

APLIKASI OTOMATISASI TRANSAKSI KAS UNTUK MENGURANGI TINGKAT KESALAHAN PEKERJA

Putu Manik Prihatini, I Ketut Gede Sudiarta

Jurusan Teknik Elektro, Politeknik Negeri Bali

Bukit Jimbaran, P.O.Box 1064 Tuban Badung – BALI

Phone:+62-361-701981, Fax:+62-361-701128 E-mail: manikprihatini@gmail.com, itutde@gmail.com

Abstrak

Kas merupakan komponen aktiva yang paling aktif. Pengamanan terhadap arus kas sangat penting untuk menjaga terjadinya kebocoran-kebocoran terhadap aliran kas. Kemajuan dunia teknologi informasi dan komunikasi telah memberikan sumbangan aplikasi untuk melakukan pengawasan dan pengamanan terhadap kebocoran kas. Kelemahan muncul ketika tidak semua aplikasi menyediakan otomatisasi proses. Kelemahan ini menjadi dasar perancangan dan pembangunan aplikasi dalam penelitian. Model sistem dibatasi hanya untuk penerimaan dan pengeluaran kas sederhana dalam ruang lingkup sekolah. Aplikasi dikembangkan melalui tahap analisis model sistem manual dalam bentuk Bagan Alir, kemudian memodelkan sistem otomatisasi dalam bentuk Bagan Alir, merancang dua puluh satu tabel, merancang tampilan dan algoritma yang diimplementasikan dalam bentuk kode program, kemudian menguji proses otomatisasi dari aplikasi yang dibangun, sehingga menghasilkan sebuah aplikasi yang mampu mengurangi jumlah proses vital yang harus dikerjakan oleh pengguna, untuk mengurangi tingkat kesalahan pekerjaan dan mempercepat waktu proses yang dilakukan.

Kata Kunci: kas, otomatisasi, aplikasi

Application of Automation Cash Transaction to Reduce User Error Rate

Abstract

Cash is the most active component of assets. Security over cash flow is essential to keep the leakage of the cash flow. The advancement of information and communication technology has contributed applications to monitoring and securing to the leakage of cash. The weakness arises when not all applications provide automation process. This weakness becomes the basis of the design and development application in research. Model of the system is limited only to simple cash receipts and payments within the scope of the school. Applications developed through the analysis phase of the manual system model in the form of Flow Chart, then modeling the automation system in the form of Flow Chart, designing twenty-one tables, designing the display and algorithms are implemented in the form of program code, then testing automation process of the application built, thus producing an application that is able to reduce the number of vital processes that must be done by the user, to reduce the error rate and speed up the processing time of work performed.

Key words: cash, automation, application

I. PENDAHULUAN

Kas dapat diartikan sebagai nilai uang kontan yang ada dalam perusahaan beserta pos-pos lain yang dalam jangka waktu dekat dapat diuangkan sebagai alat pembayaran kebutuhan financial yang mempunyai sifat paling tinggi tingkat likuiditasnya [1]. Sistem penerimaan kas adalah sistem yang dirancang untuk menangani transaksi yang berkaitan dengan sumber pemasukan kas yang diterima perusahaan. Sistem pengeluaran kas adalah sistem yang dirancang untuk membiayai berbagai transaksi yang berkaitan dengan pengeluaran kas dalam perusahaan [2]. Kas merupakan jenis aktiva yang sangat mudah untuk diselewengkan, sehingga perlunya pengawasan yang ketat agar tidak terjadi penyelewengan atau penyalahgunaan atas kas. Pengawasan dan pengamanan terhadap arus kas sangat penting terutama untuk menjaga terjadinya kebocoran-kebocoran terhadap aliran kas, seperti kurangnya pengamanan, kurangnya pengaturan pelayanan dan kesalahan pekerjaan. Dari sisi kurangnya pengamanan seperti penerimaan kas yang tidak tercatat seluruhnya atau sengaja tidak dicatat. Dari sisi kurangnya

pengaturan pelayanan disebabkan karena kondisi perusahaan yang sibuk tetapi jumlah karyawan tidak cukup untuk memberikan pelayanan sehingga menimbulkan antrian yang panjang. Dari sisi kesalahan pekerjaan, semakin sibuknya seorang pekerja dalam melakukan pekerjaannya dapat menimbulkan kesalahan seperti kesalahan hitung dalam penerimaan kas. Oleh karena itu perlu diciptakan sistem dan prosedur yang dapat menjaga pengamanan terhadap kebocoran kas.

Kemajuan dalam dunia teknologi informasi dan komunikasi yang telah diaplikasikan dalam bidang akuntansi dan keuangan telah memberikan sumbangan aplikasi untuk melakukan pengawasan dan pengamanan terhadap kebocoran kas. Aplikasi yang dibangun ada yang ditujukan hanya untuk kebutuhan satu pengguna maupun kebutuhan banyak pengguna, dari tampilan yang sederhana sampai dengan tampilan yang penuh dengan kombinasi teks, gambar dan suara.

Akan tetapi, apakah aplikasi yang dibangun telah mempertimbangkan sisi kemudahan bagi pengguna dalam menggunakan aplikasi tersebut? Apakah beberapa

prosedur yang biasanya dilakukan secara manual, juga dilakukan melalui beberapa kali proses pada aplikasi tersebut, dengan kata lain tidak otomatis, sehingga menimbulkan kesan aplikasi yang dibangun hanya memindahkan pekerjaan dari selembar kertas ke sebuah komputer? Hal ini seringkali tidak menjadi perhatian dari suatu perusahaan ketika membangun atau membeli aplikasi berbasis komputer. Dari sisi pihak yang membangun aplikasi, hal ini seringkali dianggap sebagai sesuatu yang merepotkan karena melibatkan proses analisis, perancangan dan pembuatan program dengan kompleksitas yang tinggi dan tidak sebanding dengan waktu pembangunan aplikasi serta biaya yang dialokasikan oleh perusahaan. Padahal jika dilihat dari sisi pemeliharaan terhadap aplikasi yang dibangun dan sisi kemudahan pemakaian oleh pengguna, pembangunan aplikasi dengan proses yang otomatis, tentu saja akan memudahkan dan mempercepat waktu dari seorang pekerja dalam melakukan pekerjaannya, sehingga mampu meningkatkan pelayanan yang diberikan. Dari sisi pengamanan dan kesalahan pekerjaan, aplikasi dengan proses yang otomatis, mampu mengurangi tingkat kesalahan yang dilakukan seorang pekerja dengan berkurangnya jumlah tombol atau jumlah angka yang harus diketikkan melalui aplikasi.

Melalui penelitian ini, penulis ingin membahas mengenai perancangan dan pembangunan aplikasi yang mampu menyediakan kebutuhan otomatisasi proses, dalam bentuk sebuah Aplikasi Transaksi Kas. Penelitian dibatasi hanya untuk penerimaan dan pengeluaran kas sederhana dengan model sistem manual yang digunakan adalah transaksi kas dalam ruang lingkup sekolah.

II. METODE PENELITIAN

2.1 Metode Pengumpulan Data

Pengumpulan data dilakukan dengan studi pustaka dan pengamatan terhadap model sistem manual untuk memperoleh informasi mengenai prosedur penerimaan dan pengeluaran kas yang terjadi di model sistem manual, yang digambarkan dalam bentuk Bagan Alir.

2.3 Metode Pengembangan Sistem

Metode ini dilakukan dengan merancang model sistem otomatisasi dalam bentuk Bagan Alir, merancang basis data dengan struktur tabel-tabel yang mendukung proses otomatisasi dari algoritma, yang diimplementasikan dengan menggunakan piranti bantu perangkat lunak berbasis web.

2.4 Metode Pengujian

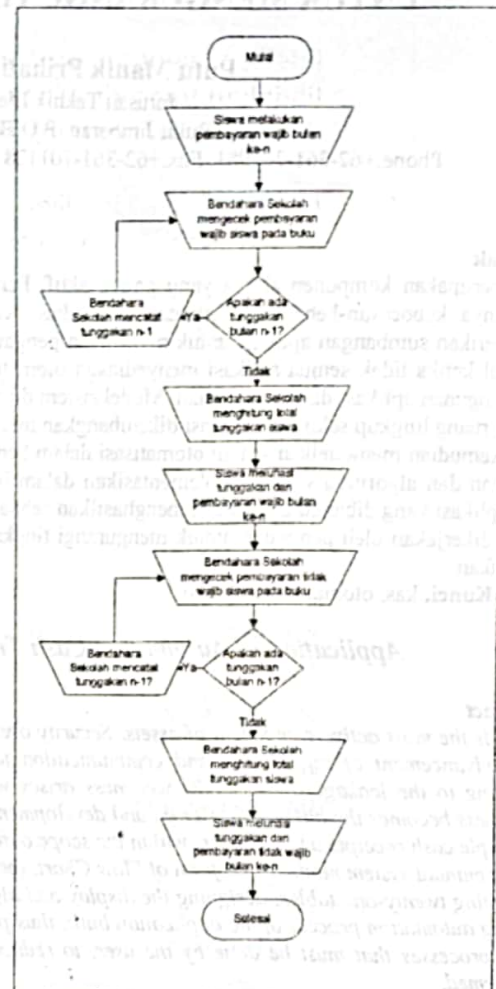
Metode pengujian dilakukan untuk menguji proses otomatisasi dari aplikasi yang dibangun.

III. HASIL DAN PEMBAHASAN

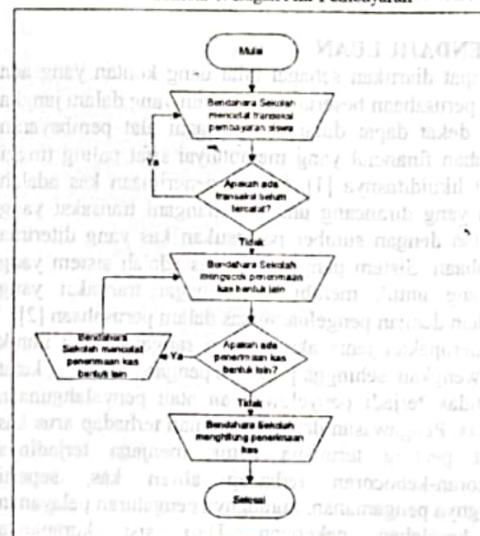
3.1 Analisis Sistem

Pada model sistem, penerimaan kas bersumber dari pembayaran siswa yang terdiri dari pembayaran wajib per bulan seperti SPP, dan pembayaran tidak wajib yang dapat dicicil seperti pembelian buku. Adapun proses pembayaran, penerimaan kas dan pengeluaran kas di

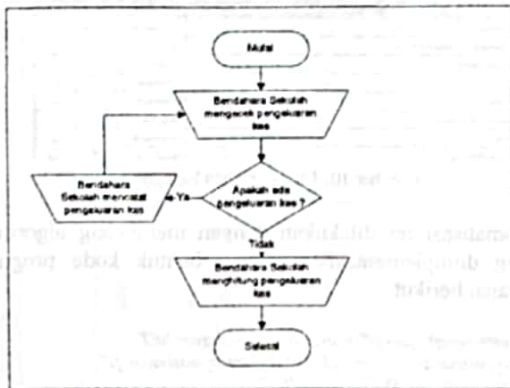
model sistem manual digambarkan dalam bentuk Bagan Alir pada Gambar 1, Gambar 2 dan Gambar 3.



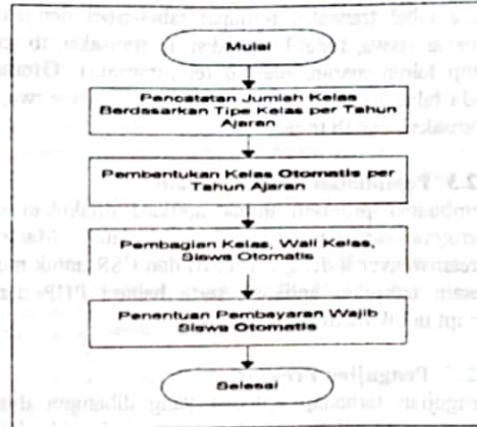
Gambar 1. Bagan Alir Pembayaran



Gambar 2. Bagan Alir Penerimaan Kas



Gambar 3. Bagan Alir Pengeluaran Kas



Gambar 5. Bagan Alir Inisialisasi Sistem

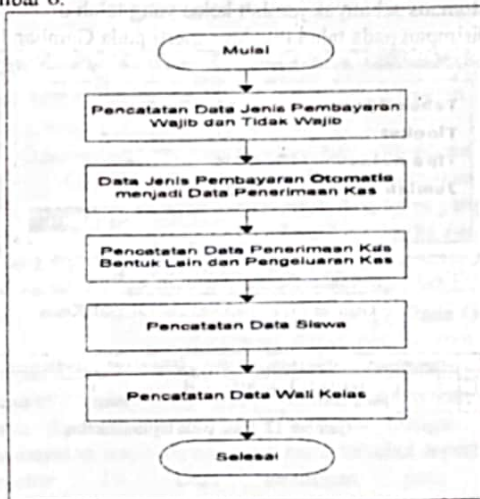
Kelemahan yang muncul dari model sistem manual tersebut antara lain:

- Bendahara Sekolah melakukan proses pengecekan dan pencatatan pembayaran wajib dan tidak wajib secara berulang-ulang terhadap seorang siswa setiap transaksi pembayaran
- Bendahara Sekolah melakukan pencatatan ulang transaksi pembayaran siswa ke dalam transaksi penerimaan kas
- Kemungkinan terjadinya kesalahan hitung pada transaksi pembayaran siswa, penerimaan kas dan pengeluaran kas
- Kesulitan dalam pembuatan laporan

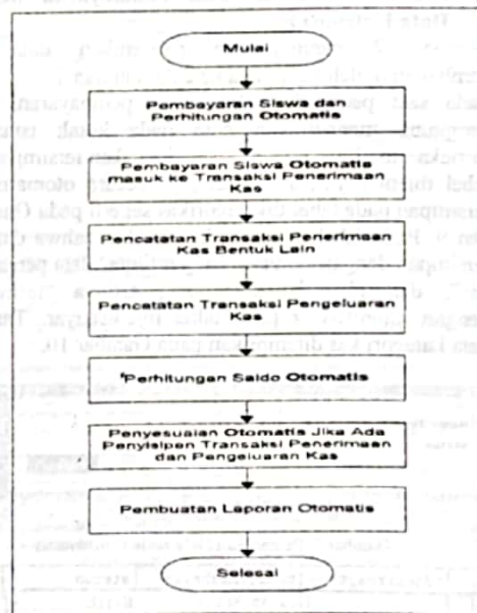
3.2 Perancangan Sistem

3.2.1 Perancangan Model Sistem

Model sistem dirancang dalam bentuk Bagan Alir meliputi pencatatan data utama, inisialisasi sistem, pembayaran, transaksi penerimaan dan pengeluaran kas, serta pelaporan seperti pada Gambar 4, Gambar 5 dan Gambar 6.



Gambar 4. Bagan Alir Pencatatan Data Utama



Gambar 6. Bagan Alir Transaksi Kas

3.2.2 Perancangan Basis Data

Tabel-tabel yang diperlukan dalam aplikasi ini dibuat dengan menggunakan piranti pengembang SQLyog Community Edition. Tabel tersebut tersimpan dalam database **dbpembayaransiswa**. Ada 21 tabel pada basis data, terdiri dari 9 tabel data utama, 6 tabel inisialisasi dan 6 tabel transaksi.

Pada tabel utama terdapat tabel-tabel dengan nama **tagama**, **tjenisbayar**, **tbkabupaten**, **tbkategorikas**, **tbpejabat**, **tbsiswa**, **tbtipekelas**, **tuser**, dan **tbwalikelas**. **Otomatisasi** pada tabel utama terletak pada **tjenisbayar** dan **tbkategorikas**.

Pada tabel inisialisasi terdapat tabel-tabel dengan nama **tbjumlahkelas**, **tbkelas**, **tbkelasasiswa**, **tjenisbayarsiswa**, **tbdetailjenisbayarsiswa**, dan **tbsubdetailjenisbayarsiswa**. **Otomatisasi** pada tabel inisialisasi terletak pada seluruh tabel yang ada.

Pada tabel transaksi terdapat tabel-tabel dengan nama tbbayar siswa, tbdetil transaksi, tb transaksi, tb temp, tb temp tahun ajaran, dan tb temptransaksi. Otomatisasi pada tabel transaksi terletak pada tb bayar siswa, tbdeti transaksi, dan tb transaksi.

3.2.3 Pembuatan Kode Program

Pembuatan program untuk aplikasi dilakukan dengan menggunakan piranti bantu yaitu Macromedia Dreamweaver 8 dengan HTML dan CSS untuk membuat desain tampilan aplikasi, serta bahasa PHP dan Java Script untuk membangun aplikasi.

3.2.5 Pengujian Program

Pengujian terhadap aplikasi yang dibangun dilakukan dengan menguji proses otomatisasi dari aplikasi sebagai berikut.

1. Otomatisasi Data Jenis Pembayaran Menjadi Data Kategori Kas

Gambar 7 menunjukkan pemasukan data jenis pembayaran oleh pengguna ke dalam aplikasi.

Pada saat pemasukan data jenis pembayaran, ketika pengguna mengetikkan data pada kotak isian dan menekan tombol Simpan, maka data akan tersimpan pada tabel tbjenisbayar, sekaligus juga secara otomatis akan tersimpan pada tabel tbkategorikas seperti pada Gambar 8 dan 9. Pada tabel tbkategorikas terlihat bahwa data SPP tersimpan dengan status=1 yang artinya "data penerimaan kas", dan idjenisbayar=2 yang artinya "terhubung" dengan idjenisbayar pada tabel tbjenisbayar. Tampilan data kategori kas ditampilkan pada Gambar 10.

Gambar 7. Pemasukan Data Jenis Pembayaran

| idjenisbayar | namajenisbayar | status |
|--------------|----------------|-------------|
| 1 | Tunas Kelate | Wajib |
| 2 | SPP | Wajib |
| 3 | OSIS | Wajib |
| 4 | Pembelian Buku | Tidak Wajib |
| (NULL) | (NULL) | (NULL) |

Gambar 8. Data pada tbjenisbayar

| idkategorikas | namakategorikas | status | idjenisbayar |
|---------------|---------------------------|--------|--------------|
| 1 | Pembelian Transaksi Detak | 0 | (NULL) |
| 2 | Pembelian Transaksi Buku | 1 | (NULL) |
| 3 | Tunas Kelate | 1 | 1 |
| 4 | SPP | 1 | 2 |
| 5 | OSIS | 1 | 3 |
| 6 | Lirik, Air, Telepon | 0 | (NULL) |
| 7 | ATP | 0 | (NULL) |
| 8 | Pembelian Buku | 1 | 4 |
| (NULL) | (NULL) | (NULL) | (NULL) |

Gambar 9. Data pada tbkategorikas

Data Kategori Kas

| No | Nama Kategori Kas | Status | idjenisbayar |
|----|---------------------------|------------|--------------|
| 1 | Pembelian Transaksi Detak | Pembayaran | 0 |
| 2 | Pembelian Transaksi Buku | Pembayaran | 1 |
| 3 | Tunas Kelate | Pembayaran | 1 |
| 4 | SPP | Pembayaran | 2 |
| 5 | OSIS | Pembayaran | 3 |
| 6 | Lirik, Air, Telepon | Pembayaran | 0 |
| 7 | ATP | Pembayaran | 0 |
| 8 | Pembelian Buku | Pembayaran | 4 |

Gambar 10. Tampilan Data Kategori Kas

Otomatisasi ini dilakukan dengan merancang algoritma yang diimplementasikan dalam bentuk kode program sebagai berikut.

```

$insert=mysql_query("Insert into tbjenisbayar SET
namajenisbayar = '".$_POST['namajenisbayar'].",
status = '".$_POST['status'].""");
$idjenisbayar=mysql_query("Select max(idjenisbayar) as id from
tbjenisbayar");
$idjenisbayar=mysql_fetch_array($idjenisbayar);
$idjenisbayar=$idjenisbayar['id'];
$insertkategori=mysql_query("Insert into tbkategorikas SET
namakategorikas = '".$_POST['namajenisbayar'].", status = '1',
idjenisbayar='".$_POST['idjenisbayar'].""");
    
```

Otomatisasi ini mengurangi proses pemasukan data jenis pembayaran siswa ke tabel tbkategorikas melalui form tampilan, sehingga mengurangi tingkat kesalahan pekerjaan dan mempercepat proses pemasukan data.

2. Otomatisasi Pembentukan Kelas

Gambar 11 menunjukkan pemasukan data jumlah kelas oleh pengguna. Pada saat pemasukan data jumlah kelas, ketika pengguna mengetikkan data pada kotak isian dan menekan tombol Simpan, maka data akan tersimpan pada tabel tbjumlahkelas seperti pada Gambar 12. Pada Gambar 13 terlihat tampilan data jumlah kelas dengan tiga tombol yaitu Edit, Hapus dan Proses. Sebelum tombol Proses ditekan, maka proses Edit dan Hapus masih dapat dilakukan. Akan tetapi, jika tombol Proses ditekan, maka tombol Edit dan Hapus akan dikunci.

Penekanan tombol Proses membentuk kelas-kelas secara otomatis sebanyak jumlah kelas yang telah diketikkan dan disimpan pada tabel tbkelas seperti pada Gambar 14.

Gambar 11. Pemasukan Data Jumlah Kelas

| idjumlahkelas | tahunajaran | tingkat | idtipkelas | jumlahkelas |
|---------------|--------------|---------|------------|-------------|
| 1 | 14 2014/2015 | XII | 1 | 2 |
| (NULL) | (NULL) | (NULL) | (NULL) | (NULL) |

Gambar 12. Data pada tbjumlahkelas

Data Jumlah Kelas Tambah Data

| No | Tahun Ajaran | Tingkat | Tipe Kelas | Jumlah Kelas | Simpan | Batal |
|----|--------------|---------|------------|--------------|--------|-------|
| 1 | 2014/2015 | III | IC | 2 | ✓ | ✗ |
| 2 | 2014/2015 | III | IC | 2 | ✓ | ✗ |

Gambar 13. Tampilan Data Jumlah Kelas

| id_kelas | id_jumlah_kelas | nama_kelas | id_val_kelas |
|----------|-----------------|------------|--------------|
| 27 | 15 | XII-TC-1 | (FULL) |
| 28 | 15 | XII-TC-2 | (FULL) |
| (NULL) | (NULL) | (NULL) | (FULL) |

Gambar 14. Data kelas pada tbkelas

Data Kelas

| No | Tahun Ajaran | Tingkat | Tipe Kelas | Jumlah Kelas | Simpan | Batal |
|----|--------------|---------|------------|--------------|--------|-------|
| 1 | 2014/2015 | III | IC | 2 | ✓ | ✗ |
| 2 | 2014/2015 | III | IC | 2 | ✓ | ✗ |

Gambar 15. Data kelas pada tbkelas

Pada tabel tbjumlahkelas terlihat bahwa data jumlah kelas dengan idjumlahkelas=15 untuk Tahun Ajaran 2014/2015 Tingkat III dengan idtipekelas 1 memiliki jumlah kelas 2. Pada tabel tbkelas terlihat bahwa idjumlahkelas=15 muncul sebanyak dua kali karena memiliki jumlahkelas=2. Hubungan ini terjadi antara idjumlahkelas pada tbjumlahkelas dengan idjumlahkelas pada tbkelas. Tampilan data kelas dapat dilihat pada Gambar 15. Otomatisasi ini dilakukan dengan merancang algoritma yang diimplementasikan dalam bentuk kode program sebagai berikut.

```

Stingkat = $_POST['tingkat'];
Sidrikelas = $_POST['idrikelas'];
Sjumlahkelas = $_POST['jumlahkelas'];
$1 = 1;
$kelas = "";
while ($1 <= $jumlahkelas)
{
    $kelas = $tingkat . "-" . $idrikelas . "-" . $1;
    $insert_kelas = mysql_query("insert into tbkelas SET
idjumlahkelas = '" . $POST['idjumlahkelas'] . "',
nama_kelas = '$kelas'");
    $1++;
}
    
```

Otomatisasi ini mengurangi proses pemasukan data kelas ke tabel tbkelas melalui form tampilan, sehingga mengurangi tingkat kesalahan pekerjaan dan mempercepat proses pemasukan data.

3. Otomatisasi Pengaturan Pembayaran Siswa

Gambar 16 menunjukkan pemasukan data pengaturan pembayaran siswa untuk data kelas yang telah dimasukkan. Pada saat pemasukan data, ketika pengguna mengetikkan data pada kotak isian dan menekan tombol Simpan, maka data akan tersimpan pada tabel tbjenisbayarsiswa seperti pada Gambar 17. Pada Gambar 18 terlihat tampilan data pengaturan pembayaran siswa dengan tombol Set Pembayaran, Set Pembayaran Wajib dan Set Pembayaran Siswa. Tombol Pembayaran Siswa harus diproses terlebih dahulu untuk mengatur jenis pembayaran wajib siswa untuk kelas tersebut seperti pada Gambar 19. Data tersimpan pada tabel tbdetailjenisbayarsiswa seperti pada Gambar 20. Ketika tombol Set Pembayaran Wajib dan Set Pembayaran Siswa ditekan, muncul form peringatan agar melengkapi

jenis-jenis pembayaran wajib siswa untuk data kelas tersebut. Penekanan tombol Set Pembayaran Wajib Siswa menyimpan data jenis pembayaran secara otomatis ke semua siswa yang ada di kelas tersebut pada tabel tbsubdetailjenisbayarsiswa seperti pada Gambar 21. Penekanan tombol Set Pembayaran Wajib mengunci tombol Set Pembayaran dan tombol Set Pembayaran Wajib. Penekanan tombol Set Pembayaran Siswa secara otomatis menyimpan data pembayaran siswa ini ke tabel tbbayarsiswa yang akan digunakan pada saat transaksi seperti pada Gambar 22, sekaligus tombol akan terkunci.

Tahun Ajaran : 2014/2015
 Tingkat : XII
 Tahun Mulai : 2014
 Bulan Mulai : Juli

Simpan

Form Tambah Data Setting Pembayaran

Gambar 16. Pemasukan Data Pengaturan Pembayaran Siswa

| id_jenisbayarsiswa | tahun_ajaran | tingkat | tahunawal | bulanawal |
|--------------------|--------------|---------|-----------|-----------|
| 1 | 2014/2015 | XII | 2014 | 7 |

Gambar 17. Data pada tbjenisbayarsiswa

Data Setting Pembayaran Tambah Data

| No | Tahun Ajaran | Tingkat | Tipe Kelas | Jumlah | Set Pembayaran | Set Pembayaran | Set |
|----|--------------|---------|------------|--------|----------------|----------------|-----|
| 1 | 2014/2015 | III | IC | 2 | ✓ | ✗ | ✗ |
| 2 | 2014/2015 | III | IC | 2 | ✓ | ✗ | ✗ |

Gambar 18. Tampilan Data Pengaturan Pembayaran Siswa

Jenis Pembayaran : SPP
 Biaya : 50000

Simpan

Gambar 19. Pemasukan Data Jenis Pembayaran Wajib Siswa

| id_detailjenisbayarsiswa | idjenisbayarsiswa | idjenisbayar | biaya |
|--------------------------|-------------------|--------------|-------|
| 1 | 1 | 1 | 50000 |

Gambar 20. Data pada tbdetailjenisbayarsiswa

| id_subdetailjenisbayarsiswa | id_detailjenisbayarsiswa | id_siswa |
|-----------------------------|--------------------------|----------|
| 1 | 1 | 1 |

Gambar 21. Data pada tbsubdetailjenisbayarsiswa

| id_bayarsiswa | id_detailjenisbayarsiswa | id_siswa | jumlahbayar | id_bayar | id_bayar |
|---------------|--------------------------|----------|-------------|----------|----------|
| 1 | 1 | 1 | 50000 | 1 | 1 |
| 2 | 1 | 2 | 50000 | 1 | 1 |
| 3 | 1 | 3 | 50000 | 1 | 1 |
| 4 | 1 | 4 | 50000 | 1 | 1 |
| 5 | 1 | 5 | 50000 | 1 | 1 |
| 6 | 1 | 6 | 50000 | 1 | 1 |
| 7 | 1 | 7 | 50000 | 1 | 1 |
| 8 | 1 | 8 | 50000 | 1 | 1 |
| 9 | 1 | 9 | 50000 | 1 | 1 |
| 10 | 1 | 10 | 50000 | 1 | 1 |
| 11 | 1 | 11 | 50000 | 1 | 1 |
| 12 | 1 | 12 | 50000 | 1 | 1 |
| 13 | 1 | 13 | 50000 | 1 | 1 |
| 14 | 1 | 14 | 50000 | 1 | 1 |
| 15 | 1 | 15 | 50000 | 1 | 1 |
| 16 | 1 | 16 | 50000 | 1 | 1 |
| 17 | 1 | 17 | 50000 | 1 | 1 |
| 18 | 1 | 18 | 50000 | 1 | 1 |
| 19 | 1 | 19 | 50000 | 1 | 1 |
| 20 | 1 | 20 | 50000 | 1 | 1 |

Gambar 22. Data pada tbbayarsiswa

Dari sisi basis data dilakukan perancangan tabel-tabel dengan field-field yang saling berhubungan. Pada field idjenisbayarsiswa pada tabel tbjenisbayarsiswa berhubungan dengan field idjenisbayarsiswa pada tabel tbdetailjenisbayar siswa, dimana pada tabel ini field idjenisbayar berhubungan dengan field idjenisbayar pada tabel tbjenisbayar. Pada field iddetailjenisbayarsiswa pada tabel tbdetailjenisbayarsiswa berhubungan dengan field

iddetailjenisbayarsiswa pada tabel tbsubdetailjenisbayarsiswa, dimana pada tabel ini field idsiswa berhubungan dengan field idsiswa pada tabel tbsiswa. Otomatisasi ini dilakukan dengan merancang algoritma yang diimplementasikan dalam bentuk kode program sebagai berikut.

```

$queryjenis=mysql_query("Select * from tjenisbayarsiswa,
tbdetailjenisbayarsiswa, tjenisbayar
where tjenisbayarsiswa.idjenisbayarsiswa =
tbdetailjenisbayarsiswa.idjenisbayarsiswa
and tbdetailjenisbayarsiswa.idjenisbayar =
tjenisbayar.idjenisbayar and
tjenisbayarsiswa.idjenisbayarsiswa =
".$POST['idjenisbayarsiswa']."' and
status = 'Wajib'");
while($rowjenis=mysql_fetch_array($queryjenis))
{
    $querysiswa=mysql_query("Select * from tjumlahkelas, tkelas,
tkelasiswa
where tjumlahkelas.idjumlahkelas= tkelas.idjumlahkelas
and
tkelas.idkelas = tkelasiswa.idkelas and
tahunajaran = ".$rowjenis['tahunajaran']."' and tingkat =
".$rowjenis['tingkat']."'");
    $jum=mysql_num_rows($querysiswa);
    if (empty($jum))
    {
        $status=0;
    }
    else
    {
        $status=1;
        while($rowsiswa=mysql_fetch_array($querysiswa))
        {
            $insert=mysql_query("Insert into
tbsubdetailjenisbayarsiswa SET
iddetailjenisbayarsiswa =
".$rowjenis['iddetailjenisbayarsiswa']."'
idsiswa = ".$rowsiswa['idsiswa']."'
");
        }
    }
}

```

Pada field idsubdetailjenisbayarsiswa pada tabel tbsubdetailjenisbayarsiswa berhubungan dengan field idsubdetailjenisbayarsiswa pada tabel tbbayarsiswa. Otomatisasi ini dilakukan dengan merancang algoritma yang diimplementasikan dalam bentuk kode program sebagai berikut.

```

while($row=mysql_fetch_array($query))
{
    $tahun = $row['tahunawal'];
    $sbln = $row['bulanawal'];
    $stglawal = '01';
    for ($si=0; $si<12; $si++)
    {
        $stgl = $tahun.'-'. $sbln.'-'. $stglawal;
        $simpan=mysql_query("Insert into tbbayarsiswa SET
idsubdetailjenisbayarsiswa =
".$row['idsubdetailjenisbayarsiswa']."'
tahunbayar = ".$tahun."',
bulanbayar = ".$sbln."',
statusbayar = '0',
tglaporan = ".$stgl."'");
        $sbln++;
        if ($sbln=='13')
        {
            $sbln='1';
            $tahun++;
        }
    }
}

```

\$i=0;

Otomatisasi ini mengurangi proses pemasukan data jenis pembayaran untuk setiap siswa ke tabel tjenisbayarsiswa, tbdetailjenisbayarsiswa, tbsubdetailjenisbayarsiswa dan tbbayarsiswa melalui form tampilan, sehingga mengurangi tingkat kesalahan pekerjaan dan mempercepat proses pemasukan data.

4. Otomatisasi Pembayaran Siswa dan Perhitungan

Gambar 23 menunjukkan tampilan transaksi pembayaran siswa. Tampilan transaksi pembayaran menampilkan seluruh daftar pembayaran wajib per bulan yang harus dibayar siswa dari kelas X sampai dengan kelas XII. Pada tampilan, tunggakan dibedakan dengan cara memberikan tulisan berwarna merah pada nominal pembayaran, sedangkan pembayaran yang sudah lunas diberikan tulisan berwarna hitam. Di sisi sebelah kanan ada nominal total jumlah pembayaran yang harus dibayar oleh siswa. Pembayaran ini dapat dilakukan untuk satu data saja dengan cara menekan langsung nominal yang mau dibayar, misalnya SPP pada bulan Juli 2013, sehingga secara otomatis mengurangi nilai nominal total yang belum dibayar, seperti terlihat pada Gambar 24.

Rekapitulasi Pembayaran dan Ageri Canggih (14)

Tahun Ajaran : 2013/2014
 Kelas : XI-IP-1
 Wali Kelas : Purta Utamiang

| Tahun | Bulan | Tanggal | Jumlah Bayar | Saldo | Status |
|-------|-----------|------------|--------------|-----------|--------|
| 2013 | Juli | 13.07.2013 | 200.000 | 200.000 | Lunas |
| 2013 | Agustus | 13.08.2013 | 200.000 | 400.000 | Lunas |
| 2013 | September | 13.09.2013 | 200.000 | 600.000 | Lunas |
| 2013 | Oktober | 13.10.2013 | 200.000 | 800.000 | Lunas |
| 2013 | November | 13.11.2013 | 200.000 | 1.000.000 | Lunas |
| 2013 | Desember | 13.12.2013 | 200.000 | 1.200.000 | Lunas |
| 2014 | Januari | 14.01.2014 | 200.000 | 1.400.000 | Lunas |
| 2014 | Februari | 14.02.2014 | 200.000 | 1.600.000 | Lunas |
| 2014 | Maret | 14.03.2014 | 200.000 | 1.800.000 | Lunas |
| 2014 | April | 14.04.2014 | 200.000 | 2.000.000 | Lunas |
| 2014 | Mai | 14.05.2014 | 200.000 | 2.200.000 | Lunas |
| 2014 | Juni | 14.06.2014 | 200.000 | 2.400.000 | Lunas |
| 2014 | Juli | 14.07.2014 | 200.000 | 2.600.000 | Lunas |

Gambar 23. Tampilan Transaksi Pembayaran

| Tahun | Bulan | Tanggal | Jumlah Bayar | Saldo | Status |
|-------|-----------|------------|--------------|-----------|--------|
| 2013 | Juli | 13.07.2013 | 200.000 | 200.000 | Lunas |
| 2013 | Agustus | 13.08.2013 | 200.000 | 400.000 | Lunas |
| 2013 | September | 13.09.2013 | 200.000 | 600.000 | Lunas |
| 2013 | Oktober | 13.10.2013 | 200.000 | 800.000 | Lunas |
| 2013 | November | 13.11.2013 | 200.000 | 1.000.000 | Lunas |

Gambar 24. Tampilan pembayaran satu data

| Tahun | Bulan | Tanggal | Jumlah Bayar | Saldo | Status |
|-------|-----------|------------|--------------|-----------|--------|
| 2013 | Juli | 13.07.2013 | 200.000 | 200.000 | Lunas |
| 2013 | Agustus | 13.08.2013 | 200.000 | 400.000 | Lunas |
| 2013 | September | 13.09.2013 | 200.000 | 600.000 | Lunas |
| 2013 | Oktober | 13.10.2013 | 200.000 | 800.000 | Lunas |
| 2013 | November | 13.11.2013 | 200.000 | 1.000.000 | Lunas |

Gambar 25. Tampilan Pembayaran Total

| Tahun | Bulan | Tanggal | Jumlah Bayar | Saldo | Status |
|-------|-------|------------|--------------|---------|--------|
| 2013 | Juli | 13.07.2013 | 200.000 | 200.000 | Lunas |

Gambar 26. Tampilan Penerimaan Kas

| No | Tanggal | Uraian | Debit | Kredit | Saldo |
|----|---------------|-------------|---------|---------|---------|
| 1 | 22 April 2014 | CSPI | — | 160.000 | 160.000 |
| 2 | 22 April 2014 | Saldo Lunas | 160.000 | — | — |
| 3 | 22 April 2014 | SPP | 200.000 | — | — |

Gambar 27. Tampilan Detail Penerimaan Kas

Dari sisi basis data, hal ini dilakukan dengan cara memperbaharui transaksi dari siswa tersebut pada bulan=Juli, tahun=2013 dan jenisbayar=SPP di tabel tbbayarsiswa. Otomatisasi ini dilakukan dengan merancang algoritma yang diimplementasikan dalam bentuk kode program sebagai berikut.

```

$lnas=mysql_query("Update tbbayarsiswa SET
tglbayar = '$_POST[\"tglbayar\"]';
statusbayar = '1';
iduser = '$_SESSION[iduser]';
tgl = '$_POST[waktu]';
where idbayarsiswa = '$_POST[idbayarsiswa]'");

```

Pembayaran juga dapat dilakukan untuk seluruh data pada satu bulan dengan cara menekan langsung nominal total belum bayar, misalnya nominal 160000 pada baris pertama, sehingga secara otomatis mengganti tulisan nominal menjadi tulisan Lunas, seperti terlihat pada Gambar 25. Dari sisi basis data, hal ini dilakukan dengan cara memperbaharui transaksi dari siswa tersebut pada bulan=Juli dan tahun=2013 di tabel tbbayarsiswa. Otomatisasi ini dilakukan dengan merancang algoritma yang diimplementasikan dalam bentuk kode program sebagai berikut.

```

$lnas=mysql_query("Update tbbayarsiswa SET
tglbayar = '$_POST[\"tglbayar\"]';
statusbayar = '1';
iduser = '$_SESSION[iduser]';
tgl = '$_POST[waktu]';
where idbayarsiswa = '$_SESSION[idbayarsiswa]'");

```

5. Otomatisasi Penerimaan Kas

Transaksi pembayaran siswa yang disimpan ke tabel tbbayarsiswa secara otomatis tersimpan juga ke tabel ttransaksi, sehingga pada tampilan transaksi penerimaan kas, data tersebut akan muncul, seperti pada Gambar 26. Otomatisasi ini dilakukan dengan merancang algoritma yang diimplementasikan dalam bentuk kode program sebagai berikut.

```

$inserttrans=mysql_query("Insert into ttransaksi SET
tanggal = '$_POST[\"tglbayar\"]';
nobukti = '$_POST[nobukti]';
iduser = '$_SESSION[iduser]';
tgl = '$_POST[waktu]';
");

```

Selain itu, data juga tersimpan ke tabel tdetailtransaksi, sehingga pada tampilan detail transaksi penerimaan kas, data tersebut akan muncul. Penyimpanan ini secara otomatis menghitung saldo berdasarkan transaksi yang ada, seperti pada Gambar 27. Otomatisasi ini dilakukan

dengan merancang algoritma yang diimplementasikan dalam bentuk kode program sebagai berikut.

```

$insertdetailtrans=mysql_query("Insert into tdetailtransaksi SET
idtransaksi = '$_SESSION[idtransaksi]';
idkategorikas = '$_SESSION[idkategorikas]';
uraian = '$_POST[uraian]';
jenis = '$_SESSION[jenis]';
biaya = '$_SESSION[biaya]';
saldo = '$_SESSION[saldo]';
iduser = '$_SESSION[iduser]';
tgl = '$_POST[waktu]';
");

```

Aplikasi juga mampu melakukan penyesuaian secara otomatis terhadap saldo. Otomatisasi dilakukan dengan merancang algoritma yang dalam bentuk kode program sebagai berikut.

```

$queryidakhir=mysql_query("Select * from ttransaksi
where idtransaksi > '$_SESSION[idtransaksi]'");
while ($rowidakhir=mysql_fetch_array($queryidakhir))
{
$querydetailakhir=mysql_query("Select * from tdetailtransaksi
where idtransaksi = '$_SESSION[idtransaksi]'");
while ($rowdetailakhir=mysql_fetch_array($querydetailakhir))
{
if($rowdetailakhir[jenis]=='1')
{
$saldo = $saldo+$rowdetailakhir[biaya];
}
else
{
$saldo = $saldo-$rowdetailakhir[biaya];
}
$detail=mysql_query("Update tdetailtransaksi SET
saldo = '$_SESSION[saldo]';
iduser = '$_SESSION[iduser]';
tgl = '$_POST[waktu]';
where iddetailtransaksi = '$_SESSION[iddetailtransaksi]'");
}
}

```

Otomatisasi ini untuk mengantisipasi kesalahan pekerja yang lupa memasukkan transaksi penerimaan atau pengeluaran kas pada suatu hari, sehingga harus melakukan penyisipan data baru. Atau, ketika pekerja melakukan kesalahan pemasukan data, sehingga harus melakukan penghapusan data.

IV. SIMPULAN DAN SARAN

4.1 Simpulan

Dari penelitian yang dilakukan dapat diperoleh kesimpulan bahwa, otomatisasi terhadap rancangan basis data dan algoritma yang diimplementasikan dengan membangun aplikasi menggunakan SQLyog Community Edition, Macromedia Dreamweaver 8 dengan bahasa PHP dan Java Script, menghasilkan sebuah aplikasi yang mampu mengurangi jumlah proses-proses vital yang harus dikerjakan oleh pengguna. Hal ini diharapkan mampu mengurangi tingkat kesalahan pekerjaan dan mempercepat waktu proses yang dilakukan, untuk menjamin keamanan dan meningkatkan pelayanan.

4.2 Saran

Untuk memperoleh hasil yang lebih baik, aplikasi disarankan agar dikembangkan pada ruang lingkup transaksi kas yang lebih kompleks.

DAFTAR PUSTAKA

- [1] Gitosudarmo, Indriyo dan Basri. 2008. Manajemen Keuangan, Edisi keempat. Yogyakarta: BPFE.
- [2] Mulyadi. 2008. Sistem Akuntansi. Jakarta: Salemba Empat.
- [3] Davis, Gordon B. 2002. Kerangka Dasar: Sistem Informasi Manajemen, Bagian I Pengantar. Seri Manajemen No. 90-A. Cetakan Kedua Belas. Jakarta: PT. Pustaka Binawan Pressindo.
- [4] Jogiyanto, H.M. 2005. Analisis dan Disain Sistem Informasi: Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis. Yogyakarta: ANDI Publisher.
- [5] Bodnar. 2005. Sistem Informasi Akuntansi, 8/e Jilid 2. Indeks.
- [6] Adi Nugroho. 2011. Perancangan dan Implementasi Sistem Basis Data. Yogyakarta: ANDI Publisher.
- [7] Abdul Kadir. 2010. Mudah Mempelajari Database MySQL. Yogyakarta: ANDI Publisher.
- [8] LPKBM MADCOMS. 2007. Seri Panduan Lengkap Macromedia Dreamweaver 8. Yogyakarta: ANDI Publisher.
- [9] Arief, Rudyanto, M. 2012. Pemrograman Web Dinamis Menggunakan PHP dan MySQL. Yogyakarta: ANDI Publisher.
- [10] Saputra, Doni. Cahyadi, Dedy. Kridalaksana, Awang Harsa. "Sistem Otomasi Perpustakaan dengan Menggunakan Radio Frequency Identification (RFID)". *Jurnal Informatika Mulawarman*, Volume 5, Nomor 3, September, 2010.
- [11] Hamid. "Pengembangan Sistem Parkir Terkomputerisasi dengan Otomatisasi Pembiayaan dan Penggunaan RFID sebagai Pengenal Unik Pengguna". *Seminar Nasional Aplikasi Teknologi Informatika 2010 (SNATI 2010)*, 19 Juni 2010, Yogyakarta, F-72.
- [12] Syafi'i, Kurniawan, Yunianto, Mochtar, Utami, Yustina Retno Wahyu. "Otomatisasi Pengolahan Data pada Sistem Pelayanan Rental CD Tisanda". *Gaung Informatika*, Volume 4, Nomer 1, 2011.
- [13] Soedjianto, Felicia. "Perancangan dan Pembuatan Sistem Informasi Akuntansi dengan Otomatisasi Pencatatan Jurnal". *Kumpulan Proseding Seminar Nasional MMT-ITS*, 28 Januari 2012.

Sumber Internet:

<http://repository.usu.ac.id/bitstream/123456789/25710/3/Chapter%2011.pdf> (24 juni 2014, 1