

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Pengelolaan surat merupakan hal yang sangat penting bagi sebuah organisasi melalui surat setiap bagian dari organisasi baik yang berada di dalam maupun yang berada diluar melakukan komunikasi melalui surat. berbagai informasi tentang organisasi disampaikan melalui surat, dalam organisasi sendiri surat dibagi menjadi surat masuk dan surat keluar. Surat adalah sebuah alat komunikasi tertulis yang digunakan oleh pengirim untuk mengirimkan berita atau informasi kepada penerima. Surat pula dapat dijadikan sebagai bukti atau dasar untuk melakukan tindakan tertentu. Oleh karena itu penanganan pengelolaan surat perlu mendapatkan perhatian yang tinggi, terutama pada kegiatan pengarsipan untuk menjaga agar surat dapat digunakan kapan saja. (Guntari & Setiawan, 2016)

Pengarsipan memegang peranan penting dalam suatu organisasi, manajemen administrasi, Pengertian Arsip di Indonesia, diatur dalam Undang-Undang No.7 Tahun 1971 tentang “Ketentuan Pokok Kearsipan” pada Bab I pasal 1 berbunyi: “Naskah-naskah yang dibuat dan diterima oleh lembaga Negara dan badan-badan pemerintahan dalam bentuk corak apapun, baik dalam keadaan tunggal maupun berkelompok dalam rangka pelaksanaan kegiatan pemerintah dan Naskah-naskah yang dibuat dan diterima oleh badan-badan swasta dan/atau perorangan dalam bentuk corak apapun, baik dalam keadaan tunggal maupun berkelompok, dalam rangka pelaksanaan kebangsaan” (Kusuma & Aryati, 2019)

Sistem Pengarsipan atau sistem kearsipan adalah cara pengaturan atau penyimpanan arsip secara logis dan sistematis dengan memakai abjad, numerik/nomor, huruf ataupun kombinasi huruf dan nomor sebagai identitas arsip yang terkait. Sistem ini dibuat untuk mempermudah dalam penyimpanan dan penemuan kembali arsip. Hal ini merupakan bagian dari pekerjaan kantor yang sangat penting. (Rustamin & Dewi, 2016)

Berdasarkan hasil wawancara peneliti dengan Kepala Sekolah SD Negeri 39 Kambo, ia mengatakan selama ini proses administrasi surat-menyurat di SD Negeri 39 Kambo masih secara manual yaitu dengan mengumpulkan surat yang masuk ke dalam suatu map begitupun dengan surat keluar, dengan banyaknya

jumlah surat yang dibuat dan diterima oleh SD Negeri 39 Kambo, maka pencarian data akan menjadi tidak efisien dalam hal waktu dan tenaga. Kekurangan dari sistem yang manual ini adalah data tersebut tidak dapat tersimpan dengan baik, apalagi jika kita ingin mencari surat, akan menjadi sulit ini dikarenakan surat-surat menumpuk sehingga memakan waktu dan dikhawatirkan data yang ada sebelumnya hilang ataupun tercecer.

Perancangan dan pembangunan aplikasi Pengarsipan Surat Masuk dan Surat Keluar merupakan salah satu langkah untuk membantu mengatasi permasalahan tersebut. Aplikasi ini diharapkan mampu mempermudah dalam pengarsipan surat yang masuk dan keluar, dan juga mempermudah dalam pencarian data dari surat masuk dan surat keluar, dengan memasukkan nomor surat kita sudah bisa melihat data surat yang kita inginkan.

Berdasarkan penjelasan latar belakang di atas maka penulis mengangkat judul “ Rancang Bangun Aplikasi Pengarsipan Surat Masuk dan Surat Keluar Di SD Negeri 39 Kambo”

## **1.2 Rumusan Masalah**

Berdasarkan latar belakang yang dijelaskan di atas maka yang menjadi rumusan masalah adalah “Bagaimana merancang dan membangun aplikasi pengarsipan surat masuk dan surat keluar pada SD Negeri 39 Kambo”?

## **1.3 Tujuan Penelitian**

Tujuan dari penelitian ini yaitu untuk merancang dan membangun sebuah aplikasi pengarsipan surat masuk dan surat keluar di SD Negeri 39 Kambo.

## **1.4 Manfaat Penelitian**

Adapun manfaat dari penelitian ini adalah:

### **1. Manfaat Bagi Peneliti**

Manfaat yang di peroleh peneliti adalah dapat berpikir mengenai penerapan teori yang telah di peroleh selama proses perkuliahan ke dalam penelitian yang sebenarnya.

## 2. Manfaat Bagi Universitas Cokroaminoto Palopo

Penelitian ini di harapkan dapat menjadi sumber referensi tambahan sebagai bahan penelitian lanjutan yang lebih mendalam pada masa yang akan datang khususnya yang berkaitan dengan Rancang Bangun di Universitas Cokroaminoto Palopo.

## 3. Manfaat Bagi Peneliti Lain

Hasil dari penelitian ini dapat dijadikan acuan oleh peneliti lain yang yang mengambil topik yang sama sebagai bahan perbandingan untuk pengembangan penelitian ini atau untuk pembuatan inovasi baru.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Kajian Teori**

Kajian teori merupakan seperangkat definisi, konsep, serta proposisi yang telah disusun dengan rapi serta sistematis tentang variable-variabel dalam sebuah penelitian. Kajian teori ini akan menjadi dasar yang kuat dalam sebuah penelitian yang akan dilakukan.

##### **1. Rancang Bangun**

Rancang bangun diartikan oleh Ariansyah dkk. (2017) adalah suatu istilah umum untuk membuat atau mendesain suatu objek dari awal pembuatan sampai akhir pembuatan. Rancang bangun berawal dari kata desain yang artinya perancangan, rancang, desain, bangun. Sedangkan merancang artinya mengatur, mengerjakan atau melakukan sesuatu dan perancangan artinya proses, cara, perbuatan merancang. Dapat disimpulkan arti kata desain adalah proses, cara, perbuatan dengan mengatur segala sesuatu sebelum bertindak atau merancang. Rancang Bangun adalah kemampuan untuk membuat beberapa alternatif pemecahan masalah. Adapun dari pendapat lain rancang bangun adalah spesifikasi umum dan terinci dari pemecahan masalah berbasis computer yang telah dipilih selama tahap analisis. Rancang bangun merupakan kegiatan menerjemahkan hasil dan analisa kedalam bentuk paket perangkat lunak kemudian menciptakan sistem tersebut ataupun memperbaiki sistem yang sudah ada sebelumnya. (Zulfiandri dkk 2014).

Berdasarkan definisi di atas penulis dapat menarik kesimpulan bahwa rancang bangun merupakan kegiatan untuk menerjemahkan hasil analisa yang digunakan untuk membangun atau memperbaiki sistem yang sudah ada atau mendesain suatu objek dari awal pembuatan sampai akhir pembuatan,

## 2. Aplikasi

Menurut Guntari & Setiawan (2016). Aplikasi adalah pengguna dalam suatu komputer, instruksi (*instruction*) atau pernyataan (*statement*) yang disusun sedemikian rupa sehingga komputer dapat memproses *input* menjadi *output*. Aplikasi adalah salah satu *unit* perangkat lunak yang dibuat untuk melayani kebutuhan akan beberapa aktivitas seperti sistem perniagaan, game pelayanan masyarakat, periklanan, atau semua proses yang hampir dilakukan manusia.

Yahya & Nurmalina (2015). Perangkat lunak aplikasi adalah suatu sub kelas perangkat lunak komputer yang memanfaatkan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pengguna. Biasanya dibandingkan dengan perangkat lunak sistem yang mengintegrasikan berbagai kemampuan komputer, tapi tidak secara langsung menerapkan kemampuan tersebut untuk mengerjakan suatu tugas yang menguntungkan pengguna. Contoh utama perangkat lunak aplikasi adalah pengolah kata, lembar kerja, dan pemutar media. Sedangkan pengertian dari aplikasi adalah penerapan, menyimpan sesuatu hal, data, permasalahan pekerjaan kedalam suatu sarana atau media yang digunakan untuk menerapkan atau mengimplementasikan hal atau permasalahan tersebut.

Secara istilah pengertian aplikasi menurut Juansyah (2015) Adalah suatu program yang siap untuk digunakan yang dibuat untuk melaksanakan suatu fungsi bagi pengguna jasa aplikasi serta penggunaan aplikasi lain yang dapat digunakan oleh suatu sasaran yang akan dituju. Menurut kamus komputer eksekutif, aplikasi mempunyai arti yaitu pemecahan masalah yang menggunakan salah satu teknik pemrosesan data aplikasi yang biasanya berpacu pada sebuah komputansi yang diinginkan atau diharapkan maupun pemrosesan data yang di harapkan.

Berdasarkan definisi di atas penulis dapat menarik kesimpulan bahwa aplikasi adalah suatu sub kelas perangkat lunak komputer yang memanfaatkan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pengguna yaitu instruksi (*instruction*) atau pernyataan (*statement*) yang disusun sedemikian rupa sehingga komputer dapat memproses *input* menjadi *output*.

### 3. Arsip

Menurut Rustamin & Dewi (2016) Arsip adalah proses pengaturan dan penyimpanan rekaman asli (*original record*), atau salinnya, sehingga rekaman tersebut dapat ditemukan dengan mudah sewaktu diperlukan. Arsip dapat berupa surat, warkat, akta, piagam, buku, dan sebagainya, yang dapat dijadikan bukti sah untuk suatu tindakan dan keputusan.

Lestanti & Susana (2016) Menyatakan bahwa Arsip berperan sangat penting dalam administrasi. Peranan penting arsip dalam administrasi adalah sebagai ingatan dan sumber informasi dalam rangka melakukan kegiatan perencanaan, *penganalisaan*, perumusan kebijaksanaan, pengambilan keputusan, pembuatan laporan, penilaian, pengendalian dan pertanggung jawaban dengan setepat-tepatnya.

Mustika dkk. (2018) Menyatakan bahwa arsip dari segi fungsinya terbagi menjadi dua yaitu arsip dinamis dan arsip statis. Arsip dinamis merupakan arsip yang masih digunakan dalam kegiatan sehari-hari. Sedangkan arsip statis merupakan arsip yang tidak digunakan secara langsung dalam proses perencanaan, penyelenggaraan kehidupan berbangsa pada umumnya maupun untuk penyelenggaran sehari-hari administrasi Negara.

Berdasarkan definisi di atas penulis dapat menarik kesimpulan bahwa arsip merupakan proses pengaturan dan penyimpanan rekaman asli (*original record*). Arsip berperan sangat penting dalam administrasi sebagai ingatan dan sumber informasi. Arsip dari segi fungsinya terbagi menjadi dua yaitu arsip dinamis dan arsip statis.

### 4. Surat

Menurut Suryatama (2017). Surat merupakan salah satu alat komunikasi tertulis yang dipergunakan dalam suatu organisasi. Surat sebagai salah satu media dalam berkomunikasi, mempunyai sasaran kegiatan pada pengumpulan bahan-bahan keterangan, baik tertulis, terekam, tercetak maupun tergambar. Surat merupakan bagian yang tidak terpisahkan dari aktivitas manusia pada zaman modern pada saat ini. Didorong oleh tuntutan kebutuhan ekonomi dan sosialnya manusia menjalin hubungan yang semakin luas dengan berbagai individu, baik yang berada disekitarnya maupun ditempat lain

Syafriadi (2015) Menyatakan bahwa surat adalah sarana komunikasi untuk menyampaikan informasi tertulis oleh suatu pihak kepada pihak lain. Fungsinya mencakup tujuh hal, yaitu sarana pemberitahuan, permintaan, buah pikiran dan gagasan, alat bukti tertulis, alat pengingat, bukti historis, dan pedoman kerja. Surat masuk adalah semua jenis surat yang diterima dari instansi lain maupun dari perorangan, baik yang diterima melalui pos (kantor pos) maupun yang diterima dari kurir (pengiriman surat) dengan mempergunakan buku pengiriman. Surat keluar ialah surat yang lengkap (bertanggal, bernomor, berstempel dan telah ditandatangani oleh pejabat yang berwenang) yang dibuat oleh instansi atau lembaga lain.

Syafriadi (2015) mengemukakan bahwa prosedur pengolahan surat masuk dan surat keluar adalah pekerjaan surat menyurat yang harus dilakukan secara tertata dan berurutan dengan kegiatan yang utama yaitu mengelola, mengatur, dan mengurus surat menyurat agar dapat memperlancar administrasi instansi tersebut.

Berdasarkan definisi di atas penulis dapat menarik kesimpulan bahwa Surat merupakan salah satu media dalam berkomunikasi, mempunyai sasaran kegiatan pada pengumpulan bahan-bahan keterangan, baik tertulis, terekam, tercetak maupun tergambar. Surat ialah sarana komunikasi untuk menyampaikan informasi tertulis oleh suatu pihak kepada pihak lainnya, prosedur pengolahan surat masuk dan surat keluar adalah pekerjaan surat menyurat yang harus dilakukan secara tertata dan berurutan.

## **5. *Visual Basic 6.0***

Menurut Andi (2009) *Visual basic* adalah bahasa pemrograman tingkat tinggi yang sudah sangat terkenal, dimulai dengan *basic* yang terdapat pada komputer “angkatan tua”, seperti AT286. Pada saat itu, bahasa *basic* merupakan sebuah bahasa yang sangat diandalkan dalam pembuatan beberapa aplikasi penting. *Basic* digemari karena susunan programnya yang membebaskan kita untuk “melompat” dari suatu baris program ke baris lainnya. Seiring dengan berkembangnya sistem operasi ke sistem operasi berbasis grafik, para pengguna DOS beralih ke *windows*. Keberadaan *Windows* mengilhami para programmer untuk menciptakan program dengan tampilan grafik yang mirip dengan *Windows*,

demikian juga *basic* yang dikembangkan dengan tambahan beberapa fungsi untuk menciptakan tampilan grafik seperti *Windows*.

Menurut Leong (2006) *Visual Basic* adalah salah satu dari berbagai bahasa pemrograman yang dimengerti oleh komputer. *Visual Basic* merupakan bahasa pemrograman yang lebih sederhana dibanding bahasa pemrograman yang lain karena *Visual Basic* membebaskan pemrogram dari penulisan perintah atau intruksi yang kompleks sehingga langkah pemrograman menjadi lebih sederhana.

Menurut Puspita & Yuliana (2016) program pada *visual basic* berfungsi menyatukan kontrol-kontrol yang ada didalam aplikasi, contohnya jika kita ingin membuat sebuah aplikasi untuk menghitung luas segitiga, maka kita harus menuliskan rumus luas segitiga tersebut ke dalam bentuk kode program *visual basic*. Dalam *visual basic* pembuatan program aplikasi harus dikerjakan dalam sebuah *project*. Sebuah projek dapat terdiri dari *file project (.vbp)*, *file form (.frm)*, *file data binary (.frx)*, *Modul Class (.cls)*, *Modul standar (.bas)*, dan *file resource tunggal (.res)*, bahasa yang digunakan adalah bahasa yang sangat populer pada era sistem operasi DOS.

Berdasarkan definisi di atas penulis dapat menarik kesimpulan bahwa *Visual Basic* merupakan bahasa pemrograman yang lebih sederhana dibanding bahasa pemrograman yang lain karena *Visual Basic* membebaskan pemrogram dari penulisan perintah atau intruksi yang kompleks sehingga langkah pemrograman menjadi lebih sederhana. *Visual Basic* adalah salah satu aplikasi pemrograman under *Windows* yang berbasis pada visual atau grafis

## 6. Microsoft Access

Menurut Ashari (2006) Aplikasi *Microsoft Access* adalah aplikasi untuk pengolahan data. Aplikasi ini merupakan aplikasi *database* yang memungkinkan pengguna untuk melakukan manipulasi data menggunakan fasilitas yang diberikan oleh *Microsoft Access*. Sebuah *database* dalam program *Microsoft Access* akan terdiri atas sekumpulan data atau informasi yang akan disimpan dalam tabel. Tabel-tabel ini akan terdiri atas baris yang disebut juga dengan *record* dan kolom (*field*).



Menurut Sarwandi & Creative (2017) *Microsoft Office Access* atau lebih populer dengan sebutan *Microsoft Access* adalah sebuah program aplikasi basis data *computer* relasional yang ditunjukkan untuk kalangan rumahan dan perusahaan kecil hingga menengah. Aplikasi ini merupakan anggota dari beberapa aplikasi *Microsoft Office*, selain tentunya *Microsoft word*, *Microsoft Excel*, dan *Microsoft PowerPoint*. Aplikasi ini menggunakan mesin basis data *Microsoft jet Database Engine*, dan juga menggunakan tampilan grafis yang intuitif untuk memudahkan pengguna. Para pengguna/*programmer* yang mahir dapat menggunakannya untuk mengembangkan perangkat lunak aplikasi yang kompleks, sementara para programmer yang kurang mahir dapat menggunakannya untuk mengembangkan perangkat lunak aplikasi yang sederhana.

Kadir (2010) *Microsoft Office Access 2007* tergolong sebagai DBMS (*DataBase Management System*). Perangkat lunak ini bermanfaat untuk mengelola data dengan cara yang sangat fleksibel dan cepat. Berikut adalah sejumlah aktivitas yang terkait dengan data yang didukung oleh perangkat lunak tersebut.

- a. Menyimpan data dengan ukuran yang tidak terbatas.
- b. Mengatur data yang tersimpan agar selalu konsisten.
- c. Mengambil data yang tersimpan dengan mudah dan cepat.
- d. Memungkinkan untuk memilih hanya data tertentu yang diambil.
- e. Memungkinkan untuk mengombinasikan teks, gambar, dan bahkan suara.
- f. Memasukkan data yang dapat ditangani melalui suatu formulir.
- g. Menyajikan informasi dengan cepat dan mudah melalui fasilitas pembangkit laporan.
- h. Mengimpor data dari format lain (misalnya dari *Excel*).
- i. Mengekspor data ke dalam format lain (misalnya ke bentuk RTF atau file teks)

Berdasarkan definisi di atas penulis dapat menarik kesimpulan bahwa *Microsoft Access* adalah sebuah sistem manajemen *database* atau (DBMS), dengan *Microsoft Access* dapat menyimpan berbagai macam informasi yang disebut data, *Access* Diciptakan untuk membuat dan menangani suatu *database*, seperti *database* Apage.

## 7. *Crystal Reports*

Menurut Madcoms (2010:234), *Crystal Reports* adalah program yang terpisah dengan program *Microsoft Visual Basic 6.0*, tetapi keduanya dapat dihubungkan (*linkage*). Membuat laporan dengan *Crystal Report* lebih baik dan lebih mudah, karena pada *Crystal Report* tersedia banyak objek dan komponen yang mudah digunakan.

*Crystal Report* merupakan aplikasi untuk merancang dan menghasilkan laporan dari berbagai sumber data (*database*) dan dapat digunakan dengan berbagai bahasa pemrograman berbasis *Windows*, seperti *Visual Basic*, *Visual Cch*, *Visual Interdev*, dan *Borland Delphi*. Beberapa kelebihan yang dimiliki *Crystal Reports* sebagai berikut:

- a. Pembuatan laporan dengan *Crystal Reports* tidak terlalu rumit dengan banyak melibatkan kode program.
- b. *Crystal Reports* banyak digunakan karena mudah terintegrasi dengan bahasa pemrograman lain.
- c. Fasilitas *impor* hasil laporan yang mendukung format-format paket program lain, seperti *Microsoft Office*, *Adobe Acrobat Reader*, *HTML*, dan sebagainya.

Pada standar *repts expert* terdiri dari delapan langkah yaitu:

- a. *Tab data*  
*Tab data* ini harus menentukan tabel atau *query* yang akan digunakan untuk membuat laporan
- b. *Tab fields*  
*Tab* ini dapat menentukan *field-field* atau kolom-kolom yang akan ditampilkan dalam laporan.
- c. *Tab sort*  
*Tab* ini dapat mengurutkan atau mengelompokkan data-data yang akan ditampilkan dalam laporan.
- d. *Tab total*  
*Tab* ini dapat menampilkan subtotal berdasarkan kriteria atau pengelompokan berdasarkan *fields* yang telah ditentukan pada *tab* sebelumnya.

e. *Tab top N*

*Tab* ini dapat menentukan data yang akan ditampilkan dengan cara memilih *N* terbesar dari data yang ada.

f. *Tab graph*

*Tab* ini dapat membuat dan memilih grafik yang diinginkan.

g. *Tab select*

*Tab* ini dapat menyaring *record-record* yang akan ditampilkan dalam laporan.

h. *Tab style*

*Tab* ini dapat memilih bentuk tampilan laporan dan menuliskan judul laporan yang diinginkan (Wali, 2017:87).

## 8. Microsoft Visio 2007

Digibook, (2007) Secara garis besar *microsoft visio* adalah program aplikasi komputer yang paling efisien untuk gambar profesional. *Microsoft visio* akan membantu mempermudah pekerjaan pembuatan gambar pada *tool-tool* menu yang *special, shape, stentil* dan *setting drawing* sehingga dapat membuat gambar sesuai keinginan.

komputer, (2013) *Microsoft visio* adalah aplikasi yang digunakan membuat diagram dan *flowchart*. Aplikasi ini banyak menyediakan banyak fasilitas untuk pembuatan diagram dan *flowchart* untuk mendeskripsikan informasi atau sistem, dari bentuk penjelasan berupa teks hingga gambar diagram yang memberikan penjelasan. Untuk mempelajari *microsoft visio* tidak perlu teknik yang tinggi karena *visio* sangat mudah digunakan dan diterapkan.

Enterprise (2008: 1) menjelaskan bahwa *visio* adalah perangkat lunak yang digunakan untuk membuat dan membantu mengungkapkan ide atau konsep apapun dalam bentuk *flowchart*, diagram dan grafik-grafik lainnya. Penggunaan *microsoft visio* akan memudahkan dalam mendesain berbagai desain diagram karena tersedianya filter dan *template* yang *special* sesuai kebutuhan.

## 9. UML (*Unified Modeling Language*)

UML menurut Munawar (2015) *Unified Modeling Language* adalah salah satu alat bentuk yang sangat handal di dunia pengembangan sistem yang berorientasi objek. Hal ini disebabkan karena UML menyediakan bahasa

pemodelan visual yang memungkinkan bagi pengembang sistem untuk membuat cetak biru atas visi mereka dalam bentuk yang baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi (*sharing*) dan mengkomunikasikan rancangan mereka dengan yang lain. UML merupakan kesatuan dari bahasa pemodelan yang dikembangkan oleh *Booch*, *Object Modeling Technique* (OMT) dan *Object Oriented Software Engineering* (OOSE).

Menurut Suhendar & Guandi (2002) merupakan sistem arsitektur yang berkerja dalam OOAD dengan satu bahasa yang konsisten untuk menentukan, visualisasi, mengkonstruksi, dan mendokumentasikan *artifact* yang terdapat dalam sistem *software*. UML merupakan bahasa pemodelan yang paling sukses dari tiga metode OO yang telah ada sebelumnya, yaitu *Booch*, OMT, and OOSE. UML merupakan kesatuan dari ketiga metode pemodelan tersebut dan ditambah kemampuan lebih karena mengandung metode tambahan untuk mengatasi masalah pemodelan yang tidak dapat ditangani ketiga metode tersebut.

Menurut Muslihudin & Oktafianto (2016) UML memiliki sintaks dan semantik. Ketika kita membuat model menggunakan konsep UML ada aturan-aturan yang harus diikuti. Bagaimana elemen pada model-model yang kita buat berhubungan satu dengan yang lainnya harus mengikuti standar yang ada. UML bukan hanya sekedar diagram, tetapi juga menceritakan konteksnya. UML diaplikasikan untuk maksud tertentu, biasanya antara lain untuk: 1. Merancang perangkat lunak, 2. Sarana komunikasi antara perangkat lunak dengan proses bisnis, 3. Menjabarkan sistem rinci untuk analisis dan mencari apa yang diperlukan sistem, 4. Mendokumentasikan sistem yang ada, proses-proses dan organisasinya.

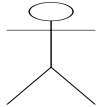


#### **a. Use Case Diagram**

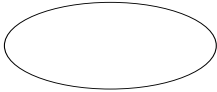
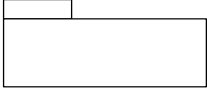


Menurut Rosa & Shalahuddin (2018) *Use case* atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih *actor* dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

Syarat penamaan pada *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut *actor* dan *use case* yaitu: 1. *Actor* merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi itu sendiri, 2. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan anatar unit atau *actor*. *Use case* nantinya akan menjadi kelas proses pada diagram kelas sehingga perlu dipertimbangkan penamaan yang dilakukan apakah sudah layak menjadi kelas atau belum sesuai dengan aturan pendefinisian kelas yang baik.

Diagram di bawah ini menunjukkan fungsionalitas suatu sistem atau kelas dan bagaimana sistem berinteraksi dengan dunia luar. *Use case diagram* dapat digunakan selama proses analisis untuk menangkap *requirements* sistem dan untuk memahami bagaimana sistem seharusnya bekerja. Selama tahap desain, *use case diagram* menetapkan perilaku (*behavior*) sistem saat diimplementasikan. Dalam sebuah model mungkin terdapat satu atau beberapa *use case diagram*. *Use case diagram* merupakan suatu diagram yang berisi *use case*, *actor*, *relationship* diantaranya. *Use case diagram* merupakan titik awal yang baik dalam memahami dan menganalisis kebutuhan sistem pada saat perancangan sistem yang diusulkan. *Use case diagram* dapat digunakan untuk menentukan kebutuhan apa saja yang diperlukan dari suatu sistem. Simbol-simbol yang digunakan dalam *use case diagram* yaitu:

Tabel 1. *Use Case Diagram*

No	Simbol	Keterangan
1		<i>Actor</i> , segala sesuatu yang berinteraksi dengan sistem aplikasi komputer.
2		<i>Include</i> , Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.
3		<i>Interaction</i> , untuk menunjukkan baik alur pesan atau informasi antara objek maupun hubungan antara objek.

4		<i>Use case</i> , menjelaskan yang dilakukan <i>actor</i> sistem untuk mencapai tujuan tertentu.
5		<i>Package</i> , komentar atau wadah konseptual yang digunakan untuk mengelompokkan elemen-elemen dari sistem yang sedang dibangun, sehingga bisa dibuat model yang lebih sederhana.
6		<i>Extend</i> , Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
7		<i>Association</i> , Apa yang menghubungkan antara objek satu dengan objek lainnya.

---


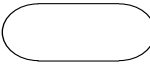



Sumber : Suhendar & Gunadi (2002)

### **b. Activity Diagram**

Memodelkan alur kerja (*workflow*) sebuah proses bisnis dan urutan aktivitas dalam suatu proses. *Diagram* ini sangat mirip dengan sebuah *flowchart* karena kita dapat memodelkan sebuah alur kerja dari satu aktivitas ke aktivitas lainnya atau dari aktivitas kedalam keadaan sesaat (*state*). Seringkali bermanfaat bila kita membuat sebuah *activity diagram* terlebih dahulu dalam memodelkan sebuah proses untuk membantu kita memahami proses secara keseluruhan. *Activity diagram* juga sangat berguna ketika kita ingin menggambarkan perilaku paralel atau menjelaskan bagaimana perilaku dalam berbagai *use case* berinteraksi.

*Activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan *actor*, jadi aktivitas yang dapat dilakukan oleh sistem. Simbol-simbol yang digunakan dalam *activity diagram* adalah:

Tabel 2. *Activity Diagram*

No	Simbol	Keterangan
1		<i>Activity</i> , Menunjukkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
2		<i>Action, State</i> dari sistem yang mencerminkan eksekusi dari suatu aksi.
3		<i>Initial Node</i> , Bagaimana objek dibentuk atau diwakili.
4		<i>Activity Final Node</i> , Bagaimana objek dibentuk atau dihancurkan.
5		<i>Activity Final Node</i> , Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

Sumber : Suhendar & Gunadi (2002)


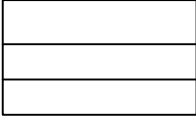



### c. *Class diagram*

Membantu kita dalam visualisasi struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai. *Class diagram* memperlihatkan hubungan antara kelas dan penjelasan detail tiap-tiap kelas didalam model desain (dalam *logical view*) dari suatu sistem. Selama proses analisis *class diagram* memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. Selama tahap desain, *class diagram* berperan dalam menangkap struktur dari semua kelas yang membentuk arsitektur sistem yang dibuat. *Class diagram* juga merupakan fondasi untuk komponen diagram dan *deployment diagram*. Dalam sebuah model mungkin terdapat beberapa kelas dengan spesifikasi tersendiri.

Menurut Rosa & Shalahuddin (2018) diagram kelas dibuat agar pembuat program atau *programmer* membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron. Banyak berbagai kasus, perancangan kelas yang dibuat tidak sesuai dengan kelas-kelas yang dibuat pada perangkat lunak, sehingga tidak ada gunanya lagi sebuah perancangan karena apa yang dirancang dan hasil jadinya tidak sesuai. Kelas-kelas yang ada pada struktur harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem sehingga pembuat perangkat lunak atau *programmer* dapat

membuat kelas-kelas di dalam program perangkat lunak sesuai dengan perancangan diagram kelas. Simbol-simbol yang digunakan dalam *class diagram* adalah:

Tabel 3. *Class Diagram*

No	Simbol	Keterangan
1		Antarmuka, sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
2		<i>Class</i> , Himpunan dari objek yang berbagi atribut serta operasi yang sama.
3		<i>Generalization</i> , Hubungan dimana objek anak berbagi perilaku data dari objek yang ada di atasnya objek induk.
4		<i>Association</i> , Apa yang menghubungkan antara objek yang satu dengan objek yang lainnya.
5		<i>Assosiasi Berarah</i> , Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .

Sumber : Suhendar & Gunadi (2002)

#### d. *Sequence diagram*


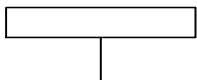
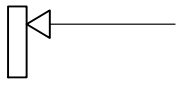
Menurut Rosa & Shalahuddin (2018) *diagram sequence* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan *diagram sequence* maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang di instansiasi menjadi objek itu. Membuat *diagram sequence* juga dibutuhkan untuk melihat skenario yang ada pada *use case*.

Banyaknya *diagram sequence* yang harus digambar adalah minimal sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada *diagram sequence* sehingga semakin banyak *use case* yang didefinisikan maka *diagram sequence* yang harus dibuat juga semakin banyak. Menjelaskan



interaksi objek yang disusun dalam suatu urutan waktu. Diagram ini secara khusus berasosiasi dengan *use case*. *Sequence diagram* memperlihatkan tahap demi tahap apa yang seharusnya terjadi untuk menghasilkan sesuatu di dalam *use case*. Tipe diagram ini sebaiknya digunakan diawal tahap desain atau analisi karena kesederhanaanya dan mudah untuk dimengerti. Simbol-simbol yang digunakan dalam *Sequence diagram* adalah:

Tabel 4. *Sequence Diagram*

No	Simbol	Keterangan
1		<i>Message</i> , Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
2		<i>Lifeline</i> , Objek entity, antarmuka yang saling berinteraksi
3		<i>Message</i> , Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

Sumber : Suhendar & Gunadi (2002)

## 10. Pengujian *Black Box*

Cholifah dkk (2018), Menyatakan bahwa pengujian *Black Box* merupakan Pengujian perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program untuk mengetahui apakah fungsi, masukan dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Metode *Black box Testing* merupakan salah satu metode yang mudah digunakan karena hanya memerlukan batas bawah dan batas atas dari data yang di harapkan, Estimasi banyaknya data uji dapat dihitung melalui banyaknya *field* data entri yang akan diuji, aturan entri yang harus dipenuhi serta kasus batas atas dan batas bawah yang memenuhi. Dan dengan metode ini dapat diketahui jika fungsionalitas masih dapat menerima masukan data yang tidak diharapkan maka menyebabkan data yang disimpan kurang valid.

Menurut Mustaqbal, dkk (2015), *black box testing* cenderung untuk menemukan hal-hal berikut:

- a. Fungsi yang tidak benar atau tidak ada.
- b. Kesalahan antarmuka (*interface error*).
- c. Kesalahan pada struktur data dan akses basis data.
- d. Kesalahan performansi (*performance error*).
- e. Kesalahan inisialisasi dan terminasi.

## 11. Pengujian *White Box*

Menurut Simarmata (2010) pengembangan melaksanakan pengujian unit untuk memeriksa apakah modul tertentu atau kode unit bekerja dengan baik. Pengujian unit berada pada tingkat yang sangat dasar seperti ketika unit kode dikembangkan atau fungsi tertentu dibangun. Pengujian unit berkaitan dengan unit secara keseluruhan. Hal ini akan menguji interaksi antara berbagai fungsi, tetapi membatasi pengujian di dalam satu unit. Lingkup yang tepat dari unit ditinggal kepada interpretasi, pendukung kode pengujian, kadang-kadang disebut perancah (*scaffolding*), mungkin diperlukan untuk mendukung setiap pengujian.

Manfaat yang dapat diperoleh dengan menggunakan teknik pengujian *white box* adalah antara lain: dapat menjamin seluruh *independent path* di dalam modul yang dikerjakan sekurang-kurangnya satu kali, dapat mengerjakan seluruh keputusan *logis*, dapat mengerjakan seluruh perulangan yang sesuai dengan batasannya, dapat mengerjakan seluruh struktur data internal yang menjamin validasi.

Menurut Utami & Asnawati (2015) *white box testing* adalah pengujian yang didasarkan pada pengecekan terhadap detail perancangan, menggunakan struktur kontrol dari desain program secara prosedural untuk membagi pengujian ke dalam beberapa kasus pengujian. Secara sekilas dapat diambil kesimpulan *white box testing* merupakan petunjuk untuk mendapatkan program yang benar secara 100%.

Pengujian testing yang digunakan peneliti yaitu pengujian testing *blackbox* merupakan pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak atau pengujian *blackbox* mengevaluasi hanya dari tampilan luarnya (*interface nya*).

## 2.2 Hasil Penelitian yang Relevan

Penelitian ini yang mengacu pada beberapa penelitian yang sudah ada, guna sebagai referensi dalam penelitian yang dilakukan, penelitian tersebut diantaranya.

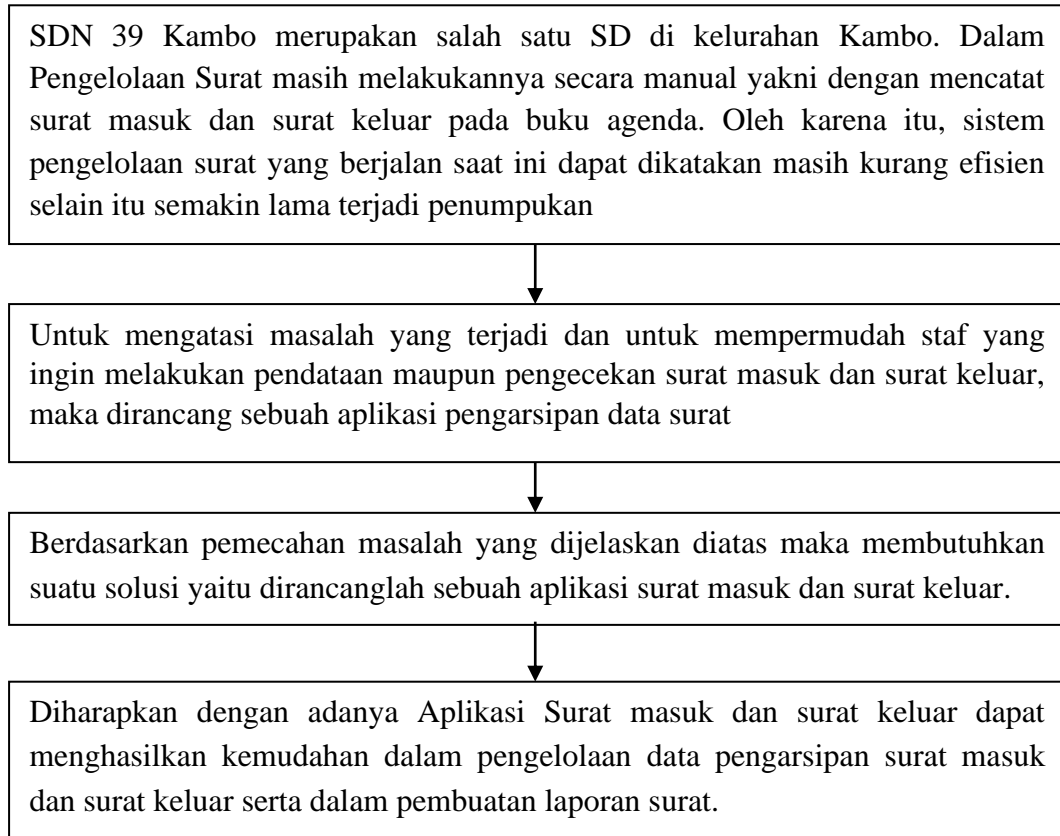
1. Irwin Supriadi, Rina Indrayani & Verra Trysda Maulydda. (2018). *Dalam Penelitiannya yang berjudul “Rancang Bangun Aplikasi Surat Masuk Dan Surat Keluar Berbasis Web pada Kantor Asuransi Jiwa Kantor Layanan Administrasi Bandung”*. Kelebihan dari penelitian ini adalah Aplikasi ini dapat menyimpan berkas surat secara aman serta berkas surat yang disimpan dapat dilihat dan dicari sesuai dengan kebutuhan. Aplikasi ini juga berbasis *web* digunakan untuk mempermudah *user* dalam pencatatan dan pengarsipan data surat masuk dan surat keluar dan Laporan surat pada aplikasi disajikan sesuai dengan tanggal atau waktu yang dibutuhkan serta menampilkan data-data surat yang sesuai dengan pengarsipan surat yang telah dimasukkan. Laporan surat masuk dan surat keluar dilihat berdasarkan periode tanggal surat dan laporan disposisi surat dilihat sesuai data disposisi yang dipilih. Aplikasi ini akan mempermudah pencatatan dan pelacakan surat masuk dan surat keluar sehingga proses akan cepat.
2. Aziz dan Cahyana, (2015). *Dalam penelitiannya yang berjudul “Pengembangan Aplikasi Pengarsipan Surat di Bagian Informatika Sekretariat Daerah Kabupaten Garut Untuk Kemudahan Dalam Pendisposisian Surat”*. Penelitian ini dapat memberikan solusi bagi permasalahan yang ada dengan mengacu pada proses perancangan yang sudah dilakukan sebelumnya, yaitu dapat mempermudah staf bagian umum informatika Kabupaten Garut dalam melakukan pengarsipan surat masuk dan surat keluar dan mempermudah pimpinan bagian informatika untuk melakukan *Development Process* dengan *Unified Modelling Language* yang dikembangkan oleh *Graddy Booch, Ivar Jacobson* dan *James Rumbaugh*. Pemodelan ini memiliki beberapa tahapan yaitu model analisis, model perancangan, model implementasi, model penyebaran dan model pengujian.
3. Putri, Hernawati dan Ananda (2019). *Dalam penelitiannya yang berjudul “Aplikasi Pengarsipan Surat Masuk dan Surat Keluar Berbasis Web pada*

*Dinas Koperasi dan Usaha Kecil Provinsi Jawa Barat*”. Hasil penelitian ini sebuah aplikasi surat masuk dan surat keluar dapat dikembangkan dengan menggunakan metode *waterfall* model dan dibangun dengan bahasa pemrograman PHP dengan menggunakan *Framwork Codeingniter*, selain itu untuk mengelola basis data menggunakan *MySQL*. Dengan tersedianya aplikasi surat masuk dan surat keluar di dinas koperasi dan usaha kecil provinsi jawa barat, dapat memudahkan sekretariat dalam melakukan pengarsipan surat, memudahkan pendesposisian pegawai, dan pemberian keputusan untuk surat keluar.

4. Zulvianita Rustamin dan Anita Puspita Dewi (2016). *Dalam penelitiannya yang berjudul “Sistem Pengarsipan Surat Masuk Dan Surat Keluar Pada Kantor Sekertariat DPRD Provinsi Sulawesi Tenggara Menggunakan Borland Delphi 7”* Penelitian ini menggunakan metode pengembangan perangkat lunak yaitu metode *Prototyping*, metode perancangan basis data dengan menggunakan metode ER dan metode perancangan sistem menggunakan DFD. Penelitian ini juga bertujuan membantu pegawai dalam melakukan pengolahan dan pengarsipan data surat sehingga pencarian surat yang dibutuhkan lebih cepat.
5. Sintia Mahmudah, Lisda Widiastuti dan Siti Ernawati (2019). *Dalam penelitiannya yang berjudul “Sistem Informasi Manajemen Pengarsipan Surat Masuk Dan Surat Keluar (Studi Kasus : Ma Darul Ihya Bogor)”*. Penelitian ini memgggunakan metode *System Development Life Cycle (SDLC)* dengan model *Waterfall*. Model air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential liniear*). Hasil dari penelitian ini adalah membuat sistem informasi manajemen berbasis *website* ini dapat mempermudah bagian tata usaha dalam proses penginputan data surat masuk dan surat keluar, pencarian data surat masuk dan keluar, pembuatan laporan data surat masuk dan keluar sehingga menghasilkan informasi yang dibutuhkan dengan cepat dan akurat.

### 2.3 Kerangka Pikir

Adapun yang menjadi landasan lebih memperjelas kerangka pikir yang akan disajikan, maka digambarkan dalam bentuk diagram sebagai berikut:



Gambar 1. Kerangka Pikir