BAB II

TINJAUAN PUSTAKA

2.1. Tinjauan Teori Umum

2.1.1. Sistem

Sistem adalah kumpulan atau himpunan dari unsur atau variabel-variabel yang saling terkait, saling berinteraksi, dan saling tergantung satu sama lain untuk mencapai tujuan. (Alpiandi, 2016).

Menurut (Iswandy, 2015) Sebuah sistem terdiri dari berbagai unsur yang saling melengkapi dalam mencapai tujuan dan sasaran. Unsur-unsur yang terdapat dalam sistem itulah yang disebut dengan subsistem. Subsistem-subsistem tersebut harus saling berhubungan dan berinteraksi melalui komunikasi yang relevan sehingga sistem dapat bekerja secara efektif dan efisien.

Menurut (Husda, 2012 :111), Sistem adalah suatu kesatuan yang terdiri dari komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi atau energi sehingga dalam sebuah sistem mempunyai syarat–syarat sebagai berikut:

1. Sistem harus dibentuk untuk menyelesaikan tujuan
2. Elemen sistem harus memepunyai rencana yang ditetapkan
3. Adanya hubungan diantara elemen sistem
4. Unsur dasar dari proses (arus informasi, energi dan material)

5. Tujuan organisasi lebih penting dari pada tujuan elemen

Sedangkan karakteristik dari sebuah sistem tersebut adalah:

1. Komponen adalah suatu sistem yang terdiri dari sejumlah komponen yang saling berinteraksi, bekerja sama membentuk satu kesatuan.
2. Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya.
3. Lingkungan luar sistem merupakan segala sesuatu diluar dari batas sistem yang mempengaruhi operasi dari suatu sistem.
4. Penghubung sistem merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya.
5. Masukan sistem merupakan energi yang dimasukan kedalam sistem.
6. Keluaran sistem merupakan hasil dari energi yang diolah oleh sistem.
7. Pengolah sistem merupakan bagian yang memperoses masukan untuk menjadi keluaran yang diinginkan.

Tujuan sistem merupakan tujuan atau sasaran yang yang mempengaruhi input yang dibutuhkan dan output yang yang dihasilkan.

2.1.2. Informasi

Dalam suatu organisasi informasi merupakan sesuatu yang sangat dibutuhkan dalam proses pengambilan keputusan oleh pihak manajemen. Maka menurut (Husda, 2012 : 117) informasi sebagai hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang

menggambarkan suatu kejadian nyata yang berguna untuk pengambilan keputusan, dimana informasi yang berkualitas harus memiliki :

1. Akurat berarti informasi harus bebas dari kesalahan - kesalahan dan tidak bisa atau menyesatkan.
2. Tepat waktu berarti informasi yang datang pada penerima tidak boleh terlambat.
3. Relevan berarti informasi tersebut mempunyai manfaat untuk pemakainya.

2.1.3. Sistem informasi

Sistem informasi merupakan sistem di dalam suatu organisasi yang berfungsi mengolah transaksi harian, mendukung operasi, serta menyediakan informasi yang diperlukan bagi pihak yang berkepentingan. Sistem informasi juga merupakan seperangkat komponen saling berhubungan dan berintegrasi yang berfungsi memproses, mendistribusikan, serta menyimpan informasi guna mendukung keputusan dan pengawasan di dalam suatu organisasi.(Iswandy, 2015).

Sistem informasi adalah suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan yaitu mengendalikan organisasi. Sistem informasi sebagai suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasasi, dan menyediakan pihak luar tertentu dengan laporan - laporan yang diperlukan. Berdasarkan beberapa pendapat para ahli tersebut dapat disimpulkan

bahwa sistem informasi adalah sebuah sistem yang terdiri dari berbagai komponen yang saling bersatu untuk mencapai suatu tujuan yakni menyediakan sebuah informasi bagi yang membutuhkan.(Hermawan, Hidayat, & Utomo, 2016)

Sistem informasi adalah kumpulan elemen yang saling berhubungan satu sama lain yang membentuk satu kesatuan untuk mengintegrasikan data, memproses dan menyimpan serta mendistribusikan informasi. Dapat pula diartikan bahwa sistem informasi adalah suatu kumpulan dari komponen. komponen dalam perusahaan atau organisasi yang berhubungan dengan proses penciptaan dan pengaliran informasi.(Soepomo, 2014)

Sistem informasi sebagai integrasi suatu cara terorganisir (mengumpulkan, memasukkan, dan memproses data, mengendalikan, dan menghasilkan informasi) dengan berbasis komputer untuk mencapai sasaran dan tujuan organisasi. Sistem Informasi mengolah data menjadi informasi dengan menggunakan alat. Alat pengolahan data menjadi informasi ini terus menerus mengalami perkembangan (Waluya & Dimas, 2017).

Menurut (Husda, 2012 : 120) Sistem informasi mempunyai enam buah komponen yaitu:

2.1.1. Komponen *input* dan kompunen masukan

2.1.2. Komponen *model*

2.1.3. Komponen *output* atau komponen keluaran

2.1.4. Komponen teknologi

2.1.5. Komponen basis data

2.1.6. Komponen *control* atau komponen pengendalian.

2.1.4. Sistem informasi akuntansi

Sistem informasi akuntansi merupakan sebuah sistem informasi yang mengubah data transaksi bisnis menjadi informasi keuangan yang berguna bagi pemakainya. Tujuan dari sistem informasi akuntansi adalah:

1. Mendukung operasi sehari-hari
2. Mendukung pengambilan keputusan manajemen
3. memenuhi kewajiban yg berhubungan dengan pertanggung-jawaban.(Rahayu & Devitra, 2017).

Sistem Informasi Akuntansi (SIA) merupakan sistem yang digunakan untuk memproses data dan transaksi guna menyediakan infomasi yang berguna oleh *user* untuk merencanakan, mengendalikan dan mengoperasikan suatu bisnis. Untuk menghasilkan informasi, SIA harus melakukan kegiatan sebagai berikut:

1. Mengumpulkan data transaksi dan data lain yang kemudian dimasukkan ke dalam sistem informasi akuntansi.
2. Memproses data.
3. Menyimpan data untuk masa yang akan datang.
4. Menyediakan informasi yang diperlukan dengan menghasilkan laporan dan memungkinkan melakukan query.
5. Mengendalikan proses, sehingga informasi yang dihasilkan akurat dan dapat dipercaya.

2.1.5. Sistem penerimaan kas

Sistem Penerimaan Kas adalah sekelompok unsur yang erat hubungannya antara satu dengan yang lain dalam melakukan penerimaan kas suatu organisasi. Sistem penerimaan kas dalam perusahaan perlu dirancang sedemikian rupa sehingga meminimalisasi kemungkinan tidak tercatat dan tidak diterimanya uang yang seharusnyaditerima.Sistem informasi akuntansi penerimaan kas digunakan dalam perusahaan untuk menjalankan kegiatan operasional yang diperlukan oleh perusahaan. (Waluya & Dimas, 2017)

Menurut (Fatmawati, 2011) prosedur penerimaan kas perlu memperhatikan hal-hal sebagai berikut :

1. Terdapat pemisahan tugas antara yang menyimpan, yang menerima dan yang mencatat penerimaan uang. Apabila untuk sebuah perusahaan kecil pemisahan sedemikian tidak dapat dilakukan, maka penggabungan antara ketiga tugas tadi hanya dapat dilakukan oleh pemilik perusahaan.
2. Setiap penerimaan uang langsung disetor ke bank sebagaimana adanya.

2.1.6. Pengertian Web

Menurut (Prabowo et al., 2017)*World Wide Web* (WWW), lebih dikenal dengan *web,* merupakan salah satu layanan yang didapat oleh pemakai komputer yang terhubung ke *internet.*

Menurut (Rahmawati & Mulyono, 2016) Web server adalah sebuah program atau applikasi yang menggunakan model klien/server dan world(HTTP),

melayanin file yang berbentuk halaman web untuk pengguna web (merespon permintaan computer klien berisi permintaan HTTP).

Sedangkan menurut (Utama, 2011) Web atau situs dapat diartikan sebagai kumpulan halaman halaman yang digunakan untuk menampilkan informasi, teks, gambar diam atau bergerak, animasi, suara, dan atau gabungan dari semuanya itu, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling berkait dimana masing masing dihubungkan dengan jaringan jaringan halaman (hyperlink). Untuk menyediakan keberadaan sebuat website, maka harus tersedia unsur-unsur penunjangnya, adalah sebagai berikut :

1. Nama domain (domain name/URL – Uniform Resource Locator) adalah alamat unik di dunia internet yang digunakan untuk mengidentifikasikan sebuah website, atau dengan kata lain domain name adalah alamat yang digunakan untuk menemukan sebuah wesite pada dunia internet.
2. Rumah Tempat Website (Web Hosting) dapat diartikan sebagai ruangan yang terdapat dalam harddisk tempat menyimpan berbagai data, file, gambar dan lain sebagainya yang akan ditampilkan di website.
3. Bahasa Program (Script Program) digunakan unuk menerjemahkan setiap perintah dalam website pada saat diakses. Jenis bahasa program sangat menentukan statis, dinamis, atau interaktifnya sebuah website.
4. Desain Website adalah menentukan kualitas dan keindahan sebuah website. Untuk membuat website biasanya dapat dilakukan sendiri atau menyewa jasa website designer. Perlu diketahui bahwa kualitas situs sangat ditentukan oleh kualitas designer.

5. Publikasi website dimana keberadaan situs tidak ada gunanya dibangun tanpa dikunjungi atau dikenal oleh pengunjung internet.Untuk mengenalkan situs kepada masyarakat memerlukan apa yang disebut publikasi atau promosi.
6. Pemeliharaan website, untuk mendukung kelanjutan dari situs diperlukan pemeliharaan setiap waktu sesuai yang diinginkan seperti penambahan informasi, berita, artikel, link, gambar dan lain sebagainya, tanpa pemeliharaan yang baik situs akan terkesan membosankan atau monoton juga akan segera ditinggalkan pengunjung

2.1.7. Konsep dasar *web* program

World Wide Web atau *www* atau juga dikenal dengan *Web* adalah salah satu layanan yang didapat oleh pemakai komputer yang terhubung ke internet. *Web* ini menyediakan informasi bagi pemakai komputer yang terhubung ke internet dari sekedar informasi "sampah" atau informasi yang tidak berguna sama sekali sampai informasi yang serius dari informasi yang gratisan sampai informasi yang komersial. *Website* atau situs dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya itu baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (*hyperlink*).

Penyajian suatu *web* harus terprogram dan dibuat sedinamis mungkin. Dengan pemanfaatan aplikasi *client-server*, pengunjung suatu *web* akan

berinteraksi secara langsung dengan *database*, jika terjadi pengolahan data di dalamnya.

2.1.8. S*ytem development life cycle*

Metode *System Development Life Cycle* atau sering disingkat dengan *SDLC* merupakan pengembangan yang berfungsi sebagai sebuah mekanisme untuk mengidentifikasikan perangkat lunak. Tahapan dalam pengembangan sistem dinamakan *System Development Life Cycle* karena pada setiap tahapan sistem akan dikerjakan secara berurut menurun dari perencanaan, analisis, desain, implementasi dan perawatan. (Hermawan et al, 2016)

SDLC atau *System Development Life Cycle* adalah proses mengembangkan atau mengubah suatu system perangkat lunak dengan menggunakan model-model dan menggunakan metodologi nyang digunakan orang untuk mengembangankan system-sistem perangkat lunak sebelumnya.

Menurut (Salahuddin & A.S, 2014 : 26) tahapan-tahapan yang ada pada SLDC secara global adalah sebagai berikut :

1. Inisiasi adalah tahapan yang biasanya ditandai dengan pembuatan proposal.
2. Pengembangan konsep sistem (*sistem concept development*) mendefenisikan lingkup konsep termasuk dokumen lingkup system, analisis manfaat biaya, manajemen rencana, dan pembelajaran kemudahan sistem.

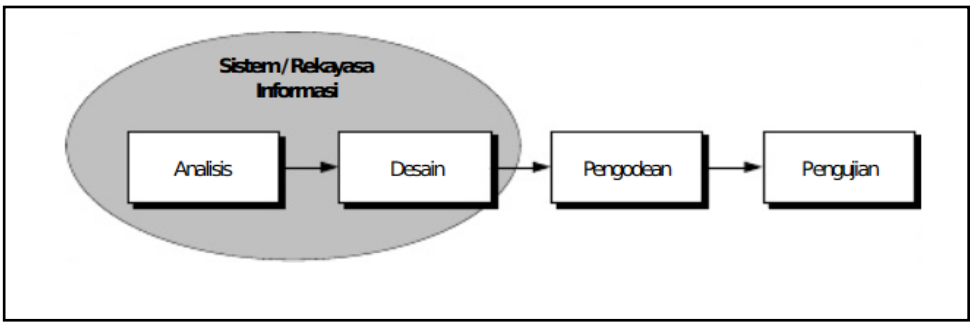
3. Perencanaan adalah mengembangkan rencana manajemen proyek dan dokumen yang lain dan menyediakan dasar untuk mendapatkan sumber daya yang dibutuhkan untuk memperoleh solusi.
4. Analisis kebutuhan adalah menganalisis kebutuhan pemakai sistem perangkat lunak (*user*) dan mengembangkan kebutuhan *user*.
5. Desain adalah mentransformasikan kebutuhan detail menjadi kebutuhan yang sudah lengkap, dokumen desain sistem fokus pada bagaimana dapat memenuhi fungsi-fungsih yang dibutuhkan.
6. Pengembangan adalah mengonversi desain kesistem informasi yang lengkap, membuat basis data, pengodean, memperbaiki dan membersihkan program.
7. Implementasi Sistem adalah tahap dimana desain sistem dibentuk menjadi suatu *coding* (program) yang siap untuk di operasikan.
8. Tahap Pemeliharaan / Perawatan merupakan tahap yang dilakukan setelah tahap implementaasi yang melipliputi penggunaan sistem, audit sistem, penjagaan sistem, perbaikan sistem dan peningkatan sistem.

2.1.9. *Waterfall*

Waterfall model adalah model klasik yang bersifat sistematis, berurutan dalam membangun *software*. Selain itu Model ini merupakan model yang paling banyak dipakai oleh para pengembang *software*. Inti dari metode *waterfall* adalah pengerjaan dari suatu sistem yang dilakukan secara berurutan atau secara linear. Jadi jika langkah satu belum dikerjakan maka tidak akan bisa melakukan

pengerjaan langkah kedua, ketiga dan seterusnya. Secara otomatis tahapan ketiga akan bisa dilakukan jika tahap pertama dan kedua sudah dilakukan. (Rachmat,Hari, 2015).

Berikut adalah gambar model air terjun menurut (Salahuddin & A.S, 2014: 29):



Gambar 0.1 Ilustrasi model *waterfall*

Deskrpsi dari gambar ilustrasi model *waterfall* diatas adalah:

1. Analisis kebutuhan perangkat lunak

   Proses pengumpulan kebutuhan dilakukan secara intensif untuk mespesifikasikan kebutuhan perangkat lunak agar dapat dipahami

perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk di dokumentasikan.

2. Desain

   Desain perangkat lunak adalah proses *multi* langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat di implementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu di dokumentasikan.

3. Pembuatan kode program

   Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

4. Pengujian

   Pengujian fokus pada perangkat lunak secara segi lojik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

5. Pendukung (*support*) atau pemeliharaan (*maintenance*)

   Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau

perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari tahap analisis spesifikasi untuk perubahan perangkat lunak baru.

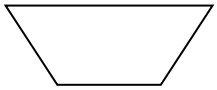
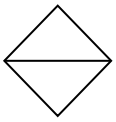
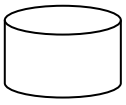
Dengan berbagai kelemahan yang dimiliki model air terjun tapi model ini telah menjadi dasar dari model-model yang lain dalam melakukan perbaikan model pengembangan perangkat lunak. Model air terjun sangat cocok digunakan kebutuhan pelanggan sudah sangat dipahami dan kemungkinan terjadi perubahan kebutuhan selama pengembangan perangkat lunak kecil. Hal positif dari model air terjun adalah struktur tahap pengembangan sistem jelas, dokumentasi di hasilkan di setiap tahap pengembangan dan sebuah tahap di jalankan setelah tahap sebelumnya selesai di jalankan (tidak ada tumpah tindih pelaksanaan tahap).

Model *waterfall* adalah model *SLDC* yang paling sederhana. Model ini hanya cocok untuk pengembangan perangkat lunak dengan spesifikasi yang tidak berubah-ubah.

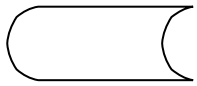
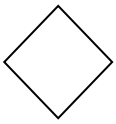
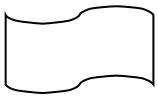
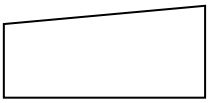
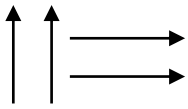
2.1.10. Aliran sistem informasi

Aliran Sistem informasi (ASI) merupakan bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem. Jadi bagan yang menggambarkan arus dari sebuah data dari mana data itu berasal sampai ke pemrosesan data hingga dihasilkan output, sampai proses akhir/arsip. Simbol yang digunakan dalam aliaran system informasi adalah sebagai berikut :

Tabel 0.1 Simbol aliran sistem informasi

GAMBAR SIMBOL	NAMA	KETERANGAN
	Dokumen	Menunjukkan dokumen yang digunakan untuk *input* dan *output*, baik secara manual, mekanik atau menggunakan komputer
	Kegiatan Manual	Menunjukkan pekerjaan yang dikerjakan secara manual
	Proses	Menunjukkan kegiatan proses dari operasi program komputer
	Pengurutan Offline	Menunjukkan proses pengurutan data di luar proses komputer
	Harddisk	Input/output dengan menggunakan harddisk

Lanjutan Tabel 2.1 Simbol aliran sistem informasi

GAMBAR SIMBOL	NAMA	KETERANGAN
	Disket	Input/output dengan menggunakan disket
	Decision	Simbol keputusan yang digunakan untuk suatu penyeleksian kondisi di dalam program
	Pita Kertas Berlubang	Input/output dengan menggunakan online keyboard
	Terminal	Output yang ditampilkan di layar terminal
	Alur Garis	Menunjukkan alur dari proses
	Simbol Penghubung	Digunakan untuk penghubung ke halaman yang masih sama atau ke halaman lain

Sumber*: James A. Kowal, Behavior Models, Specifying User's Expectations*

2.2. Tinjauan Teori Khusus

Aplikasi ini dibangun menggunakan bahasa pemograman *PHP* dan *database MySQL*. Untuk cara mengakses databasenya menggunakan *PDO* (*PHP Data Objects*).

Materi yang dapat dipelajari dari aplikasi ini selain proses *create*, *read*, *update*, *delete* dan search adalah bagaimana menggunakan *JQuery Data Tables* untuk membuat *table* yang dinamis, meng*upload* gambar kedalam *database*, membuat grafik menggunakan *Higtchart*. Selain itu juga dapat dipelajari bagaimana mendesain tampilan aplikasi menggunakaan *CSS Bootstrap*.

2.2.1. *PHP*

Menurut (Herkules & Utariani, 2017) " *PHP* merupakan singkatan dari *PHP Hypertext Preprocessor*. Ia merupakan bahasa berbentuk skrip yang ditempatkan dalam server dan diproses di *server*. Hasilnya yang akan dikirimkan ke klien, tempat pemakai menggunakan *browser*". Bahasa pemrograman ini dirancangan khusus untuk membuat *web* dinamis, artinya pemrograman *PHP* dapat membentuk suatu tampilan berdasarkan permintaan terkini, misalnya halaman yang menampilkan daftar buku tamu. Halaman tersebut akan selalu mengalami perubahan mengikuti jumlah data tamu yang telah mengisi buku tamu. *PHP* dikenal sebagai bahasa pemrograman *web*, terutama di lingkungan *linux*, tetapi sebenarnya *PHP* juga dapat digunakan pada *server-server* yang berbasis *Unix*, *Windows NT*, *Windows 95/98* dan *Machintosh*. Pada awalnya, *PHP* dirancang untuk diintegrasikan dengan *web server Apache*. Namun belakangan, *PHP* juga

dapat bekerja dengan *web server* seperti *PWS (personal web server)*, *IIS (internet information server)*.

Menurut (Azizah, Ariessanti, & Dwiyanti, 2015), "*PHP* merupakan bahasa pemrograman yang ditujukan untuk membuat aplikasi *web*". Ditinjau dari pemrosesannya, *PHP* tergolong berbasis *server side*. Artinya pemrosesan dilakukan di *server*. Hal ini berkebalikan dengan bahasa pemrograman seperti *JavaScript* yang pemrosesannya dilakukan di sisi klien *(side client)*.

2.2.2. *MySQL*

Menurut (Herkules & Utariani, 2017) *MySQL* adalah sebuah perangkat lunak sistem manajemen basis data *SQL* atau *DBMS* yang *mult ithread* dan *multi user*. *MySQL* adalah *Relational Database Managemen Sistem (RDBMS)* yang didistribusikan secara gratis dibawah lisensi *GPL (General Public License)*.

MySQL menurut (Susanti, 2016) adalah salah satu jenis *database server* yang menggunakan *SQL* sebagai bahasa dasar untuk mengakses basis datanya. Dengan menggunakan *script PHP* dan *PERL Software database* ini dapat berfungsi atau berjalan pada semua *platform* sistem operasi yang biasa digunakan (*Windows, Linux, OS/2*, berbagai varian *Unix*).

Sedangkan menurut (Djaelangkara, Sengkey, & LAntang, 2015) *MySQL* adalah sebuah implementasi dari sistem manajemen basis data relasional (*RDBMS*) yang didistribusikan secara gratis dibawah lisensi *GPL (General Public License)*. Setiap pengguna dapat secara bebas menggunakan *MySQL*, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan

yang bersifat komersial. *MySQL* sebenarnya merupakan turunan salah satu konsep utama dalam basis data yang telah ada sebelumnya, *SQL (Structured Query Language). SQL* adalah sebuah konsep pengoperasian basis data, terutama untuk pemilihan atau seleksi dan pemasukkan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

2.2.3. *PDO (php data object)*

Menurut (Djaelangkara et al., 2015) *PDO (PHP Data Objects)* merupakan salah satu ekstensi dari *PHP* versi 5 (lima) yang mendefinisikan koneksi database *(DBMS)* dalam sebuah perpustakan secara abstrak. *PDO* ditulis berdasarkan kompilasi bahasa *C / C++*. *PDO* menjadi pilihan karena mendukung berbagai jenis *database*, dimana untuk untuk memanggil seluruh fungsi setiap *database* tidak memperlukan perubahan *script* pada fungsi *PDO*. *PDO* menggunakan *data-access abstaction layer* untuk berhubungan dengan database.

Sedangkan menurut (Sari, Sholeh, & Informatika, 2014) *PHP Data Objects* adalah ekstensi *PHP* untuk meresmikan koneksi database *PHP* dengan menciptakan antarmuka seragam. Hal ini memungkinkan pengembang untuk membuat kode yang portabel di banyak *database* dan *platform*. *PDO* dibangun menggunakan bahasa *C/C++* dan *PDO* menawarkan sebuah paradigma pemrograman berorientasi objek (*Object Oriented Programming/OOP*) di dalam *script PHP* sehingga dapat berjalan lebih cepat. *PDO* bertujuan untuk membuat satu buah *interface* yang seragam untuk koneksi ke beragam jenis *database*. Ketika menggunakan *database* yang didukung oleh *PDO* seperti *MySQL*, *Oracle*,

MS. SQL Server dan *PostgreSQL* cukup membuat *script* yang sama menggunakan *PDO,* yang berbeda hanya string koneksi untuk setiap *database*, sehingga akan meningkatkan produktivitas pemrograman. Manfaat lain adalah dapat dengan mudah bermigrasi ke *database* lain.

2.2.4. *UML*

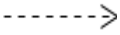
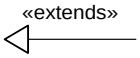
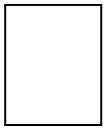
Unified Modeling Language (*UML*) adalah himpunan struktur dan teknik untuk pemodelan desain program berorientasi objek (*OOP*) serta aplikasinya. *UML* adalah metodologi untuk mengembangkan sistem *OOP* dan sekelompok perangkat *tool* untuk mendukung pengembangan sistem tersebut. *UML* mulai diperkenalkan oleh *Object Management Group*, sebuah organisasi yang telah mengembangkan model, teknologi, dan standar *OOP* sejak tahun 1980-an. Sekarang *UML* sudah mulai banyak digunakan oleh para praktisi *OOP*. *UML* merupakan dasar bagi perangkat (*tool*) desain berorientasi objek dari IBM. (Rachmat Hidayat, 2015)

UML memiliki diagram grafis untuk membuat suatu model, yaitu:

1. *Use-case Diagram*

Use Case Diagram digunakan untuk mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. *Use Case* Diagram juga menjelaskan manfaat sistem jika dilihat menurut pandangan orang yang berada diluar sistem (*actor*).

Tabel 0.2 Simbol diagram *use case*

NO	GAMBAR	NAMA	KETERANGAN
1		*Actor*	Menspesifikasikan himpuan peran yang pengguna mainkan ketika berinteraksi dengan *use case*.
2		*Dependency*	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri *(independent)* akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (*independent*).
3		*Generalization*	Hubungan dimana objek anak (*descendent*) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*).
4		*Include*	Menspesifikasikan bahwa *use case* sumber secara *eksplisit*.
5		*Extend*	Menspesifikasikan bahwa *use case* target memperluas perilaku dari *use case* sumber pada suatu titik yang diberikan.
6		*Association*	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		*System*	Menspesifikasikan paket yang menampilkan sistem secara terbatas.

Lanjutan Tabel 2.2 Simbol *Diagram Use Case*

NO	GAMBAR SIMBOL	NAMA	KETERANGAN
8		*Use Case*	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
9		*Collaboration*	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan prilaku yang lebih besar dari jumlah dan elemen-elemennya
10		*Note*	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

Sumber: Rosa dan Shalahuddin, 2013: 156

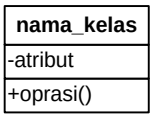
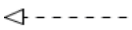
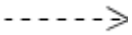
2. Diagram Kelas (*Class Diagram)*

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem.

Tabel 0.3 Simbol diagram kelas

NO	GAMBAR	NAMA	KETERANGAN
1		*Generalization*	Hubungan dimana objek anak *(descendent)* berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*).

Lanjutan Tabel 2.3 Simbol diagram kelas

NO	GAMBAR SIMBOL	NAMA	KETERANGAN
2		*Nary Association*	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		*Class*	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		*Collaboration*	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
5		*Realization*	Operasi yang benar-benar dilakukan oleh suatu objek.
6		*Dependency*	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri *(independent)* akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri

Lanjutan Tabel 2.3 Simbol *Diagram Use Case*

NO	GAMBAR SIMBOL	NAMA	KETERANGAN
7	———	*Association*	Apa yang menghubungkan antara objek satu dengan objek lainnya

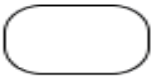
Sumber: Rosa dan Shalahuddin, 2013: 146

3. Diagram Aktifitas (*Activity Diagram*)

Diagram yang menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak.

Simbol yang digunakan dalam diagram aktifitas adalah:

Tabel 0.4 Simbol Diagram Aktifitas

NO	GAMBAR	NAMA	KETERANGAN
1		*Actifity*	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		*Action*	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		*Initial Node*	Bagaimana objek dibentuk atau diawali.

Lanjutan Tabel 2.4 Simbol Diagram Aktifitas

NO	GAMBAR	NAMA	KETERANGAN
4		*Actifity Final Node*	Bagaimana objek dibentuk dan dihancurkan
5		*Fork Node*	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

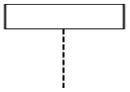
Sumber: Rosa dan Shalahuddin, 2013: 162

4. *Sequence* Diagram

Diagram ini menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek.

Simbol yang digunakan dalam diagram sekuen adalah:

Tabel 0.5 Simbol Diagram Sekuen

NO	GAMBAR	NAMA	KETERANGAN
1		*LifeLine*	Objek *entity*, antarmuka yang saling berinteraksi.

Lanjutan Tabel 2.5 Simbol diagram sekuen

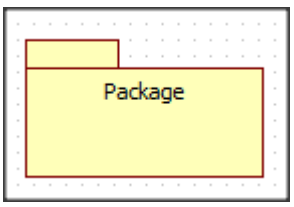
NO	GAMBAR	NAMA	KETERANGAN
2		*Message*	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
3		*Message*	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

Sumber: Rosa dan Shalahuddin, 2013: 165

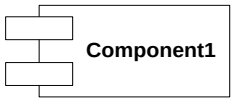
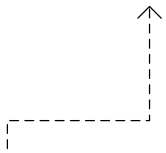
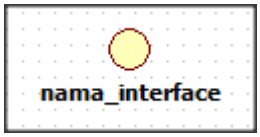
5. *Diagram Komponen (Component Diagram)*

Diagram ini dibuat untuk menunjukkan organisasi dan ketergantungan diantara kumpulan komponen dalam sebuah sistem. Diagram ini berfokus pada komponen sistem yang dibutuhkan dan ada di dalam sistem.

Tabel 0.6 Simbol Diagram Komponen

Simbol	Nama	Deskripsi
	*Package*	Package merupakan sebuah bungkusan dari satu atau lebih komponen.

Lanjutan Tabel 2.6 Simbol diagram komponen

GAMBAR SIMBOL	NAMA	KETERANGAN
	Komponen	Komponen system
	Ketergantungan (*Dependency*)	Kebergantungan antar komponen, arah panah mengarah pada komponen yang dipakai
	Antarmuka/ interface	Sama dengan interface pada pemrograman berbasis objek, yaitu sebagai antarmuka komponen agar tidak mengakses langsung komponen.

Sumber: Rosa dan Shalahuddin, 2013: 149

2.2.5. *CSS bootstrap*

Menurut (Ilmiah, Purba, Tarigan, Dameri, & Kom, 2013) *Bootstrap* merupakan sebuah *CSS framework* yang dikembangkan pengembang *Twitter* pada pertengahan tahun 2010. Sebelum resmi menjadi *open-source*, *Bootstrap* dikenal sebagai *Twitter Blueprint*. Hingga saat ini *bootstrap* sudah dirilis *versi 3.3.7* dan telah menjadi salah satu *front-end framework* yang paling populer serta menjadi

proyek *open source* di dunia. *Bootstrap* digambarkan sebagai *CSS* sederhana namun dibangun dengan *pre-processor* yang menyediakan lebih banyak daya dan fleksibilitas dibanding *CSS* standar. *Bootstrap* sudah menyediakan *class-class CSS* dan terintegrasi dengan *JQuery*. Responsive *layout* pada *CSS bootstrap* dengan 12 kolom grid sistem menghasilkan *layout website* yang secara otomatis menyesuaikan dengan lebar aplikasi *browser* pengguna. Hal ini yang menyebabkan *bootstrap* mendukung semua jenis perangkat seperti *smartphone, tablet*, laptop ataupun PC *Desktop*. Di samping itu, *bootstrap* juga sudah *support* untuk *HTML 5* dan *CSS 3*.

2.2.6. Pengertian kas

Menurut (Merystika Kabuhung, 2013) Kas adalah salah satu unsur aktiva yang paling penting karena merupakan alat pertukaran atau pembayaran yang siap dan bebas digunakan untuk membiayai kegiatan operasional perusahaan. Hampir setiap transaksi perusahaan dengan pihak luar menggunakan kas. Oleh karena itu, kas mempunyai sifat mudah dipindahtangankan dan tidak dapat dibuktikan pemiliknya maka uang kas yang keluar akan mudah disalah gunakan. Melihat kondisi kas yang demikian beresiko maka setiap perusahaan harus punya sistem dan prosedur penerimaan dan pengeluaran yang baik, dimana manajemen bertanggung jawab atas penerimaan dan pengeluaran kas. Kas meliputi uang tunai dan instrumen atau alat-alat pembayaran yang diterima oleh umum, baik yang ada di dalam perusahaan maupun yang disimpan di bank (uang tunai kertas dan logam, cek, wesel cek, rekening bank yang berbentuk tabungan dan giro).

Sedangkan menurut (Pakadang, 2013 : 212-223) Kas merupakan aset yang paling lancar dan hampir setiap transaksi dipengaruhi oleh kas. Karena sifat kas yang *liquid,* maka mudah digelapkan, sehingga diperlukan pengendalian *intern* untuk mengelolahnya, dengan memisahkan fungsi-fungsi penyimpanan, pelaksanaan dan pencatatan. Pengendalian *intern* yang baik terhadap kas, dimana manajemen harus menetapkan tanggung jawab secara jelas dan tiap orang memiliki tanggung jawab untuk tugas yang diberikan padanya. Pengendalian *intern* yang baik terhadap kas memerlukan prosedur yang memadai untuk melindungi penerimaan kas.

2.2.7. Pengertian *jquery*

JQuery menurut (Yatini, 2014) adalah *JavaScript library* yang dirancang untuk meringkas kode-kode *JavaScript*, sehingga dapat menyederhanakan penulisan skrip program, sesuai dengan slogan "*write less, do more*" , J*Query* pertama kali dirilis oleh *John Resig* pada tahun 2006, pada perkembangannya *JQuery* tidak hanya sebagai *framework JavaScript*, namun memiliki kehandalan dan kelebihan yang cukup banyak. Hal tersebut menyebabkan banyak developer web menggunakannya.

2.3. Penelitian Terdahulu

Di sumber yang lain mengatakan, penelitian terdahulu adalah analisa berupa kritik (membangun maupun menjatuhkan) dari penelitian yang sedang dilakukan terhadap topik khusus atau pertanyaan terhadap suatu bagian dari keilmuan.

Penelitian terdahulu merupakan cerita ilmiah terhadap suatu permasalahan tertentu. Penelitian terdahulu berisi ulasan, rangkuman, dan pemikiran penulis tentang beberapa sumber pustaka (artikel, buku, slide, informasi dari internet, dll) tentang topik yang dibahas.

Penelitian terdahulu ini menjadi salah satu acuan penulis dalam melakukan penelitian sehingga penulis dapat memperkaya teori yang digunakan dalam mengkaji penelitian yang dilakukan. Dari penelitian terdahulu, penulis tidak menemukan penelitian dengan judul yang sama seperti judul penelitian penulis. Namun penulis mengangkat beberapa penelitian sebagai referensi dalam memperkaya bahan kajian pada penelitian penulis.

Beberapa penelitian yang telah dilakukan dan memiliki korelasi yang searah dengan penelitian skripsi ini antara lain:

1. Penelitian yang telah dilakukan oleh Ade Mubarok dan Sri Handayani Jurnal Informatika, Vol.III No.1 Tahun 2016

   Dalam jurnalnya yang berjudul “Perancangan Program Transaksi Penerimaan dan Pengeluaran Kas Berbasis Web”. Metode menggunakan metode Air Terjun (Waterfall) Informasi yang dihasilkan dari sistem ini bertujuan pembuatan aplikasi lebih terstruksur dan aplikasi yang dihasilkan bisa membantu pencatatan transaksi penerimaan dan pengeluaran kas kecil lebih efektif dan efesien.(Mubarok & Hadianti, 2016).

2. Penelitian yang telah dilakukan oleh Umy Mukhofifah, Migunani, dan Arief Hidayat IJSE – Indonesian Journal on Software Engineering Volume 2 No 1 – 2016

   Dalam jurnalnya yang berjudul "Perancangan Sistem Pelaporan Keuangan Berbasis Web (Studi Kasus Pada PT.EMKL Tirtasari Abadi Sejahtera Semarang)" metode yang digunakan metode *system development life cycle (SDLC).* Penelitian ini bertujuan menyediakan sistem pelaporan keuangan di PT.EMKL Tirtasari Abadi Sejahtera yang masih mengalami keterlambatan didalam menghasilkan laporan keuangan serta kesalahan dalam pencatatan dan penghitungan dalam menyusun laporan keuangan, maka perlu dikembangkan suatu sistem pelaporan keuangan yang dapat menyediakan laporan keuangan dengan cepat, tepat dan akurat bagi pengguna sistem.(Mukhofifah, Migunani, & Arief, 2016).

3. Penelitian yang telah dilakukan oleh Novianto Yosua Tampi, dan Inge Handriani JURNAL ILMIAH FIFO P-ISSN 2085-4315 / E-ISSN 2502-8332 Volume IX/No.1/Mei/2017

   Dalam jurnalnya yang berjudul "Aplikasi Sistem Pengeluaran Kas Pada PT Ithaca Resources" dengan menggunakan metode *waterfall*. Penelitian ini bertujuan aplikasi ini dapat mengontrol pengeluaran kas berdasarkan anggaran yang telah ditentukan sehingga pengeluaran kas lebih tertib dan aplikasi ini dapat membantu proses verifikasi pengajuan *cash advance* menjadi lebih baik. Memberikan kemudahan informasi tentang tata cara,

prosedur, ketentuan dan etika pengeluaran kas kepada karyawan dalam mengajukan *cash advance*. (Tampi & Handriani, 2017).

4. Penelitian yang telah dilakukan oleh Sarip Hidayatuloh, dan Indah Sari Agustin Jurnal Sistem Informasi, 2015 p-ISSN 1979-0767

   Dalam jurnalnya yang berjudul "Analisis Dan Perancangan Sistem Informasi Pencatatan Keuangan Pada Koperasi Lancar Jaya" Analisis dan Desain Sistem Informasi menggunakan metode *Waterfall* yang mengacu pada teori Jeffery L, Whitten Dkk. (2004) dengan *Unified Modeling Language* (*UML*) sebagai alat pemodelan sistem, sementara *Microsoft Visio* 2013 untuk desain antarmuka dan membuat grafik. Metodologi penelitian yang digunakan untuk menggunakan observasi, wawancara, sastra dan untuk pengembangan sistem informasi koperasi menggunakan metodologi pengembangan *waterfall*. Hasil penelitian ini adalah Analisis dan Desain Sistem Informasi Pencatatan Keuangan berbasis desktop. Desain ini diharapkan dapat memfasilitasi aliran informasi rekaman keuangann, serta mendukung pelaksanaan proses bisnis koperasi. (Dan & Sistem, 2015).

5. Penelitian yang telah dilakukan oleh Annisa Dwiyanti (2015)

   Dalam jurnalnya yang berjudul "Perancangan Sistem Informasi Penggunaan Dana Kas Kecil Pada PT Jumbo Power Internasional. Penulis mengusulkan

sistem informasi penggunaan dana kas kecil dengan menggunakan metode SWOT dan mengusulkan sistem dengan rancangan layar (user interface) menggunakan metode prototype. Penelitian ini bertujuan membuat laporan harian kas kecil dengan adanya sistem pengecekan saat pengajuan dana kas kecil, menginput satu per satu rincian pengeluaran dana kas kecil, membuat laporan harian kas kecil secara akurat dengan memonitoring dengan baik sehingga tidak terjadi kecurangan dalam penyalahgunaan. (Dan & Sistem, 2015).