

## BAB II

### LANDASAN TEORI

#### 2.1 Tinjauan Studi

Dalam penelitian, penelitian terdahulu sangatlah penting bagi peneliti untuk dapat dijadikan sebagai pedoman untuk menentukan perbedaan maupun persamaan dengan penelitian yang dilakukan sebelumnya dan dapat dijadikan pegangan peneliti untuk mempermudah peneliti melakukan penelitian sesuai dengan tema. Beberapa referensi sebagai dasar pelaksanaan penelitian ini diantaranya :

Pada Jurnal pertama yang telah dilakukan oleh Sri Indah Darmawati dan Hendrie Noprisson , dengan judul **Analisa dan Perancangan Aplikasi Monpelsis untuk Monitoring Pelanggaran Siswa** dalam penelitian ini dijelaskan bahwa permasalahan yang terjadi adalah belum adanya sistem informasi pelanggaran siswa dalam lingkungan sekolah. Tidak adanya sistem informasi ini peneliti membuat perancangan dalam hal sistem akademik guna untuk mmbantu dan mempermudah pihak sekolah memantau pelanggaran siswa. Pada penelitian ini dijelaskan menggunakan metode *Waterfall* untuk membangun aplikasi ini. Dalam menganalisis masalah peneliti menggunakan *Diagram Fishbone* yang pertama kali dikembangkan oleh Dr, Kaoru Ishikawa sekitar tahun 1960-an. Diagram ini merupakan alat unruk mengidentifikasi, mengeksplorasi dan secara grafik menggambarkan secara detail semua penyebab yang berhubungan dengan suatu permasalahan. Diagram ini juga akan menunjukkan dampak atau akibat dari suatu permasalahan, dengan berbagai penyebabnya. Untuk itu peneliti menerapkan *Diagram Fishbone* dalam sebuah permasalahan yang ada yaitu terjadinya perbedaan antara point pelanggaran yang dilakukan siswa dengan pencatatan point pelanggaran yang dilakukan oleh guru sehingga terjadi kesalahan pencatatan, kebutuhannya adalah membuat form transaksi berisi tentang kegiatan transaksi yang ada dan bisa melakukan perhitungan poin secara otomatis. Hasil dari penelitian ini adalah Aplikasi Monpelsis Monitoring Pelanggaran Siswa untuk mempermudah dalam pemantauan pelanggaran siswa . [1]

**Informasi Bimbingan Konseling Siswa pada SMP Negeri 1 Panarukan** dalam penelitian ini dijelaskan bahwa permasalahan yang terjadi adalah pencatatan point pelanggaran siswa yang masih manual. Manualnya pencatatan menggunakan buku besar dapat mengakibatkan pencarian siswa yang pointnya sudah melebihi batas akan sulit ditemukan karena terlalu banyak data siswa. Pada penelitian ini menggunakan metode *waterfall*, dan dalam proses perancangan sistem pada penelitian ini menghasilkan *class diagram*, *physical data model*, perancangan komponen dan perancangan antarmuka. Dan juga terdapat 3 *class diagram*, dan 12 *sequence diagram* yang didokumentasikan. Dan dalam proses pengujian menggunakan 3 metode yaitu *basis path testing*, *validation testing*, dan *compatibility testing*. Sistem ini dibangun menggunakan Bootstrap sebagai *framework HTML* dan *CSS*, *Codeigniter* sebagai *framework PHP* dan *MySQL* sebagai database sistem. Hasil dari peneliti ini adalah sistem informasi bimbingan konseling untuk mempermudah dalam pencatatan point pelanggaran siswa. [2]

Pada jurnal terakhir yang telah dilakukan oleh Renaldi, Tri Sagirani, dan Agus Dwi Churniawan dengan judul **Rancang Bangun Aplikasi E-Poin Untuk Pencatatan Data Pelanggaran Dan Prestasi Akademik Siswa Pada SMK Negeri 10 Surabaya** dalam penelitian ini dijelaskan bahwa permasalahan yang dihadapi adalah dalam proses pencatatan yang dilakukan selama ini belum terdokumentasi dengan baik, masih menggunakan cara manual dengan buku pelanggaran dan prestasi akademis sehingga tidak dapat dilakukan analisa tentang pengaruh pelanggaran terhadap prestasi akademis siswa. Disini peneliti menggunakan metode *System Development Life Cycle* yang merupakan salah satu metodologi pengembang sistem, dalam desain sistem peneliti menggunakan *Context Diagram* dan *Data Flow Diagram*, aplikasi ini berbasis website. Hasil dari peneliti ini adalah Aplikasi E-Poin untuk pencatatan Data Pelanggaran dan Prestasi Akademik Siswa. [3]

Penelitian – penelitian tersebut diatas, tidak sama dengan penelitian yang akan dipakai dalam penerapan Aplikasi Pencatatan point pelanggaran siswa menggunakan *Android* untuk aplikasi di SMP N 2 Kembang ini, tetapi secara umum aspek-aspek yang diperoleh peneliti-peneliti terdahulu memberi dukungan informasi yang diperlukan.

## 2.2 Tinjauan Pustaka

### 2.2.1 Profil Sekolah Menengah Pertama (SMP) 2 Kembang

Sekolah Menengah Pertama (SMP) 2 Kembang merupakan sebuah instansi pendidikan negeri yang terletak di Desa Cepogo Kecamatan Kembang Kabupaten Jepara Jawa Tengah, letaknya di Jalan Raya Gambiran Cepogo berada pada lintang -6.5088254737911875 dan pada bujur 110.83594545722008 dengan ketinggian 174. Bangunan SMP N 2 Kembang ini berdiri diatas tanah Pemerintah Daerah yang memiliki Luas Tanah 17630 m<sup>2</sup>. Kepala SMP Negeri 2 Kembang yaitu Bapak Marsahid, S.Pd dan juga 17 Guru, terdapat 540 siswa yakni 329 putra dan 212 putri. SMP Negeri 2 Kembang Memiliki beberapa ruangan yakni 15 Ruang Kelas, 3 Laboratorium, 1 Perpustakaan, dan 2 ruang sanitasi siswa, dan juga memiliki beberapa ekstrakurikuler seperti pramuka, PMR (Palang Merah Indonesia), Kesenian Tari, dan kegiatan Olah Raga (Voli, Futsal, Hoki). (Sumber : SMP Negeri 2 Kembang)

### 2.2.2 Siswa

Siswa (Peserta Didik) adalah anggota masyarakat yang berusaha mengembangkan potensi diri melalui proses pembelajaran pada jalur pendidikan baik pendidikan informal, pendidikan formal maupun pendidikan nonformal, pada jenjang pendidikan dan jenis pendidikan tertentu. [4]

Siswa merupakan salah satu komponen manusiawi yang memiliki posisi dalam proses belajar mengajar, dalam proses mengajar siswa sebagai pihak yang ingin meraih cita-cita memiliki tujuan dan ingin mencapainya dengan optimal. Siswa akan menjadi faktor penemu, sehingga dapat mempengaruhi segala sesuatu yang diperlukan untuk mencapai tujuan belajarnya.

### 2.2.3 Point Pelanggaran

Point pelanggaran merupakan suatu alternatif yang dapat diberlakukan di sekolah sebagai upaya untuk menegakkan disiplin sekolah. terdapat enam pedoman yang mengilhami pembeelakuan Sistem Point

Pelanggaran seperti berikut [1] :

1. Hukuman itu harus jelas dan terang
2. Hukuman harus konsisten
3. Hukuman diberikan dalam waktu secepatnya

4. Bentuk-bentuk hukuman yang diberikan sebaiknya melibatkan Siswa.
5. Pemberi hukuman harus objektif
6. Hukuman Sebaiknya tidak bersifat fisik.

Poin pelanggaran ini mempunyai peranan sangat penting dalam menjaga tata tertib dan suasana nyaman di sekolah. Poin pelanggaran ini mempunyai batas nilai akhir bagi setiap siswa siswi. Dalam pembuatan ini sekolah bekerja sama dengan seluruh pihak-pihak antara lain Kepala Sekolah, Guru, dan Orang tua siswa.

#### **2.2.4 Aplikasi Mobile**

Aplikasi mobile berasal dari dua kata, yaitu aplikasi dan *mobile*. Secara istilah, aplikasi adalah program siap pakai yang dibuat untuk menjalankan suatu fungsi sedangkan *mobile* adalah sebuah perpindahan dari tempat satu ke tempat yang lain. Secara lengkap, aplikasi mobile adalah program siap pakai yang mes

##### **2.2.4.1 Pengertian Android**

Android merupakan sistem operasi yang digunakan untuk perangkat mobile berbasis linux yang mencakup sistem operasi, middleware, dan aplikasi. Android menyediakan platform terbuka bagi pengembang untuk membuat aplikasi mereka, Awalnya, Google Inc, membeli Android Inc.

Kemudian didunia ini ada dua jenis distributor sistem operasi Android. Pertama yang mendapat dukungan penuh dari Google atau Google Mall Services dan kedua adalah yang benar- benar bebas distribusinya tanpa dukungan langsung Google atau dikenal sebagai Open Handset Distribution (OHD), sebuah konsorsium dari beberapa perusahaan, yaitu Texas Instrument, Broadcom Corporation, Google, HTC, Intel, LG, Marvell Technology Group, Motorola, Nvidia, Qualcomm, Samsung Electronics, Sprint Nextel, dan T-Mobile dengan tujuan untuk mengembangkan standar terbuka untuk perangkat mobile Smartphone.[5]

Berdasarkan data tersebut peneliti dapat menyimpulkan bahwa saat ini android masih menjadi pilihan utama pada system operasi yang masih beredar di kalangan masyarakat Indonesia terutama dalam sistem informasi di dalam dunia pendidikan. Oleh sebab itu peneliti memilih System Operasi Android dalam perancangan aplikasi pencatatan poin pelanggaran siswa berbasis android.

##### **2.2.4.2 Versi Android**

Dalam perjalanannya Android telah banyak merilis versi dengan berbagai macam fitur dan perbaikan disetiap versi yang baru. Berikut versi-versi Android yang telah dirilis hingga saat ini [5]:

1. Android 1.0 Astro (Alpha)
2. Android 1.1 Bender (Beta)
3. Android 1.5 Cupcake
4. Android 1.6 Donut
5. Android 2.0 - 2.1 Eclair
6. Android 2.2 Froyo (Frozen Yoghurt)
7. Android 2.3 Gingerbread
8. Android 3.0 – 3.2 Honeycomb
9. Android 4.0 Ice Cream Sandwich
10. Android 4.1 – 4.3 Jelly Bean
11. Android 4.4 Kitkat
12. Android 5.0 – 5.1 Lollipop
13. Android 6.0 Marshmallow
14. Android 7.0 – 7.1 Nougat
15. Android 8.0 – 8.1 Oreo
16. Android 9.0 Pie

### 2.2.5 Java Development Kit (JDK)

*Java Development Kit (JDK)* adalah sekumpulan perangkat lunak yang dapat di gunakan untuk mengembangkan perangkat lunak yang berbasis java, sedangkan JRE adalah implementasi dari *Java Virtual Machine* yang digunakan untuk menjalankan program java. Biasanya setiap JDK berisi satu atau lebih JRE dan berbagai alat pengembangan lain seperti sumber *compiler java, bundling, debuggers, development libraries* dan lain-lain. [6]

JDK telah menjadi salah satu platform pengembangan yang paling banyak diunduh dan digunakan sejak dirilisnya platform pengembangan Java. Komponen JDK meliputi Java API Classes, Java Compiler, dan The Java Virtual Machine.

### 2.2.6 Android Software Development Kit (SDK)

Android SDK yaitu *tools API (Application Programming Interface)* yang diperlukan untuk mengawali pengembangan aplikasi pada platform android dengan menggunakan bahasa pemrograman java. Android merupakan sebuah subset perangkat lunak untuk ponsel yang meliputi sistem operasi *middleware*, dan aplikasi kunci yang di rilis oleh Goolge.

Android SDK terdiri dari debugger, libraries, handset emulator, dokumentasi, contoh kode program dan tutorial. Saat ini Android telah mendukung arsitektur x86 pada Linux (, Mac OS X 10.4.8 atau lebih, Windows XP atau Vista. Persyaratan mencakup JDK, Apache Ant dan Python 2.2 atau lebih. IDE yang didukung secara resmi adalah Eclipse 3.2 atau lebih dengan menggunakan plugin Android Development Tools , dengan ini pengembang dapat menggunakan IDE untuk mengubah dokumen Java dan XML serta menggunakan peralatan command line untuk menciptakan, membangun, melakukan debug aplikasi Android dan pengendalian perangkat Android (misalnya reboot, menginstal paket perangkat lunak). [6]

### 2.2.7 Firebase

Firebase adalah layanan *DbaaS (Database as a Service)* dengan menggunakan konsep secara *realtime*. *Firebase* merupakan penyedia layanan *cloud* dengan *backend* sebagai servis. *Firebase* terdiri dari fitur pelengkap yang bisa dipadupadankan sesuai dengan kebutuhan. *Firebase* memberikan perlengkapan dan infrastruktur untuk membangun suatu aplikasi yang lebih baik dengan tujuan untuk meningkatkan kesuksesan bisnis. [7]

Dengan *firebase* aplikasi pengembangan fokus mengembangkan aplikasi tanpa harus memberikan usaha yang besar. Dua fitur yang menarik pada *firebase* yaitu *Firebase Config* dan *Firebase Realtime Database*. Contoh dari aplikasi *realtime* saat ini seperti *Whatsapp*, *Facebook*, *Instagram* dll, dapat di aplikasikan menggunakan *firebase* untuk database maupun tools lainnya yang ada pada *firebase*. Apabila anda telah membuat aplikasi namun memerlukan suatu masalah tersebut dengan menggunakan tools yang ada di *firebase*.

### 2.2.8 Android Studio

Android Studio adalah sebuah (Integrated Development Environment) IDE yang bisa digunakan untuk pengembangan aplikasi berbasis Android, dan dikembangkan oleh google inc. Android Studio merupakan pengembangan dari Eclipse IDE, yang dibuat berdasarkan IDE Java populer, yaitu adalah IntelliJ IDEA (Java Development Environment). Android Studio yang Di rencanakan untuk menggantikan Eclipse kedepannya sebagai IDE resmi untuk pengembangan aplikasi berbasis Android. Sebagai Pengembangan dari Eclipse Android Studio yang mempunyai banyak fitur-fitur baru dibandingkan dengan Eclipse IDE. Berbedadengan Eclipse yang menggunakan Android Development Tools (ADT), Android Studio Menggunakan gradle sebagai *build environment* dalam proses pengembanganya. [6] Berikut merupakan contoh kode bahasa pemrograman kotlin:

```
fun main(args: Array<String>) {
    println("Lita Latihan")
}
```

pada contoh kode tersebut menunjukkan suatu fungsi yang di dalamnya terdapat perintah untuk mencetak string Lita Latihan.

### 2.2.9 Kotlin

*Kotlin* merupakan Bahasa pemrograman dengan pengetik anstatis dan dapat berjalan pada mesin *Virtual Java*, atau menggunakan *Compiler LLVM* yang dapat menyusun kedalam bentuk kode sumber *JavaScript*. Memiliki keunggulan untuk mudah dipelajari, sederhana, dan efisien. *Kotlin* diumumkan sebagai salah satu Bahasa pemrograman yang dapat digunakan untuk membuat aplikasi *Android* pada Google I/O 2017 berdampingan dengan *Java* dan *C++* yang telah terlebih dahulu sebagai Bahasa resmi untuk pengembang aplikasi *Android*. [8].

*Kotlin* bisa juga diartikan sebuah bahasa pemrograman yang menjadi bahasa penyempurnaan dari *Java*. *Kotlin* awalnya dikembangkan oleh JetBrains, perusahaan dibalik IntelliJ IDEA yang berjalan pada JVM ataubisadisebut *Java Virtual Machine*. Bahasa ini dirancang untuk memenuhi industry bahasa objek-oriented dengan standart yang lebih baik dibandingkan java.

### 2.2.10 Pemodelan UML (Unified Modeling Language)

UML (Unified Modeling Language) adalah kumpulan alat yang digunakan untuk melakukan abstraksi pada sistem atau perangkat lunak. UML menjadi bagian untuk mempermudah pengembangan suatu aplikasi yang bisa ditinjau lanjut.

UML sendiri juga memberikan standar penulisan sebuah sistem blue print, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema database, dan komponen-komponen yang diperlukan dalam sistem software. [13].

Ada beberapa building block pada UML yang terdiri dari 3 macam, yaitu:

1. Things

Adalah hal yang sangat mendasar dalam model UML juga merupakan bagian paling dasar dari sebuah model, serta menjelaskan elemen-elemen lainnya dari sebuah konsep.

2. Relationship

Terdiri dari 4 macam penggunaan UML, yaitu

- a. **Dependency** adalah hubungan antara 2 benda yaitu benda berubah menjadi benda satunya akan berubah juga. Digambarkan dengan garis putus-putus.
- b. **Association** adalah hubungan antara benda yang tersambung dengan obyek .kesatuan yang terhubung dalam obyek akan menjadi hubungan khusus.
- c. **Generalization** adalah gambaran khusus dalam suatu obyek anak/child untuk mengubah obyek perrent/induk.
- d. **Realization** adalah hubungan antara pengelompokan yang di jamin adanya ikatan diantaranya, hubungan ini dapat diwujudkan diantaranya interface dan kelasatau element.

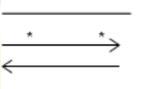
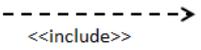
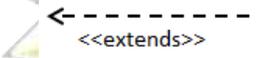
3. Diagram

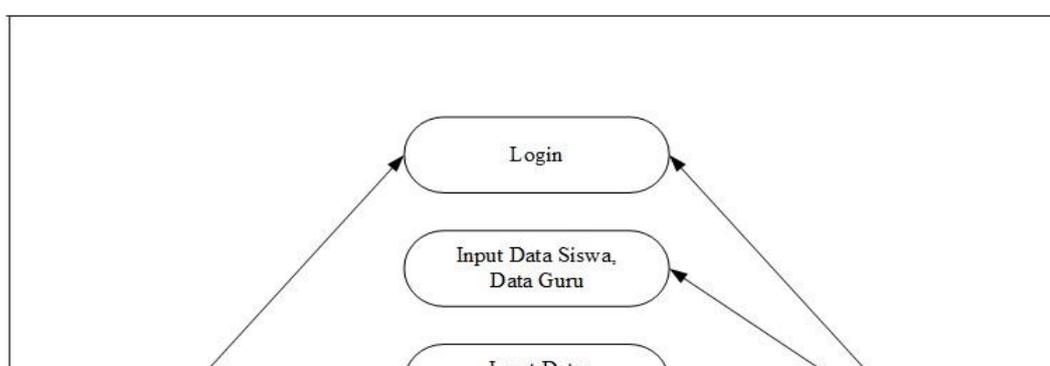
UML terdiri dari pengelompokan diagram darisuatu system menurut aspek maupun sudut pandang tertentu. Diagram merupakan gambaran alur masalah maupun solusi dari suatu model. UML mempunyai beberapa diagram yaitu:

### 2.2.11 Use Case Diagram

Use Case Diagram digunakan untuk menggambarkan sebuah sistem dari sudut pandang pengguna sistem (*user*), sehingga pembuatan lebih di beratkan pada fungsionalitas yang ada pada sistem dan bukan berdasarkan alur sistem. Sebuah *use case* diagram mempresentasikan sebuah interaksi antara aktor dengan sistem.[9]

Tabel 2. 1 Simbol *Usecase Diagram*

Simbol	Keterangan
	Menggambarkan proses / kegiatan yang dapat dilakukan oleh aktor.
	Menunjukkan entitas/subjek yang melakukan suatu proses.
	Menunjukkan hubungan antara case dengan actor ataupun case dengan case.
	<i>Include</i> , di dalam <i>use case</i> lain ( <i>required</i> ) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain.
	<i>Extend</i> , perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.



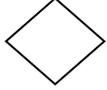
Gambar 2. 1 Contoh Use Case Diagram

### 2.2.12 Activity Diagram

*Activity Diagram* atau diagram aktivitas adalah salah satu jenis diagram pada UML yang memodelkan proses yang terjadi pada sistem. Tipe khusus dari diagram status yang memperlihatkan aliran dari suatu aktivitas ke aktivitas lainnya dalam suatu sistem. Dihubungkan dengan tanda panah yang menggambarkan urutan aktivitas yang terjadi dari awal sampai akhir. Jadi aliran ini dapat berurutan, bercabang atau bersamaan.[9]

Tabel 2. 2 Simbol Activity Diagram

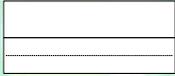
Simbol	Keterangan
	<i>Activities</i> , menggambarkan suatu proses kegiatan bisnis
	<i>Start Point</i> , diletakkan di pojok kiri atas dan sebagai awal aktifitas
	<i>End Point</i> , akhir aktivitas

	<i>Flow dan Control</i> , sebagai penghubung aliran aktivitas dari elemen satu ke elemen lain
	<i>Decision Points</i> , menggambarkan pilihan untuk mengambil keputusan, <i>true</i> atau <i>false</i>

### 2.2.13 Class Diagram

*Class Diagram* merupakan sebuah spesifikasi yang akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class diagram* menggambarkan struktur dan deskripsi *class*, *package* dan objek serta hubungan satu sama lain seperti pewarisan, asosiasi, *containtment*, dan lain-lain. [10]

Tabel 2. 3 Simbol *Class Diagram*

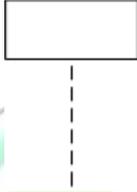
Simbol	Keterangan
	<b><i>Class</i></b> , merupakan blok-blok pembangun pada pemrograman berorientasi objek
	<b><i>Association</i></b> , merupakan suatu garis yang digunakan untuk menghubungkan atau merelasikan antar kelas
	<b><i>Dependency</i></b> , digunakan untuk menunjukkan operasi pada suatu kelas yang menggunakan kelas yang lain.
	<b><i>Aggregation</i></b> , simbol yang menghubungkan antar kelas dengan makna untuk semua bagian.

### 2.2.14 Squence Diagram

*Squence Diagram* adalah salah satu jenis diagram pada UML yang sudah menjelaskan tentang interaksi objek didasari oleh urutan waktu. Diagram ini

digunakan untuk menggambarkan scenario atau langkah-langkah yang dilakukan untuk menghasilkan *output*. Proses ini diurutkan dari kiri kekanan berdasarkan waktu terjadinya. *Sequence diagram* dapat berupa sebuah *sequencediagram* yang bersifat umum yang menunjukkan semua skenario untuk semua *use case*, atau terpisah untuk skenario tunggal didalam *use case* [10]

Tabel 2. 4 Simbol *Sequence Diagram*

SIMBOL	KETERANGAN
	<p><b>Object</b>, berfungsi menggambarkan pos-pos objek yang mengirim dan menerima pesan</p>
	<p><b>Message</b>, berfungsi untuk menggunakan aliran pesan yang dikirim oleh pos-pos objek</p>

### 2.2.15 Metode Prototype

Prototyping adalah sebuah metode yang mengembangkan perangkat lunak yang digunakan dalam pembuatan system aplikasi. Dengan menggunakan prototyping pengembang dan pelanggan bisa saling berinteraksi atau saling komunikasi selama proses pembuatan system. Sistem *prototype* memperbolehkan pengguna untuk mengetahui bagaimana sistem itu berjalan dengan baik. Penggunaan metode Prototyping di dalam penelitian ini bertujuan supaya peneliti mendapatkan gambaran aplikasi yang akan dibangun melalui tahap pembangunan aplikasi *prototype* terlebih dahulu yang akan dievaluasi oleh *user*. Aplikasi *prototype* yang telah dievaluasi oleh selanjutnya akan dijadikan acuan untuk membuat aplikasi yang dijadikan produk akhir sebagai *output* dari penelitian ini[11]:

### 2.2.16 Black Box Testing

*Blackbox* Testing merupakan pengujian yang berfokus pada persyaratan fungsional pada perangkat lunak. Dalam pengujian *Blackbox testing* ini pun dapat menemukan kesalahan seperti fungsi yang salah, kesalahan *interface*, kesalahan dalam struktur data atau *database* eksternal, kesalahan kinerja dan instalasi dan kesalahan terminasi.[12]

Kegiatan yang dilakukan pada pengujian *BlackBox Testing* adalah :

1. Mengujifungsi-fungsi pada aplikasi dengan membuat *test case*.
2. Mengujikesesuaian alur kerja suatu fungsi system dengan kebutuhan yang dibutuhkan oleh respondendengan membuat *test case*.
3. Mencari *error* dari tampilan (*interface*) aplikasi.



### 2.3 Kerangka Pemikiran

