

## **BAB II**

### **KAJIAN PUSTAKA**

#### **2.1 Teori Dasar**

Teori dasar merupakan kumpulan teori-teori yang akan di kumpulkan oleh peneliti untuk dijadikan sebagai referensi penelitian sehingga hasil yang didapat pada penelitian lebih akurat. Teori yang telah di kumpulkan berasal dari jurnal-jurnal, buku yang berkaitan dengan *Rapid Application Develoment (RAD)*.

#### **2.2 Perancangan**

Perancangan sistem melibatkan penentuan proses dan data yang diperlukan oleh sistem yang akan dibangun. Tahap ini berguna untuk memberikan gambaran kepada programmer mengenai rancangan sistem yang akan dibangun untuk untuk membuat sebuah aplikasi. Sesuai dengan komponen sistem yang dikomputerisasikan, maka yang harus didesain dalam tahap ini mencakup perangkat keras atau perangkat lunak komputer, basis data dan aplikasi (Tambunan & Zetli, 2020).

#### **2.3 Aplikasi**

Aplikasi merupakan program atau prangkat lunak komputer yang berasal dari kata *application* yang berarti penerapan, lamaran, penggunaan dengan cara menggabungkan beberapa fitur. Menurut Putra & Fauziah (2018), aplikasi adalah program jadi yang bertujuan untuk melakukan suatu fungsi yang bermamfaat bagi

pengguna, seperti penggunaannya sebagai media pembelajaran, media hiburan atau sebagai alat yang mempermudah pekerjaan seseorang.

## **2.4 Kamus**

Kamus merupakan buku yang menyimpan kata-kata dari bahasa tertentu yang biasanya disusun menurut abjad yang disertai dengan arti, ejaan dan sebagainya. Dalam kamus yang ideal, semestinya dilengkapi juga dengan informasi tentang pemenggalan kata, asal-usul kata dan informasi tentang kata baku. Penggunaan kamus dimaksudkan untuk keperluan akademik dan sebagai rujukan dalam berbahasa (Statistik, 2020). Berdasarkan laman resminya Badan Pengembangan dan Pembinaan Bahasa Kementerian Pendidikan dan Kebudayaan sedang mengembangkan kamus daring yang diberi nama kamus 4.0. Kamus baru dikembangkan saat ini dapat menterjemahkan Bahasa Indonesia ke dalam berbagai bahasa daerah seperti, Bima, Ternate, Madura, Melayu, Minangkabau, Sunda, Toraja, Batak, Tidore, Mandailing, Jawa, Melayu Palembang dan lain-lain.

## **2.5 *Android***

*Android* adalah sebuah sistem operasi seluler yang modifikasi berdasarkan sistem dari kernel linux dan perangkat lunak lain. Pertama kali diperkenalkan pada tahun 2008 dan mulai dikenal dari tahun ke tahun, dengan berbagai peningkatan hingga tahun 2017. *Android* adalah sistem operasi gratis yang bersifat *open source*, yang artinya *Google* mengizinkan pengguna untuk mengembangkan sistem operasi tersebut (Statistik, 2020). Pengguna *smartphone* yang menjalankan sistem

*android* juga dapat bebas mendownload aplikasi dan *game* dari toko aplikasi *android* yang bernama *Google Play Store*. Hingga saat ini *android* telah meliris banyak versi, mulai dari *Android 1.0* sampai yang terbaru adalah *Android 11*. Beberapa versi *android* memiliki nama yang berasal dari jenis *desert*, seperti *Cup Cake*, *Donut*, *Froyo*, *Jelly Bean*, *KitKat*, *Marshmallow*, *Oreo* dan *Pie*. Namun, dimasa yang akan datang *android* mungkin akan menggunakan sistem nomor yang telah terbukti, seperti *Android 1.0* dan *Android 11* (Lesmana & Silalahi, 2020).

#### 1. Versi 1.0 – 1.1 *Android*

Pertama kali dirilis pada tanggal 23 September 2008 oleh *Google* dan OHA (*Open Handset Alliance*) *android* versi alpha beta ini sudah dilengkapi dengan fitur dasar seperti *gmail*, *google talk*, *youtube* dan lainnya.



**Gambar 2.1** Versi 1.0 – 1.1 *Android*  
**Sumber:** (Shiddiq & Witanti, n.d.)

#### 2. Versi 1.5 *android*

*Android* versi ini pertama kali dikenal dengan nama *Cupcake* dirilis pada tanggal 30 April 2009 yang diambil dari nama kue. Perubahan pada aplikasi ini terdapat pada *UI* yang mendukung *keyboard virtual* pihak ketiga bisa menampilkan *widget*, memutar dan merekam video format *MPEG-4*



**Gambar 2.2** *Versi 1.5 Android*  
**Sumber:** (Kasema et al., 2018)

### 3. Versi 1.6 *android*

Pada tanggal 15 September 2009, versi 1.6 pertama kali dirilis dengan nama Donut. Fokus yang diberikan *Android* ini adalah pada beberapa celah dari versi sebelumnya, seperti pengoperasian yang lebih *fleksibel* diberbagai ukuran dan resolusi layar. Selain itu, tampilan antarmuka mengalami pengguna juga bertambah yang lebih *user friendly*.



**Gambar 2.3** *Versi 1.6 Android*  
**Sumber:** (Pasinggil & Palelleng, 2018)

### 4. Versi 2.0 – 2.1 *android*

Versi android yang diberi nama *Éclair* dirilis pada tanggal 26 Oktober 2009. Pada versi ini, tampilan antarmuka berubah menjadi lebih efektif dan menarik dengan berbagai warna. Selain itu navigasi belokan demi belokan dengan

panduansuara dan info *real-time* sekarang disebut dengan *Google Maps*, juga dihadirkan pada sistem operasi ini. Selain *Navigasi*, *Eclair* juga menghadirkan *wallpaper* animasi ke *Android* serta fungsi *Speech to text*



**Gambar 2.4** *Versi 2.0-2.1 Android*  
**Sumber:** (Pasinggil & Palelleng, 2018)

#### 5. Versi 2.2 android

Android yang diberi nama *Froyo* dirilis pada tanggal 10 Mei 2010 yang sudah memiliki fitur *USB tethering*, *wifi* dan *hotspot* yang memiliki kemampuan memindahkan data ke eksterna. Fitur-fitur ini masih digunakan hingga saat ini.



**Gambar 2.5** *Versi 2.2 Android*  
**Sumber:** (Pasinggil & Palelleng, 2018)

#### 6. Versi 2.3 android

*Gingerbread* dirilis pada tanggal 06 Desember 2010 memiliki simbol kue jahe berbasis kernal linux 2.6.3.5. *google* memperbaiki desain antar muka pengguna

dengan meningkatkan performa dan kesederhana tampilan.



**Gambar 2.6** *Versi 2.3 Android*  
**Sumber:** (Pasinggil & Palelleng, 2018)

7. *Versi 3.0-3.2 android*

*Android Honeycomb* dirilis pada tanggal 22 Februari 2011, dimana versi memiliki tampilan yang mewah dengan tampilan *multi-core*.



**Gambar 2.7** *Versi 3.0-3.2 Android*  
**Sumber:** (Pasinggil & Palelleng, 2018)

8. *Versi 4.0 android*

Android yang diberi nama *Ice cream sandwich* dirilis pada tanggal 19 Oktober 2011. Dalam versi ini *smartphone* dan tablet sudah mendukung *user interface* yang minimalis menggunakan NFC untuk mengirim data dengan mudah.



**Gambar 2.8** *Versi 4.0 Android*  
**Sumber:** (Pasinggil & Palelleng, 2018)

9. *Versi 4.1 android*

*Jelly Bean* dirilis pada tanggal 09 Juli 2012 dengan baterai lebih awet pada saat digunakan dan kamera yang didukung resolusi UHD 4K, serta fokus pada keamanan.



**Gambar 2.9** *Versi 4.1 Android*  
**Sumber:** (Pasinggil & Palelleng, 2018)

10. *Versi 4.4 android*

Kitkat dirilis pada tanggal 31 Oktober 2013 yang memiliki model *interface update*. Pada saat layar terkunci tampilan akan berubah menjadi albumpemutar musik.



**Gambar 2.10** *Versi 4.4 Android*  
**Sumber:** (Pasinggil & Paelleng, 2018)

11. *Versi 5.0 android*

Pada tanggal 17 Oktober 2014, sistem operasi *Lollipop* dengan fitur baru yang mendukung penggunaan dual sim *card* dan tampilan yang baru serta keamanan yang melindungi perangkat ketika dicuri atau hilang.



**Gambar 2.11** *Versi 5.0 Android*  
**Sumber:** (Pasinggil & Paelleng, 2018)

12. *Versi 6.0 android*

Pada tanggal 28 Mei 2015, *Marshmallow* diliris dengan kemampuan yang terintegrasi seperti *fingerprint* dan kemampuan dalam mengenali perintah suara





**Gambar 2.12** *Versi 6.0 Android*  
**Sumber:** (Pasinggil & Palelleng, 2018)

13. *Versi 7.0-7.1 Nougat*

Pada tanggal 22 Agustus 2016, *Nougat* diluncurkan dengan fitur *replay* untuk pemberitahuan dari *windows* dan kemampuan untuk membuka lebih dari satu *windows* secara bersamaan.



**Gambar 2.13** *Versi 7.0-7.1 Android*  
**Sumber:** (Pasinggil & Palelleng, 2018)

14. *Versi 8.0-8.1 android*

*Oreo*, yang dirilis pada tanggal 21 Agustus 2017 dilengkapi dengan fitur emoji yang telah diperbaharui, ketahanan baterai lebih lama, kecepatan dan efisien yang lebih baik.



**Gambar 2.14** *Versi 8.0 Android*  
**Sumber:** (Pasinggil & Palelleng, 2018)

15. *Versi 9 android*

Merupakan urutan android yang ke 15 dan dirilis pertama kali pada 6 Agustus 2018 yang memiliki fitur adaptive battery yang dapat mengatur pola daya dan aktifitas aplikasi.



**Gambar 2.15** *Versi 9 Android*  
**Sumber:** (Pasinggil & Palelleng, 2018)

16. *Versi 10 Android*

Pertama kali dirilis pada 3 September 2019, dengan fitur teks otomatis yang bisa memberikan teks pada video atau *podcast* yang direkam tanpa harus terhubung oleh *wifi*.



**Gambar 2.16** *Versi 10 Android*  
**Sumber:** (Pasinggil & Palelleng, 2018)

17. *Versi 11 Android*

Diberi nama dengan *Red velved* yang resmi dirilis pada 8 September 2020 dengan fitur balon yang bisa membuat pengguna melanjutkan percakapan setelah mengakses aplikasi lain.



**Gambar 2.17** *Versi 11 Android*  
**Sumber:** (Pasinggil & Palelleng, 2018)

18. *Versi 12 Android*

*Snow Cone* yang dirilis pertama kali pada 4 Oktober 2021, menawarkan fitur kecerdasan buatan yang secara otomatis menghasilkan tema warna untuk menu sistem dan aplikasi berdasarkan *wallpaper* pengguna.



**Gambar 2.18** *Versi 12 Android*  
**Sumber:** (Pasinggil & Paelleng, 2018)

#### 19. *Versi 13 Android*

Diperkenalkan pada 10 Februari 2022 dengan meningkatkan privasi, keamanan dan optimalisasi *UI*.



**Gambar 2.19** *Versi 13 Android*  
**Sumber:** (Pasinggil & Paelleng, 2018)

## 2.6 *UML (Unified Modeling Language)*

*UML (Unified Modeling Language)* adalah standar bahasa pemrograman yang digunakan untuk membuat gambaran dari program berorientasi objek. *UML* membantu memenuhi kebutuhan pengguna dengan menggambarkan pemodelan (Riyowati & Fadlilah, 2019).

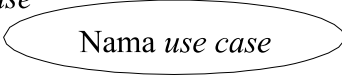

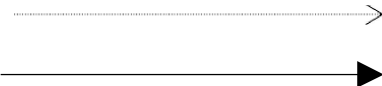
Fitur-fitur *UML* yang sering digunakan meliputi

## 1. Use Case Diagram

*Use case* digunakan untuk menggambarkan hak dan kewajiban dari setiap aktor yang saling berinteraksi. Untuk membuat diagram lebih mudah dimengerti, penamaannya harus menggunakan bahasa yang mudah dipahami orang luas. *Use case* juga menggambarkan tindakan dari setiap aktor di dalam sistem, seperti pengguna yang tidak dapat merubah aturan dari sistem.

*Diagram Use Case* memiliki Aktor sebagai komponen utama, yang mewakili pelaku atau orang, tempat dan interaksi dengan sistem. Menggunakan *Use Case* adalah cara untuk memberikan informasi agar setiap aktor memiliki aktivitas yang jelas, sehingga penyalahgunaan sistem dapat dicegah karena telah mematuhi alur yang sudah ditentukan (Hasan et al., 2022).

**Tabel 2. 1** *Use case diagram*

Simbol	Deskripsi
<i>Use case</i> 	Interaksi antara aktor adalah fungsi sistem.
Aktor / <i>Actor</i>  nama actor _____	Aktor merupakan bagian yang saling terkait dengan sistem, orang, dan proses di dalam desain sistem. Mereka berperan dalam menentukan bagaimana sistem bekerja untuk memenuhi tujuan pengembangannya (Hasan et al., 2022).
<i>Asosiasi / association</i>	Asosiasi yang tercipta antara aktor dan <i>use case</i> di dalam sistem.
Ektensi / <i>extend</i> 	Menciptakan use case baru yang berdiri sendiri dengan memperluas perilaku yang biasanya ditandai oleh kesamaan nama depan.
Generalisasi / <i>generalization</i>	Generalisasi dan spesialisasi adalah hubungan antara dua <i>use case</i> , baik secara umum maupun secara khusus.

**Tabel Lanjutan**



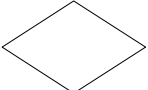


<p>Menggunakan / <i>include</i> / <i>uses</i></p> <p>-----&gt;</p> <p style="text-align: center;">&lt;&lt;include&gt;&gt;</p> <p>-----&gt;</p> <p style="text-align: center;">&lt;&lt;uses&gt;&gt;</p>	<p>Include: <i>Use case</i> dapat dipanggil jika terjadi proses peningkatan terhadap use case.</p> <p>Include: <i>Use case</i> akan memastikan bahwa penambahan telah dilakukan dengan benar.</p>
--	---

**Sumber:** (Hasan et al., 2022)

## 2. *Activity Diagram*

Diagram Aktivitas (*Workflow*) berfungsi untuk menjabarkan tahapan dari seluruh proses bisnis dan untuk menampilkan aliran kerja dengan susunan yang terpisah. Penggunaan *activity diagram* merupakan perwakilan dari proses yang akan dilakukan sistem, yang disajikan dalam bentuk diagram atau bagan kegiatan. Ini menyediakan gambaran dari struktur menu dari sebuah sistem terkait dan memungkinkan pengujian relatif mudah.

**Tabel 2. 2 Activity Diagram**



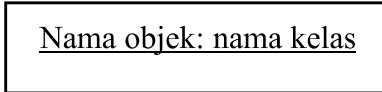

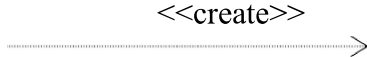

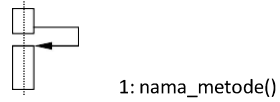
Simbol	Deskripsi
Simbol awal 	Keadaan awal sebuah sistem
Aktivitas 	Aktivitas tidak dapat dipisahkan dari tindakan yang dilakukan oleh sistem.
Percabangan / <i>decision</i> 	Penggolongan menurut kondisi khusus yang akan menghasilkan berbagai aktivitas.
Penggabungan / <i>join</i> 	Aktivitas dapat digabungkan menjadi satu.
Status akhir 	Kondisi akhir yang telah diproses oleh sistem.

**Sumber:** (Pasinggil & Palelleng, 2018)


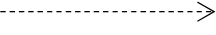
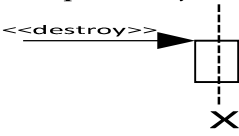
### 3. *Sequence Diagram*

*Sequence diagram* adalah ilustrasi dari aktivitas objek yang terkait dengan satu *use case* tertentu, serta memperhatikan durasi aktif dan pesan yang dikirimkan atau diterima oleh objek. Langkah-langkah aktivitas ini ditentukan dengan jelas dalam sebuah diagram secara berturut-turut dengan menggunakan metode yang telah ditetapkan. Semakin luas penggambaran *use case* semakin banyak *sequence* yang dipetakan dan dijelaskan oleh langkah-langkah aktivitas yang dapat dilakukan oleh aktor (Kasema et al., 2018).

**Tabel 2. 3** *Sequence Diagram*

Simbol	Deskripsi
Aktor  nama aktor	Aktor tak dapat dipisahkan dari kata benda, karena aktor bisa digambarkan sebagai individu, proses, atau sistem.
Garis hidup / <i>lifeline</i> 	<i>Lifeline</i> adalah suatu gambaran yang menunjukkan jalur hidup suatu objek dalam sistem.
Objek 	Objek adalah media yang digunakan aktor untuk berkomunikasi.
Waktu aktif 	Keadaan objek dinyatakan dalam jangka waktu berlangsung.
Pesan tipe <i>create</i> 	Ketika membuat siklus baru dengan tujuan yang berbeda, maka panah akan menunjuk pada objek itu sendiri.
Pesan tipe <i>call</i> 1 : nama_metode() 	Pemanggilan diri sendiri atau objek lain didalam sistem. 

**Tabel Lanjutan**

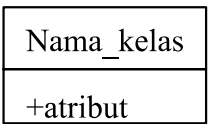


Pesan tipe <i>send</i> 1 : masukan 	Saat mengirim pesan berisi informasi atau instruksi, panah akan menuju pada objek terkait.
Pesan tipe <i>return</i> 1 : keluaran 	Saat mengirim balik respon dari pesan masuk berupa informasi, panah akan bergerak ke arah objek tersebut.
Pesan tipe <i>destroy</i> 	Menyatakan objek yang akan mengakhiri objek lain ditandai oleh panah yang menuju objek yang akan diakhiri, dengan pesan <i>destroy</i> sebagai kebalikannya dari <i>create</i> .

Sumber: (Kasema et al., 2018)

#### 4. *Class Diagram*



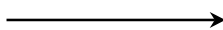

Class diagram menjelaskan struktur sebuah sistem dengan menggambarkan keterkaitan antar kelas-kelas yang ada melalui atribut dan metode. Susunan arsitektur Class yang baik berfokus pada menyederhanakan susunan kelas, meningkatkan modifikasi dan memperbaiki layanan. Pemisahan antar kelas, *Collaboration* dan Implementasi dapat digunakan untuk menerapkan susunan arsitektur Class yang baik. Berikut adalah susunan arsitektur *Class* yang baik (Pasinggil & Palelleng, 2018).

**Tabel 2. 4 *Class Diagram***

	<b>Simbol</b>	<b>Deskripsi</b>
Kelas		Variabel-variabel yang dimiliki sistem dan digunakan sebagai atribut.
Antarmuka/ <i>interface</i> Nama_ <i>interface</i>		Sama seperti konsep antarmuka dalam sebuah aplikasi.
Asosiasi/ <i>association</i>		Hubungan yang biasanya disertai dengan arti <i>multiplicity</i> .



**Tabel Lanjutan**

Assosiasi berarah / directed association 	Realsi, di mana satu kelas menggunakan informasi dari kelas lain untuk menyelesaikan masalah.
Generalisasi 	Generalisasi, ketika sebuah kelas memberikan informasi umum yang dapat diadaptasikan untuk situasi khusus.
Kebergantungan /dependency 	Pengaitan, di mana satu kelas bergantung pada informasi yang disediakan oleh kelas lain.
Agregasi 	Ketergantungan alur, di mana satu kelas memerlukan informasi dari kelas lain di setiap tahap.

**Sumber:** (Pasinggil & Palelleng, 2018)

### 2.7. RAD (*Rapid Application Development*)

Pengembangan aplikasi dengan RAD merupakan metode yang berfokus pada keseluruhan proses pengembangan aplikasi dengan menggunakan *feedback* berulang-ulang untuk mencapai pembuatan aplikasi secara cepat, namun tetap mematuhi tahap-tahap dalam pembuatan aplikasi (Trimahardhika & Sutinah, 2017).

### 2.8 *Software* Pendukung

Peneliti menggunakan perangkat tambahan, yang disebut *software* pendukung, untuk mendukung dan menyelesaikan penelitiannya. Berikut adalah *software* pendukung yang digunakan dalam penelitian ini:

### 2.8.1 Bahasa Pemrograman PHP

Rasmus Lerdorf membuat PHP pada awalnya untuk memantau orang-orang yang mengunjungi halaman webnya. Sering dengan berjalannya waktu, PHP kemudian terkenal dengan cepat dan makin populer, Lerdorf akhirnya mengeluarkannya sebagai proyek *open source*, dimana para pengembang bisa menggunakan, memperbaiki dan meyempunakan kode-kode ini. PHP telah berkembang menjadi. Walaupun dianggap tidak memiliki tujuan khusus, PHP telah berkembang menjadi penulisan bahasa skrip yang banyak dipakai. PHP memiliki sebuah fitur yang sangat berguna, yaitu kemampuan untuk disematkan ke file HTML. PHP merupakan singkatan dari *Hypertext Preprocessor*, yaitu bahasa pemrograman yang mirip dengan *JavaScript* dan *Python*. Perbedaan dari keduanya yaitu, PHP lebih berfokus pada komunikasi sisi server, sedangkan *JavaScript* dapat digunakan untuk *frontend* dan *backend*. *Python* hanya digunakan untuk sisi *server (backend)* (Hasan et al., 2022).

### 2.8.2 Framework Bootstrap

*Bootstrap* adalah *framework web development* gratis bersifat *open-source*, yang dibangun menggunakan HTML, CSS, dan *JavaScript*. Ini dirancang untuk mempercepat proses pengembangan *web* yang responsif dan *mobile-first* (memprioritaskan perangkat seluler). *Framework* ini bisa digunakan untuk mengembangkan *website* dengan lebih cepat. *Bootstrap* menyediakan *skrip* dan *syntax* yang bisa diterapkan untuk berbagai komponen dalam desain *web*. Tujuannya adalah untuk membuat *website* responsif dan *mobile-first*. Dengan

begitu, semua elemen antarmuka *website* akan bekerja secara optimal di semua ukuran layar, baik di desktop maupun di perangkat seluler (Putra & Fauziah, 2018).

### **2.8.3 Database MySQL**

*MySQL* adalah Sistem Manajemen Basis Data Relasional yang bersifat *open source*. Ini adalah perangkat lunak basis data yang bisa dipakai dengan bahasa pemrograman *server web*. Raharjo (dalam Muchtar, 2019) menyatakan bahwa *MySQL* menyatakan bahwa sebuah *server database* yang dapat memproses dan menampung data dalam jumlah yang besarsecara cepat dan dapat diakses oleh banyak pengguna. *MySQL* adalah *software open source* yang dipakai untuk membuat *database*.

## **2.9 Variabel Penelitian**

Variabel merupakan suatu objek pada penelitian yang akan di teliti oleh peneliti. Penelitian ini dilakukan pada narasumber yang berasal dari suku toraja dan aktif menggunakan bahasa Toraja.

### **1. Bahasa Toraja**

Bahasa Toraja Bahasa Toraja adalah bahasa yang dituturkan oleh etnis Toraja yang mendiami wilayah kabupaten Toraja Utara, Luwu, Mamasa, dan Tana Toraja termasuk salah satu dari banyak bahasa yang ada di Indonesia (Noorduyn, 1991). Bukan hanya sebagai alat komunikasi, tetapi Bahasa Toraja juga merupakan salah satu warisan budaya yang perlu dilestarikan sebagai bagian dari kekayaan budaya Indonesia. Berdasarkan sensus pada tahun 2000, Bahasa Toraja berada pada

level developing yang jumlah penuturnya diperkirakan sekitar 750.000 jiwa (Pasinggil & Palelleng, 2018).

## 2.10 Penelitian Terdahulu

Pada penelitian ini peneliti menggunakan beberapa penelitian terdahulu sebagai referensi tambahan penelitian.

1. Penelitian Siregar & Handoko (2021) menyoroti tentang “Perancangan aplikasi pengelolaan stock material dengan QR kode di PT SP Manufacturing Batam Berbasis *Android*”. Dalam kajian ini, aplikasi yang dibangun adalah program siap pakai yang bisa digunakan untuk mencapai tujuan tertentu oleh pengguna. Selain itu, aplikasi ini juga dapat dimanfaatkan sebagai media pembelajaran, hiburan, dan alat untuk memudah.
2. Penelitian Pasinggil & Palelleng (2018) tentang “Digitalisasi Kamus Bahasa Toraja dan Perancangan Basis Data” menyatakan bahwa bahasa Toraja merupakan salah satu warisan budaya yang harus dilindungi. Status bahasa Toraja berada pada tingkat yang sedang berkembang, dengan jumlah penutur sekitar 750.000 orang (sensus tahun 2000).
3. Menurut penelitian Tambunan & Zetli (2020) berjudul "Perancangan Sistem Peramalan Cuaca Berbasis Logika *Fuzzy* Di Kota Batam", proses dan data yang diperlukan oleh sistem baru harus ditentukan agar bisa berjalan dengan baik. Manfaat dari tahap perancangan sistem ini adalah memberikan petunjuk gambaran rancangan bangun yang lengkap bagi programmer dalam mengembangkan aplikasi. Komponen sistem yang dikomputerisasikan,

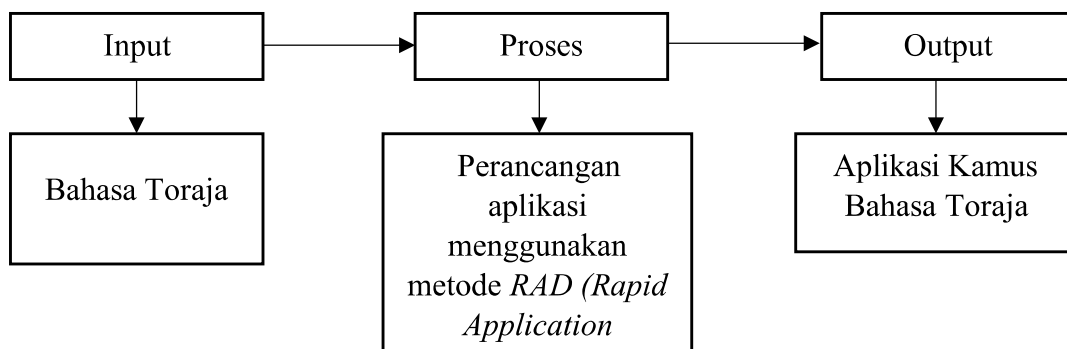
seperti perangkat keras atau perangkat lunak, basis data dan aplikasi, juga harus didesain pada tahap ini.

4. Menurut penelitian Hasan et al. (2022) berjudul “Digitalisasi Kamus Bahasa Daerah Papua Menggunakan Metode *Rapid Application Development*”, PHP awalnya diciptakan oleh Rasmus Lerdorf untuk mengawasi orang-orang yang mengunjungi laman *web* miliknya. Selama bertahun-tahun, bahasa ini menjadi semakin populer dan Lerdorf akhirnya merilisnya sebagai proyek *open-source*.
5. Menurut penelitian Mubarak (2019) berjudul “Rancang Bangun Aplikasi *Web* Sekolah Menggunakan UML”, PHP adalah bahasa pemrograman server yang memungkinkan integrasi antara *script* PHP dengan HTML (*server side HTML embedded scripting*). Ini merupakan bahasa pemrograman *Open Source* yang dapat digunakan pada berbagai sistem operasi seperti *Linux*, *Windows*, *Unix*, dan sebagainya.
6. Menurut penelitian Putra & Fauziah (2018) berjudul “Perancangan Aplikasi Presensi Dosen *Realtime* dengan Metode *Rapid Application Development (RAD)* Menggunakan *Fingerprint* Berbasis *Web*”, *Rapid Application Development (RAD)* adalah metode yang berfokus pada pengembangan aplikasi dalam waktu singkat dengan berbagai tahap pengembangan dan *feedback*. Metode ini dirancang untuk memenuhi target waktu pengembangan aplikasi yang singkat tanpa menghilangkan tahapan pembuatan aplikasi.
7. Menurut penelitian Statistik (2020) berjudul "Perancangan Aplikasi Kamus Sinonim dan Antonim Berbahasa Indonesia Berbasis Android", UML adalah

standar bahasa pemrograman yang digunakan untuk membuat visualisasi dari sebuah program berorientasi objek. UML membantu memenuhi keinginan pengguna dengan secara visual menggambarkan modelnya.

### 2.11 Kerangka Pemikiran

Kerangka pemikiran merupakan penjelasan mengenai hubungan antara variabel yang akan diteliti. Berikut di bawah ini merupakan kerangka penelitian pada penelitian ini:



**Gambar 2.20** Kerangka Pemikiran  
**Sumber:** (Data Penelitian 2022)