

## BAB II

### LANDASAN TEORI

#### 2.1 Tinjauan Studi

Penelitian mengenai laporan berbasis *web* telah banyak dilakukan oleh berbagai peneliti dengan masalah dan kerumitan yang bervariasi. Untuk mengetahui lebih jauh sistem yang telah dibuat, ada beberapa penelitian diantaranya adalah sebagai berikut:

Penelitian yang dilakukan oleh Pangkrego Talita Andreta, Pungus Richard Stenly (2016), Penelitian dengan judul “Perancangan Aplikasi Pelaporan Kegiatan Berbasis *Web* Pada BPJN XI Satker Wilayah II Sulawesi Utara”. Tujuan dari penelitian ini adalah untuk membangun aplikasi laporan kegiatan berbasis *web* yang digunakan untuk menangani proses pelaporan kegiatan dari tim teknis pada PPK satker wilayah II Provinsi Sulawesi Utara. Khususnya laporan rekonstruksi atau perbaikan jalan. Metode yang digunakan adalah metode rekayasa perangkat lunak dengan model *Rational Unified Process* (RUP). Metode perancangan aplikasi yang digunakan adalah *Rational Unified Process* (RUP). Berdasarkan pengujian *black box testing* menunjukkan bahwa aplikasi ini dapat berjalan baik, form-form yang terdapat pada aplikasi ini berjalan dengan semestinya dan setiap validasi yang terdapat pada aplikasi ini sesuai dengan perancangan aplikasi. Hasil dari penelitian ini adalah dengan pembuatan aplikasi laporan pada Satker II Provinsi Sulawesi Utara mempermudah pihak tim teknis dalam membuat dan mengakses laporan, serta pihak PPK dapat mengakses laporan tim teknis di mana saja selama tim PPK masih terkoneksi dengan internet dan mengakses alamat *website* laporan ini melalui *smartphone*, laptop maupun PC.

Kemudian dari Lustia Febrita (2015), Penelitian ini berjudul “Aplikasi Pencatatan dan Pelaporan Akuntansi Pada CV.Bunda Membangun Palembang”. Tujuan dari pembuatan aplikasi ini adalah untuk

menghasilkan laporan keuangan yang akurat dan cepat, memudahkan karyawan melakukan pencatatan, pengidentifikasian, dan pengikhtisaran transaksi serta menghasilkan laporan keuangan dengan waktu yang relative cepat. Metode perancangan aplikasi yang digunakan adalah *Iterative* (pengulangan tidak beraturan) pengerjaan dapat dilakukan berulang-ulang. Jika ditemukan kesalahan pada tahap sebelumnya, maka pengerjaan dapat dilakukan langsung pada tahap yang terjadi kesalahan tersebut. Berdasarkan pengujian *blackbox testing* menunjukkan bahwa aplikasi ini dapat berjalan baik, form-form yang terdapat pada aplikasi ini berjalan dengan semestinya dan setiap validasi yang terdapat pada aplikasi ini sesuai dengan perancangan aplikasi. Hasil dari penggunaan aplikasi ini dapat memberi kemudahan pada CV. Bunda Membangun Palembang dalam merekap absensi, menghitung jurnal secara otomatis serta membuat laporan keuangan pada perusahaan, dengan lebih mempersingkat waktu dalam pencarian data yang di perlukan sewaktu-waktu, serta dapat menjaga keamanan data karena di simpan didalam databases.

Selanjutnya Peneliti Madja Hibatur Nur Rahman, Muryani Sri (2017), Penelitian dengan judul “Aplikasi Laporan untuk Menyusun Laporan Keuangan Pada Koperasi Amanah Jakarta”. Dengan adanya aplikasi ini di harapkan dapat meminimalisir kesalahan dalam mengelola data keuangan menjadi laporan keuangan. Metode penelitian ini menggunakan metode riset lapangan, studi pustaka dan wawancara untuk mengumpulkan data. Selanjutnya menggunakan metode *deskriptif*, dimana peneliti mengumpulkan data, menyusun, mengklarifikasi dan menganalisa sehingga dapat memberikan gambaran *objektif* tentang aplikasi laporan keuangan pada Koperasi Amanah ini. Berdasarkan pengujian *blackbox testing* menunjukkan bahwa aplikasi ini dapat berjalan baik, form-form yang terdapat pada aplikasi ini berjalan dengan semestinya dan setiap validasi yang terdapat pada aplikasi ini sesuai dengan perancangan aplikasi. Hasil dari aplikasi ini dapat meminimalisir kesalahan yang ada, mempercepat

penyusunan laporan keuangan, data lebih terorganisir sehingga proses pencarian data lebih cepat, akurat dan efisien.

Berdasarkan masalah yang ada di Balai Pengelolaan Sumber Daya Air Seluna peneliti akan melakukan penelitian guna mempermudah dalam melakukan pekerjaan di seksi OP (Operasi dan Pemeliharaan) khususnya. Aplikasi “Sistem Informasi Pengendalian Operasi dan Pemeliharaan Pada Balai Pengelolaan Sumber Daya Air Seluna Menggunakan *CodeIgniter*”. Diharapkan dengan adanya aplikasi ini dapat mempermudah dalam pengelolaan data Operasi dan Pemeliharaan pada Balai PSDA Seluna. Metode *waterfall* di pilih karena di rasa lebih terstruktur sehingga dapat mengurangi kerugian akibat kesalahan pada proses sebelumnya.

Perancangan aplikasi ini menggunakan *framework CodeIgniter* (CI) karena memiliki dokumentasi yang bagus dan ukuran file yang kecil, selain itu CI sangat muda di intergrasikan dengan library. *CodeIgniter* merupakan *framework* terbaik karena waktu eksekusi yang cepat, file management juga memiliki sub-folder yang dapat bekerja sebagai *config*, model, *controller*, *view* dan lainnya. Keamanan *framework CodeIgniter* juga bagus karena ada fitur yang bisa digunakan pada proyek pengembangan aplikasi ini selanjutnya. Salah satu alasan utama penggunaan *CodeIgniter* karena konfigurasi yang mudah, dengan kelebihan yang di miliki oleh CI ini diharap aplikasi “Sistem Informasi Pengendalian Operasi dan Pemeliharaan Pada Balai Pengelolaan Sumber Daya Air Seluna Menggunakan *CodeIgniter*” bisa lebih mempermudah dalam melakukan pekerjaan di Balai Pengelolaan Sumber Daya Air Seluan.

## **2.2 Tinjauan Pustaka**

### **2.2.1 Sistem Informasi**

Sistem adalah kesatuan bagian yang saling berhubungan berada pada suatu wilayah serta memiliki item penggerak. Kata sistem sering digunakan dalam kehidupan sehari-hari, forum diskusi maupun dokumen ilmiah. Karena kata ini sering digunakan dalam banyak hal, dan pada banyak

bidang, maka arti sistem menjadi beragam. Sehingga pengertian sistem secara umum adalah sekumpulan benda yang memiliki hubungan di antara mereka[3].

Informasi adalah kumpulan pesan/informasi yang di kelola menjadi sebuah laporan sehingga menjadikan sebuah informasi yang berguna bagi semua orang, sifat informasi bisa berupa berita, pengetahuan, peristiwa dan kejadian. Sistem informasi adalah kombinasi teknologi informasi dan aktivitas orang menggunakan teknologi sebagai pendukung operasi dan manajem. Dalam artian luas, istilah informasi yang sering digunakan merujuk pada interaksi antara orang, proses algoritmik, data dan teknologi. Informasi adalah data yang berhubungan dengan keputusan. Menurut Burch & Strater informasi didefinisikan sebagai proses pengumpulan dan pengelolaan data yang ditujukan untuk memberikan keterangan atau pengetahuan tertentu mengenai suatu hal[5].

Aspek manusia dan teknologi memiliki peran yang sangat penting sehingga. Kombinasi dengan elemen pendukung akan membentuk kolaborasi sistem informasi yang didapat. Dengan sistem informasi ini maka dapat mempermudah melakukan aktifitas lebih cepat, efektif dan akurat.



Gambar 2.1 Konsep Dasar Sistem Informasi

Sumber : <http://www.pintarkomputer.com/penjelasan-lengkap-tentang-cara-kerja-sistem-komputer/>

### 2.2.2 Laporan

Laporan adalah suatu bentuk penyampaian berita, keterangan, pemberitahuan dan pertanggung jawaban baik secara lisan maupun tertulis kepada atasan atau pihak berwenang. Adapun jenis laporan yaitu :

Fungsi laporan adalah sebagai berikut[6]:

- 1) Pertanggung jawaban bagi orang yang diberi tugas.
- 2) Landasan pimpinan dalam mengambil kebijakan/keputusan.
- 3) Alat untuk melakukan pengawasan.
- 4) Dokumen sebagai bahan studi dan pengalaman bagi orang lain.
- 5) Sedangkan definisi laporan di dalam seksi Operasi dan Pemeliharaan adalah:

a. Laporan Rencana Anggaran Biaya (RAB)

Dalam laporan RAB diperlukan perhitungan biaya untuk bahan dan upah, serta biaya lainnya. Hal dasar yang digunakan dalam laporan RAB ialah :

1. Volume Pekerjaan
2. Harga Satuan dan Upah
3. Kurva Pekerjaan
4. Rekapitulasi

b. Laporan Harga Perkiraan Sendiri (HPS)

Penyusunan laporan HPS untuk mendapatkan harga penawaran yang wajar sesuai spesifikasi yang telah ditetapkan dan dapat dipertanggungjawabkan oleh penyedia barang sesuai harga di pasaran masing-masing wilayah pekerjaan.

c. Laporan Surat Perintah Kerja (SPK)

Laporan SPK sendiri adalah syarat dan ketentuan yang dijelaskan dalam suatu perintah kerja. Dalam laporan SPK ini setiap instansi mempunyai syarat-syarat yang berbeda-beda. Dasar dalam pembuatan SPK biasanya adalah :

1. Laporan Hasil Pekerjaan
2. Jadwal Pengiriman Barang
3. Absensi Pekerja
4. Laporan Kekuatan Beton
5. Curah Hujan
6. Dokumentasi

### 2.2.3 Balai Seluna

Balai Pengelolaan Sumber Daya Air Seluna atau yang biasa di singkat BPSDA Seluna ini adalah salah satu Instansi pemerintah yang bergerak di bidang Pekerjaan Umum yang berpusat pada sistem irigasi umunya yang bertempat di Kabupaten Kudus. Sesuai dengan Permen PUPR No.20/PRT/M/2016 tentang Organisasi dan Tata Kerja Unit Pelaksanaan Teknis Kementrian Pekerjaan Umum dan Perumahan Rakyat Balai PSDA Seluna bertugas melaksanakan pengelolaan sumber daya air yang meliputi perencanaan, pelaksanaan konstruksi, operasi dan pemeliharaan dalam rangka konservasi dan pendayagunaan sumber daya air dan pengendalian daya rusak air pada sungai, pantai, bendungan, danau, situ, embung dan tampungan air lainnya, irigasi, rawa, tambak, air tanah, dan air baku serta pengelolaan drainase utama perkotaan[1].

Satuan wilayah Sungai Jratun seluna terdiri dari 2 Daerah aliran Sungai (DAS) utama yaitu DAS Jratun (Serang, Lusi, dan Juana). Pada DAS Seluna terdapat 3 lokasi mata air utama, yang tersebar berasal dari pegunungan di Kabupaten Boyolali mengalir Sungai Serang, dan selanjutnya berasal dari pegunungan kapur di Kabupaten Blora dan Kabupaten Grobogan mengalir Sungai Lusi, serta yang ketiga berasal dari pegunungan Muria di Kabupaten Kudus, Kabupaten Pati dan Kabupaten Jepara mengalir Sungai Juana dan drainase Serang Welahan Drain 2 (SWD.2). Sedangkan akhir sungai-sungai pada DAS Seluna bermuara di 4 lokasi yang seluruhnya di Pantai Utara Jawa, tiga berada di pantai Kabupaten Demak yaitu SWD.2, SWD.1 (Sungai Serang Lama), Sungai Wulan (Sungai Serang Hilir) dan muara yang lainnya berada di pantai Kabupaten Pati (Sungai Juana).

Sampai dengan saat ini DAS Seluna masih mengalami beberapa kejadian banjir setiap musim hujan, sehingga ancaman bencana banjir setiap tahun selalu menghantui masyarakat terutama di sekitar sungai Juana yang dikenal dengan Lembah Juana di Kabupaten Kudus dan Kabupaten Pati.

Pada sungai Serang telah dibangun bendungan Kedungombo di Kabupaten Grobogan yang mampu menampung air sebanyak 723 juta m<sup>3</sup> dengan fungsi utama untuk mengairi sawah seluas 60.000 ha tersebar di Kabupaten Grobogan, Kabupaten Demak, Kabupaten Kudus dan Kabupaten Pati. Dengan demikian pengguna air irigasi dalam satu jaringan melibatkan beberapa daerah, hal ini dapat memicu konflik antar daerah.

Untuk meminimalisir kerugian akibat banjir dan konflik sosial diperlukan adanya upaya antisipasi dan manajemen pengelolaan sungai/irigasi dengan prinsip satu sungai/jaringan irigasi satu manajemen oleh satu lembaga atau institusi yang berkemampuan teknis di bidang tersebut.

#### 2.2.4 XAMPP

XAMPP berfungsi sebagai server yang terdiri dari *Apache* HTTP Server, *MySQL database* dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan *Perl*. XAMPP dikembangkan oleh perusahaan *apache friend* yang biasanya digunakan untuk simulasi pengembangan *website*. XAMPP juga dilengkapi dengan fitur manajemen database *phpMyAdmin* seperti pada server hosting sungguhan, sehingga pengembang *web* dapat mengembangkan aplikasi web dengan mudah[7]. Di dalam menu XAMPP ada menu *Apache*. *Apache* adalah aplikasi *web server*. Tugas utama *Apache* adalah menampilkan halaman web yang benar kepada user berdasarkan kode HTML, PHP atau yang lainnya[8]. Dalam pembuatan *web* atau sebuah aplikasi yang menggunakan databases XAMPP sangat membantu.

*Apache* (server Web/WWW) merupakan server yang dapat digunakan untuk memfungsikan dan melayani fasilitas Web/WWW sedangkan server *apache* sendiri dapat di jalankan di banyak sistem operasi seperti (LINUX, *Microsoft Windows*, *Voxel Netware*, *Unix*, BSD serta platform lainnya). Sedangkan *MySQL* adalah perangkat lunak sejuta umat dengan sekitar 6 juta instalasi diseluruh dunia, *MySQL* merupakan perangkat lunak berbasis data SQL atau DBMS yang *multithread*, *multi-*

*user*. Kedua program ini (*Apache* dan *MySQL*) paling sering digunakan untuk melakukan pekerjaan *web* dan aplikasi yang berhubungan dengan *XAMPP*.

*phpMyAdmin* mendukung berbagai operasi *MySQL* melalui *word wide web* (*WWW*) dan bahasa yang digunakan juga cukup familiar di kalangan programmer yaitu bahasa pemrograman *PHP* dan *HTML*. *phpMyAdmin* mendukung berbagai operasi *MySQL* diantaranya mengelola tabel-tabel, perijinan (*permission*), relasi, indeks, *user*, basis data dan lain-lain.

*phpMyAdmin* bukan satu-satunya perangkat lunak yang bisa digunakan untuk membuat suatu *databases*, banyak sekali perangkat lunak yang dapat dimanfaatkan untuk mengelola basis data dalam *MySQL*. Dengan *phpMyAdmin* seorang dapat membuat *databases*, mengisi data, membuat tabel dan lain-lain tanpa harus menghafal baris perintahnya. Untuk membukanya cukup dengan membuka *browser* lalu menetik alamat <http://localhost/phpmyadmin>, maka akan langsung diarahkan ke halaman *phpMyAdmin* di dalam sini nanti kita bisa membuat basis data dan mengelolanya.

### 2.2.5 Database

Kumpulan file yang berelasi ditunjukna dengan kunci dari tiap file. Database dmenunjukkan suatu kumpulan data dan dipakai dalam pembuatan suatu aplikasi untuk pemerintah, swasta dan organisasi.

### 2.2.6 PHP (Hypertext Preprocessor)

*PHP* merupakan Bahasa pemrograman untuk membuat suatu website yang dinamis. *PHP* mampu berinteraksi menuju *database*, folder dan file seperti forum, blog, Website Social Networking dan sebagainya.

## 2.2.7 Framework CodeIgniter

### 1) CodeIgniter

*CodeIgniter* adalah aplikasi *Open Source* berarti *framework* dengan model MVC (*Model, View, Controller*) untuk membuat *web* dinamis dengan menggunakan PHP[9].

Kelebihan *CodeIgniter* adalah:

#### 1. Aplikasi bersifat Open Source

*CodeIgniter* dilisensikan dibawah *Apache/BSD style Open Source*

#### 2. Support PHP 4 dan PHP 5

Meskipun PHP sekarang sudah sampai PHP 6 tapi masih banyak orang yang menggunakan PHP 4. Oleh karena itu *CodeIgniter* dikembangkan agar mampu berjalan baik PHP 4 dan PHP 5

#### 3. Ukuran File Kecil dan Cepat

Kenapa bisa cepat? Karena *CodeIgniter* hanya me-load fungsi atau *library* yang digunakan saja.

#### 4. Dokumentasi

*Framework* yang baik pasti dilengkapi dengan dokumentasi yang lengkap dan mendukung. Didukung oleh *User Guide* yang mudah dimengerti, dari mulai install sampai fungsi-fungsinya semua ada.

#### 5. Memakai Konsep MVC (*Model, View, Controller*)

Konsep modern yang banyak digunakan oleh *framework* PHP lainnya.

#### 6. Komunitas

*Framework CodeIgniter* memiliki komunitas yang bebas untuk siapa saja yang mau bergabung ([situsCodeIgniter.com/formus](http://situsCodeIgniter.com/formus))[10].

### 2) Framework.

*Framework* dapat diartikan kumpulan fungsi/prosedur dan *class* dengan tujuan tertentu untuk mempermudah pekerjaan seorang *programer*[11].

Kelebihan *framework* adalah :

#### 1. Mempercepat dan mempermudah pembangunan aplikasi.

2. Relatif memudahkan dalam proses *maintenance* karena sudah ada pola tertentu didalamnya.
3. Menyediakan fasilitas yang umum dipakai sehingga kita tidak perlu membangun dari awal.
4. Lebih bebas dalam pengembangan.

Kekurangan *framework* adalah:

1. *Update core engine* tidak secepat *framework* lainnya.
2. Masih banyak kelonggaran dalam hal *coding*.
3. Tidak mencerminkan MVC sesungguhnya.

### 3) MVC (*Model, View, Controller*)

MVC memisahkan komponen utama membangun sebuah aplikasi seperti memanipulasi data, *user interface*, dan bagian yang menjadi control aplikasi. Ada 3 jenis komponen dalam membangun MVC yaitu :

#### 1. *View*

*View* merupakan bagian yang menangani *prestation logic*. Pada suatu aplikasi *web* berupa file *templete* HTML, yang di atur oleh *Controller*. *View* berfungsi sebagai penerima dan mempresentasikan data pada *user*

#### 2. *Model*

*Model* biasanya berhubungan langsung dengan *databases* untuk manipulasi data (*Insert, Update, Delete, Searce*).

#### 3. *Controller*

*Controller* merupakan bagian yang mengatur hubungan antara bagian *model* dan *view*. *Controller* berfungsi untuk menerima *request* data dari *user* kemudian menentukan apa yang akan diproses oleh aplikasi [12].

Dengan menggunakan MVC suatu aplikasi dapat dikembangkan sesuai dengan kemampuan *developernya*, yaitu program yang menangani bagian *model* dan *controller*, sedangkan *design* yang menangani bagian *view*, sehingga penggunaan arsitektur MVC dapat meningkatkan *maintability* dan organisasi kode.

### 2.2.8 *Unified Modeling Language (UML)*

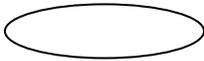
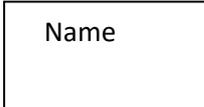
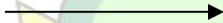
Definisi UML adalah himpunan struktur dan teknik untuk permodelan desain program berorientasi objek (OOP) serta aplikasinya. UML adalah suatu bahasa yang digunakan untuk menentukan, memvisualisasikan, membangun dan mendokumentasikan suatu sistem. UML di kembangkan oleh Grady Boch, Jum Rumbaugh dan Ivar Jacobson. Penggunaan UML dalam penggunaan industry terus meningkat ini menjadikan sabagai bahasa permodelan umum dalam industri peranti lunak dan pengembangan sistem[3].

UML terbuat dari beberapa model yang menjelaskan bagaimana sistem dilakukan desain, model-model digambarkan dalam bentuk diagram-diagram yaitu sebagai berikut:

#### 1) *Use Case Diagram*

*Use case* adalah *abstraksi* dari interaksi antara sistem dan *actor*. *Use case* bekerja dengan cara mendeskripsikan tipe interaksi antara *user* sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai. *Use case* merupakan kontruksi untuk mendeskripsikan bagaimana sistem akan terlihat di mata *user*. Sedangkan *use case* diagram memfasilitasi komunikasi diantara analis dan pengguna serta antara analis dan *client*[3].

Tabel 2.1 Simbol *Use Case Diagram*

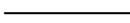
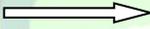
No	Simbol	Nama
1.		Use Case : Interaksi antar sistem.
2.		Actor : Alat komunikasi dengan use case
3.		Subsistem
4.		Association : Penghubung antar actor dan use case.
5.		Generalization : Actor berpartisipasi dengan use case.
6.		Include : fungsionalitas dari use case lainnya.
7.		Extend : tambahan fungsional dari use case lainnya.

Sumber : Rosa A.S dan M. Shalahuddin (2014 : 156)

## 2) *Class Diagram*

*Class Diagram* adalah deskripsi kelompok obyek-obyek dengan *property*, perilaku (operasi) dan relasi yang sama. Sehingga dengan adanya *class diagram* dapat memberikan pandangan global atas sebuah sistem. Hal tersebut tercermin dari *class-class* yang ada dan relasinya satu dengan yang lainnya. Sebuah sistem biasanya mempunyai beberapa *class diagram*. *Class diagram* sangat membantu dalam visualisasi struktur kelas dari suatu sistem [13].

Tabel 2.2 Simbol *Class Diagram*

No	Simbol	Keterangan
1.		Class : menambah class baru pada diagram
2.		Interface : Menambah antar muka pada diagram
3.		Association : Menggambar relasi asosiasi.
4.		Association Class: Menghubungkan class asosiasi.
5.		Generalization : Menggambarkan relasi generalisasi.
6.		Action : actor pada diagram class
7.		Control : kendali pada diagram
8.		Entity : Class entitas pada diagram.

Sumber : Rosa A.S dan M. Shalahuddin (2014 : 156)

### 3) *Activity Diagram*

*Activity diagram* atau disebut dengan aktivitas diagram adalah diagram yang menggambarkan berbagai alir aktivitas dalam sistem yang sedang di rancang, bagaimana masing-masing alur itu berawal, hingga bagaimana mereka berakhir. *Activity diagram* dapat digunakan untuk alur kerja model, *use case* individual, atau logika keputusan yang terkandung dalam metode individual. *Activity diagram* juga menyediakan pendekatan untuk proses pemodelan *paralle*[14].

Tabel 2.3 Simbol *Activity Diagram*

No	Symbol	Keterangan
1.		State : menambah objek
2.		Aktiviti: menambah activity baru
3.		End state : aliran kerja terahir
4.		Detision Point:titik keputusan aliran kerja
5.		State transition : aliran kerja terahir
6.		Start state : aliran kerja awal.

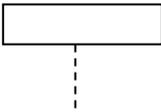
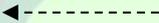
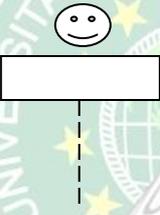
Sumber : Rosa A.S dan M. Shalahuddin (2014 : 156)

#### 4) *Sequence Diagram*

*Sequence diagram* digunakan untuk menggambarkan perilaku pada sebuah *scenario*. Kegunaannya untuk menunjukan rangkaian pesan yang dikirim antar *object* juga interaksi *object*, suatu yang terjadi pada titik tertentu dalam eksekusi sistem [15].

Pesan yang dipertukarkan antara obyek gambar sebagai sebuah anak panah antara action pengirim dan penerima. Kemudian di atasnya diberikan label pesan.

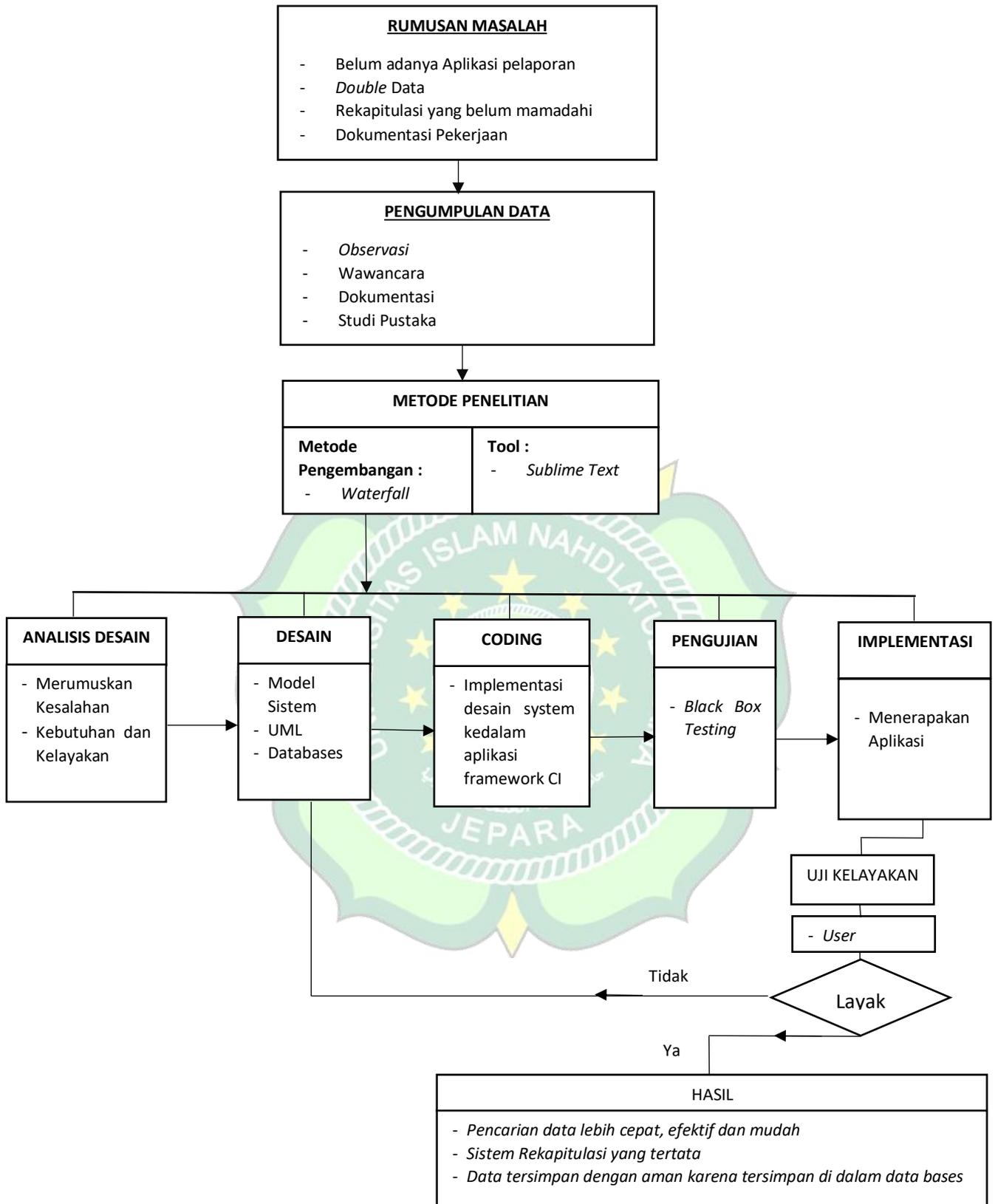
Tabel 2.4 Simbol *Sequence Diagram*

No	Symbol	Keterangan
1.		Object lifeline : menambah objek baru
2.		Activation : mengaktifkan objek baru
3.		Message : menggambar pesan antar dua objek
4.		Return Message : pesan kembali
5.		Self Message : pesan ke dirinya sendiri.
6.		Actor lifeline : menambah aktor baru.

Sumber : Rosa A.S dan M. Shalahuddin (2014 : 156)

### 2.3 Kerangka Pemikiran

Kerangka pemikiran merupakan tahapan yang akan dilakukan dalam proses penyelesaian penelitian dalam rangka memudahkan dalam memecahkan masalah dari awal hingga tercapainya suatu tujuan. Adapun kerangka pemikiran adalah sebagai berikut:



Gambar 2.2 Kerangka Pemikiran