

BAB II

TINJAUAN PUSTAKA

2.1. Lanadsan Teori

2.1.1. Sistem

Dalam arti terminologi, sistem berasal dari bahasa Yunani “*system*”, yang artinya “kesatuan”. Sistem dapat terdiri dari beberapa subsistem yang saling berhubungan untuk membentuk satu kesatuan sehingga tujuan atau sasaran sistem dapat dicapai.

Sutrisno (2013 : 21) sistem didefinisikan oleh Frederick H.W.U, Sistem adalah suatu sistem beroperasi dan berinteraksi dengan lingkungannya untuk mencapai sasaran tertentu, suatu sistem menunjukkan tingkah lakunya melalui interaksi diantara komponen-komponen di dalam sistem dan diantara lingkungannya.

Menurut Jogiyanto (2009:34), sistem dapat didefinisikan dengan pendekatan prosedur dan dengan pendekatan komponen.

Menurut Sutarman (2009:5), sistem adalah kumpulan elemen yang saling berhubungan dan saling berinteraksi dalam satu kesatuan untuk menjalankan suatu proses pencapaian suatu tujuan utama.

Menurut Jimmy L.Gaol (2008:9), sistem adalah hubungan satu unit dengan unit - unit lainnya yang saling berhubungan satu sama lainnya dan yang tidak dapat dipisahkan serta menuju satu kesatuan dalam rangka mencapai tujuan yang telah ditetapkan. Apabila suatu unit macet atau terganggu, unit lainnya pun akan terganggu untuk mencapai tujuan yang telah ditetapkan tersebut.

Jadi dapat disimpulkan bahwa sistem adalah suatu ketentuan yang memiliki

beberapa komponen pendukung yang digunakan untuk mencapai tujuan tertentu.

2.1.1.1 Karakteristik Sistem

Menurut Sutabri (2012:20), “Model umum sebuah sistem adalah input, process, output, hal ini merupakan konsep sebuah sistem yang sangat sederhana sebab sebuah sistem dapat mempunyai beberapa masukan dan keluaran.”

Selain itu, sebuah sistem mempunyai karakteristik atau sifat-sifat tertentu yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem. Adapun karakteristik yang dimaksud adalah sbb:

a. Komponen Sistem (*Components*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, artinya saling bekerjasama membentuk satu kesatuan. Komponen-komponen sistem tersebut dapat berupa suatu bentuk subsistem, setiap subsistem memiliki sifat dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan, suatu sistem dapat mempunyai sistem yang lebih besar atau sering disebut “supra sistem”.

b. Batasan Sistem (*Boundary*)

Ruang lingkup sistem merupakan daerah yang membatasi antar sistem dengan sistem yang lain atau sistem dengan lingkungan luarnya, batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisahkan.

c. Lingkungan Luar Sistem (*Environment*)

Bentuk apapun yang ada diluar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut lingkungan luar sistem, lingkungan luar sistem ini dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut. Dengan demikian, lingkungan luar tersebut harus tetap dijaga dan dipelihara, lingkungan luar yang merugikan harus dikendalikan, jika tidak akan mengganggu kelangsungan hidup sistem tersebut.

d. Penghubung Sistem (*Interface*)

Media yang menghubungkan sistem dengan subsistem lain disebut penghubung sistem atau *interface*, penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lain. Bentuk keluaran dari satu subsistem akan menjadi masukan untuk subsistem lain melalui penghubung tersebut, dengan demikian dapat terjadi suatu integrasi sistem yang membentuk suatu kesatuan.

e. Masukkan Sistem (*Input*)

Energi yang dimasukkan ke dalam sistem disebut masukkan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal Input*). Contoh, di dalam suatu unit sistem komputer “program” adalah *maintenance input* yang digunakan untuk mengoperasikan komputernya dan “data” adalah *signal input* untuk diolah menjadi informasi.

f. Keluaran Sistem (*Output*)

Hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna, keluaran ini merupakan masukan bagi subsistem yang lain seperti sistem informasi. Keluaran yang dihasilkan adalah informasi, informasi ini dapat

digunakan sebagai masukan untuk pengambilan keputusan atau hal-hal yang menjadi input bagi subsistem lain.

g. Pengolahan Sistem (*Proses*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran, contohnya adalah sistem akuntansi. Sistem ini akan mengolah data transaksi menjadi laporan-laporan yang dibutuhkan oleh pihak manajemen.

h. Sasaran Sistem (*Objective*)

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat *deterministic*, jika suatu sistem tidak memiliki sasaran maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan.

2.1.1.2. Klasifikasi Sistem

Menurut Sutabri (2012:22), “Sistem merupakan suatu bentuk integrasi antara satu komponen dengan komponen lainnya karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi yang ada di dalam sistem tersebut”. Oleh karena itu, sistem dapat diklasifikasikan dari beberapa sudut pandang, diantaranya:

- a. Sistem abstrak adalah sistem yang berupapemikiran 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.

- b. Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem perputaran bumi, terjadinya siang malam, dan pergantian musim. Sedangkan sistem buatan manusia merupakan sistem yang melibatkan interaksi manusia dengan mesin yang disebut *human machine* sistem. Sistem informasi berbasis komputer merupakan contoh *human machine* sistem karena menyangkut penggunaan komputer yang berinteraksi dengan manusia.
- c. Sistem yang berinteraksi 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 probabilistik adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur *probabilistic*.
- d. 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 sub sistem lainnya.

2.1.1.3. Data

Menurut Supriyadi dkk dalam Jurnal CCIT Vol. 6 No 3 (2013:310), “Data merupakan sekumpulan keterangan atau bukti mengenai sesuatu kenyataan yang masih mentah, masih berdiri sendiri, belum diorganisasikan, dan belum diolah.”

Menurut Sutabri (2012:1), data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Kejadian-kejadian adalah sesuatu yang

terjadi pada saat tertentu di dalam dunia bisnis. Kesatuan nyata adalah suatu objek yang nyata seperti tempat, benda, dan orang yang betul-betul ada dan terjadi.

2.1.2. Informasi

Menurut Agus Mulyanto (2009 : 12), Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya, sedangkan data merupakan sumber informasi yang menggambarkan suatu kejadian yang nyata.

Menurut Jimmy L.Gaol (2008 :8), Informasi adalah data yang telah diproses atau diolah ke dalam bentuk yang berarti untuk penerimanya dan merupakan nilai yang sesungguhnya atau dipahami dalam tindakan atau keputusan yang sekarang atau nantinya.

Menurut Jogiyanto (2008:36), Informasi adalah data yang diolah menjadi bentuk yang berguna bagi para pemakainya.

Sutrisno (2013 : 21) informasi didefinisikan oleh Hartanto yaitu, Informasi adalah sebagai suatu data yang diolah menjadi bentuk yang lebih berguna bagi pemakainya (user). Sumber informasi adalah data.

Dari beberapa definisi di atas, dapat disimpulkan bahwa informasi adalah kumpulan dari beberapa yang telah diolah sehingga menjadi sebuah kesatuan lengkap yang dibutuhkan untuk maksud tertentu.

2.1.2.1. Siklus Informasi

Menurut Sutabri (2012:33) data diolah menjadi suatu model informasi, kemudian si penerima akan menangkap informasi tersebut untuk membuat suatu keputusan dan melakukan tindakan. Tindakan si penerima menjadi sebuah data baru. Data tersebut akan ditangkap sebagai input dan diproses kembali lewat suatu

model, dan seterusnya sehingga membentuk suatu siklus. Siklus inilah yang disebut “Siklus Informasi” (*information Cycle*).

2.1.2.2. Nilai Informasi

Nilai dari informasi (*value of information*) ditentukan dari dua hal, yaitu manfaat dan biaya mendapatkannya.

Menurut Sutabri (2012:38) Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya, akan tetapi perlu diperhatikan bahwa informasi yang digunakan didalam suatu sistem informasi umumnya digunakan untuk beberapa kegunaan.

Sehingga tidak memungkinkan dan sulit untuk menghubungkan suatu bagian informasi pada suatu masalah yang tertentu dengan biaya untuk memperolehnya, karena sebagian besar informasi dinikmati tidak hanya oleh satu pihak didalam perusahaan.

Lebih lanjut sebagian besar informasi tidak dapat persis ditaksir keuntungannya dengan suatu nilai uang, tetapi dapat ditaksir nilai efektifitasnya. Nilai informasi ini didasarkan atas 10 (sepuluh) sifat, yaitu :

a. Mudah diperoleh (*easily obtained*)

Sifat ini menunjukkan informasi dapat diperoleh dengan mudah dan cepat, kecepatan memperoleh dapat diukur misalnya 1 menit versus 24 jam. Akan tetapi, beberapa nilainya bagi pemakai informasi sulit mengukurnya.

b. Luas dan lengkap (*extensive and complete*)

Sifat ini menunjukkan lengkapnya isi informasi, hal ini tidak berarti hanya mengenai volumenya, tetapi juga mengenai keluaran informasinya. Sifat ini sangat kabur, karena itu sulit mengukurnya.

c. Ketelitian (*accuracy*)

Sifat ini menunjukkan minimnya kesalahan dalam informasi, dalam hubungannya dengan volume data yang besar terjadi dua jenis kesalahan, yakni kesalahan pencatatan dan kesalahan perhitungan.

d. Kecocokan (*suitability*)

Sifat ini menunjukkan seberapa baik keluaran informasi dalam hubungan dengan permintaan para pemakai, isi informasi harus ada hubungannya dengan masalah yang sedang dihadapi semua keluaran lainnya tidak berguna tetapi mahal mempersiapkannya, sifat ini sulit mengukurnya.

e. Ketepatan waktu (*timeliness*)

Menunjukkan tak ada keterlambatan jika ada yang sedang ingin mendapatkan informasi masukan, pengolahan, dan pelaporan keluaran kepada pemakai biasanya tepat waktu.

f. Kejelasan (*clarity*)

Sifat ini menunjukkan keluaran informasi yang bebas dari istilah-istilah yang tidak jelas, membetulkan laporan dapat memakan biaya yang besar.

g. Keluwesan (*flexibility*)

Sifat ini berhubungan dengan dapat disesuaikannya keluaran informasi tidak hanya dengan beberapa keputusan, tetapi juga dengan beberapa pengambilan keputusan. Sifat ini sulit diukur, tetapi dalam banyak hal dapat diberikan nilai yang dapat diukur.

h. Dapat dibuktikan (*can be proved*)

Sifat ini menunjukkan kemampuan beberapa pemakai informasi untuk menguji keluaran informasi dan sampai pada kesimpulan yang sama.

i. Tidak ada prasangka (*no prejudice*)

Sifat ini berhubungan dengan tidak adanya keinginan untuk mengubah informasi guna mendapatkan kesimpulan yang telah dipertimbangkan sebelumnya.

j. Dapat diukur (*can be measured*)

Sifat ini menunjukkan hakikat informasi yang dihasilkan dari sistem informasi formal.

2.1.2.3. Kualitas Informasi

Menurut Sutabri (2012:41) Kualitas dari suatu informasi (*quality of information*) tergantung dari tiga hal, yaitu informasi harus akurat (*accurate*), tepat pada waktunya (*time liness*) dan relevan (*relevance*).

a. Akurat (*accurate*)

Informasi harus bebas dari kesalahan-kesalahan dan tidak menyesatkan. Informasi harus akurat karena biasanya dari sumber informasi sampai penerima

informasi ada kemungkinan terjadi gangguan (*noise*) yang dapat mengubah atau merusak informasi tersebut.

b. Tepat Pada Waktunya (*time liness*)

Informasi yang datang pada penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai lagi, karena informasi merupakan landasan didalam pengambilan keputusan. Bila pengambilan keputusan tersebut terlambat, maka dapat berakibat fatal untuk organisasi.

c. Relevan (*relevance*)

Informasi tersebut mempunyai manfaat untuk pemakainya. Relevansi informasi untuk tiap–tiap orang satu dengan yang lainnya berbeda.

2.1.2.4. Sistem Informasi

Sistem informasi adalah kumpulan dari komponen yang saling terkait yang berfungsi bersama untuk mencapai suatu hasil tertentu. Sebuah sistem informasi adalah suatu kumpulan komponen yang saling terkait yang mengumpulkan, memproses, menyimpan, dan menampilkan *output* berupa informasi yang diperlukan (Satzinger, dkk., 2012: 4).

Menurut O'Brien dan Marakas (2013: 6), sistem informasi adalah gabungan yang terorganisasi dari manusia, perangkat keras, perangkat lunak, jaringan komunikasi dan sumber data dalam mengumpulkan, mengubah, dan menyebarkan informasi dalam suatu organisasi.

Sistem informasi adalah salah satu alat utama untuk mencapai keunggulan operasional, mengembangkan produk dan layanan baru, meningkatkan

pengambilan keputusan, dan mencapai keunggulan kompetitif (Laudon, dkk., 2013: 3).

Dari ketiga pendapat di atas dapat disimpulkan bahwa sistem informasi adalah suatu kumpulan komponen yang saling terkait dan terintegrasi dalam proses pengolahan suatu data mulai dari input, proses, kemudian menghasilkan output berupa informasi, dengan tujuan untuk mencapai dan memenuhi kebutuhan pengguna.

Adapun 5 komponen sistem informasi berdasarkan buku *Introduction to Information Systems* yang dibuat oleh O'Brien (2013:29-33) yaitu sebagai berikut:

1. Manusia (pemakai akhir): terdiri dari orang-orang yang memakai sistem informasi atau informasi yang dihasilkan sistem tersebut seperti pelanggan, *manager*, bagian administrasi, maupun akuntan.
2. *Software* (program dan prosedur): meliputi semua prosedur dan program *software* yang digunakan selama proses input data, pengolahan data, dan *output* hasil pengolahan data.
3. *Hardware* (mesin dan media): meliputi semua perangkat fisik yang digunakan selama pemrosesan informasi. Contohnya seperti sistem komputer *desktop*, *laptop*, *printer*, *keyboard*, *mouse*, *hard disk*.
4. *Database* (dasar data dan pengetahuan): merupakan kumpulan dari data-data yang diinput untuk kemudian diproses menjadi informasi sesuai kebutuhan pengguna sistem. Data merupakan sumber daya informasi paling berharga sebab tanpa data pengguna sistem tidak bisa mendapatkan informasi yang diinginkan.

5. Jaringan media komunikasi dan dukungan jaringan teknologi: jaringan komunikasi seperti internet, intranet dan extranet menjadi dasar pengguna sistem untuk berkomunikasi. Adapun contoh dari jaringan adalah modem, kabel optic, viber, satelit, dan lain – lain.

Suatu sistem informasi dibuat untuk suatu keperluan tertentu atau untuk memenuhi permintaan penggunaan tertentu, maka struktur dan cara kerja sistem informasi berbeda-beda bergantung kepada keperluan dan permintaan yang harus dipenuhi, oleh karena kepentingan yang harus dilayani sangat beraneka ragam, maka sistem informasipun semakin beraneka ragam. Pengembangan sistem dapat berarti menyusun sistem yang baru untuk menggantikan sistem yang lama secara keseluruhan atau untuk memperbaiki sistem yang sudah ada. Sistem yang sudah lama perlu diperbaiki atau bahkan diganti, disebabkan oleh beberapa hal, diantaranya yaitu:

1. Kesalahan yang tidak sengaja, yang menyebabkan kebenaran data kurang terjamin.
2. Kurang efisiensinya operasi pengolahan data tersebut.
3. Adanya instruksi-instruksi atau kebijaksanaan yang baru baik dari pemimpin atau dari luar organisasi seperti peraturan pemerintah.

Dengan sistem informasi, masyarakat jadi lebih mudah untuk memperoleh informasi dengan cepat. Perkembangan sistem informasi pun dari tahun ke tahun

berkembang semakin cepat, seiring dengan perkembangan teknologi saat ini,

2.1.3. Pembayaran

Menurut Tirto Waluyo (2010:1) pembayaran adalah suatu tindakan menukarkan sesuatu uang atau barang dengan maksud dan tujuan yang sama yang dilakukan oleh dua orang atau lebih.

Dalam kamus besar bahasa Indonesia definisi pembayaran adalah : pembayaran proses, cara, perbuatan membayar.

2.1.4. SPP

SPP (Sumbangan Pembinaan Pendidikan) merupakan iuran wajib bagi siswa atau siswi yang dipergunakan oleh pihak sekolah untuk memfasilitasi segala kegiatan pembelajaran yang dilakukan siswa/siswi, dengan waktu pembayaran ditentukan sebelumnya (Nur, 2010)

2.1.5. Pembayaran SPP

Pengertian pembayaran SPP adalah proses membayar SPP yang dilakukan berulang-ulang, sekali dalam satu bulan. SPP merupakan iuran rutin sekolah yang mana pembayarannya dilakukan setiap sebulan sekali. SPP merupakan salah satu bentuk kewajiban setiap siswa yang masih aktif disekolah tersebut.

2.1.6. Sistem Informasi Pembayaran SPP

Sistem informasi pembayaran spp adalah suatu kumpulan komponen yang saling terkait dan terintegrasi dalam proses pengolahan pembayaran spp mulai dari input nis, proses pembayaran, kemudian menghasilkan output berupa informasi laporan pembayaran, dengan tujuan untuk mengelola pembayaran SPP di sekolah.

2.1.7. Barcode

Menurut Jaja Jamaludin (2010, p.2) , *barcode* merupakan sejenis kode yang mewakili data atau informasi tertentu, biasanya jenis dan harga barang , contohnya seperti untuk makanan dan buku. *Barcode* berbentuk batangan balok dan berwarna hitam putih ini mengandung satu kumpulan kombinasi batang yang berlainan ukuran yang disusun sedemikian rupa.

2.1.7.1 Manfaat *barcode*

Menurut Jaja Jamaludin (2010, p.2) *barcode* memiliki banyak manfaat yang beraneka ragam, diantaranya :

1. Akurasi

Barcode mampu meningkatkan akurasi dengan mengurangi kesalahan manusia ketika melakukan pemasukan data secara manual atau barang yang salah baca.

2. Kemudahan pemakaian

Dengan perangkat keras dan perangkat lunak yang tepat, *barcode* bisa memaksimalkan proses otomatisasi pengumpulan data.

3. Keseragaman pengumpulan data

Beragam standar pemenuhan dan simbologi *barcode* yang terstandarisasi menjamin informasi diterima dan disampaikan dengan cara yang benar sehingga dapat diterima dan dipahami secara umum.

4. Umpan balik yang tepat waktu

Barcode menawarkan umpan balik yang tepat waktu. Begitu muncul, data bisa diterima dengan cepat sehingga memungkinkan pengambilan keputusan yang cepat berdasarkan informasi terbaru.

5. Keamanan

Dengan menggunakan *barcode*, perusahaan mampu memperkecil kesalahan yang terjadi. Misalnya penukaran harga label produk yang dilakukan konsumen di suatu supermarket.

6. Meningkatkan produktivitas

Barcode mampu membuat aktivitas operasional dalam bisnis menjadi lebih cepat sehingga produktivitas dapat meningkat.

7. Meningkatkan profit

Peningkatan efisiensi dan produktivitas yang diberikan *barcode* memungkinkan perusahaan menghemat biaya dan waktu, sehingga profit perusahaan meningkat.

2.1.7.2 Jenis-jenis *barcode*

Menurut Teguh Wahyono (2010, p.6) *barcode* memiliki banyak sekali standard dan tidak terpaku pada satu model tertentu. Tetapi meski demikian, secara umum *barcode* terbagi menjadi dua kelompok besar, yaitu *barcode* satu dimensi dan *barcode* dua dimensi.

1. Barcode satu dimensi

barcode ini dinamakan satu dimensi atau ada juga yang menyebut *linear barcodes* karena kodenya hanya terdiri dari baris- baris. Berikut ini adalah beberapa jenis *barcode* satu dimensi :

a. Tipe code 39

Merupakan *barcode alphanumeric* (full ASCII) yang dapat mewakili abjad (A-Z) dan angka (0-9), serta beberapa karakter lain, seperti \$, /, +, % titik dan spasi dan juga memiliki jumlah digit maksimal 16. Kode seperti ini biasanya di gunakan untuk *barcode* buku maupun untuk *barcode* anggota perpustakaan. Dan juga beberapa aplikasi lain seperti untuk inventori, *asset tracking* dan digunakan pada tanda pengenalan identitas. Seperti yang ditunjukkan pada gambar 2.2 dibawah ini :



BC 1 3 4

Gambar 2.1. Barcode 1 dimensi tipe kode 39

b. Tipe Code 128

Barcode tipe code 128 juga tergolong di dalam *barcode alphanumeric* (full ASCII), tetapi memiliki kerapatan yang lebih tinggi dan juga panjang baris yang bervariasi. *Barcode* Tipe Code 128 biasanya di gunakan untuk aplikasi, seperti pengaturan maskapai pelayaran dan pengelolaan gudang.

Setiap karakter pada code 128 47 dikodekan oleh 3 bar dan 3 spasi (atau 6 elemen) dengan ketebalan masing-masing elemen 1 sampai 4 kali ketebalan minimum (module). . Seperti yang ditunjukkan pada gambar 2.3 dibawah ini :



Gambar 2.2. *Barcode* 1 dimensi tipe kode 128

c. Code 25 (*Interleaved*)

Merupakan kode *barcode* yang hanya untuk angka (0-9), maksimum 32 digit. Jadi *barcode* ini berbentuk numerik dan memiliki panjang baris yang bervariasi. *Barcode* yang juga disebut sebagai *interleaved 2 of 5* biasa digunakan untuk aplikasi dalam dunia industry dan laboratorium. . Seperti yang ditunjukkan pada gambar 2.4 dibawah ini :



Gambar 2.3. *Barcode* 1 dimensi tipe kode 25

d. EAN 13

Simbologi *barcode* model ini dikeluarkan EAN untuk identitas suatu produk. Standarisasi EAN menggunakan 3 digit pertama adalah untuk kode Negara asal produk, 4 digit berikutnya adalah *Manufacture Number*, 5 digit berikutnya adalah kode produk atau nomor urut produk sedangkan 1 digit terakhir adalah cek digit atau angka untuk melakukan tes validasi *barcode*.

Kode EAN sering digunakan di Indonesia untuk identifikasi produk nasional. Seperti yang ditunjukkan pada gambar 2.5 dibawah ini :



Gambar 2.4. *Barcode* 1 dimensi tipe kode EAN 13

e. UPC (*Universal Product Code*)

Barcode UPC terdiri dari angka (0-9) namun *barcode* harus mempunyai panjang tepat 11 atau 12 digit, Kurang atau lebih dari angka tersebut tidak bisa digunakan. Jadi *barcode* ini berbentuk numerik dan memiliki panjang baris yang tetap. UPC biasanya digunakan untuk pelabelan pada produk- produk kecil atau eceran. Simbol UPC dibuat untuk kemudahan pemeriksaan keaslian suatu produk dan bilangan UPC harus diregistrasikan atau terdaftar di *Uniform Code Council*. Seperti yang ditunjukkan pada gambar 2.6 dibawah ini :



Gambar 2.5. *Barcode* 1 dimensi tipe kode UPC

2. *Barcode* dua dimensi

Barcode dua dimensi adalah *barcode* yang tidak hanya terbuat dari garis-garis saja, tetapi lebih mendekati bentuk gambar tertentu. Dengan menggunakan dua dimensi, informasi atau data yang besar dapat di simpan di

dlaam suatu ruang yang lebih kecil. Berikut ini adalah beberapa jenis *barcode* dua dimensi :

a. QR code

QR merupakan kependekan dari *Quick Response*, sebuah harapan dari pembuatnya bahwa code ini akan cepat di-decode. Tidak seperti *barcode* yang satu sisinya saja yang mengandung data, QR code mempunyai dua sisi yang berisi data. Hal tersebut membuat QR code dapat lebih banyak memuat informasi dibandingkan *barcode*.

b. Kode PDF417

Model ini merupakan pengembangan dari *barcode* satu dimensi linear, tepatnya disebut sebagai *stacked linear barcode*.PDF sendiri di artikan sebagai *Portable Data File*. Model ini memiliki kemampuan untuk menyimpan lebih dari 2000 karakter di dalam ruang berukuran 4 inch persegi. Seperti yang ditunjukkan pada gambar 2.8 dibawah ini :

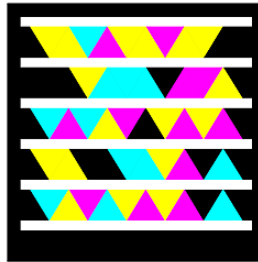


Gambar 2.6. *Barcode* 2 dimensi tipe PDF417

c. HCCB (*High Capacity Color Barcode*)

Contoh lain dari *barcode* dua dimensi adalah HCCB yang mana didevelop oleh Microsoft. Microsoft mengimplementasikan HCCB menggunakan 4 warna pada ukuran 5x10 *grid*.

Dengan *barcode* HCCB data akan mampu disimpan dalam ukuran yang lebih besar. Selain itu tampilan kode juga lebih enak untuk dilihat karena menggunakan tampilan berwarna(*color barcode*). Seperti yang ditunjukkan pada gambar 2.9 dibawah ini :



Gambar 2.7. Barcode 2 dimensi tipe HCCB

d. *Data Matrix*

Data Matrix adalah *barcode* dua dimensi yang terdiri dari sel hitam dan putih atau modul titik-titik di sekitar *rectangular patern*. *Data matrix* menggunakan ukuran data sampai pada 2 kilobytes. Sebuah *data matrix* dapat menyimpan sampai dengan 2.335 karakter *alphanumeric*.

Salah satu contoh penggunaan *data matrix* ini yang sering dijumpai seperti *barcode* pada *Mini PCI Card* dengan ukuran symbol yang bervariasi mulai dari 8x8 sampai 144x144 seperti yang ditunjukkan pada gambar 2.10 dibawah ini :



Gambar 2.8. *Barcode 2 dimensi tipe Data Matrix*

2.1.8. Pengembangan Perangkat Lunak

Dalam proses pengembangan perangkat lunak terdapat beberapa tahapan atau proses yang harus dilalui atau disebut juga dengan SDLC (Software Development Life Cycle). Menurut Rosa dan Salahuddin (2013:26), SDLC merupakan proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya (berdasarkan best practice atau cara-cara yang sudah teruji dengan baik). SDLC memiliki beberapa model diantaranya yaitu model Waterfall, Prototipe, RAD (Rapid Application Development), Iteratif dan Spiral (Rosa & Salahuddin, 2013).

2.1.8.1. Waterfall

Model Waterfall menyediakan pendekatan pengembangan perangkat lunak yang sistematis dan sekuensial yang mulai pada tingkat dan kemajuan sistem pada analisis, desain, pengodean, pengujian, dan pemeliharaan (Pressman, 2001:37).

Tahap-tahap dalam model Waterfall menurut Rosa & Salahuddin (2013:29-30) antara lain:

1. Analisis kebutuhan perangkat lunak

Merupakan proses pengumpulan kebutuhan yang dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak sesuai kebutuhan user. Spesifikasi perangkat lunak pada tahap ini perlu didokumentasikan.

2. Desain

Desain perangkat lunak merupakan proses multi langkah yang berfokus pada struktur data, arsitektur perangkat lunak, representasi interface, dan prosedur pengkodean. Tahap ini mentranslasi kebutuhan dari tahap analisis agar dapat diimplementasikan.

3. Pembuatan kode program

Mentranslasikan desain menjadi kode program. Hasil dari tahap ini adalah program komputer yang sesuai dengan desain.

4. Pengujian

Pengujian fokus pada perangkat lunak dari segi logika dan fungsional serta memastikan bahwa semua bagian sudah diuji untuk meminimalisir kesalahan.

5. Pemeliharaan

Tahap ini dilakukan jika terjadi perubahan ketika perangkat lunak dikirim ke user. Tahap pemeliharaan dapat mengulangi proses pengembangan dari mulai analisis kebutuhan, akan tetapi hanya untuk perubahan perangkat lunak yang sudah ada, bukan untuk membuat perangkat lunak baru.

Dari kenyataan yang terjadi sanagat jarang model air terjun dapat dilakukan sesuai alurnya karena sebab berikut :

- a. Perubahan sepesifikasi perangkat lunak terjadi di tengah alur pengembangan
- b. Sangat sulit bagi pelanggan untuk mendefinisikan semua spesifikasi di awal alur pengembangan. Pelanggan sering kali butuh contoh (*prototype*) untuk menjabarkan spesifikasi kebutuhan lebih lanjut.
- c. Pelanggan tidak mungkin bersabar mengakomodasi perubahan yang diperlukan di akhir alur pengembangan

Dengan berbagai kelemahan yang dimiliki model *waterfall* tapi model ini telah menjadi dsar dari model-model yang lain dalam melakukan perbaikan model pengembangan perangkat lunak.

Model *waterfall* sangat cocok digunakan apabila kebutuhan dari user sudah dipahami dan perangkat lunak yang akan dikembangkan spesifikasinya tidak berubah-ubah (Rosa dan Salahuddin, 2013:29). Selain itu, kelebihan dari model *waterfall* menurut Rosa dan Salahuddin (2013:30-31) yaitu struktur dan pengembangan sistem jelas, dokumentasi dihasilkan di setiap tahap pengembangan dan tidak ada tumpang tindih pelaksanaan tahap.

2.1.9. UML

Menurut Pressman (2010:841) *Unified Modeling Language* atau disingkat UML merupakan bahasa standar yang digunakan untuk memvisualisasikan, menspesifikasikan, menkonstruksikan, serta mendokumentasikan sebuah sistem *software*. Sementara menurut Whitten dan Bentley (2007:381) *Unified Modeling*

Language (UML) adalah suatu kumpulan konvensi pemodelan untuk menentukan atau menggambarkan suatu sistem piranti lunak yang berhubungan dengan objek.

UML merupakan metode analisis yang lebih baru dari DFD yang lebih menitik beratkan analisisnya di sisi pengguna atau *actor* atau orang-orang di luar sistem yang terlibat dengan sistem.

Whitten dan Bentley (2007:382) menyatakan bahwa UML dideskripsikan oleh beberapa diagram, diantaranya:

1. Fase Analisis Kebutuhan:
 - a. *Use Case Diagram*

2. Fase Desain Logis:
 - a. *Activity Diagrams*
 - b. *System Sequence Diagrams*
 - c. *Class Diagrams*

3. Fase Desain Fisik:
 - a. *Sequence Diagrams*
 - b. *Class Diagrams*
 - c. *State Machine Diagrams*
 - d. *Communication Diagrams*
 - e. *Component Diagrams*
 - f. *Deployment Diagrams*

Berdasarkan pengelompokan diagram UML di atas, penulis hanya menjelaskan beberapa diagram UML:

2.1.9.1. *Use Case Diagram*

Menurut Whitten dan Bentley (2007:246) *use case diagram* dipakai untuk menggambarkan relasi antara sistem dan sistem eksternal dan *user*, dengan kasus yang disesuaikan dengan langkah-langkah yang telah ditentukan. *Use Case Diagram* merupakan cara / metode yang dapat menggambarkan interaksi yang jelas antara sistem dengan pengguna.

a. *Use Case*

Use case mendeskripsikan fungsi dari sebuah sistem dilihat dari sudut pandang pengguna (Whitten dan Bentley, 2007:246).



Gambar 2.9. Use Case

b. *Actor*

Actor merupakan sesuatu yang berinteraksi dengan sistem untuk saling bertukar informasi (Whitten dan Bentley, 2007:247). *Actor* tidak harus berupa manusia, tetapi dapat berupa suatu organisasi atau sistem informasi.



Gambar 2.10. Actor

c. *Relationship*

Sebuah relasi antar sistem dan sistem atau *user* dan sistem digambarkan dengan sebuah garis di antara keduanya. Arti relasi yang digambarkan bisa dengan tergantung pada bagaimana garis itu digambarkan dan apa yang mereka hubungkan. Ada beberapa macam relasi, antara lain *associations*, *extends*, dan *users*.

1. *Associations*

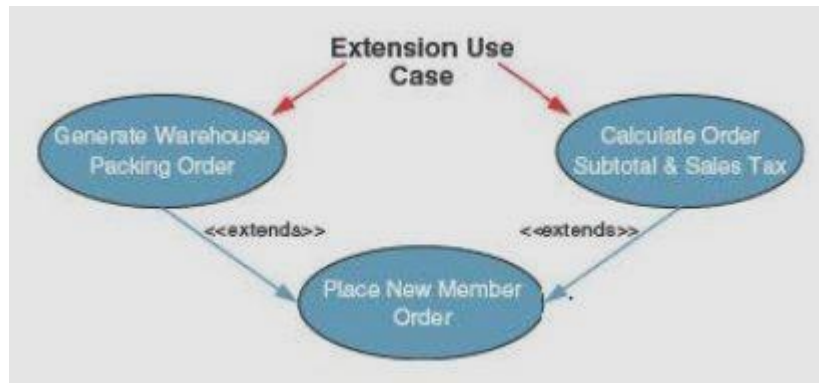
Associations adalah sebuah relasi antara seorang *actor* dengan sebuah *use case* dimana terjadi interaksi antar mereka (Whitten dan Bentley, 2007:248). Asosiasi dengan panah tertutup di ujung yang menyentuh *use case* mengindikasikan bahwa *actor* di ujung yang satu lagi melakukan *use case* tersebut. Sedangkan asosiasi tanpa panah mengindikasikan sebuah interaksi dari *use case* ke *actor* yang menerima hasil dari *use case* tersebut.



Gambar 2.11. Associations dalam Use Case Diagram

2. *Extends*

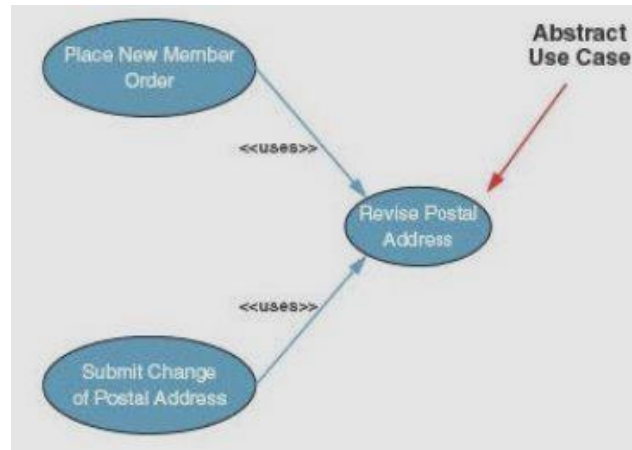
Extends bertujuan untuk menyederhanakan *use case* dengan fungsionalitas yang kompleks seperti beberapa langkah yang perlu dilakukan menjadi lebih mudah dipahami (Whitten dan Bentley, 2007:249).



Gambar 2.12. *Extends* dalam *Use Case Diagram*

3. *Uses (or Include)*

Uses bertujuan untuk mengurangi redundansi di antara dua *use case* atau lebih dengan menggabungkan langkah-langkah yang sama tersebut (Whitten dan Bentley, 2007:249).



Gambar 2.13. *Uses dalam Use Case Diagram*

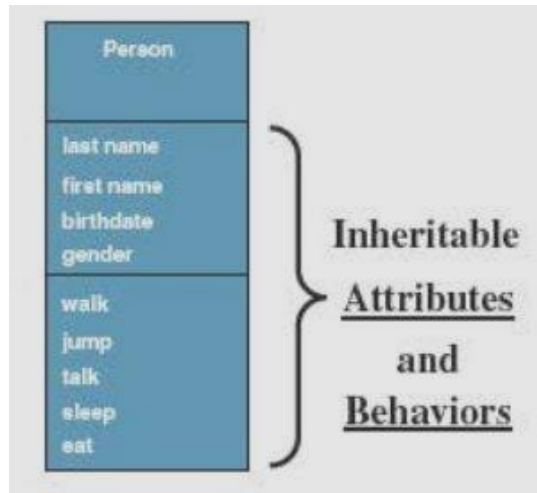
2.1.9.2. *Class Diagram*

Menurut Whitten dan Bentley (2007:400), *class diagram* digunakan untuk menggambarkan struktur objek statis dalam sebuah sistem, menunjukkan sistem tersusun dari kelas-kelas apa saja dan hubungan apa saja yang terbentuk di antara kelas tersebut.

Hubungan yang terbentuk bisa berupa *associations*, *aggregation*, atau *composition*. *Associations* adalah konsep hubungan antar kelas objek yang terkadang memiliki batasan minimal dan maksimal untuk jumlah objek yang terbentuk dari masing-masing kelas. *Aggregation* merupakan hubungan dimana satu kelas “*whole*” mengandung satu atau lebih banyak kelas “*part*”. Sedangkan *composition* merupakan hubungan *aggregation* dimana kelas “*whole*” bertanggung jawab untuk masa aktif kelas “*part*” miliknya.

Class diagram di atas terbentuk dari beberapa notasi, antara lain *attribute*, *behavior*, *arrow head*, dan *solid diamond*.

1. *Attribute dan Behavior*



Gambar 2.14. Attribute and Behavior

2. Arrow Head

Arrow head melambangkan generalisasi atau spesialisasi dari sebuah relasi (Whitten dan Bentley, 2007:376).



Gambar 2.15. Arrow Head

3. Solid Diamond

Solid diamond melambangkan komposisi agregasi dari sebuah relasi (Whitten dan Bentley, 2007:379).



Gambar 2.16. *Solid Diamond*

2.1.9.3. *Activity Diagram*

Menurut Whitten dan Bentley (2007:391), *activity diagram* merupakan gambaran dari alur yang berurutan dari aktivitas *use case* atau proses bisnis. *Activity Diagram* juga bisa dipakai untuk memodelkan berbagai aksi yang dilakukan saat sebuah operasi dieksekusi, dan memodelkan hasil dari aksi tersebut. Dari diagram ini, kita dapat melihat bagaimana aktivitas dalam suatu sistem, dari mulai hingga saat sistem berakhir.

Activity diagram dibentuk oleh beberapa notasi, antara lain *initial*, *actions*, *flow*, *decision*, *merge*, *fork*, *join*, dan *activity final*, dan terkadang digunakan *swimlane* untuk mempertisi aksi yang terjadi berdasarkan pelaku.

1. *Initial Node*

Initial node berupa lingkaran penuh yang menggambarkan titik mulai suatu proses (Whitten dan Bentley, 2007:392).



Gambar 2.17. *Initial Node*

2. *Actions*

Actions adalah notasi segiempat bersudut tumpul yang menggambarkan langkah-langkah yang terjadi (Whitten dan Bentley, 2007:392).



Gambar 2.18. Actions

3. *Flow*

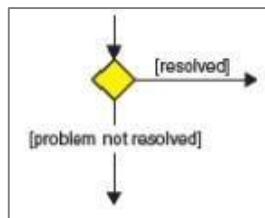
Flow (alur) merupakan panah dalam diagram yang mengindikasikan alur antar *actions* (Whitten dan Bentley, 2007:392).



Gambar 2.19. Flow

4. *Decision*

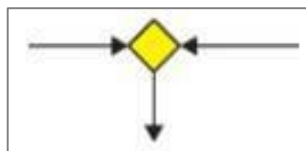
Decision memiliki bentuk seperti wajik dengan satu alur masuk dan dua atau lebih alur keluar, alur keluar ditentukan dengan kondisi tertentu (Whitten dan Bentley, 2007:392).



Gambar 2.20. Decision

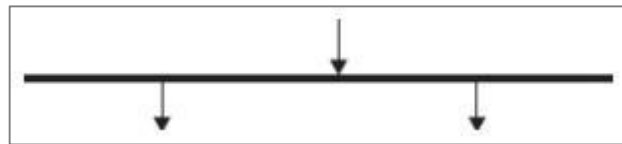
5. *Merge*

Merge adalah wajik dengan dua atau lebih alur masuk dan satu alur keluar untuk menggabungkan alur yang sebelumnya terpisah oleh *decision* (Whitten dan Bentley, 2007:392).

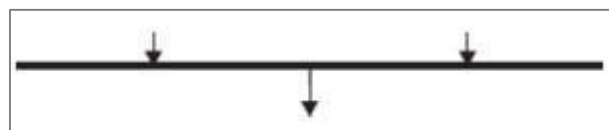


Gambar 2.21. Merge**6. Fork**

Fork adalah bar hitam dengan satu alur masuk dan dua atau lebih alur keluar, aksi di bawah percabangan dapat terjadi dalam urutan apapun atau bahkan secara bersamaan (Whitten dan Bentley, 2007:392).

**Gambar 2.22. Fork****7. Join**

Join adalah bar hitam dengan dua atau lebih alur masuk dan satu alur keluar untuk menyatukan lagi alur aksi yang dipisahkan oleh *fork* (Whitten dan Bentley, 2007:392).

**Gambar 2.23. Join**

8. *Activity Final*

Activity final berbentuk lingkaran penuh dengan satu lingkaran di luarnya untuk menggambarkan titik akhir proses (Whitten dan Bentley, 2007:393).



Gambar 2.24. *Activity Final*

2.1.10. *Database Management System*

Kristanto (1994:2) mengemukakan bahwa “*Database Management System* (DBMS) berisi satu koleksi data yang saling berelasi dan satu set program untuk mengakses data tersebut”. Jadi DBMS terdiri dari *database* dan *set* program pengelola untuk menambahkan data, menghapus data, mengambil dan membaca data.

“*Database* adalah kumpulan *file-file* yang saling berelasi, relasi tersebut biasa ditunjukkan dengan kunci dari tiap *file* yang ada” (Kristanto, 1994:2). Satu *database* menunjukkan satu kumpulan data yang dipakai dalam satu lingkup perusahaan / instansi.

Dalam satu *file* terdapat *record-record* yang sejenis, sama besar, sama bentuk, merupakan satu kumpulan *entity* yang seragam. Satu *record* terdiri dari *field-field* yang saling berhubungan untuk menunjukkan bahwa *field* tersebut dalam satu pengertian yang lengkap dan direkam dalam satu *record*.

Untuk menyebut isi dari *field* maka digunakan *attribute* atau merupakan judul dari satu kelompok *entity* tertentu, misalnya *attribute* alamat menunjukkan *entity* alamat dari siswa. *Entity* adalah suatu objek yang nyata dan akan direkam.

Beberapa *software* atau perangkat lunak DBMS yang sering digunakan dalam aplikasi program antara lain adalah DB2, Microsoft SQL Server, Oracle, Sybase, Interbase, Teradata, Firebird, MySQL, dan PostgreSQL.

Dari beberapa *software* DBMS di atas, penulis menggunakan MySQL sebagai perangkat lunak untuk menyimpan informasi atau data dari aplikasi yang akan dibuat. “MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: *database management system*) atau DBMS yang *multithread*, dan *multiuser*” (Solichin, 2005:85). MySQL merupakan perangkat lunak yang bersifat *open source* dengan bahasa standar yang digunakan adalah SQL. Saat ini MySQL banyak digunakan untuk membangun aplikasi-aplikasi *web* yang menggunakan *database*, karena MySQL memiliki kinerja, kecepatan proses dan ketangguhan yang tidak kalah dibanding *database-database* besar lainnya yang komersil.

Beberapa keunggulan yang dimiliki oleh MySQL, yaitu *open source / gratis*, fleksibel dengan berbagai pemrograman, *security* yang baik, kemudahan *management database*, mendukung transaksi, perkembangan *software* yang cukup cepat, dan dapat dijalankan di beberapa sistem operasi (Solichin, 2005:85).

2.1.11. User Acceptance Tes (UAT)

Menurut Perry (2006:70), *User Acceptance Testing* merupakan pengujian yang dilakukan oleh *end-user* dimana *user* tersebut adalah *staff/* karyawan

perusahaan yang langsung berinteraksi dengan sistem dan dilakukan verifikasi apakah fungsi yang ada telah berjalan sesuai dengan kebutuhan/fungsinya.

Menurut Lewis (2009:134), setelah dilakukan *system testing*, *acceptance testing* menyatakan bahwa sistem *software* memenuhi persyaratan. *Acceptance testing* merupakan pengujian yang dilakukan oleh pengguna yang menggunakan teknik pengujian *black box* untuk menguji sistem terhadap spesifikasinya. Pengguna akhir bertanggung jawab untuk memastikan semua fungsionalitas yang relevan telah diuji.

Menurut Black (2002:7), *acceptance testing* biasanya berusaha menunjukkan bahwa sistem telah memenuhi persyaratan-persyaratan tertentu. Pada pengembangan *software* dan *hardware* komersial, *acceptance test* biasanya disebut juga "*alpha tests*" (yang dilakukan oleh pengguna *in-house*) dan "*beta tests*" (yang dilakukan oleh pengguna yang sedang menggunakan atau akan menggunakan sistem tersebut). *Alpha* dan *beta test* biasanya juga menunjukkan bahwa produk sudah siap untuk dijual atau dipasarkan. *Acceptance testing* mencakup data, *environment* dan skenario yang sama atau hampir sama pada saat *live* yang biasanya berfokus pada skenario penggunaan produk tertentu.

Dari definisi di atas, *user acceptance testing* adalah pengujian yang dilakukan oleh pengguna dari sistem tersebut untuk memastikan fungsi-fungsi yang ada pada sistem tersebut telah berjalan dengan baik dan sesuai dengan kebutuhan pengguna.

2.2. Penelitian Yang Relevan

Untuk menghasilkan suatu sistem yang baik dan sesuai kebutuhan pengguna, maka penulis melakukan tinjauan pustaka dari beberapa penelitian sebelumnya yang sudah ada.

1. Sistem Komputerisasi Pembayaran SPP SD Negeri Selomulyo Sukoharjo Ngaglik Sleman Yogyakarta.

Tugas Akhir oleh Akhirowati (2006), mahasiswa Universitas Teknologi Yogyakarta dengan judul “Sistem Komputerisasi Pembayaran SPP SD Negeri Selomulyo Sukoharjo Ngaglik Sleman Yogyakarta” mencantumkan data inputan yakni data siswa, data guru, setup bayar, transaksi pembayaran SPP dan memiliki outputan antara lain laporan pembayaran SPP, laporan berdasarkan bulan, laporan belum bayar, laporan data guru, laporan data siswa, laporan data setup bayar.

2. Sistem Informasi Pembayaran Pada SMK Muhammadiyah Kalibawang

Tugas Akhir oleh Sulistyono (2010), mahasiswa Universitas Teknologi Yogyakarta dengan judul “Sistem Informasi Pembayaran Pada SMK Muhammadiyah Kalibawang” dengan data inputan meliputi data siswa, data guru, transaksi pembayaran SPP, transaksaksi pembayaran uang gedung dengan outputan laporan data guru, laporan data biaya spp, laporan data siswa dan laporan pembayaran uang gedung.

3. Perancangan dan Implementasi Sistem Informasi Pembayran Biaya Sekolah berbasis Client-Server

Tugas Akhir mengenai sistem pembayaran biaya sekolah oleh Eva Sulasmi (2012) yang mengambil judul “Perancangan dan Implementasi Sistem Informasi Pembayaran Biaya Sekolah berbasis Client-Server”. Pada penelitian ini membahas mengenai alokasi pendanaan *incoming* dan *outgoing* pembayaran administrasi sekolah perbulannya.

4. Rancang Bangun Sistem Pengolahan Data Santri Dan Pembayaran SPP Berbasis Dekstop Dan SMS Gateway

Penelitian Kerja Praktik yang dilakukan oleh Khasbuloh (2013) mahasiswa Teknik Informatika, Universitas Teknologi Yogyakarta, yang dalam penelitiannya mengambil judul “Rancang Bangun Sistem Pengolahan Data Santri Dan Pembayaran SPP Berbasis Dekstop Dan SMS Gateway” (Studi kasus di Pondok Pesantren Islam Salafiah Al Falah).

5. Membangun Aplikasi Pembayaran Keuangan Siswa Di SMK Ma’arif 1 Wates Yogyakarta

Laporan Kerja Praktik mengenai pembangunan aplikasi pembayaran keuangan siswa oleh Sugeng Triyanto (2013), mahasiswa Universitas Teknologi Yogyakarta yang mengambil judul “Membangun Aplikasi Pembayaran Keuangan Siswa Di SMK Ma’arif 1 Wates Yogyakarta”. Pada penelitian ini terdapat inputan data petugas, data siswa, data keuangan, data kelas, dan data inputan pembayaran siswa, serta memiliki outputan berupa laporan kekurangan pembayaran siswa, laporan pembayaran per-periode, laporan pembayaran per-kelas dan bukti pembayaran.