aplikasi keuangan dan melakukan *input* data serta melihat *input*-nya apakah sesuai dengan aturan – aturan akuntansi yang telah ditetapkan. Berikut adalah tabel pengujian

### Blackbox Testing

Blackbox testing adalah suatu metode pengujian perangkat lunak yang tidak memerlukan pengetahuan tentang struktur internal atau kode program. penguji hanya perlu berfokus pada input dan output aplikasi, tanpa melihat bagaimana sistem itu diproses dibelakang layar. Adapun karakteristik Blackbox Testing yaitu tidak perlu untuk akses ke kode sumber dengan hanya berinteraksi dengan antarmuka, berdasarkan spesifikasi kebutuhan dengan mengecek apakah aplikasi berfungsi, melakukan simulasi sebagai seorang pengguna sebenarnya saat menggunakan sistem. dan melakukan uji fungsionalitas fitur fitur didalamnya

TABEL 1 BLACKBOX TEXTING

No	Pengujian	Interface yang diharapkan	Hasil pengujian
1.	<i>Interface</i> halaman login	Halaman menu login	OK
2.	<i>Interface</i> halaman utama <i>admin</i>	Halaman utama <i>admin</i>	OK
3.	<i>Interface</i> halaman utama <i>user</i>	Halaman utama <i>user</i>	OK
4.	<i>Interface</i> data <i>user</i>	Halaman data <i>user</i>	OK
5.	<i>Interface</i> tambah <i>user</i>	Halaman tambah <i>user</i>	OK
6.	<i>Interface</i> ganti <i>user</i>	Halaman ganti <i>user</i>	OK
7.	<i>Interface</i> hapus <i>user</i>	Halaman hapus <i>user</i>	OK
8.	<i>Interface</i> data subklasifikasi akun	Halaman data subklasifikasi akun	OK
9.	<i>Interface</i> tambah subklasifikasi akun	Halaman tambah subklasifikasi akun	OK
10.	<i>Interface</i> ganti subklasifikasi akun	Halaman ganti subklasifikasi akun	OK
11.	<i>Interface</i> hapus subklasifikasi akun	Halaman hapus subklasifikasi akun	OK
12.	<i>Interface</i> data rekening	Halaman data rekening	OK
13.	<i>Interface</i> tambah rekening	Halaman tambah rekening	OK
14.	<i>Interface</i> ganti rekening	Halaman ganti rekening	OK
15.	<i>Interface</i> hapus rekening	Halaman hapus rekening	OK
16.	<i>Interface</i> cari rekening	Halaman cari rekening	OK
17.	<i>Interface</i> data tipe jurnal	Halaman data tipe jurnal	OK
18.	<i>Interface</i> tambah tipe jurnal	Halaman tambah tipe jurnal	OK



19.	<i>Interface ganti tipe jurnal</i>	Halaman ganti tipe jurnal	OK
20.	<i>Interface hapus tipe jurnal</i>	Halaman hapus tipe jurnal	OK
21.	<i>Interface jurnal umum</i>	Halaman jurnal umum	OK
22.	<i>Interface tambah jurnal umum</i>	Halaman tambah jurnal umum	OK
23.	<i>Interface jurnal penyesuaian</i>	Halaman jurnal penyesuaian	OK
24.	<i>Interface tambah jurnal penyesuaian</i>	Halaman tambah jurnal penyesuaian	OK
25.	<i>Interface jurnal penutup</i>	Halaman jurnal penutup	OK
26.	<i>Interface tambah jurnal penutup</i>	Halaman tambah jurnal penutup	OK
27.	<i>Interface tampil data jurnal</i>	Halaman tampil data jurnal	OK
28.	<i>Interface cari jurnal</i>	Halaman cari jurnal	OK
29.	<i>Export data jurnal</i>	Export data jurnal ke dalam Microsoft Excel	OK
30.	<i>Interface jurnal koreksi</i>	Halaman jurnal koreksi	OK
31.	<i>Interface tambah jurnal koreksi</i>	Halaman tambah jurnal koreksi	OK
32.	<i>Interface buku besar</i>	Halaman buku besar	OK
33.	<i>Interface buku besar detail</i>	Halaman buku besar detail	OK
34.	<i>Interface grafik harta</i>	Halaman grafik harta	OK
35.	<i>Interface grafik kewajiban</i>	Halaman grafik kewajiban	OK
36.	<i>Interface grafik modal</i>	Halaman grafik modal	OK
37.	<i>Interface grafik pendapatan</i>	Halaman grafik pendapatan	OK
38.	<i>Interface grafik beban</i>	Halaman grafik beban	OK
39.	<i>Interface grafik pendapatan dan beba</i>	Halaman grafik pendapatan dan beban	OK
40.	<i>Interface neraca saldo</i>	Halaman neraca saldo	OK



41.	<i>Exported neraca saldo</i>	Neraca saldo dalam Microsoft Excel	OK
42.	<i>Interface Laporan Laba Rugi</i>	Halaman Laporan Laba Rugi	OK
43.	<i>Exported Laporan Laba Rugi</i>	Laporan Laba Rugi dalam Microsoft Excel	OK
44.	<i>Interface Laporan Perubahan Modal</i>	Halaman Laporan Perubahan Modal	OK
45.	<i>Exported Laporan Perubahan Modal</i>	Laporan Perubahan Modal dalam Microsoft Excel	OK

## KESIMPULAN DAN SARAN

### Kesimpulan

Aplikasi akuntansi berbasis web ini dapat diakses dengan mudah oleh siapa saja dan di mana saja secara online sehingga memudahkan pihak untuk memasukan, mengolah dan mengorganisir data dan informasi. Selain itu, manajer atau stakeholder lainnya dimudahkan dalam memeriksa keadaan keuangan untuk mencapai sebuah pengambilan keputusan.

### Keterbatasan

Meskipun PHP adalah salah satu bahasa yang sering digunakan dalam membangun web, salah satu masalah terbesarnya adalah kerentanan dalam keamanan. Sama halnya dengan menggunakan database MYSQL, database bisa menjadi target serangan jika koneksi database tersebut tidak diamankan dengan baik.

Dari segi performa PHP bersifat server-side scripting, sehingga setiap request diproses ulang. sedangkan MYSQL akan mengalami penurunan kecepatan jika *query* tidak dioptimalkan.

Terakhir, kombinasi antara program dengan bahasa pemograman PHP berbasis data MYSQL, memiliki kelemahan tidak dirancang untuk real-time updates seperti fitur chat atau notifikasi instan, karena keduanya bersifat *request-response*.

### Saran

Beberapa rekomendasi berikut dapat diberikan kepada pihak-pihak yang ingin melakukan penelitian serupa di kemudian hari.

1. Diharapkan pada pengembangan selanjutnya aplikasi ini terdapat jurnal yang lebih detail
2. Diharapkan pada pengembangan selanjutnya aplikasi dapat melakukan penyimpanan data untuk menyimpan data pada rentang waktu lebih lama.
3. Dilakukan pengembangan lebih lanjut sehingga dapat dibuat laporan arus kas secara otomatis untuk memudahkan pekerjaan bagian keuangan.

## REFERENSI

- Febrian, Jack. 2007. *Kamus Komputer dan Teknologi Informasi*. Bandung : Informatika Bandung.
- Hakim, Lukmanul. 2010. *Bikin Website Super Keren Dengan PHP dan jQuery*. Yogyakarta : Lokomedia.
- Hariyanto, Bambang. 2004. *Sistem Manajemen Basis Data*. Bandung : Informatika Bandung.
- Husni.2007. *Pemrograman Database Berbasis Web*.Yogyakarta : Graha Ilmu
- Ibrahim, Ali. 2008.*Cara Praktis Membuat Website Dinamis Menggunakan XAMPP*. Yogyakarta : Neotekno.
- Jusup, Al Haryono. 2005. *Dasar – Dasar Akuntansi*. Yogyakarta : Sekolah Tinggi Ilmu Ekonomi YKPN.
- Kardiman, Agus Suranto.2007. *Prinsip – Prinsip Akuntansi 1 : SMA Kelas XI*.



- Jakarta : Yudhistira
- Komputer, Wahana.2006.*Seri Panduan Lengkap Menguasai Pemrograman Web Dengan PHP5*.Yogyakarta : Andi.
- Ladjamuddin, Al-bahra. 2005. *Analisis dan Desain Sistem Informasi*. Yogyakarta : Graha Ilmu.
- Nader, J. C. 1992. *Prentice Hall's Illustrated Dictionary of Computing*. New South Wales : Prentice Hall Inc.
- Nugroho, Bunafit.2005.*Database Relasional Dengan MySQL*.Yogyakarta : Andi
- Peranginangin, Kasiman. 2006. *Aplikasi Web dengan PHP dan MySQL*. Yogyakarta : Andi.
- Post,Gerald V, David L. Anderson. 2003.*Management Information Systems : Solving Bussiness Problems With Information Technology*. New York : McGraw-Hill.
- Pressman,Roger. 2010. *Software Engineering, A Practitioner's Approach*. USA : McGraw-Hill
- Rizki Amalia, Tri. 2010. *Sistem Informasi Akuntansi Pengeluaran Kas (Studi Kasus : BNI Syariah Fatmawati Jakarta Selatan)*. Jakarta : Universitas Islam Negeri SyBu Hidayatullah.
- Setiawati, Wina. *Aplikasi Laporan Keuangan Berbasis Web Untuk Kelurahan Dukuh*. Jakarta : Universitas Gunadarma.
- Simamora, Henry. 2000. *Akuntansi, Basis Pengambilan Keputusan*. Jakarta : Salemba Empat.
- Simarmata, Janner. 2007. *Perancangan Basis Data*. Yogyakarta : Andi
- Turban, Efraim, *et all*. 2004. *Information Technology For Management : Transforming Organizations In The Digital Economy*. USA : John Wiley And Sons.
- Whitten,Jeffrey L,*et all*. 2004. *System Analysis And Design Methods*. USA : McGraw-Hill
- Wismakarma, Komang. 2010. *Panduan Lengkap Menguasai Pemrograman CSS*. Yogyakarta : Lokomedia.