

BAB II

TINJAUAN PUSTAKA

2.1. Teori Dasar

2.1.1. Pengertian Sistem

Sistem adalah sekelompok orang yang bekerja sama dengan seperangkat peraturan atau aturan yang sistematis dan terstruktur untuk membentuk suatu kesatuan yang melakukan tindakan untuk mencapai tujuan (Anggraeni, 2017). Sistem adalah sekumpulan komponen yang terhimpun dan berhubungan satu sama lain, baik fisik maupun non fisik, untuk bekerja sama mencapai tujuan yang disepakati secara harmonis. Sistem memiliki pendekatan dimana proses kerja saling terhubung, dikelompokkan dan bekerja sama untuk mendapat tujuan yang diinginkan (Prehanto, 2020).

Berdasarkan buku yang berjudul Analisa dan Perancangan Sistem Informasi, memaparkan bahwa terdapat beberapa karakteristik agar dapat dikatakan sebagai sebuah sistem (Anggraeni, 2017), yaitu :

a. Komponen Sistem

Sistem memiliki beberapa bagian yang saling berinteraksi satu sama lain, sehingga sistem tidak dapat berada dalam lingkungan yang kosong.

b. Batasan Sistem (*Boundary*)

Batas sistem adalah pembatas atau garis pemisah antara sistem tersebut dengan sistem lain di luar lingkungannya. Ini ditunjukkan oleh ruang lingkup sistem.

c. Lingkungan (*Environment*)

Lingkungan berarti segala sesuatu di luar batas-batas sistem yang dapat mempengaruhi pengoperasian sistem baik merugikan maupun menguntungkan.

d. Penghubung sistem (*Interface*)

Penghubung sistem yang relevan adalah media yang digunakan sebagai penghubung antara suatu subsistem dengan subsistem lainnya, seperti antarmuka.

e. Masukan (*Input*)

Input atau masukan adalah data yang dimasukkan ke dalam sistem berupa pemeliharaan (*maintenance*) agar sistem bekerja dengan baik, dan sinyal (*signal input*) berupa proses untuk menghasilkan keluaran atau output.

f. Pengolahan (*processing*)

Pengolahan atau *processing* adalah salah satu bagian dari sistem yang melakukan perubahan berdasarkan *input* untuk menghasilkan *output* sesuai dengan tujuan dari sebuah sistem.

g. Tujuan dan Sasaran

Agar suatu sistem dapat berguna, maka sistem itu harus mempunyai tujuan atau maksud. Target ini dapat digunakan untuk memantau sistem yang sedang berjalan.

h. Keluaran (*output*)

Keluaran atau *output* adalah hasil yang berupa informasi dari pemrosesan suatu sistem berdasarkan *input*.

i. Umpan balik (*feedback*)

Sistem kendali atau control panel membutuhkan umpan balik atau *feedback* untuk mengontrol proses yang tidak normal tersebut sehingga dapat mengembalikannya seperti semula.

2.1.2. Pengertian Informasi

Informasi adalah data yang diproses untuk memainkan peran yang berguna dalam membuat keputusan tentang situasi (Anggraeni, 2017). Informasi adalah hasil pengolahan informasi atau fakta yang berharga dalam pengambilan keputusan (Aasinjery, 2020). Definisi informasi menurut pakar di Bidang Teknologi, Raymond Mc. Leod menyatakan bahwa informasi adalah informasi yang telah diolah dalam format yang berarti bagi penerimanya dan berguna untuk keputusan saat ini atau masa depan (Nugroho, 2017). Ciri-ciri sebuah informasi yang memiliki kualitas tinggi menurut Mc Leod adalah sebagai berikut :

a. Akurat

Makna dari akurat pada sebuah informasi yaitu, informasi dapat mencerminkan atau menjelaskan informasi sesuai dengan keadaan yang sebenarnya.

b. Tepat Waktu

Informasi yang disajikan dapat tersedia sesuai pada waktunya saat diperlukan.

c. Relevan

Relevan memiliki arti yaitu, informasi yang disajikan sesuai dengan apa yang dibutuhkan.

d. Lengkap.

Sebuah informasi harus disajikan secara lengkap dan rinci.

2.1.3. Sistem Informasi

Bonnie Soeherman dan Marion Pinontoan menyebutkan bahwa sistem informasi adalah sekumpulan orang, proses, data dan teknologi, seperti komputer, yang digunakan untuk melakukan proses pengambilan keputusan bila berguna untuk mendukung keberhasilan organisasi dalam mencapai tujuan tersebut. Sistem informasi memiliki beberapa elemen proses, diantaranya adalah Pengumpulan data (*data collection*), pengelolaan data yang tersimpan, diseminasi data (Aasinjery, 2020).

Sistem informasi memiliki beberapa komponen, dimana komponen sistem informasi tersebut diharapkan dapat berfungsi seperti pada mestinya (Aasinjery, 2020). Berikut adalah penjelasan dari konsep sistem informasi :

- a. Perangkat keras yang terdiri dari komputer, *printer* dan jaringan.
- b. Perangkat lunak yang terdiri dari kumpulan beberapa perintah atau *query* yang ditulis agar komputer dapat bekerja sesuai dengan perintah yang diberikan.
- c. Data adalah komponen fundamental dari sistem informasi dimana data selanjutnya diproses untuk menghasilkan informasi.
- d. Manusia, pada komponen sistem informasi ini, manusia yang terlibat seperti operator dan pimpinan.
- e. Prosedur, dalam komponen sistem informasi ini, prosedur yaitu dokumentasi dari proses sebuah sistem, atau buku penuntun operasional sebuah aplikasi maupun teknis.

2.1.4. Pengertian Perangkat Lunak

Perangkat lunak bukan hanya sekedar sebuah program komputer, namun termasuk didalamnya terdapat dokumen-dokumen yang menyertainya, seperti dokumen *planning, analysis, design, implementation* (Bahar et al., 2011). Pengembangan perangkat lunak adalah ilmu yang berhubungan dengan semua aspek produksi, dari pemeriksaan spesifikasi hingga pemeliharaan sistem. Perangkat lunak juga merupakan elemen logis dan bukan bagian dari sistem fisik, maka demikian perangkat lunak memiliki karakteristik, yaitu :

- a. *Maintainability* atau dapat dirawat, perangkat lunak harus dapat diubah sewaktu-waktu sesuai dengan kebutuhan pengguna.
- b. *Dependability* atau dapat dipercaya, sehingga perangkat lunak dapat sepenuhnya digunakan dan tidak menyebabkan kerusakan fisik atau finansial jika terjadi kegagalan sistem.
- c. Efisiensi perangkat lunak harus dapat di pertanggung jawabkan, sehingga efisien dalam penggunaan *resource*.
- d. *Usability*, di mana perangkat lunak dapat sepenuhnya digunakan sebagaimana dimaksud, dalam hal ini diperlukan antarmuka pengguna yang bagus dan dokumentasi yang tepat.

2.1.5. Pengertian Inventaris

Inventaris atau persediaan adalah barang, atau dapat disebut sebagai bahan, yang digunakan perusahaan atau organisasi untuk menjalankan bisnisnya. Inventaris bagi organisasi adalah hal yang dilakukan untuk mengantisipasi kebutuhan pelanggan. Selain itu, inventaris juga dapat dikatakan sebagai kegiatan yang melakukan penyusunan hingga pencatatan barang, dimana hal tersebut

dilakukan untuk memudahkan penggelaran atau pengawasan dan dalam penggunaan barang serta dalam mengukur tanggung jawab pemeliharaan dan penyimpanan barang. Jika ada kebutuhan informasi tentang properti, kami dapat dengan mudah menemukannya karena disimpan dengan benar (Siregar, 2018).

2.1.6. Pengertian Web

Halaman web atau *world wide web* yang sering disingkat “www” adalah sebuah sistem yang berisi informasi dengan format skrip HTML (*Hyper Text Markup Language*). Website adalah fasilitas hypertext untuk menampilkan data berupa teks, gambar, suara, animasi dan data multimedia lainnya (Limbong, 2021). Seluruh informasi dari *website* akan diterjemahkan dalam bahasa HTML dan ditampilkan melalui *web browser*.

Umumnya website terbagi menjadi tiga jenis, yaitu website statis yang isinya tidak perlu diperbarui secara berkala, lalu ada *website* dinamis yang isi dari *web* nya perlu dikelola atau diperbarui secara berkala, dan yang terakhir *website* interaktif yang indormasinya dilakukan oleh penggunanya sendiri seperti *marketplace*, jejaring sosial, dan lain-lain (Abdulloh, 2018) Dari sini dapat disimpulkan bahwa website adalah kumpulan situs web yang dapat diakses secara publik, serta saling berhubungan dengan menggunakan nama domain. Website sendiri merupakan campuran halaman yang menampilkan informasi berupa teks, suara, video, animasi yang diterjemahkan dalam bahasa HTML dan ditampilkan melalui *web browser*.

2.1.7. Pengertian Database

Basis data atau database adalah kumpulan atau kombinasi data, berupa data dengan konteks logis, disimpan dalam susunan tertentu pada penyimpanan komputer. Basis data sering digunakan untuk mengolah data untuk menghasilkan informasi tertentu. Basis data adalah kumpulan file terkait dari mana basis data dapat menampilkan berbagai informasi yang digunakan baik di dalam perusahaan maupun organisasi. Database dapat digambarkan seperti sebuah lemari arsip yang menyimpan berbagaimacam data dan perlu adanya pengelolaan dari data-data tersebut (Pratama, 2020).

Informasi berupa data yang ada pada database disimpan secara sistematis dalam komputer sehingga dapat diperiksa oleh program komputer untuk memperoleh informasi dari database (Ahmad et al., 2021). Program komputer atau perangkat lunak yang mengelola database disebut dengan *Database Management System* (DBMS). DBMS terdiri dari basis data dan set program pengelola data seperti Tambah data, hapus data, cari dan baca data. Dalam sebuah sistem yang besar, DBMS tentu akan membantu pengguna dan *third-party software* untuk menyimpan, mengelola, serta mengambil data (Pratama, 2020).

2.1.8. *System Development Life Cycle* (SDLC)

SDLC atau *System Development Life Cycle* biasa disebut siklus pengembangan sistem. SDLC adalah proses pengembangan atau pembuatan perangkat lunak menggunakan model dan metode yang telah digunakan di masa lalu (Pratama, 2020).

SDLC sebagai alat komunikasi antara tim pengembangan dan pemangku kepentingan, selain memungkinkan operasi SDLC memberikan input dan output yang jelas dari satu fase ke fase lainnya.. SDLC memiliki empat tahap fundamental yang digunakan untuk menghasilkan sistem yang sebenarnya. Berikut adalah beberapa fase dari SDLC yang dikutip dari (Dennis, 2021):

a. Perencanaan (*Planning*)

Fase perencanaan adalah fase fundamental atau fase yang paling penting agar paham mengapa sistem informasi harus dibangun dan menentukan bagaimana tim dalam proyek ini akan membangun sistem tersebut.

b. Analisis (*Analysis*)

Selanjutnya adalah fase analisis, pada tahap ini akan menjawab siapa yang menggunakan sistem, apa yang dilakukan sistem, dan di mana dan kapan sistem itu digunakan. Dia juga memantau sistem yang ada, mengidentifikasi kemungkinan untuk perbaikan dan mengembangkan konsep untuk sistem baru. Hasil dari analisis ini adalah kelebihan dan kekurangan sistem, karakteristik sistem dan pembaharuan yang berlaku. Biasanya pada tahap analisis mereka melakukan penelitian literatur, brainstorming, menganalisis persyaratan sistem dan membuat batasan sistem dan menentukan persyaratan sistem.

c. *Design*

Setelah analisis, lanjutkan ke fase desain di mana Anda memutuskan atau menentukan bagaimana sistem akan bekerja dalam hal perangkat lunak, antarmuka pengguna, formulir, dan laporan dan database serta file yang diperlukan. Jadi pada fase

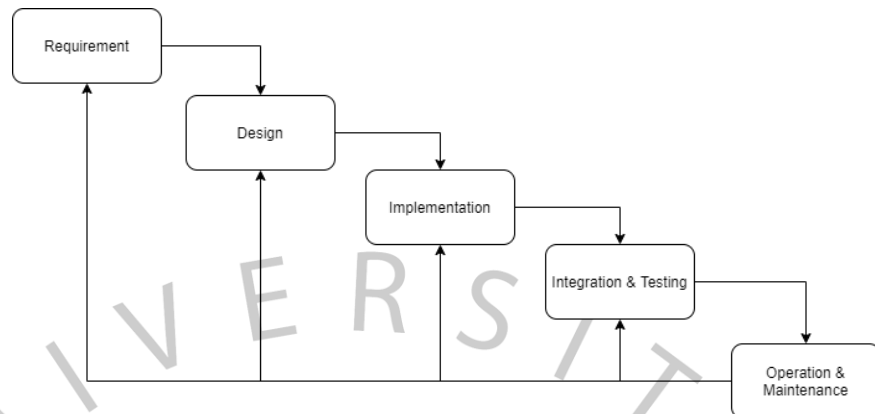
ini ada fase yang secara tepat ditentukan bagaimana sistem yang akan digunakan nantinya.

d. Implementasi

Tahap terakhir pada SDLC adalah implementasi, setelah hasil dokumentasi dari tahap perancangan design telah selesai, masuk pada tahap implementasi. Pada tahap ini mulai mengimplementasikan desain sistem untuk situasi nyata dengan perangkat keras dan mempersiapkannya dengan aplikasi. Berbagai kegiatan dilakukan pada tahap implementasi, seperti: cara membuat database sesuai dengan diagram desain, membangun aplikasi berdasarkan desain sistem, serta menguji dan memperbaiki aplikasi.

2.1.9. Pengertian *Waterfall Model*

Waterfall model atau model air terjun dalam mengembangkan sistem informasi merupakan tahap pengembangan yang menekankan fase berurutan dan sistematis, sehingga jenis model *waterfall* ini termasuk ke dalam *classic life cycle*. Herbert D. Benington adalah yang pertama kali memperkenalkan penggunaan *waterfall* model di *Symposium on Advanced Programming Method for Digital Computers* pada 29 Juni 1956. *Waterfall* model adalah metodologi original yang masih digunakan sampai sekarang. Salah satu kelebihan dari *waterfall* model adalah dapat meminimalisir kesalahan, karena kesalahan tersebut dapat teridentifikasi dalam setiap tahapnya dan dapat kembali ke tahap sebelumnya apabila terdapat perubahan. (Nugroho, 2017) Berikut adalah gambaran dari *waterfall* model menurut Ian Sommerville (2011) :



Gambar 2.1 Waterfall

Sumber : (Nugroho, 2017)

• 2.1.10. Object Oriented Analysis and Design (OOAD)

Object-Oriented Analysis and Design (OOAD) adalah metode untuk menganalisis dan mendesain perangkat lunak dengan pendekatan berorientasi objek. Metode berorientasi objek adalah paradigma baru dalam pengembangan sistem sebagai kumpulan objek yang digerakkan oleh perilaku dan interaktif (Purwaningtias, 2018). OOAD merupakan metode analisis yang mengkaji persyaratan dari sudut pandang kelas dan objek dalam konteks masalah sistem dan subsistem. (Nugroho, 2017). Perancangan OOAD membutuhkan teknik khusus yang menggunakan UML (*Unified Modeling Language*) sebagai alat bantu standar dalam dunia pengembangan sistem perangkat lunak berorientasi objek. dipinjam (Purwaningtias, 2018)

UML adalah bahasa yang banyak digunakan dalam industri untuk menggambarkan persyaratan, melakukan analisis dan desain untuk menggambarkan arsitektur dalam pemrograman berorientasi

objek. UML juga termasuk bahasa yang fleksibel dalam melakukan analisis dan pengembangan karena secara keseluruhan memiliki notasi yang konsisten (Dennis, 2021). Berikut adalah beberapa pemodelan yang biasanya digambarkan pada UML:







a. *Use Case Diagram*

Use case diagram merupakan model untuk memproses sistem informasi. *Use case* bekerja dengan menggambarkan interaksi tipikal antara pengguna dan sistem (Wira et al., 2019). Membuat diagram *use case* yang komprehensif sangat penting dalam fase analisis, karena mendapatkan banyak informasi penting tentang aturan bisnis yang akan dibuat, dalam *use case* diagram akan menerangkan bagaimana sistem digunakan, sehingga dalam melakukan pengembangan analisis dimulai dengan menggunakan *use case diagram* (Nugroho, 2017).

Secara khusus, *use case diagram* dibuat secara sederhana, mudah digunakan dalam komunikasi antar *user* mengenai apa yang akan dilakukan oleh sistem, dengan menggambarkan fungsi utama sistem, serta berbagai jenis *user* yang akan berinteraksi dengan sistem. Oleh karena itu, *use case diagram* digunakan saat mengumpulkan dan menentukan persyaratan dari sebuah sistem, dan agar pengguna dapat memberikan masukan atau tambahan mengenai sebuah sistem. Seluruh *use case* akan dilakukan didetailkan pada dokumen terpisah, dimana pada dokumen tersebut menjelaskan secara detail informasi dari masing-masing diagram kasus yang dituangkan pada dokumen spesifikasi *use case* (Dennis, 2021).

Hasil dari dokumen spesifikasi *use case* akan menjadi landasan dalam penggambaran UML lainnya. Dokumen spesifikasi *use case* berisi *success scenario* bagaimana sistem bekerja sesuai dengan diagram *use case* tersebut, dimana spesifikasi ini akan menjelaskan urutan interaksi antara actor dengan sistem. Berikut adalah komponen dari *use case* diagram yang terdapat pada Tabel 2.1 di bawah :

Tabel 2.1 Komponen *Use Case* Diagram

Simbol	Nama	Keterangan
	Aktor	Pelaku utama yang berinteraksi dengan sistem dalam melakukan <i>input</i> hingga menghasilkan <i>output</i> melalui suatu sistem.
	<i>Use Case</i>	Bagian dari fungsionalitas sistem secara keseluruhan.
	<i>Association</i>	Alur penghubung interaksi antara aktor dengan <i>use case</i> .
	<i>Include</i>	Fungsionalitas terhadap proses <i>use case</i> .
	<i>Extend</i>	Penambahan fungsionalitas terhadap proses.
	<i>Boundaries System</i>	Batasan dari sebuah sistem




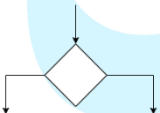
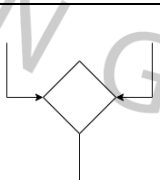
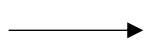
Sumber : (Dennis, Wixom, & Tegarden, 2021)

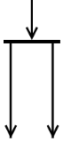

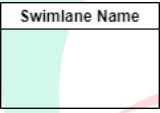
b. Diagram Aktivitas (*Activity Diagram*)

Pada *activity diagram* akan menggambarkan *flow* atau alur dari suatu proses bisnis perangkat lunak terlepas dari *class*, *activity diagram* pada dasarnya akan menunjukkan hal yang

serupa seperti pada scenario, lebih tepatnya *activity diagram* menggambarkan skenario secara grafis (Syarif et al., 2020). *Activity diagram* biasanya digunakan untuk memodelkan sebuah perilaku terhadap proses bisnis, dan menggambarkan hubungan antara aktifitas pada sebuah proses (Dennis, 2021). Berikut adalah komponen dari *activity diagram* yang terdapat pada Tabel 2.2:

Tabel 2.2 Komponen *Activity Diagram*

Simbol	Nama	Keterangan
	<i>Initial State</i>	<ul style="list-style-type: none"> - Menggambarkan dimulainya sebuah aktifitas. - Hanya ada satu pada sebuah <i>activity diagram</i>.
	<i>Final State</i>	<ul style="list-style-type: none"> - Digunakan untuk menghentikan semua alur aktifitas.
	<i>Activity</i>	<ul style="list-style-type: none"> - Digunakan untuk menggambarkan sebuah aktifitas. - Ditandai dengan nama aktifitasnya sendiri.
	<i>Decision</i>	<ul style="list-style-type: none"> - Digunakan untuk menggambarkan pilihan kondisi. - Ditandai dengan kriteria keputusannya sendiri.
	<i>Merge</i>	<ul style="list-style-type: none"> - Digunakan untuk menggabungkan kembali perbedaan keputusan yang telah dipilih sebelumnya
	<i>Association</i>	<ul style="list-style-type: none"> - Menunjukkan aliran aktifitas.

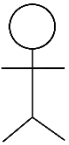

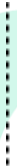



Simbol	Nama	Keterangan
	<i>Fork</i>	- Digunakan untuk aktifitas yang dilakukan secara parallel atau bersamaan.
	<i>Join Fork</i>	- Digunakan untuk menggabungkan aktifitas yang dilakukan secara parallel atau bersamaan.
	<i>Swimlane</i>	- Digunakan untuk memisahkan aktifitas pada baris tertentu. - Dinamakan sesuai dengan objeknya.


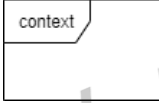
Sumber : (Dennis, Wixom, & Tegarden, 2021)

c. *Sequence Diagram*

Sequence Diagram adalah diagram yang mencakup seluruh interaksi pada *use case*, karena *sequence diagram* menggambarkan interaksi antar objek pada *use case* diagram seperti pengguna, *display*, pesan yang digambarkan terhadap waktu. Sehingga, semakin banyak *use case* maka semakin banyak juga *sequence diagram* yang perlu dibuat (Wira et al., 2019). *Sequence diagram* menggambarkan ilustrasi *actor* dan objek yang saling berhubungan pada *use case* dan terdapat pesan yang melewati antar hubungan tersebut. *Sequence diagram* juga menjadi urutan *generic* yang menggambarkan semua kemungkinan pada *scenario* terhadap suatu *use case* (Dennis, 2021). Berikut adalah komponen dari *sequence diagram* diagram yang terdapat pada Tabel 2.3:

Tabel 2.3 Komponen *Sequence Diagram*

Simbol	Nama	Keterangan
	Aktor	- Aktor yang berinteraksi pada sistem.
	<i>An object</i>	- Ikut serta sebagai pengirim atau penerima pesan
	<i>Lifeline</i>	- Menunjukkan garis hidup atau aktifitas dari sebuah object pada alur. - Berisi symbol X dipaling bawah apabila sebuah <i>class</i> tidak lagi berinteraksi
	<i>Activation Box</i>	- Menunjukkan aktifitas objek ketika mengirim atau menerima pesan.
	<i>Message</i>	- Menyampaikan informasi dari satu objek ke objek lainnya. - Dinamai dengan pesan yang dikirim dan dengan anak panah yang solid. - Untuk pesan kembali (<i>return message</i>) dinamai dengan nilai yang dikembalikan dan digambarkan dengan anak panah garis putus-putus.
	<i>Object Destruction</i>	- Ditempatkan diakhir sebuah <i>lifeline</i> dari sebuah objek

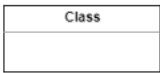

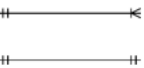
Simbol	Nama	Keterangan
		- Menunjukkan bahwa objek tersebut keluar dari kehidupan <i>sequence</i> .
	<i>Frame</i>	- Mengindikasikan sebuah konteks pada <i>sequence diagram</i> .

Sumber : (Dennis, Wixom, & Tegarden, 2021)

d. Diagram Kelas (*Class Diagram*)

Class diagram merupakan visualisasi dari struktur sistem program yang dibentuk. Diagram kelas juga merupakan gambaran struktur sistem dalam hal kelas-kelas yang dibuat untuk membangun sistem (Putra, 2019). Berikut adalah komponen dari *class diagram* diagram yang terdapat pada Tabel 2.4:

Tabel 2.4 Komponen *Class Diagram*

Simbol	Nama	Keterangan
	<i>Class</i>	Himpunan terhadap objek dengan atribut serta operasi yang sama
	<i>Association</i>	Penghubung antara objek satu dengan objek lainnya
 0..* 1..* 1..1	<i>Cardinality</i>	Jumlah maksimum entitas yang saling berelasi.

Sumber : (DENNIS, WIXOM, & ROTH, 2012)

2.2. Tinjauan Studi

Literatur review atau tinjauan studi yang dilakukan penulis sebagai pendukung penulisan riset penelitian, literature review ini berkaitan dengan topik yang dibahas penulis. Berikut beberapa referensi literature review :

1. Jurnal penelitian yang disusun oleh Muhammad Faizal dan Dhea Astrina Januar pada tahun 2013 dengan judul **“APLIKASI SISTEM PERBANTUAN KOMPUTER UNTUK PEMINJAMAN BARANG (STUDI KASUS LIPI SUBANG)”**, pada jurnal penelitian ini membahas mengenai masalah yang terjadi pada Pusat Penelitian Informatika LIPI-Subang terkait kurangnya pemanfaatan komputer secara optimal pada Memberikan informasi tentang perangkat terkait peminjam atau mengembalikan perangkat. Dengan tidak memanfaatkan komputer secara optimal, sering kali menimbulkan berbagai permasalahan seperti kesulitan untuk mengetahui keberadaan barang dan persediaan stock. Selain itu dengan tidak adanya formulir peminjaman serta pengembalian barang, menyebabkan bidang sarana kehilangan barang karena tidak adanya bukti identitas peminjam. Padahal seluruh divisi yang ada di Pusat Penelitian Informatika LIPI-Subang saling membutuhkan sarana yang ada di LIPI, namun dengan tidak adanya pemanfaatan komputer dengan baik, petugas dalam peminjaman peralatan agak sulit untuk mengelola semuanya. Oleh karena itu diperlukannya sebuah sistem perbantuan komputer untuk peminjaman barang agar mempermudah pekerjaan user dan petugas dalam mengklasifikasikan data serta mengelola peralatan. Penelitian ini menggunakan metode *Field Research* dan *Library Research*. Perancangan dalam sistem ini meliputi flowmap, ERD dan DFD.
2. Jurnal penelitian yang disusun oleh Aji Wicaksono pada tahun 2019 dengan judul **“SISTEM INFORMASI INVENTARIS ASET BERBASIS JAVA PADA SMK BOARDING SCHOOL BREBES”**, pada jurnal penelitian ini membahas mengenai masalah yang terjadi pada SMK Boarding School

Brebes, yaitu masih memakai metode pengolahan data secara manual, terutama pada pengolahan data inventaris asset. Metode tersebut dirasa kurang efektif karena sering memperlambat proses pengolahan atau penyusunan data serta pengontrolan atau penjagaan inventaris. Berdasarkan permasalahan tersebut peneliti menggunakan metode penelitian deskriptif kualitatif dan untuk menghasilkan produk berupa *software* sistem informasi pengarsipan dokumen dengan metode pengembangan *waterfall*. Perancangan pada sistem meliputi *use case diagram*, *activity diagram* dan *sequence diagram* untuk menjelaskan *behaviour* yang terjadi pada sistem.

3. Jurnal penelitian yang disusun oleh Erni Indah Saputri dan Paulus Hartanto pada tahun 2014 dengan judul **“SISTEM INFORMASI PENGELOLAAN INVENTARIS BERBASIS CLIENT SERVERPADA PT. CAHAYA AGUNG CEMERLANG SEMARANG”**, pada jurnal penelitian ini membahas mengenai kesulitan dalam mendapatkan data barang inventaris yang masih di dokumentasikan secara manual, sehingga memperlambat proses permintaan data yang diperlukan karena petugas harus menelusuri satu per satu arsip catatan yang telah disimpan sesuai dengan tanggal pembelian, selain itu penyimpanan kondisi barang-barang inventaris belum dilakukan pengecekan secara berkala, sehingga kondisi barang tidak memiliki status. Oleh karena itu penulis mencoba mengusulkan solusi untuk PT. Cahaya Agung Cemerlang dengan membangun sistem informasi inventaris berbasis komputer client-server yang bisa diakses oleh admin, *accounting* dan pimpinan. Dibangunnya sistem informasi inventaris ini agar dapat meningkatkan kualitas pengendalian barang inventaris secara berkala dan memudahkan dalam pembuatan laporan. Metode penelitian ini menggunakan metode *Research and Development* untuk menemukan pengetahuan baru melalui *basic research*.

4. Jurnal penelitian yang disusun oleh Victor Marudut Mulia Siregar pada tahun 2018 dengan judul **“PERANCANGAN SISTEM INFORMASI INVENTARIS BARANG PADA SEKOLAH SMA NEGERI 4 PEMATANGSIANTAR”**, jurnal penelitian ini membahas kurang optimalnya pengolahan data inventaris seperti penyimpanan dan pencarian data inventaris barang karena masih dilakukan secara manual seperti pencatatan data ke dalam buku besar. Hal tersebut mengakibatkan kewalahan para petugas yang menangani inventaris jika ada perpindahan dan kehilangan barang, maka metode pencatatan tersebut dirasa kurang efektif. Sehingga, penulis merancang sistem informasi inventaris barang untuk mendata penyimpanan barang di SMA Negeri 4 Pematangsiantar. Sistem dirancang menggunakan bahasa pemrograman *Visual Basic Net* dan *database MySQL* yang akan mempermudah dalam *record* data inventaris serta pelaporannya. Metode penelitian yang dilakukan meliputi wawancara untuk pengumpulan data. Tahap perancangan dilakukan dengan membuat *Data Flow Diagram* hingga *form* elisitasi. Hasil implementasi dari sistem informasi inventaris adalah proses tata kelola data inventaris barang pada SMA Negeri 4 Pematangsiantar, dapat dilakukan dengan lebih cepat. Serta sistem yang dibangun dapat menghasilkan laporan yang tepat dan akurat.

5. Jurnal penelitian yang disusun oleh Uci Rahmalisa pada tahun 2018 dengan judul **“PERANCANGAN SISTEM INFORMASI INVENTARIS BARANG PADA KANTOR DINAS PENDIDIKAN PROVINSI RIAU BERBASIS WEB”**, pada jurnal penelitian ini membahas mengenai UPT Teknologi dan Komunikasi Pendidikan Dinas Pendidikan yang mempunyai kesulitan dalam pengolahan data inventaris barang, dimana pendataan inventaris barang sudah menggunakan komputer, namun aplikasi yang digunakan hanya Microsoft Excel dan belum menggunakan bahasa pemrograman khusus. Dengan ini penulis merumuskan masalah yaitu “Bagaimana cara untuk mengetahui informasi barang di kantor dinas Pendidikan dengan cepat dan akurat?”. Berdasarkan analisa permasalahan

yang dihadapi, pengolahan data inventaris menggunakan software dengan bahasa pemrograman PHP memiliki peranan penting sebagai alat bantu kesuksesan proses bisnis yang ada.

6. Jurnal penelitian yang disusun oleh Wahyu Nugraha, Muhamad Syarif, Weiskhy Steven Dharmawan pada tahun 2018 dengan judul **“PENERAPAN METODE SDLC WATERFALL DALAM SISTEM INFORMASI INVENTORY BARANG BERBASIS DESKTOP”**, pada jurnal penelitian ini membahas mengenai perusahaan manufaktur yang berada di Kota Pontianak, yaitu De Lapisa Cakes Pontianak. Dimana perusahaan manufaktur ini memiliki pangsa pasar yang semakin luas sehingga meningkatkan penjualan, selain itu diketahui juga bahwa sistem pencatatan persediaan barang belum ada, karena masih menggunakan media kertas untuk mencatat setiap transaksi pada perusahaan tersebut. Hal tersebut mengakibatkan laporan yang dihasilkan sering terjadi kesalahan karena hilang dan rusaknya bukti transaksi. Oleh karena itu penulis merumuskan masalah untuk mengharuskan adanya sistem informasi *inventory* berbasis desktop agar dapat mengatur persediaan barang yang dimiliki oleh De Lapisa Cakes. Selain itu dengan adanya sistem informasi *inventory* ini diharapkan dapat mengurangi *humans error* pada proses pembuatan laporan maupun proses pencatatan transaksi. Rancangan yang dibuat menggunakan bahasa pemrograman Visual Studio 2010 dan menerapkan metode SDLC *Waterfall* sebagai metode pengembangan perangkat lunak.