

## **BAB II**

### **KAJIAN PUSTAKA**

#### **A. Kajian Teori**

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

Rancang bangun merupakan proses yang mengubah hasil analisis ke dalam bentuk perangkat lunak untuk menciptakan atau meningkatkan sistem yang ada. Secara esensial, rancang bangun melibatkan penggambaran, perencanaan, dan penyusunan komponen-komponen terpisah menjadi satu kesatuan yang terintegrasi (Wulandari, 2021).

Sedangkan menurut Nizar (2021), “rancang bangun sistem merupakan kegiatan yang mengubah hasil analisis menjadi perangkat lunak dengan menentukan proses dan data yang diperlukan untuk membuat sistem baru atau memperbaiki sistem yang sudah ada”.

Dari kedua kutipan tersebut mendapat kesimpulan bahwa proses perencanaan dan penggambaran yang mengubah hasil analisis menjadi perangkat lunak dikenal sebagai rancang bangun. Dalam proses ini, komponen yang berbeda diatur menjadi satu set yang berfungsi untuk membangun ataupun membetulkan sistem yang ada.

##### **2. Aplikasi**

Aplikasi adalah program yang terpasang di dalam komputer atau handphone untuk menjalankan program yang telah dikembangkan sebelumnya (Dewi, 2021).

Aplikasi adalah perangkat lunak yang melakukan pengolahan data melalui komputasi. Istilah “aplikasi” berasal dari bahasa Inggris “*application*”, yang mengacu pada penerapan atau penggunaan sesuatu. Secara umum, aplikasi menggambarkan program yang telah dipersiapkan untuk menjalankan fungsi tertentu bagi pengguna, dan dapat diakses oleh pengguna target serta pihak lainnya. (Parjito, 2023).

Dari Kedua penjelasan tersebut menunjukkan bahwasanya aplikasi adalah *software* yang dapat digunakan untuk memproses data menggunakan komputasi. Istilah ini berasal dari kata "*application*" dalam bahasa Inggris, yang artinya "penerapan" bisa disebut juga "penggunaan". Secara praktis, "aplikasi" merujuk pada program yang dapat digunakan oleh pengguna dan sasaran yang dituju dan dimaksudkan untuk melakukan tugas tertentu.

### **3. Pengaduan**

Menurut (Payong, 2020) “Pentingnya layanan pengaduan masyarakat bagi pemerintah adalah untuk mengukur partisipasi masyarakat dalam proses pemerintahan, memenuhi kebutuhan mereka, serta memperkuat hubungan antara pemerintah dan masyarakat. Pengaduan masyarakat merupakan elemen krusial dalam struktur pemerintahan daerah karena berfokus pada perbaikan terhadap kegiatan yang telah dilaksanakan.”

Pengaduan merupakan menyampaikan aduan masyarakat kepada pemerintah mengenai layanan yang tidak selaras dengan SOP yang

ditetapkan. Penanganan aduan mencakup serangkaian kegiatan seperti penerimaan, pencatatan, tindak lanjut, pengarsipan, dan pelaporan (Aldisa, 2022).

Dari penjelasan di atas, pengaduan adalah proses di mana orang mengadukan tentang pelayanan yang tidak memenuhi oleh persyaratan. Penerimaan, pencatatan, tindak lanjut, pengarsipan, dan pelaporan adalah beberapa tindakan yang dilakukan dalam penanganan pengaduan untuk memastikan bahwa pengaduan ditangani dengan benar.

#### **4. Website**

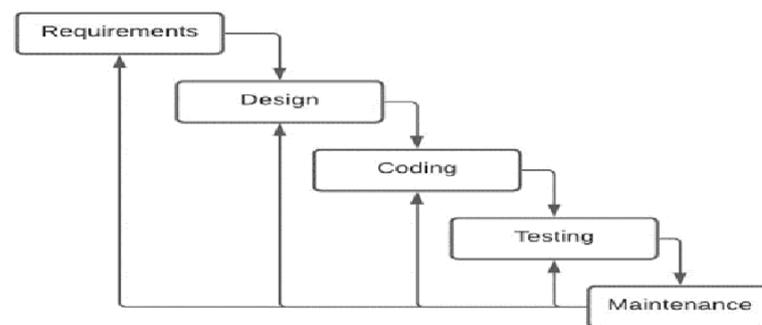
Menurut (Arafat, 2022) “*Website*, yang juga dikenal sebagai situs *web*, merupakan sebuah koleksi halaman digital yang berisi informasi dalam format-format yang beragam seperti teks, gambar, video, audio, dan animasi. Secara lebih rinci, sebuah situs web terdiri dari halaman-halaman yang dapat ditampilkan oleh *browser* seperti *Mozilla Firefox*, *Google Chrome*, ataupun peramban web lainnya.”.

Didasarkan pada pernyataan di atas, *Website* adalah kumpulan halaman digital yang mencakup informasi pada bermacam format, seperti audio, video, gambar, teks, serta animasi, yang dapat diakses melalui *browser* yang terhubung ke internet.

#### **5. Metode Waterfall**

Menurut Pressman dalam Buku “ODOL (*one desa one produk unggulan online*) penerapan metode *Naïve Baiyes* pada pengembangan

aplikasi *e-commerce* menggunakan *Codeigniter*” karya Sanubari (2020:41). Dalam pengembangan perangkat lunak, metode *waterfall* atau air terjun merupakan siklus hidup klasik yang dikenal. Metode ini melukiskan proses yang terstruktur serta berurutan, yang dimulai dari spesifikasi keperluan pemakai, diikuti oleh tahapan penyerahan, konstruksi, pemodelan, perencanaan sistem yang telah selesai kepada pelanggan atau pemakai. Tahap ini berakhir dengan penyediaan dukungan untuk *software* yang sudah dikembangkan secara lengkap.



Gambar 2.1 Tahapan Metode Waterfall

(Sanubari, 2020:41)

Menurut Sanubari (2020:41-42), metode *waterfall* dikembangkan dalam beberapa tahapan, yaitu:

a. *Requirements* (Kebutuhan)

Tahap ini melibatkan komunikasi yang bertujuan untuk memahami kebutuhan perangkat lunak yang diinginkan oleh pengguna dan batasan yang terkait. Informasi ini biasanya diperoleh melalui wawancara, diskusi, atau survei langsung. Data yang terkumpul dianalisis untuk memahami kebutuhan pengguna secara mendalam.

b. *Design* (Desain)

Setelah memenuhi persyaratan dari tahap sebelumnya, desain sistem membantu dalam menentukan perangkat keras yang dibutuhkan, menguraikan persyaratan sistem, dan merancang arsitektur sistem secara keseluruhan.

c. *Coding* (Pengkodean)

Pada tahap ini, sistem dikembangkan dalam bentuk unit-unit kecil yang disebut modul. Setiap modul dikodekan dan diuji secara terpisah untuk memastikan fungsionalitasnya dalam pengujian unit.

d. *Testing* (Pengujian)

Setelah semua modul diimplementasikan dan diuji, mereka diintegrasikan menjadi satu sistem. Pengujian integrasi dilakukan untuk memastikan bahwa seluruh sistem berjalan dengan baik dan memenuhi semua persyaratan yang telah ditentukan.

e. *Maintenance* (Pemeliharaan)

Tahap terakhir dari model *waterfall* di mana perangkat lunak sudah siap untuk dioperasikan oleh pengguna akhir. Pemeliharaan meliputi perbaikan kesalahan yang mungkin terjadi setelah penerapan sistem, serta pembaruan atau peningkatan sistem sesuai dengan kebutuhan baru yang muncul.

Kelebihan dari menggunakan metodologi *waterfall* adalah kemampuannya untuk departementalisasi dan pengendalian yang baik. Tahap peningkatan dilaksanakan secara bertahap yang lebih lanjut, sehingga mengurangi kesalahan. Konsep adalah titik awal pengembangan,

yang diikuti oleh instalasi, pengujian, implementasi, desain, penyelesaian masalah, serta akhirnya operasi serta pemeliharaan.

Namun, kekurangan dari metodologi *waterfall* ialah tidak fleksibel terhadap revisi yang banyak jika terjadi *error* pada tahapannya. Sesudah aplikasi memasuki tahap pengujian, sulit untuk kembali serta melakukan perubahan terhadap hal-hal yang tidak terdokumentasi dengan baik dalam fase konseptual sebelumnya.

## 6. XAMPP

Menurut (Tanko, 2022) “XAMPP, yang dikembangkan oleh *Apache Friends*, adalah paket solusi tumpukan *server web* lintas platform yang gratis dan sumber terbuka yang terdiri utamanya dari *Apache HTTP Server*, *database MariaDB*, dan *interpreter* untuk *skrip PHP* dan *Perl*. *Cross-Platform (X)*, *Apache (A)*, *MariaDB (M)*, *PHP (P)*, dan *Perl (P)* adalah nama distribusi *Apache* yang ringan dan mudah digunakan yang memungkinkan pengembang membuat *server web* lokal untuk pengujian.” Dengan demikian, XAMPP (*Cross-Platform, Apache, MariaDB, PHP, dan Perl*) ialah paket *server web* lintas platform yang gratis serta sumber terbuka yang ditingkatkan oleh *Apache Friends*. Paket ini terdiri dari *Apache HTTP Server*, *database MariaDB*, serta *interpreter* guna *skrip PHP dan Perl*.

Dalam bukunya Lusita Violita Aprilian (2020: 51-52), menyebutkan pendukung keputusan *XAMPP* meliputi:

a. Apache

*Apache* merupakan *software* sumber terbuka yang menjadi alternatif bagi *server web Netscape*. Berperan sebagai *server HTTP* atau *World Wide Web*, *Apache* bisa beroperasi pada beragam sistem operasi serta mempunyai kegunaan untuk mengelola situs *web*. Pengembangan *Apache* dilakukan oleh komunitas terbuka di bawah pengawasan *Apache Software Foundation*.

b. MariaDB

*MariaDB* ialah sistem manajemen basis data yang berfungsi sebagai server untuk menyediakan akses *multi-user* ke berbagai *database*. *MariaDB* biasanya dipakai oleh *software* sumber terbuka yang membutuhkan fitur lengkap dalam sistem manajemen basis data, seperti *WordPress*, *phpBB*, serta aplikasi lain yang menggunakan *software LAMP*. *MariaDB* juga dipakai secara luas di *World Wide Web*, termasuk oleh produk dari *Google* serta *Facebook*.

c. PHP

*PHP* ialah bahasa pemrograman skrip yang sering dipakai guna mengembangkan situs web dinamis, walaupun dapat juga dipakai guna berbagai keperluan lainnya. Contoh aplikasi terkenal yang menggunakan *PHP* adalah *phpBB* dan *MediaWiki*, yang merupakan perangkat lunak di balik Wikipedia.

Lusia Violita Aprilian (2020:54), kelebihan dari *XAMPP* meliputi:

a. Gratis dan Populer.

*Engine* penyimpanan database ini banyak dipakai oleh programmer dan pengembang *web* dikarenakan gratis. Versi yang lebih canggih tersedia dengan biaya untuk pengguna yang mahir.

b. Kapasitas dan Keandalan

*XAMPP* dapat menangani sekitar 60.000 tabel dengan jumlah record menggapai 5.000.000.000 atau lebih, menunjukkan kapasitas yang handal, terutama pada versi terbaru.

c. Keamanan yang Memadai

Meskipun tidak sekuat *PostgreSQL* atau *Oracle*, *XAMPP* menyediakan tingkat keamanan yang cukup untuk sebagian besar pengguna.

d. *Multiplatform*

*Engine* ini mendukung bermacam sistem operasi, sehingga tepat digunakan aplikasi di kelas menengah serta kecil.

e. Kecepatan

Salah satu kelebihan utama *XAMPP* adalah kecepatannya dalam mengelola dan mengakses data.

Kekurangan dari *XAMPP* (Lusia, 2020:54), meliputi:

a. Keterbatasan pada Data Besar

Tidak cocok guna menangani volume data besar baik dalam penyimpanan maupun pemrosesan data.

b. Kinerja Terbatas pada Skala Besar

*XAMPP* memiliki keterbatasan kinerja pada server ketika kapasitas penyimpanan melebihi batas maksimal, karena tidak menerapkan konsep *Technology Cluster*.

## 7. *Visual Studio Code*

*Visual Studio Code* ialah software yang cukup ringan tetapi mampu sebagai editor kode sumber yang berguna di *desktop*. *Visual Studio Code* digunakan untuk mengembangkan berbagai jenis program, *Visual Studio Code* memerlukan aplikasi yang handal (Hartati, 2020). Perangkat ini mendukung berbagai bahasa pemrograman seperti *JavaScript*, *HTML*, *CSS*, *PHP*, *Python*, *C++*, serta lainnya. *Visual Studio Code* bisa difungsikan di bermacam sistem operasi layaknya *Windows*, *macOS*, serta *Linux*. Fitur *Live Share* yang disediakan oleh *Visual Studio Code* memungkinkan beberapa pengembang bekerja secara bersamaan pada satu proyek dari lokasi yang berbeda (Nanda, 2023).

## 8. *CodeIgniter*

*CodeIgniter* adalah *framework* pengembangan *web PHP* yang populer yang mengikuti pola arsitektur *Model-View-Controller* (MVC). *Framework* ini menyediakan beragam alat dan fitur lengkap bagi pengembang untuk membangun aplikasi *website* yang kuat dan *scalable* dengan cepat dan efektif. *CodeIgniter* menyederhanakan proses pengembangan *web* dengan menawarkan sintaksis yang bersih dan

ekspresif, struktur modular, serta berbagai komponen bawaan. Di dalam *CodeIgniter*, Anda dapat memanfaatkan lapisan abstraksi basis data untuk berinteraksi dengan *database* dengan mudah. *Framework* ini menyediakan antarmuka yang mudah digunakan untuk bekerja dengan *database*, memungkinkan Anda melakukan operasi-operasi basis data tanpa perlu menulis *kueri SQL* yang kompleks. Hal ini membantu menyederhanakan tugas penanganan data dan meningkatkan efisiensi pengembangan (Gangurde, 2023).

Selain itu, *CodeIgniter* juga dilengkapi dengan mesin templating yang kuat yang memudahkan pembuatan tampilan yang dapat digunakan ulang dan dinamis. Dengan mesin templating ini, yang disebut *Blade*, Anda dapat memisahkan logika aplikasi dan lapisan presentasi secara bersih. *Template Blade* mudah dipelajari dan menyediakan cara yang langsung untuk membuat tampilan yang dinamis dan menarik secara visual. *CodeIgniter* juga menawarkan berbagai fungsionalitas bawaan seperti *otentikasi*, *otorisasi*, *caching*, dan *routing*. Fitur-fitur ini membantu pengembang fokus pada pembangunan fitur inti dari aplikasi mereka tanpa perlu khawatir tentang detail implementasi dari fungsionalitas umum ini (Gangurde, 2023).

Menurut Muhammad Yusril Helmi Setyawan dalam bukunya yang berjudul *CodeIgniter : “Implementasi Metode Entropy Pada Pemrograman PHP (Belajar Dengan Praktek)”* (2020:3) *CodeIgniter* adalah *toolkit* bagi mereka yang ingin membangun aplikasi *web* menggunakan bahasa

pemrograman *PHP*. Berikut adalah beberapa keunggulan yang ditawarkan oleh *CodeIgniter*:

- a. *CodeIgniter* yakni *framework* gratis serta *open-source*.
- b. *CodeIgniter* mempunyai ukuran file yang relatif kecil diperbandingkan dengan *framework PHP* lainnya. Sesudah diunduh dan diekstrak, *CodeIgniter* memiliki total ukuran sekitar 11 MB, dengan folder user guide (dokumentasi *CodeIgniter*) sekitar 9 MB dan folder aplikasi serta sistem sekitar 2 MB.
- c. Aplikasi yang dikembangkan dengan *CodeIgniter* dapat berjalan dengan cepat.
- d. *CodeIgniter* memakai pola desain *Model-View-Controller* (MVC), sehingga memungkinkan untuk memisahkan kode ke dalam bagian-bagian yang terstruktur. Dalam membuat kode menjadi lebih mudah dipahami, dibaca, serta dikembangkan, serta memudahkan dalam pemeliharaan di masa mendatang.
- e. *CodeIgniter* bisa diperluas sesuai dengan keperluan pengembangan.
- f. *CodeIgniter* memiliki dokumentasi yang baik. Informasi mengenai kelas serta fungsi yang tersedia dalam *CodeIgniter* bisa ditemukan melalui dokumentasi yang disertakan pada paket distribusinya.

## 9. *Android Studio*

Menurut (Juanda, 2022) “*Android Studio* merupakan *integrated development environment* (IDE) yang khusus dirancang untuk pengembangan aplikasi *Android*. IDE ini berbasis pada perangkat lunak

*IntelliJ IDEA* dari *JetBrains*, dan telah menggantikan *Eclipse Android Development Tools* (ADT) sebagai IDE utama untuk pengembangan aplikasi Android. Sebagai *platform open source*, Android memungkinkan pengembangan oleh siapapun, dan Android Studio adalah alat yang digunakan untuk menciptakan aplikasi tersebut.”

Menurut Jatmiko Indriyanto (2022:4) dalam bukunya yang berjudul “Aplikasi *Database* Di *Android Studio* Menggunakan *Database* Di Internet”, *Android* merupakan sistem operasi yang dibuat oleh *Google*, dirancang khusus untuk perangkat *smartphone* dan *tablet*. Banyak produsen telah mengadopsi *Android* sebagai sistem operasi utama untuk perangkat mereka. *Android* juga memiliki platform distribusi aplikasi yang dikenal sebagai toko aplikasi, dengan lebih dari 2,3 miliar pengguna aktif pada tahun 2018. Di Indonesia, pada bulan Maret 2019, sekitar 93,5% dari pangsa pasar menggunakan *Android* sebagai sistem operasi pada perangkat *mobile* mereka.

Alasan *Developer* Memilih *Android*:

a. Fitur Canggih, Tampilan Menarik:

*Android* menawarkan fitur-fitur *modern* serta *design* yang menarik. Sistem *Android* dapat juga dipakai sebagai media multimedia layaknya pemutar video serta musik. Selain itu, *Android* mendukung *hardware* seperti *akselerometer*, *giroskop*, dan sensor lainnya.

b. Sistem Operasi Smartphone Terpopuler:

Sejak peluncuran pada tahun 2008, *Google* sudah meluncurkan beberapa versi *Android*, dengan versi terbaru "Q" yang diperkenalkan pada 7 Agustus 2019. *Android Q* adalah versi ke-10 dari sistem operasi ini dan telah tersedia untuk berbagai smartphone, seperti *Google Pixel*. *Android 10* atau *Android Q* difokuskan untuk pengembangan tema gelap secara menyeluruh.

c. Distribusi Aplikasi:

Aplikasi *Android* dapat didistribusikan melalui web, salinan APK, dan toko aplikasi. *Google Play Store* adalah cara mudah bagi para developer guna mendistribusikan aplikasi mereka ke miliaran pengguna di seluruh dunia.

## 10. *MySQL*

*MySQL* ialah sistem manajemen basis data relasional *open-source* yang populer. Sistem ini umumnya dipakai untuk aplikasi *web* dan mendukung berbagai bahasa pemrograman. *MySQL* dapat menangani jumlah data yang besar dan dikenal karena skalabilitasnya. *MySQL* menawarkan fitur ketersediaan tinggi dan keamanan untuk memastikan integritas data. *MySQL* terus berkembang dengan pembaruan dan peningkatan fitur serta kinerjanya secara berkala (Deravi, 2023).

Menurut Namruddin dalam bukunya yang berjudul *Belajar Database Dengan Mudah Menggunakan MySql* (2023:3-4), *MySQL* mempunyai keunggulan dan kekurangan, diantaranya:

Adapun beberapa keunggulan *DBMS MySQL*:

a. *Software* yang Portabel:

Keunggulan pertama dari *MySQL* adalah sifatnya yang portabel. Ini berarti *MySQL* dapat dijalankan untuk mengelola basis data di berbagai platform. Sistem operasi seperti *Windows*, *Linux*, *Mac*, dan lainnya dapat menggunakan *DBMS MySQL* ini, sehingga meningkatkan efisiensi dan fungsionalitasnya.

b. *Open Source*:

Keunggulan utama *MySQL* adalah bahwa perangkat lunak ini gratis. Ya, *MySQL* versi dasar atau sederhana tersedia secara gratis karena merupakan perangkat lunak *open source*. Meski demikian, *MySQL* memiliki lisensi GPL, sehingga kualitasnya tidak perlu diragukan lagi.

c. Mendukung *Multi-User*:

Seperti program *DBMS* lainnya, meskipun perangkat lunak *open source*, *MySQL* memiliki kemampuan yang baik untuk mendukung kepentingan *multi-user*, dimana banyak pengguna dapat mengaksesnya secara bersamaan tanpa mengalami masalah seperti crash.

d. Tipe Data yang Bervariasi:

*MySQL* menawarkan berbagai tipe data. Berikut tipe data yang didukung oleh *MySQL* adalah *integer*, *float*, *double*, *char*, *text*, *date*, *timestamp*, dan masih banyak lagi. Dengan dukungan berbagai tipe data ini, *MySQL* sangat berguna untuk *DBMS*.

e. Keamanan yang Baik:

Kelebihan *MySQL* adalah fitur keamanannya yang baik. Meskipun merupakan perangkat lunak *open source* dan gratis, fitur keamanan yang ditawarkan oleh *MySQL* sudah sangat mumpuni.

Kekurangan dari *MySQL* ( Namruddin 2023:6-7):

a. Sulit Diterapkan pada Instansi atau Perusahaan dengan Basis Data Besar:

Karena *MySQL* dirancang agar ramah terhadap spesifikasi komputer yang lebih rendah, fitur-fitur yang ditawarkan tidak sekomplit *Oracle*. Ini berhubungan dengan kemampuan *MySQL* dalam mengelola basis data yang sangat besar, yang diragukan efektivitasnya. Oleh karena itu, *MySQL* kurang cocok untuk digunakan di instansi atau perusahaan besar yang membutuhkan manajemen basis data dalam jumlah besar.

b. Dukungan Teknis yang Terbatas:

Dukungan teknis dari *MySQL* dianggap kurang memadai. Hal ini mungkin terkait dengan status *open source* yang dimilikinya, sehingga pengguna bisa mengalami kesulitan dalam menghubungi dukungan teknis *MySQL* ketika menghadapi masalah atau kendala dalam penggunaannya.

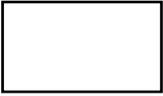
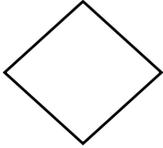
c. Kurang Populer untuk Aplikasi Game dan Mobile:

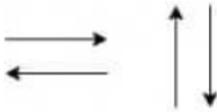
*MySQL* kurang cocok digunakan untuk aplikasi game dan aplikasi mobile, jadi mungkin bukan pilihan yang baik jika ingin mengembangkan dua jenis aplikasi ini..

## 11. Flowchart

*Flowchart* adalah contoh representasi visual yang mampu menyajikan, mengorganisir secara grafis, dan menunjukkan metode untuk memecahkan masalah dengan memberikan pemahaman visual terhadap informasi yang rumit. *Flowchart* dapat meningkatkan pemahaman yang mendalam, daya ingat, pengenalan kembali, dan kemampuan untuk membangun koneksi yang bermakna dengan materi. Sifat visual dan diagramatis dari *flowchart* menyederhanakan informasi yang kompleks, sehingga mempermudah dalam proses pengambilan keputusan (Zimmermann, 2024).

Tabel 2.1 Simbol simbol *flowchart*.

Simbol	Nama	Fungsi
	<i>Terminal</i>	Menandai mulai atau selesainya suatu proses atau program.
	<i>Input Output</i>	Menunjukkan proses <i>input</i> atau <i>output</i> terlepas dari jenis perangkat yang digunakan.
	<i>Process</i>	Menandai langkah atau proses operasional komputer dalam <i>flowchart</i> .
	<i>Decision</i>	Menunjukkan bahwa suatu kondisi tertentu mengarah pada dua kemungkinan atau lebih dalam alur <i>flowchart</i> .
	<i>Offline Connector</i>	Menunjukkan bahwa ada koneksi atau hubungan antara dua bagian <i>flowchart</i> yang mungkin tidak langsung.

Simbol	Nama	Fungsi
	<i>Connector</i>	Menghubungkan dua bagian <i>flowchart</i> yang terpisah tanpa mengganggu alur visual.
	<i>Document</i>	Menandai titik di mana dokumen atau laporan dihasilkan sebagai <i>output</i> dari proses.
	<i>Disk</i>	Simbol yang menyatakan <i>input</i> yang berasal dari <i>disk</i> atau disimpan ke <i>disk</i>
	<i>Flow</i>	Menunjukkan arah aliran dari satu simbol <i>flowchart</i> ke simbol berikutnya, menggambarkan urutan atau alur dari proses.

## 12. UML

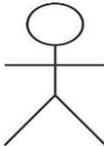
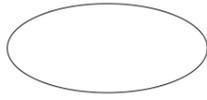
UML dipilih sebagai metode untuk mendukung pemodelan kolaborasi elektronik pada sistem keras karena memiliki tingkat kesederhanaan yang tinggi dan formalisme yang baik, sehingga menjadi alat pemodelan yang terstandarisasi dengan baik. Dengan formalisme UML, untuk mentransmisikan model konseptual kepada berbagai aktor yang merupakan user aplikasi dengan mudah (Purwaningsih, 2024). Pada tahap pengembangan sistem, dipilihlah beberapa perancangan yang akan digunakan untuk membuat sistem antara lain, *Use case Diagram*, *Activity Diagram*, *Sequence Diagram*, *Class Diagram*.

### 13. Use Case Diagram

*Use case* merupakan diagram dari *UML* yang sering digunakan oleh para *developer* untuk menggambarkan bagaimana pengguna akan berinteraksi dengan sistem dalam bentuk visual agar mudah dipahami oleh pengguna.

*Use case* adalah penjelasan tentang bagaimana sebuah sistem berinteraksi dengan pengguna dari sudut pandang pengguna itu sendiri. *Use case* berfungsi dengan cara menjelaskan secara rinci bagaimana pengguna berinteraksi dengan sistem, menggambarkan interaksi tipikal antara pengguna sistem dan sistem itu sendiri melalui sebuah narasi tentang penggunaan sistem tersebut (Saputra, 2021).

Tabel 2.2 Simbol *Use Case Diagram*

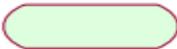
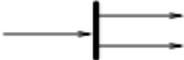
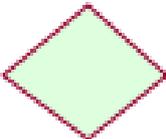
Simbol	Nama	Fungsi
	Aktor	Entitas (seseorang atau sesuatu) yang berinteraksi dengan sistem yang sedang dikembangkan.
	<i>Use Case</i>	Menggambarkan bagaimana pengguna akan menggunakan atau memanfaatkan sistem.
	Relasi Asosiasi	Relasi yang menghubungkan aktor dengan <i>use case</i> , biasanya digambarkan sebagai garis lurus

Simbol	Nama	Fungsi
	<i>Relation</i>	dengan kepala panah di salah satu ujungnya. Penghubung antara aktor dengan <i>use case</i> , serta antara <i>use case</i> dengan <i>use case</i> lainnya.
	<i>Include Relationship</i>	Relasi cakupan yang memungkinkan suatu <i>use case</i> untuk menggunakan fungsionalitas yang telah didefinisikan oleh <i>use case</i> lainnya.
	<i>Extend Relationship</i>	Relasi yang memungkinkan suatu <i>use case</i> untuk memperluas atau menambahkan fungsionalitas dari <i>use case</i> yang lainnya.

#### 14. Activity Diagram

Diagram aktivitas atau *activity diagram* menggambarkan alur kerja atau aktivitas dari sebuah sistem atau proses bisnis, termasuk menu-menu yang tersedia dalam perangkat lunak. Fokus utama diagram aktivitas adalah pada representasi visual dari aktivitas-aktivitas sistem atau aktivitas yang dapat dilakukan oleh sistem tersebut, bukan pada aktivitas yang dilakukan oleh actor (Maria, 2021). *Activity diagram* biasanya akan digambarkan secara *vertical* untuk memvisualkan aliran control yang terjadi dalam suatu sistem dengan cara yang mudah untuk dipahami dengan cara mengurutkan proses berjalannya sistem informasi.

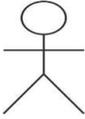
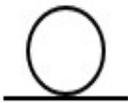
Tabel 2.3 Simbol *Activity Diagram*

Simbol	Nama	Fungsi
	<i>Start Point</i>	Menggambarkan titik awal dari suatu aktivitas atau alur kerja.
	<i>End Point</i>	Menggambarkan titik akhir dari suatu aktivitas atau alur kerja.
	<i>Activity</i>	Menggambarkan suatu proses atau kegiatan bisnis yang dilakukan dalam aktivitas atau alur kerja.
	<i>Fork</i>	Menggambarkan titik percabangan di mana satu aktivitas membagi alur kerja menjadi dua atau lebih jalur yang dikerjakan secara bersamaan.
	<i>Decision</i>	Menggambarkan keputusan atau pilihan yang harus dibuat dalam alur kerja, biasanya mengarahkan alur kerja ke jalur yang berbeda berdasarkan kondisi tertentu.
	<i>State Transition</i>	Menggambarkan transisi atau perpindahan kontrol antara state dalam suatu sistem atau proses.
	<i>Swimlane</i>	Menggambarkan pemisahan atau pengelompokan aktivitas berdasarkan aktor atau unit organisasi yang bertanggung jawab atas aktivitas tersebut.

## 15. Sequence Diagram

Diagram urutan atau *Sequence Diagram* menggambarkan perilaku objek pada sebuah skenario *use case* dengan menjelaskan siklus hidup objek dan pesan yang dikirim dan diterima antar objek (Noviantoro, 2022). *Sequence Diagram* menggambarkan perilaku objek dalam skenario *use case* dengan menjelaskan siklus hidup objek, Diagram ini menunjukkan interaksi antara objek dalam urutan waktu, membantu menggambarkan bagaimana proses terjadi secara dinamis. *Sequence Diagram* sangat berguna untuk memvisualisasikan alur komunikasi dan koordinasi antar komponen dalam sistem.

Tabel 2.4 Simbol *Sequence Diagram*

Simbol	Nama	Fungsi
	<i>Actor</i>	Menggambarkan orang yang sedang berinteraksi dengan sistem
	<i>Entity Class</i>	Menggambarkan hubungan yang akan dilakukan
	<i>Boundary Class</i>	Menggambarkan sebuah gambaran dari <i>foem</i>
	<i>Control class</i>	Menggambarkan penghubung antara <i>boundary</i> dengan <i>table</i>

Simbol	Nama	Fungsi
	<i>A Focus of Control &amp; A Life Line</i>	Menggambarkan tempat mulai dan berakhirnya <i>message</i>
	<i>A Message</i>	Menggambarkan pengiriman pesan

## 16. Class Diagram

Diagram kelas atau *Class Diagram* menggambarkan struktur sistem dari segi definisi kelas-kelas yang dibuat untuk membangun sistem (Maria, 2021).

Tabel 2.5 Simbol *Class Diagram*

Simbol	Nama	Fungsi
	Kelas	Kelas pada struktur sistem
	<i>Association</i>	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
	<i>Inheritance</i>	Relasi antar kelas dengan makna generalisasi-spesialisasi
	<i>Aggregation</i>	Relasi antar kelas dengan makna semua bagian( <i>whole-part</i> )

## B. Kajian Empiris

Penelitian yang dilakukan oleh Khoirul Fuadi (2023) penelitian ini menjabarkan proses pembuatan sistem informasi pelayanan dan pengaduan berbasis *web* di sekolah dengan menggunakan metode *extreme programming* dan *CodeIgniter* sebagai *framework*. Hasil penelitian dari sistem ini dapat mempermudah proses pelaporan aktivitas siswa yang melanggar aturan. Siswa dapat menggunakan sistem ini untuk melihat informasi mengenai jenis pengaduan yang telah diproses oleh administrator, serta untuk mengajukan pengaduan mereka sendiri. Selain itu, siswa juga dapat memberikan kritik dan saran, serta melihat riwayat kritik dan saran yang pernah mereka ajukan.

Penelitian selanjutnya dilakukan oleh Yoki Firmasyah, Reza Maulana, dan Nadiyah Fatin (2020). dalam penelitian ini akan menjelaskan bagaimana pembuatan proses pengaduan berbasis *website* di kelurahan. Sistem informasi akan dibuat dengan menggunakan metode *prototype*. Hasil dari sistem yang sudah dibuat dapat mempermudah warga dalam melaporkan atau melakukan pengaduan secara cepat dan efisien serta menjadi pemecah masalah dalam kelurahan, para warga dapat melihat berita terbaru dari kelurahan yang akan ditempatkan pada *website*.

Penelitian ketiga yakni dari Josua Armando Silalahi dan Arief Budiman (2023). Menjelaskan bagaimana membuat sistem informasi pengaduan berbasis web di polsek. Dalam pembuatan sistem informasi menggunakan metode *extreme programmer* dan *PHP* sebagai media coding serta *MySQL* sebagai basis data. Hasil dari penelitian ini adalah mempermudah para warga dalam

menyampaikan keluhan yang sedang dialami tanpa mendatangi langsung ke tempat pengaduan.

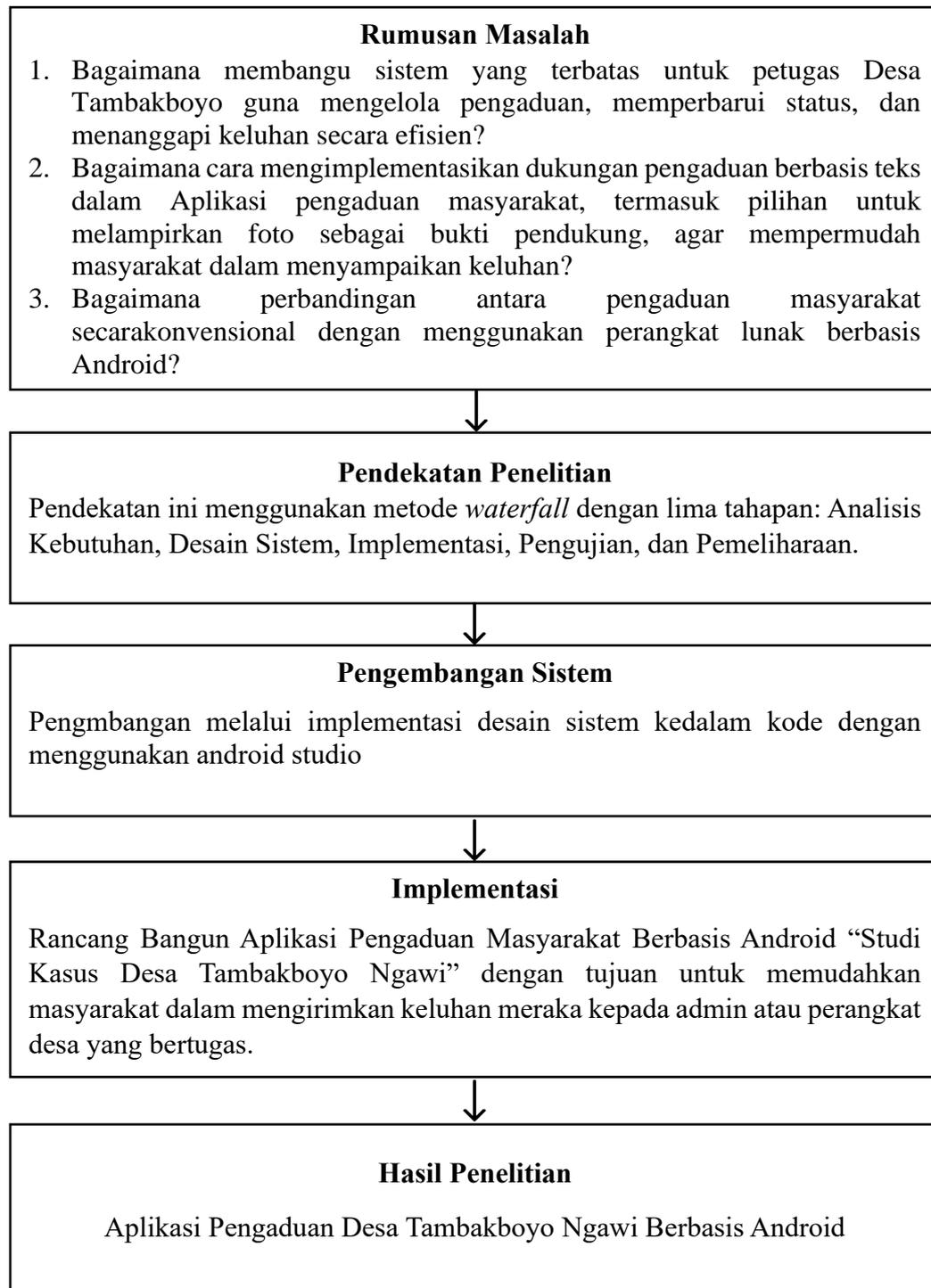
Kesimpulan yang dapat diambil dari ketiga penelitian yang telah dilakukan oleh peneliti sebelumnya yang berkaitan dengan Rancang Bangun Aplikasi Pengaduan Masyarakat Berbasis Android (Studi Kasus Tambakboyo Ngawi). Hasil yang diperoleh dalam dari kajian empiris yakni sistem informasi pengaduan yang sudah berhasil dibuat akan memudahkan para pengguna untuk menyampaikan apa yang menjadi keluhan mereka dalam kegiatan bermasyarakat. Dan sistem informasi menjadi tempat yang tepat untuk menyampaikan sebuah pengumuman maupun sebuah berita yang penting untuk diketahui oleh khalayak ramai.

### **C. Kerangka Berfikir**

Dalam pembuatan aplikasi terdapat rancangan untuk kerangka berpikir yang akan digunakan untuk memudahkan dalam pembuatan aplikasi dengan menggunakan metode *waterfall*. Tahapan pertama dalam pembuatan aplikasi yakni menganalisa kebutuhan yang akan digunakan dalam pembuatan aplikasi, karena ini sangat penting untuk kemajuan penelitian. Selanjutnya melakukan design sistem menggunakan *UML* sebagai konsep sistem kedalam bentuk visual. Tahap implementasi dilanjutkan dengan memasukkan *design* sistem kedalam kode dengan menggunakan *android studio* dan *visual studio code*, dilanjutkan dengan pengujian awal untuk memastikan aplikasi berjalan dengan baik.

Dalam kerangka berpikir ini, tahapan metode pengembang dirancang untuk memastikan keberhasilan dalam pembuatan aplikasi. Dalam setiap tahapan yang dilakukan akan memastikan bahwa aplikasi yang dikembangkan akan berhasil. Aplikasi pengaduan dikembangkan dengan aplikasi android, setelah melalui pengujian. Evaluasi aplikasi akan dilakukan dengan menggunakan respons dari pengguna yang akan digunakan sebagai perbaikan dan peningkatan aplikasi. Hal ini dilakukan untuk menjamin bahwa aplikasi bermanfaat dan sesuai dengan keinginan pengguna.

Untuk memahami konsep dan alur pemikiran yang mendasari penelitian ini, berikut disajikan gambar kerangka berpikir. Kerangka berpikir ini membantu dalam menjelaskan alur logis yang diikuti dalam penelitian. Silakan perhatikan gambar berikut untuk pemahaman yang lebih mendalam:



Gambar 2.2 Kerangka Berrpikir