

BAB I

PENDAHULUAN

A. Latar Belakang

Perkembangan teknologi yang cepat telah menyebabkan dominasi penggunaan smartphone dalam penggunaan telepon seluler. Dengan kemunculan smartphone, fungsi dasar dari ponsel tradisional telah digantikan secara bertahap. Saat ini, ponsel tidak hanya digunakan untuk komunikasi melalui panggilan dan pesan singkat, tetapi juga dilengkapi dengan berbagai aplikasi menarik. Pengguna smartphone, terutama yang menggunakan sistem operasi Android, dapat mengunduh aplikasi yang mereka butuhkan melalui Google Play Store, yang menyediakan beragam layanan aplikasi.

Kamus adalah sejenis buku yang berfungsi untuk memuat definisi dari kata-kata atau istilah yang menjelaskan makna (Khairul, K., 2018). Dalam hal istilah nama latin hewan, kamus menjelaskan arti dari istilah-istilah tersebut, dan saat ini, keberadaan kamus sangatlah penting untuk meningkatkan pengetahuan dan pemahaman. Meskipun penggunaan kamus dalam bentuk buku masih dominan, namun hal ini menyebabkan kurangnya minat masyarakat karena buku tersebut tebal dan pencariannya manual, sehingga menurunkan minat dalam memperoleh pengetahuan dan pemahaman mengenai nama latin hewan.

Namun, penting untuk diperhatikan bahwa keefektifan aplikasi tersebut sangat bergantung pada cara pencarian informasi yang diterapkan di dalamnya. Metode

pencarian yang tidak efisien dapat menghambat pengalaman pengguna dan mengurangi daya tarik aplikasi tersebut. Dalam hal ini, implementasi algoritma binary search menjadi relevan. Algoritma ini telah terbukti efisien dalam mencari elemen tertentu dalam kumpulan data yang diurutkan. Dengan menerapkan algoritma binary search pada aplikasi kamus nama latin hewan, diharapkan dapat meningkatkan kinerja pencarian informasi, mengurangi waktu yang diperlukan untuk menemukan data yang diinginkan, dan meningkatkan kepuasan pengguna.

Oleh karena itu diperlukan pembuatan sebuah aplikasi sebagai upaya untuk meningkatkan pengetahuan dan pemahaman masyarakat mengenai kamus nama latin hewan. Aplikasi ini akan menyajikan informasi mengenai nama latin hewan dengan desain yang menarik untuk menarik minat masyarakat. Dengan adanya aplikasi ini, masyarakat akan lebih mudah mencari informasi mengenai nama latin hewan secara efisien, karena satu aplikasi dapat digunakan oleh banyak orang.

Dari penjelasan masalah yang telah disampaikan, penulis merasa tertarik untuk mengangkat permasalahan dengan judul “**Implementasi Algoritma Binary Search Pada Aplikasi Kamus Nama Latin Hewan Berbasis Android**”.

B. Rumusan Masalah

Berdasarkan latar belakang masalah yang di jelaskan, rumusan masalah dalam penelitian ini adalah bagaimana membuat aplikasi pencarian nama latin hewan berbasis android dengan algoritma binary search ?

C. Tujuan Penelitian

Berdasarkan yang telah di jelaskan, maka tujuan dari penelitian ini adalah untuk mengimplementasikan algoritma *binary search* pada aplikasi kamus nama latin hewan.

D. Batasan Masalah

Agar penelitian ini lebih terarah, maka penulis membatasi masalah yang akan dibahas yaitu:

1. Aplikasi ini dirancang dengan berfokus pada penggunaan nama latin hewan sebagai sumber informasi. Dalam upaya memberikan informasi yang mendalam dan bermanfaat kepada pengguna.
2. Peneliti membatasi kamus latin hewan yang terdapat dalam aplikasi android yang berisi teks, gambar, dan tidak menampilkan audio dan video

E. Manfaat Penelitian

Adapun manfaat dari dari penelitian ini adalah sebagai berikut:

1. Bagi akademik

Penulis dapat menerapkan ilmu yang dipreoleh selama proses perkuliahan.

2. Bagi penulis

Sebagai bahan referensi bagi penulis lain untuk mengembangkan kemampuan dibidang yang sama.

3. Bagi Masyarakat

Memberikan kemudahan akses terhadap kamus nama latin hewan yang memungkinkan masyarakat untuk belajar dan mendapat informasi

BAB II

TINJAUAN PUSTAKA

A. Kajian Teori

1. Kamus

Menurut Kamus Besar Nama Indonesia, “Kamus adalah sejenis buku rujukan yang menerangkan makna kata-kata yang berfungsi untuk membantu seseorang mengenal perkataan baru”. Selain itu membantu seseorang memahami kata-kata yang baru. Selain memberikan definisi, kamus juga bisa mencakup panduan pelafalan, asal-usul kata (etimologi), dan contoh penggunaan kata-kata tersebut. Jenis-jenis kamus meliputi kamus nama, kamus istilah, kamus budaya, dan kamus nama latin. Namun, dalam konteks ini, kamus yang dimaksud adalah kamus elektronik, bukan buku fisik.

2. Hewan

Hewan, juga dikenal sebagai binatang, margasatwa, atau satwa, merupakan kelompok organisme yang tergolong dalam kerajaan Animalia atau Metazoa, dan merupakan salah satu dari berbagai jenis makhluk hidup yang ada di alam semesta. Hewan dapat berupa organisme uniselular atau multiselular. Semua hewan memiliki sifat heterotrof, yang berarti mereka memperoleh energi dengan mengonsumsi materi organik dari lingkungan sekitarnya (Rahmanto, Y. 2021). Adapun klasifikasi hewan yaitu :

a. Vertebrata

Hewan vertebrata, yang memiliki tulang belakang atau punggung, dipelajari dalam bidang vertebrata dengan fokus pada pemahaman sistematika berdasarkan analisis struktur anatomi dan fisiologi. Ini mencakup kajian tentang otot dan rangka, sistem peredaran darah dan pernapasan, sistem pencernaan, sistem reproduksi, sistem ekskresi, sistem syaraf dan indera, serta sistem endokrin. Phylum chordata, yang terdiri dari dua superclass yaitu Pisces dan Tetrapoda (amfibi, reptil, burung, dan mamalia), merupakan bagian dari subjek tersebut. Hewan vertebrata memiliki struktur tubuh yang lebih kompleks dibandingkan dengan hewan invertebrate (Maya dan Nur, 2021).

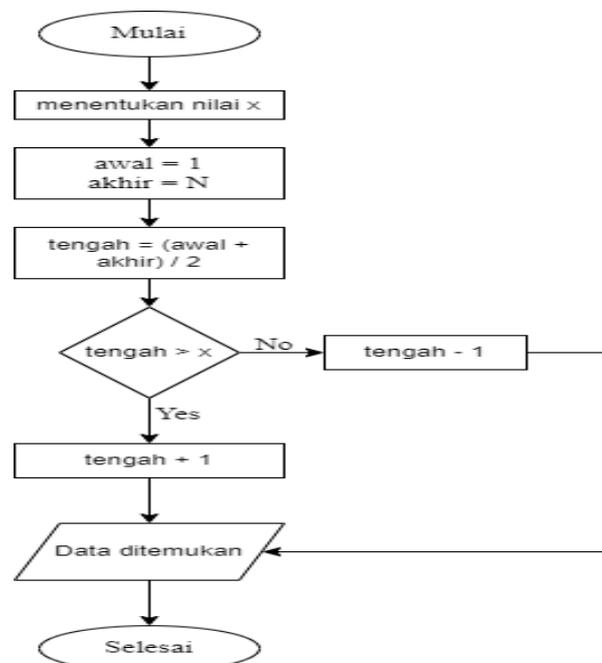
b. Invertebrata

(Maya dan Nur, 2020) Invertebrata adalah kelompok hewan yang tidak memiliki tulang belakang atau punggung, yang mencakup berbagai spesies seperti artropoda (serangga, laba-laba, dan krustasea), moluska (siput, gurita, dan kerang), echinodermata (bintang laut dan teripang), dan lainnya. Invertebrata dapat ditemukan di berbagai habitat, baik di air maupun di darat, dengan beragam ukuran tubuh. Selain hidup secara bebas, beberapa di antaranya, seperti protozoa, cacing, dan artropoda, juga dapat hidup secara parasit, memanfaatkan organisme lain untuk kelangsungan hidup mereka (Maya dan Nur, 2020).

3. Binary Search

Binary search merupakan metode pencarian yang merupakan salah satu variasi dari pendekatan *decrease and conquer*. Prinsip dasar *decrease and*

conquer adalah mengurangi suatu masalah menjadi sub-masalah yang lebih kecil. Dalam pendekatan ini, setiap sub-masalah dipecahkan secara terpisah tanpa mengintegrasikan solusi dari setiap sub-masalah tersebut. Variasi dari pendekatan decrease and conquer dapat dibagi menjadi tiga jenis, yaitu *decrease by a constant*, *decrease by a constant factor*, dan *decrease by a variable size*. Binary search adalah contoh dari algoritma yang menerapkan pendekatan *decrease by a constant factor*. Dalam *decrease by a constant factor*, ukuran masalah dikurangi secara konsisten dengan faktor tetap pada setiap iterasi (Andriani, A., 2021). Biasanya, faktor konstan yang digunakan adalah 2, jarang sekali nilai yang lebih besar dari 2 digunakan. Binary pada "*binary search*" tidak berarti biner dalam arti bilangan biner (binary numbers). Jadi kata "binary" di sini tidak terkait dengan bilangan biner, tapi merujuk pada proses membagi menjadi dua bagian yang digunakan dalam algoritma pencarian ini.



Gambar 2.1 Flowchart *Binary Search*

(Wijaya, A., 2021) *Binary search* dapat dilakukan hanya jika data sudah dalam keadaan terurut. Artinya, jika data belum terurut, maka pencarian biner tidak dapat dilakukan secara efisien. *Binary Search* digunakan untuk :

- a. Mengurangi jumlah operasi perbandingan antara data yang ingin ditemukan dan data yang ada dalam tabel, terutama saat data tersebut memiliki jumlah yang sangat besar.
- b. Kurangnya beban komputasi karena pencarian dilakukan dengan berbagai arah.
- c. Melakukan pembagian ruang pencarian secara berulang hingga data ditemukan atau tidak ada lagi ruang untuk dibagi (berarti kemungkinan data tidak ditemukan).
- d. Membutuhkan persyaratan bahwa data dalam tabel harus telah diurutkan.

Adapun prinsi dari binary search yang dijabarkan sebagai berikut :

- a. Menentukan data yang dicari = X.
- b. Langkah pertama adalah menetapkan posisi awal = 1 dan posisi akhir = N.
- c. Kemudian, mencari posisi data tengah dengan menggunakan rumus (posisi awal + posisi akhir) / 2.
- d. Setelah itu, data yang dicari dibandingkan dengan data tengah.
- e. Jika data yang dicari lebih kecil, proses dilanjutkan kembali dengan menetapkan posisi akhir sama dengan posisi tengah - 1.
- f. Sedangkan jika data yang dicari lebih besar, proses dilanjutkan kembali dengan menetapkan posisi awal sama dengan posisi tengah + 1.
- g. Proses ini terus berlanjut hingga data tengah sama dengan data yang dicari.

4. Sistem Bilangan Biner

Bilangan biner juga dikenal sebagai sistem bilangan basis 2 merupakan jenis bilangan yang terdiri dari dua simbol, yaitu nol dan satu. Sehingga setiap digit dalam bilangan biner dinyatakan sebagai kelipatan dari 2. Dalam penghitungan bilangan biner, digunakan metode 2^n . Setiap digit biner biasanya disebut sebagai bit, sementara empat bit membentuk sebuah nibble, dan delapan bit membentuk satu byte (Wurara, D., 2020).

5. Dasar Pemrograman

Dasar pemrograman adalah konsep dan prinsip fundamental yang perlu dipahami oleh seseorang yang ingin belajar atau bekerja dalam bidang pemrograman. Berikut adalah beberapa dasar pemrograman yang penting :

a. Algoritma

Algoritma adalah serangkaian langkah atau instruksi yang terstruktur dan logis yang digunakan untuk menyelesaikan masalah atau mencapai tujuan tertentu. Dalam konteks pemrograman, algoritma adalah blueprint atau rencana yang digunakan untuk menulis program komputer yang dapat memproses data dan menghasilkan output yang diinginkan.

b. Flowchart

Flowchart dalam bahasa indonesia diterjemahkan sebagai diagram alir. Dari dua kata ini, maka dapat kita bayangkan bahwa flowchart ini berbentuk diagram yang bentuknya dapat mengalirkan sesuatu. Flowchart adalah representasi grafik dari langkah-langkah yang harus

diikuti dalam menyelesaikan suatu permasalahan yang terdiri atas sekumpulan simbol, dimana masing-masing simbol mempersentasikan suatu kegiatan tertentu (Ade Hastuty Hasyim, 2021).

6. Android



Gambar 2.2 Logo *Android*

Android adalah sebuah sistem operasi untuk perangkat seluler yang berbasis pada kernel Linux, terdiri dari sistem operasi, middleware, dan aplikasi. Awalnya dikembangkan oleh perusahaan kecil di Silicon Valley bernama Android Inc., kemudian diadopsi oleh Google pada tahun 2005 dan dibuat menjadi sistem operasi "*Open Source*". Hal ini membuatnya tersedia bagi siapa pun tanpa biaya, termasuk kode sumber yang digunakan untuk mengkompilasi sistem operasi tersebut. Sebagai hasilnya, Android telah menjadi sistem operasi paling banyak digunakan di dunia, karena sifat terbukanya bagi para pengembang untuk meningkatkan fungsionalitasnya (Mide, B., & Masnur, M., 2021).

Android menyediakan beragam lingkungan untuk pengembangan aplikasi, di mana setiap aplikasi diberikan level yang setara. Sistem ini tidak membedakan antara aplikasi inti dan aplikasi pihak ketiga. Melalui Application Programming Interface (API) yang disertakan, pengembang dapat mengakses perangkat keras, data ponsel cerdas, atau bahkan data sistem. Pengguna juga memiliki fleksibilitas untuk menghapus aplikasi inti dan menggantinya dengan aplikasi pihak ketiga.

Adapun Kelebihan dari android yaitu:

- a. Ketersediaan platform yang luas. Android merupakan sistem operasi yang digunakan oleh berbagai jenis perangkat seluler, termasuk smartphone, tablet, smartwatch, dan banyak lagi. Ini memberi pengembang akses ke pasar yang besar dengan jutaan pengguna potensial.
- b. Lingkungan pengembangan terbuka. Android memiliki lingkungan pengembangan yang terbuka, yang memungkinkan pengembang untuk mengakses dan memodifikasi kode sumber sistem operasi sesuai kebutuhan mereka. Hal ini memungkinkan kreativitas dan inovasi yang lebih besar dalam pengembangan aplikasi.
- c. Fleksibilitas dalam pengembangan. Android memberikan fleksibilitas dalam pengembangan aplikasi, memungkinkan pengembang untuk membuat aplikasi dengan berbagai macam fitur dan fungsionalitas.
- d. Platform gratis. Pengembang dapat mengembangkan, mendistribusikan, dan memperdagangkan sistem operasi ini tanpa perlu membayar royalti untuk lisensi

7. *Android Studio*



Gambar 2.3 *Logo Android Studio*

Android Studio adalah sebuah Integrated Development Environment (IDE) yang digunakan untuk pengembangan aplikasi Android. Perkenalan Android Studio terjadi pada acara Google I/O 2013 oleh Google. IDE ini merupakan pengembangan dari Eclipse IDE, dan didasarkan pada IntelliJ IDEA, sebuah IDE

Java yang terkenal. Sebagai IDE resmi untuk pengembangan aplikasi Android, Android Studio menyediakan berbagai fitur baru yang tidak ada di Eclipse IDE. Salah satu perbedaan utama antara keduanya adalah penggunaan Gradle sebagai lingkungan build oleh Android Studio, sedangkan Eclipse menggunakan Ant.

Android Studio sendiri dikembangkan berdasarkan IntelliJ IDEA, mirip dengan Eclipse, disertai dengan plugin ADT (*Android Development Tools*). Android studio memiliki fitur:

1. Proyek menggunakan *Gradle Build* sebagai basisnya.
 2. Refactoring dan debugging yang cepat untuk memperbaiki bug.
 3. Lint adalah sebuah alat baru, diklaim dapat secara efisien memantau kecepatan, kegunaan, dan kompatibilitas aplikasi.
 4. Mendukung Proguard dan penandatanganan aplikasi untuk meningkatkan keamanan.
 5. Memiliki antarmuka grafis (GUI) yang lebih intuitif untuk pengembangan aplikasi Android.
 6. Dapat diintegrasikan dengan Google Cloud Platform untuk setiap aplikasi yang dikembangkan.
8. Java



Gambar 2.4 Logo Java

Java adalah nama pemrograman berbasis objek yang memungkinkan pembuatan dan eksekusi perangkat lunak pada berbagai perangkat dan platform komputer. Java memiliki kemampuan multiplatform, yang berarti aplikasi Java dapat dijalankan di berbagai jenis platform selama Java Virtual Machine (JVM) sudah terinstal di platform tersebut. Dengan demikian, program yang dikembangkan di platform Windows juga dapat dijalankan di platform Linux, Unix, dan lainnya (Kasema, L. O., 2018).

Nama pemrograman Java pertama kali dikembangkan melalui Proyek Hijau, yang berlangsung selama 18 bulan, mulai dari awal tahun 1991 hingga musim panas 1992. Pada awalnya, proyek ini belum menggunakan nama "Java". Proyek tersebut didirikan oleh Patrik Naughton, Mike Sheridan, James Gosling, dan Bill Joy, bersama sembilan programmer lain dari Sun Microsystems. Salah satu hasil dari proyek ini adalah penciptaan maskot Duke yang dibuat oleh Joe Cross. Mereka mengembangkan browser (Mosaic) sebagai fondasi awal untuk menciptakan browser Java pertama yang dinamai Web Runner, terinspirasi oleh film Blade Runner dari tahun 1980-an. Pada rilis pengembangan pertama, nama Web Runner diubah menjadi Hot Java.

Pada sekitar Maret 1995, kode sumber Java versi 1.0a2 pertama kali dibuka. Kesuksesan mereka kemudian dilaporkan oleh surat kabar San Jose Mercury News pada 23 Mei 1995. Nama Oak tidak digunakan untuk versi rilis Java karena masalah merek dagang dengan perangkat lunak lain. Sebagai gantinya, nama "Java" diadopsi. Nama ini terinspirasi oleh kopi murni yang disukai oleh Gosling, yang dikenal sebagai "java coffee", yang biasanya ditanam di Pulau Jawa. Oleh

karena itu, nama nama pemrograman Java berasal dari kata "Java", yang juga merupakan nama nama inggris untuk Pulau Jawa.

9. MySQL



Gambar 2.5 Logo MySQL

MySQL adalah sistem manajemen basis data relasional (RDBMS) yang sumber terbukanya dan mendukung multithreading. Diciptakan oleh Michael "Monty" Widenius pada tahun 1995. Pada tahun 2000, MySQL dirilis dengan lisensi ganda, memungkinkan penggunaan gratis di bawah lisensi GNU GPL (General Public License), yang mengakibatkan meningkatnya popularitasnya (Silalahi, M. 2018). Perusahaan yang bertanggung jawab atas pengembangan MySQL adalah MySQL AB, yang sekarang menjadi bagian dari Sun Microsystems. MySQL AB melaporkan adanya lebih dari 6 juta instalasi MySQL di seluruh dunia, dengan rata-rata 50.000 unduhan instalasi per hari dari situs resmi dan mirror. Keberhasilan MySQL sebagai sistem basis data terkemuka tidak hanya karena harganya yang terjangkau, tetapi juga karena kehandalan, kinerja, dan fitur-fiturnya.

Banyak fitur yang tersedia di MySQL membuatnya tetap menjadi sistem basis data yang unggul. Kecepatan adalah salah satu fitur yang paling menonjol. Dalam perbandingan oleh eWeek antara beberapa sistem basis data seperti MySQL, Oracle, MS SQL, IBM DB2, dan Sybase ASE, MySQL dan Oracle menunjukkan

kinerja dan skalabilitas terbaik. MySQL mampu menangani dengan cepat dan lancar jumlah tabel yang mencapai puluhan ribu dan miliaran baris data.). MySQL menggunakan Structured Query Language (SQL) sebagai nama untuk mengakses dan mengelola data.

Komponen penyimpanan, yang mengelola query dan memfasilitasi pernyataan SQL pengguna ke dalam penyimpanan data, merupakan elemen krusial dalam setiap Sistem Manajemen Basis Data (DBMS). MySQL menawarkan berbagai opsi komponen penyimpanan dengan kelebihan unik masing-masing. Sebagai contoh, ada komponen penyimpanan yang transaction-safe yang memungkinkan pengembalian data ke keadaan sebelumnya (rollback). Selain itu, MySQL memiliki beragam fungsi internal. Kecepatan dan peningkatan stabilitas juga merupakan ciri khas yang membuat MySQL sangat terkenal.

10. XML

Menurut Sudirman (dalam Ramadhani, R. Z, 2019) XML (Extensible Markup Language) adalah nama markup yang menggunakan tag untuk mengkategorikan dan menjelaskan data secara spesifik. Dalam XML, terdapat tag pembuka <tag> dan tag penutup </tag>, dengan perbedaan karakter "/" pada tag penutup. Fungsinya utama adalah untuk komunikasi antara aplikasi, integrasi data, dan komunikasi aplikasi eksternal dengan pihak luar. Dengan standar XML, aplikasi yang berbeda dapat berkomunikasi dengan mudah satu sama lain.

XML singkatan dari Extensible Markup Language, merupakan salah satu bagian dari Standar Generalized Markup Language (SGML) yang telah ditetapkan oleh World Wide Web Consortium (W3C). XML masuk dalam kategori nama

yang dapat diperluas karena memungkinkan pengguna untuk menentukan elemen-elemen markup mereka sendiri. Ini berbeda dengan HTML, di mana elemen-elemen tertentu harus ditentukan untuk tampilan di peramban web.

XML juga merupakan dokumen yang fleksibel yang strukturnya mencerminkan struktur data yang terdapat di dalamnya. Secara keseluruhan, struktur dokumen XML dapat dibayangkan sebagai hierarki pohon yang terdiri dari elemen-elemen seperti akar (root), induk (parent), dan anak (child).

11. UML

UML (*Unified Modeling Language*) adalah alat desain sistem berorientasi objek. Secara filosofis kemunculan UML terinspirasi dari konsep yang sudah ada yaitu konsep pemodelan *Object Oriented* (OO), karena konsep ini menganalogikan sistem seperti kehidupan nyata yang didominasi oleh objek dan digambarkan atau dicatat dalam simbol-simbol yang cukup spesifik. proses standar dan independen (Masnur, M., & Difla, D. I., 2021).

UML (*Unified Modeling Language*) adalah nama visual untuk memodelkan sistem atau perangkat lunak yang berbasis paradigma berorientasi objek (Kasema, L. O., 2018). UML membantu pengembang untuk menggambarkan aliran sistem yang akan dibangun. Aliran sistem tersebut direpresentasikan oleh simbol-simbol dalam diagram-diagram.

Fokus utama dari diagram UML adalah memfasilitasi komunikasi di antara tim pengembangan proyek, mengeksplorasi berbagai desain yang mungkin, dan memvalidasi arsitektur perangkat lunak atau kode yang ditulis. Komponen atau notasi dalam UML berasal dari tiga notasi yang ada sebelumnya: Grady Booch

dengan OOD (Object-Oriented Design), James Rumbaugh dengan OMT (Object Modeling Technique), dan Ivar Jacobson dengan OOSE (Object-Oriented Software Engineering).

UML dibagi menjadi tiga kategori utama yaitu diagram struktur, diagram perilaku, dan diagram interaksi. Setiap kategori memiliki berbagai diagram yang menggambarkan arsitektur sistem dan saling terhubung satu sama lain

Adapun daftar simbol UML yaitu :

Tabel 2.1 *Symbol Usecase Diagram*

No	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menentukan kumpulan peran yang dijalankan oleh pengguna saat berinteraksi dengan <i>Use Case</i> .
2		<i>Dependency</i>	Hubungan di mana modifikasi pada suatu elemen yang berdiri sendiri (<i>Independent</i>) akan berdampak pada elemen-elemen yang bergantung padanya, yaitu elemen-elemen yang tidak berdiri sendiri.
3		<i>Generalization</i>	Hubungan di mana objek turunan (<i>Descendent</i>) mengadopsi perilaku dan struktur data dari objek yang berada di atasnya, yaitu objek induk (<i>Ancestor</i>).
4		<i>Include</i>	Menyatakan secara eksplisit sumber use case.
5		<i>Extend</i>	Menentukan paket yang memperlihatkan sistem dengan cakupan yang terbatas.
6		<i>Association</i>	Apa yang menghubungkan satu objek dengan objek lainnya.
7		<i>System</i>	Menentukan paket yang memperlihatkan sistem dengan cakupan yang terbatas.
8		<i>Use Case</i>	Deskripsi dari serangkaian tindakan yang dilakukan oleh sistem yang menghasilkan hasil yang dapat diukur bagi seorang aktor.

No	GAMBAR	NAMA	KETERANGAN
9		<i>Collaboration</i>	Kolaborasi antara aturan-aturan dan elemen lain yang berkerja bersama untuk menghasilkan perilaku yang lebih besar daripada jumlah dan elemen-elemennya (sinergi).
10		<i>Note</i>	Objek fisik yang ada ketika aplikasi berjalan dan mencerminkan sumber daya komputasi.

Tabel 2.2 *Symbol Class Diagram*

No.	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan di mana objek turunan (<i>descendent</i>) memiliki perilaku dan struktur data yang berasal dari objek di atasnya, yaitu objek induk (<i>ancestor</i>).
2		<i>Nary Association</i>	Upaya untuk menghindari keterkaitan dengan lebih dari dua objek.
3		<i>Class</i>	Kumpulan objek yang memiliki atribut dan operasi yang serupa.
4		<i>Collaboration</i>	Kolaborasi antara aturan-aturan dan elemen lain yang berkerja bersama untuk menghasilkan perilaku yang lebih besar daripada jumlah dan elemen-elemennya (sinergi).
5		<i>Realization</i>	Operasi yang sebenarnya dijalankan oleh suatu objek.
6		<i>Dependency</i>	Hubungan di mana modifikasi pada suatu elemen yang berdiri sendiri (<i>Independent</i>) akan berdampak pada elemen-elemen yang bergantung padanya, yaitu elemen-elemen yang tidak berdiri sendiri.
7		<i>Association</i>	Apa yang menghubungkan satu objek dengan objek lainnya

Tabel 2.3 *Symbol Sequence Diagram*

No	GAMBAR	NAMA	KETERANGAN
----	--------	------	------------

1		<i>LifeLine</i>	Objek entity, antarmuka yang berinteraksi bersama..
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktivitas yang terjadi.
3		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktivitas yang terjadi.

Tabel 2.4 *Symbol State Chart Diagram*

No	GAMBAR	NAMA	KETERANGAN
1		<i>State</i>	Nilai atribut dan hubungan pada suatu waktu tertentu, yang dimiliki oleh suatu objek.
2		<i>Initial Pseudo State</i>	Bagaimana objek diinisialisasi atau dimulai.
3		<i>Final State</i>	Bagaimana objek dibuat dan dihapus
4		<i>Transition</i>	Suatu peristiwa yang mengubah status suatu objek dengan mengubah satu atau lebih nilai atributnya.
5		<i>Association</i>	Apa yang menghubungkan satu objek dengan objek lainnya.
6		<i>Node</i>	Objek fisik yang ada ketika aplikasi berjalan dan mencerminkan sumber daya komputasi.

Tabel 2.5 *Activity Diagram*

No.	GAMBAR	NAMA	KETERANGAN
1		<i>Actifity</i>	Menunjukkan interaksi antarmuka kelas yang saling berhubungan satu sama lain.
2		<i>Action</i>	Keadaan sistem yang mencerminkan pelaksanaan suatu tindakan.

3		<i>Initial Node</i>	Bagaimana objek diinisialisasi atau dimulai.
4		<i>Activity Final Node</i>	Bagaimana objek dibuat atau dihapus
5		<i>Fork Node</i>	Suatu aliran yang pada suatu titik menjadi beberapa aliran.

B. Kajian Hasil Penelitian Terdahulu

Dalam penelitian ini, mengacu pada penelitian terdahulu yang relevan dengan penelitian yang akan dilakukan. Berikut ini beberapa hasil penelitian yang berhubungan dengan penelitian yang dijadikan bahan acuan bagi penelitian yang akan dilakukan, antara lain :

Deva, A. R. S. (2020). “Penerapan Algoritma Binary Search Pada Aplikasi E-Order (Studi Kasus Paris Van Java Kota Bengkulu)”. Universitas Muhammadiyah Bengkulu. Tujuan dari penelitian ini adalah membuat aplikasi E-Order yang mampu mengelola data pesanan makanan, minuman, dan tambahan lainnya dengan sukses.

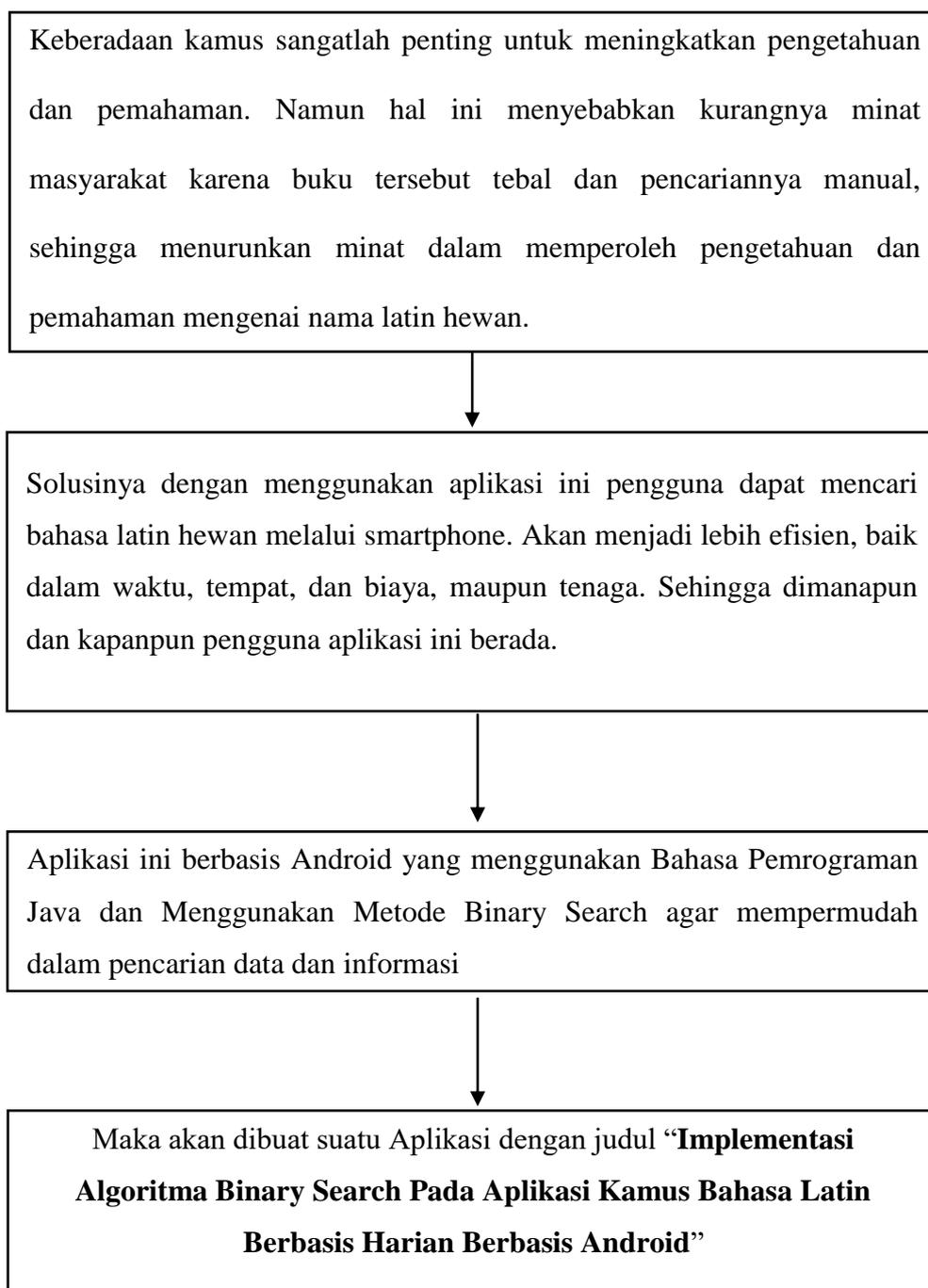
Rahmanto, Y. (2021). “Penerapan Algoritma Sequential Search pada Aplikasi Kamus Nama Latin Tumbuhan”. Universitas Tenokrat Indonesia. Tujuan penelitian ini dilakukan untuk merancang aplikasi nama latin tumbuhan sebagai sarana belajar dan alternatif untuk memperluas pengetahuan pengguna dalam mengenal nama-nama latin yang terdapat pada setiap tumbuhan.

Sulistio, B. (2019). “Aplikasi Kamus Nama Taliabu Berbasis Android Dengan Menggunakan Metode Binary Search”. Universitas Khairun. Tujuan penelitian ini

yaitu mempermudah pengguna arti terjemahan kosakata dalam nama Indonesia ke nama Taliabu maupun sebaliknya. Aplikasi ini juga disertai dengan contoh kalimat dalam Nama Taliabu.

C. Kerangka Berpikir

Untuk Memahami alur penelitian diatas, diuraikan ke dalam kerangka berpikir yang akan disajikan dalam bentuk diagram ini:



BAB III

METODE PENELITIAN

A. Jenis Penelitian

Jenis penelitian ini merupakan jenis penelitian deskriptif kualitatif, di mana data yang terkumpul berupa teks, gambar, bukan data angka. Penelitian kualitatif melibatkan proses pengumpulan data deskriptif berupa wawancara atau pengamatan perilaku. Penelitian deskriptif adalah jenis penelitian yang bertujuan untuk menggambarkan atau menjelaskan fenomena yang ada, termasuk fenomena alami maupun hasil karya manusia.

B. Lokasi dan Waktu

Penelitian ini akan dilakukan di perpustakaan umum kota Parepare, Ujung Sabbang, Kec. Ujung, Kota Parepare, Sulawesi Selatan 91114. Data akan dikumpulkan dari buku-buku yang terkait dengan penelitian ini. Adapun waktu penelitian belum ditentukan.

C. Alat dan Bahan

Dalam melakukan penelitian, penulis memerlukan alat dan bahan penelitian yang mendukung kegiatan penelitian tersebut antara lain:

1. Perangkat keras (*Hardware*)

Adapun spesifikasi perangkat keras yang digunakan dalam penyusunan proposal ini :

- a. Laptop Acer Aspire A515-44

- b. *Processor* Amd Ryzen 5 4500U
 - c. RAM 8,00 GB
 - d. HDD 512 GB
2. Perangkat lunak (Software)

Perangkat lunak (*software*) yang digunakan untuk membangun aplikasi ini dapat dilihat pada spesifikasi sebagai berikut:

- a. Windows 10
- b. Android Studio
- c. Java
- d. MySQL
- e. Ld Player

D. Teknik Pengumpulan Data

1. Studi pustaka

Metode studi pustaka adalah teknik pengumpulan data yang melibatkan pencarian informasi dari berbagai sumber literatur seperti buku, internet, artikel, jurnal, dan sumber lain yang relevan dengan penelitian.

2. Observasi

Observasi merupakan teknik pengumpulan data yang melibatkan pengamatan langsung di lapangan atau terhadap suatu proses.

3. Dokumentasi

Dokumentasi adalah teknik pengumpulan data yang melibatkan pengumpulan informasi dalam bentuk dokumen atau catatan tertulis.

E. Metode Pengujian

Dalam penelitian ini, digunakan 3 (tiga) metode dalam pengujian yaitu *blackbox testing*, *whitebox testing*, dan pengujian algoritma :

1. Pengujian algoritma

Pengujian algoritma bertujuan untuk memastikan bahwa algoritma tersebut berfungsi seperti yang diharapkan dalam berbagai situasi dan kondisi. Proses pengujian ini melibatkan pembuatan kasus uji, eksekusi algoritma pada kasus uji, dan analisis hasil kinerja algoritma.

2. *Blackbox testing*

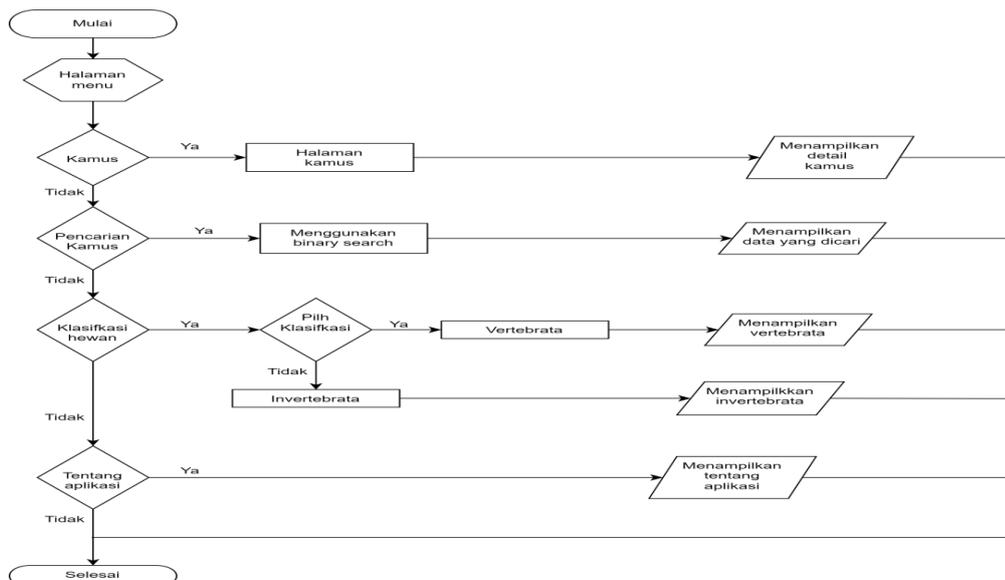
Blackbox testing berpusat pada fungsi dari program yang sedang diuji. Dalam *Blackbox testing*, program dijalankan dan kemudian hasilnya diamati untuk menilai apakah program tersebut berhasil atau tidak. Pendekatan *Blackbox testing* menggunakan teknik *equivalence partitions*, yang melibatkan pengujian berdasarkan input yang diberikan ke setiap fitur program. Setiap fitur diuji melalui klasifikasi dan pengelompokan berdasarkan fungsi yang dijalankannya.

3. *Whitebox testing*

Whitebox testing digunakan untuk mengetahui apakah struktur pada aplikasi yang dibuat sudah sesuai dengan ketentuan. *Whitebox testing* fokus pada pemeriksaan detail perancangan perangkat lunak. Pendekatan ini melibatkan definisi semua alur yang ada dalam perangkat lunak, pembangunan kasus uji yang sesuai, dan eksekusi kasus uji tersebut untuk mengevaluasi hasilnya.

F. Diagram Alir

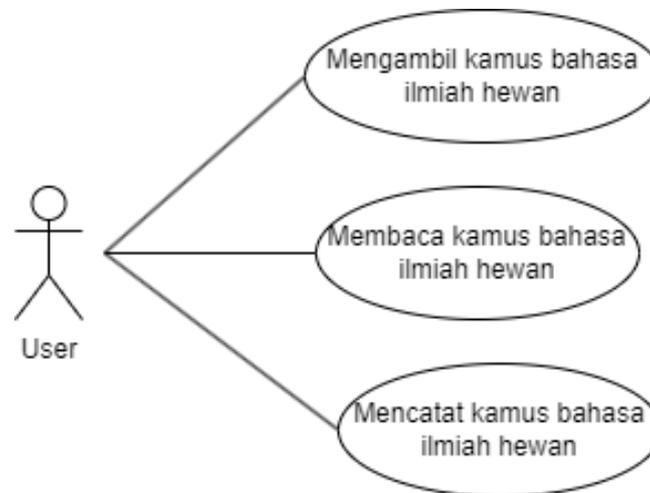
Tahap dari implementasi algoritma *binary search* pada aplikasi kamus nama latin hewan berbasis android dengan alur yang sederhana. Ketika pengguna membuka aplikasi maka pengguna akan langsung ditampilkan dengan halaman utama yang menampilkan 4 opsi pilihan. Di antaranya kamus latin hewan, pencarian kamus, klasifikasi hewan, dan tentang aplikasi. Dimana opsi kamus latin yang akan menampilkan kamus nama latin hewan dar A – Z. Kemudian opsi pencarian kamus untuk mempermudah pengguna dalam mencari suatu data pada kamus secara tepat dengan implementasi algoritma *binary search*. Selanjutnya terdapat opsi klasifikasi hewan yang terbagi dari hewan vertebrata dan invertebrata dan terakhir menjelaskan tentang aplikasi. Dengan demikian, aplikasi ini memberikan pengalaman pengguna yang mudah dalam mengakses dan mengelola informasi terkait seputar kamus latin hewan.



Gambar 3.1 Flowchart

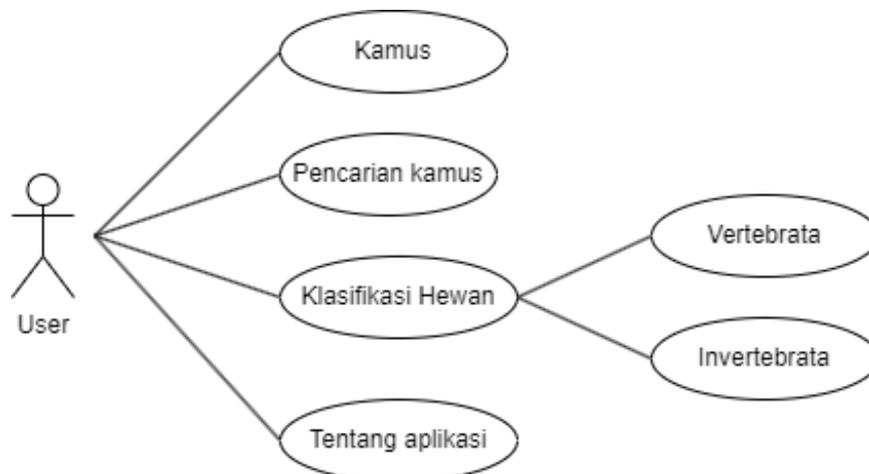
G. Desain Sistem

1. Desain sistem yang berjalan



Gambar 3.2 Desain Sistem Yang Berjalan

2. Desain sistem yang diusulkan



Gambar 3.3 Desain Sistem Yang Diusulkan

BAB IV

HASIL DAN PEMNAMAN

A. Analisis Aliran Data UML

Analisis aliran data menggunakan Unified Modeling Language (UML) adalah langkah penting dalam merancang sistem informasi yang kompleks. UML merupakan bahasa pemodelan yang digunakan untuk memvisualisasikan, menentukan, membangun, dan mendokumentasikan artefak dari sistem perangkat lunak. Salah satu aspek penting dalam UML adalah kemampuan untuk memetakan aliran data di dalam sistem, yang memungkinkan pengembang untuk memahami bagaimana informasi bergerak dan diolah di berbagai bagian sistem.

Aliran data dalam konteks UML sering diwakili melalui berbagai diagram, seperti Diagram Aktivitas (Activity Diagram) dan Diagram Urutan (Sequence Diagram). Diagram Aktivitas digunakan untuk memodelkan aliran kontrol atau aktivitas dalam sebuah proses bisnis atau sistem, termasuk bagaimana data berpindah dari satu aktivitas ke aktivitas lainnya. Misalnya, dalam proses pemesanan barang, data pesanan mungkin berpindah dari aktivitas input pesanan ke aktivitas pengecekan stok, lalu ke aktivitas konfirmasi pesanan, dan seterusnya.

Diagram Urutan, di sisi lain, fokus pada interaksi antara objek atau komponen dalam sistem, menunjukkan urutan pesan yang dikirimkan di antara mereka. Dalam analisis aliran data, diagram ini membantu mengidentifikasi jalur data

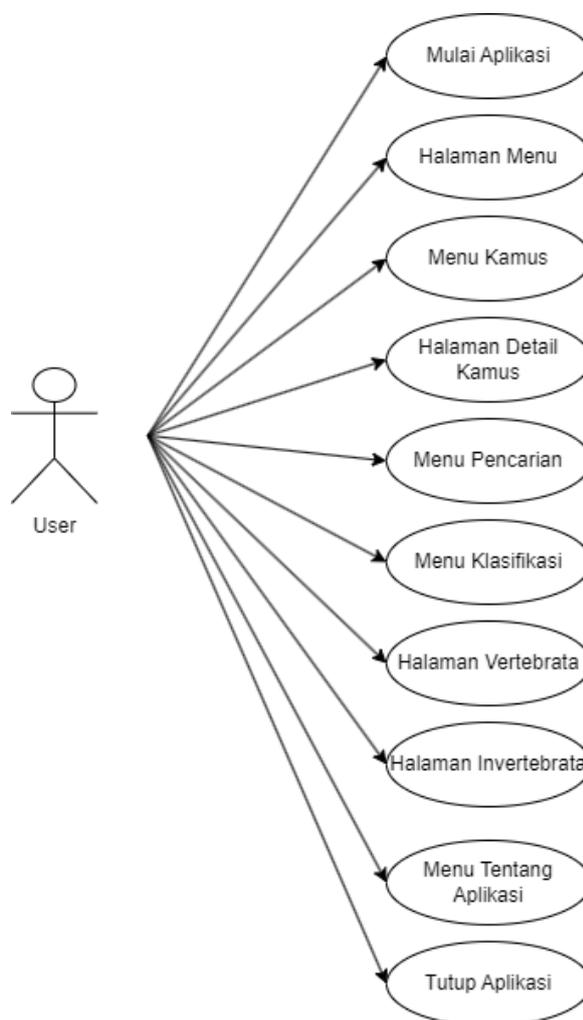
antara objek-objek sistem dan menunjukkan bagaimana data tersebut diproses di berbagai tahapan interaksi.

Menggunakan UML untuk analisis aliran data memberikan keuntungan dalam hal visualisasi yang jelas, memfasilitasi komunikasi antar pemangku kepentingan, dan membantu dalam identifikasi potensi masalah atau ketidakefisienan dalam desain sistem. Dengan demikian, pengembang dapat merancang sistem yang lebih efisien, terstruktur, dan mudah dipelihara.

Proses analisis ini tidak hanya memastikan bahwa semua data yang dibutuhkan dapat diakses oleh setiap komponen yang memerlukannya, tetapi juga membantu dalam mengidentifikasi kemungkinan redundansi data atau jalur aliran data yang tidak diperlukan, sehingga sistem dapat dioptimalkan untuk kinerja yang lebih baik.

1. *Use Case Diagram*

Use case diagram merupakan gambaran dari interaksi antara aktor dengan sistem pada suatu rancangan sistem.



Gambar 4.1 *Use Case*

Tabel 4.1 *Use case scenario*

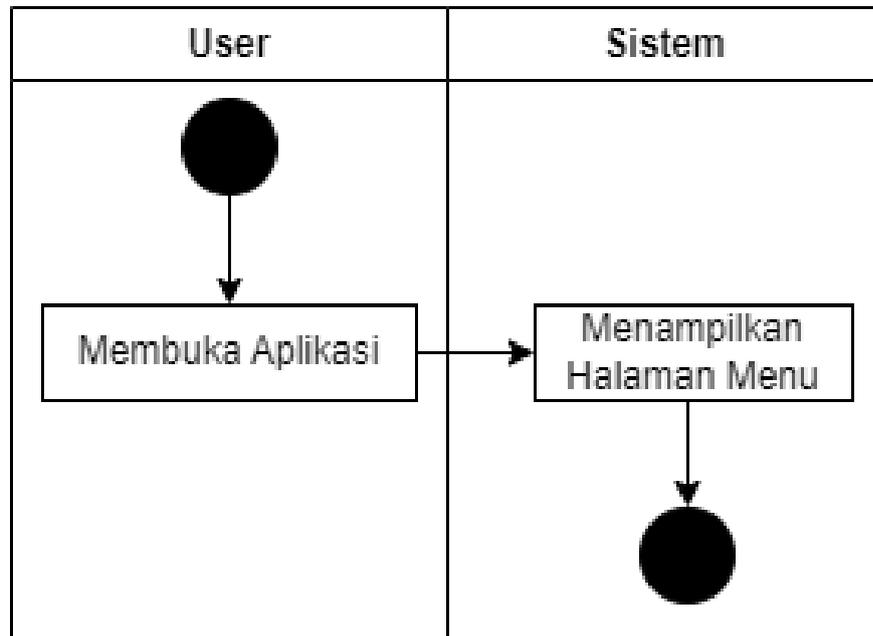
Nama Use Case	Deskripsi
Halaman Menu	Use case ini menjelaskan proses di mana pengguna mengakses menupage dan bertujuan untuk menyambut pengguna, memberikan gambaran umum tentang isi aplikasi, dan menawarkan button ke bagian-bagian utama aplikasi.
Menu Kamus	Use case ini menjelaskan proses di mana pengguna mengakses daftar kamus dalam aplikasi. Tujuannya adalah untuk menampilkan ringkasan tentang berbagai kamus kepada pengguna.
Halaman Detail Kamus	Use case ini menjelaskan proses di mana pengguna melihat informasi detail tentang

	kamus yang dipilih dari daftar.
Menu Pencarian	Use case ini menjelaskan proses di mana pengguna mencari data kamus yang ingin dicari pada aplikasi
Menu Klasifikasi	Use case ini menjelaskan proses di mana pengguna mengakses daftar klasifikasi kamus hewan dalam aplikasi. Tujuannya adalah untuk mengelompokkan klasifikasi hewan vertebrata dan invertebrata.
Halaman Vertebrata	Use case ini menjelaskan proses di mana pengguna mengakses daftar hewan vertebrata dalam aplikasi. Tujuannya adalah untuk menampilkan ringkasan tentang hewan vertebrata kepada pengguna
Halaman Invertebrata	Use case ini menjelaskan proses di mana pengguna mengakses daftar hewan invertebrata dalam aplikasi. Tujuannya adalah untuk menampilkan ringkasan tentang hewan invertebrata kepada pengguna
Menu Tentang Aplikasi	Use case ini menjelaskan proses di mana pengguna mengakses tentang aplikasi. Tujuannya adalah untuk menampilkan ringkasan info pembuatan aplikasi
Tutup Aplikasi	Use case ini menjelaskan fitur untuk menutup aplikasi yang telah di akses.

2. Activity Diagram

Activity diagram yaitu diagram yang dapat memodelkan proses-proses yang terjadi pada sebuah sistem.

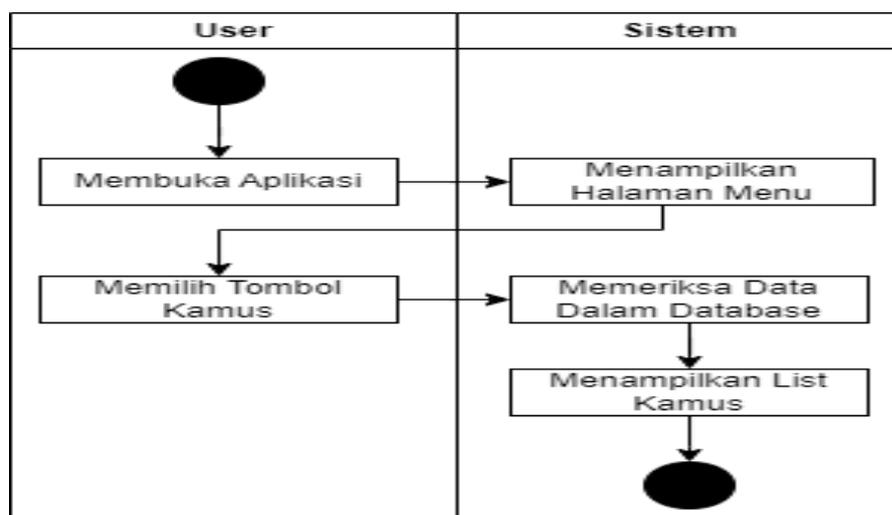
a. Activity diagram Halaman Menu



Gambar 4.2 Activity diagram Halaman Menu

Pada Gambar 4.2 menjelaskan proses menampilkan halaman menu pada aplikasi kamus nama latin, yang dimana pada halaman ini menampilkan menu-menu yang lain.

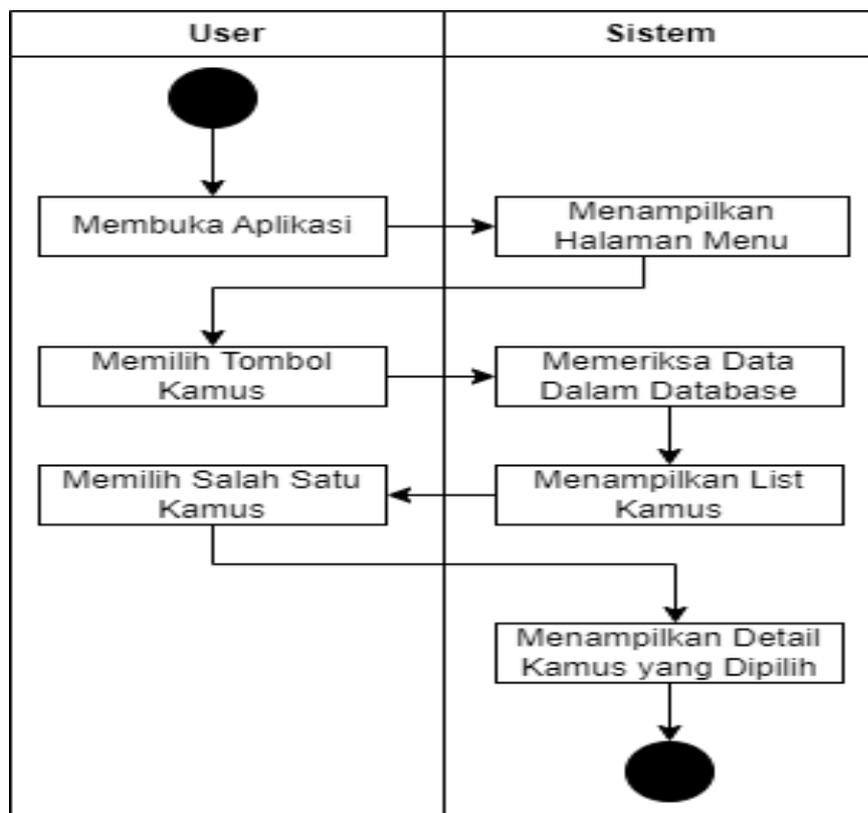
b. Activity diagram Menu Kamus



Gambar 4.3 Activity diagram Menu Kamus

Pada Gambar 4.3 menjelaskan proses menampilkan halaman daftar kamus pada aplikasi kamus latin hewan, yang dimana ketika user menekan tombol kamus maka sistem memeriksa apakah ada data dalam database. Setelah itu menampilkan kamus latin hewan dalam halaman daftar list kamus.

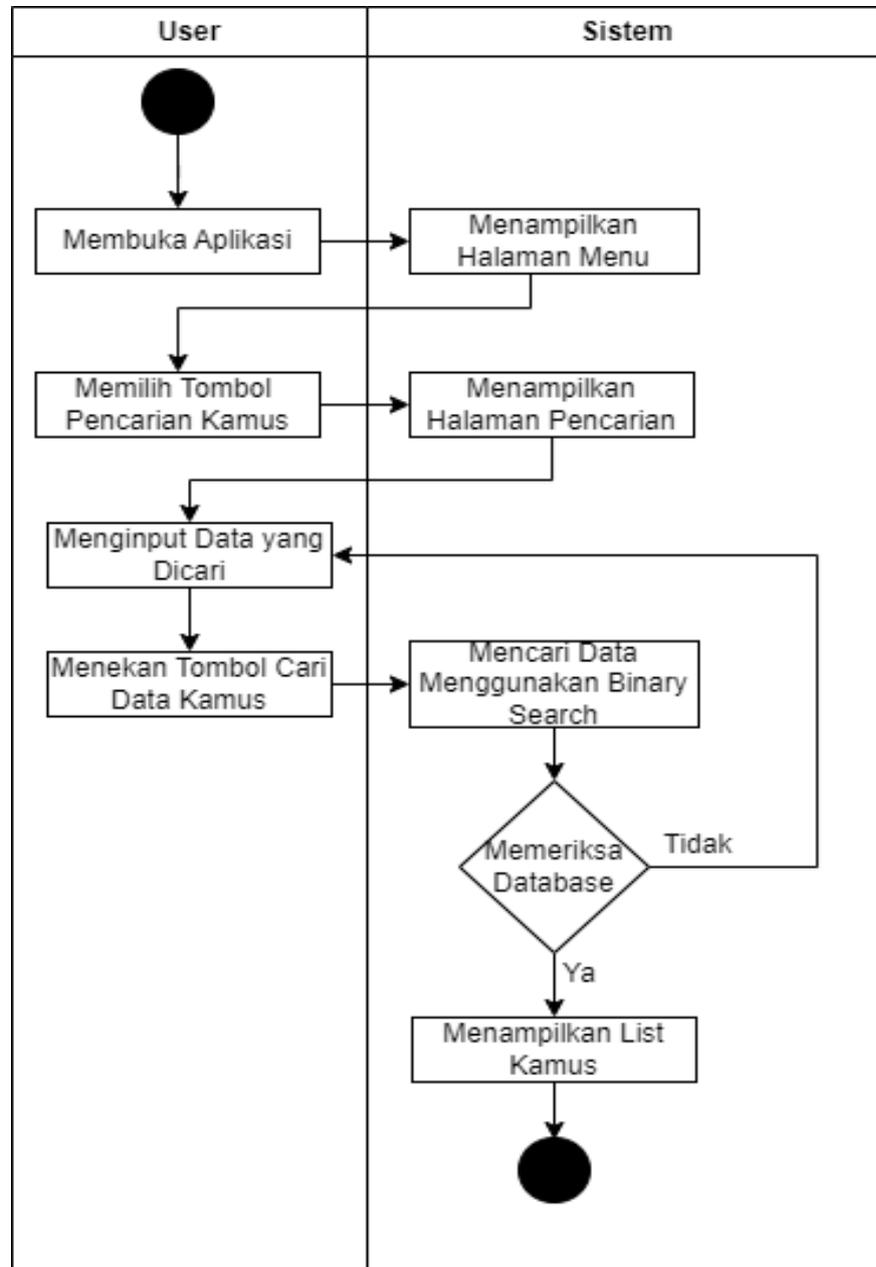
c. *Activity diagram detail Kamus*



Gambar 4.4 *Activity diagram detail Kamus*

Pada Gambar 4.4 menjelaskan proses menampilkan halaman detail kamus pada aplikasi kamus nama latin hewan. Dimana saat user berada di menu kamus lalu memilih salah satu daftar kamus, maka sistem akan menelusuri database detail kamus yang dipilih oleh user. Halaman detail kamus ini berisi informasi tentang nama hewan, nama latin hewan, klasifikasi hewan, deskripsi hewan serta menampilkan gambar hewan.

d. *Activity diagram Menu Pencarian*

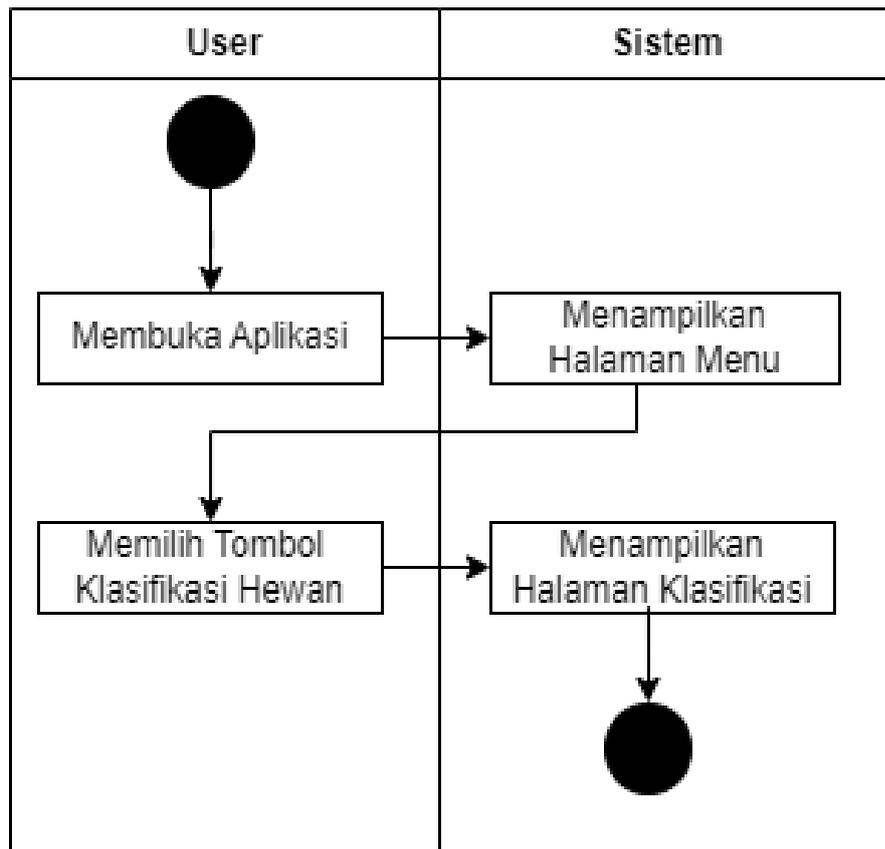


Gambar 4.5 *Activity diagram Menu Pencarian*

Pada Gambar 4.5 menjelaskan proses menampilkan halaman pencarian kamus pada aplikasi kamus nama latin hewan. Dimana saat user berada di menu pencarian lalu menuliskan salah satu data kamus, maka sistem akan menampilkan data kamus yang dicari oleh user dengan menggunakan binary search. Setelah itu

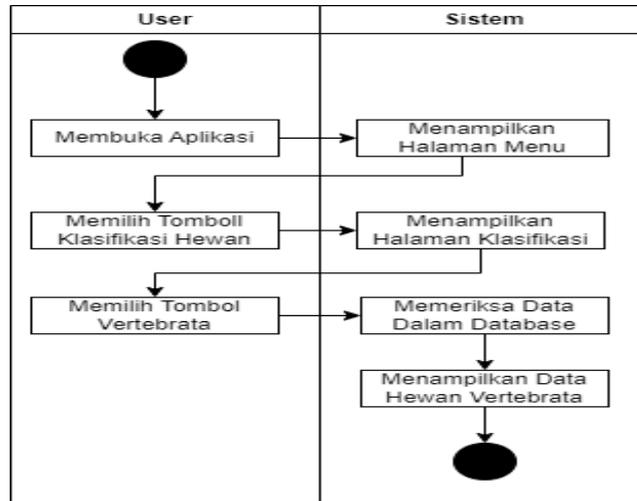
sistem memeriksa data jika data ditemukan maka akan menampilkan data kamus yang dicari, tapi jika data tidak di temukan maka sistem akan mengembalikan user untuk menginput data yang ingin dicari.

e. *Activity diagram Menu Klasifikasi*



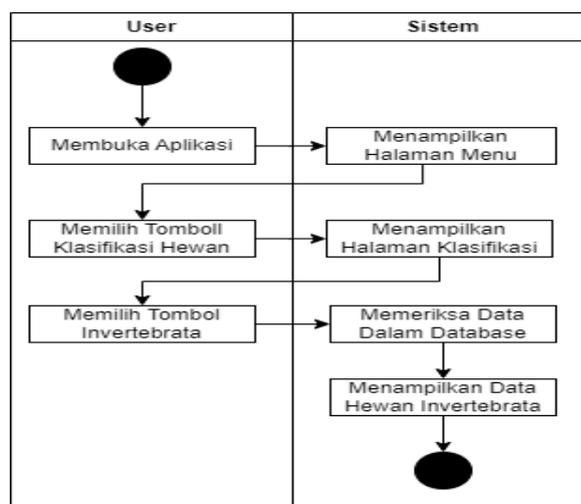
Gambar 4.6 *Activity diagram Klasifikasi Hewan*

Pada Gambar 4.6 menjelaskan proses menampilkan halaman klasifikasi hewan pada aplikasi kamus latin hewan, yang dimana ketika user menekan tombol klasifikasi maka sistem menampilkan klasifikasi hewan pada kamus latin hewan.

f. *Activity diagram Halaman Vertebrata*

Gambar 4.7 *Activity diagram Halaman Vertebrata*

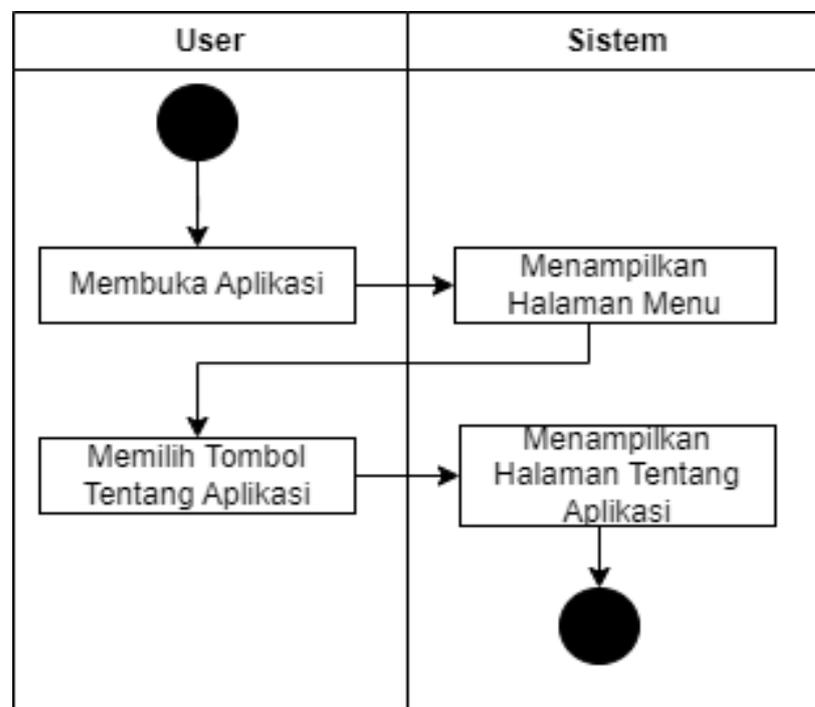
Pada Gambar 4.7 menjelaskan proses menampilkan halaman vertebrata pada aplikasi kamus latin hewan, yang dimana ketika user menekan tombol vertebrata maka sistem memeriksa apakah ada data dalam database. Setelah itu menampilkan kamus latin hewan yang berklasifikasi vertebrata dalam halaman kamus.

g. *Activity diagram Halaman Invertebrata*

Gambar 4.8 *Activity diagram Halaman Invertebrata*

Pada Gambar 4.8 menjelaskan proses menampilkan halaman invertebrata pada aplikasi kamus latin hewan, yang dimana ketika user menekan tombol invertebrata maka sistem memeriksa apakah ada data dalam database. Setelah itu menampilkan kamus latin hewan yang berklasifikasi invertebrata dalam halaman kamus.

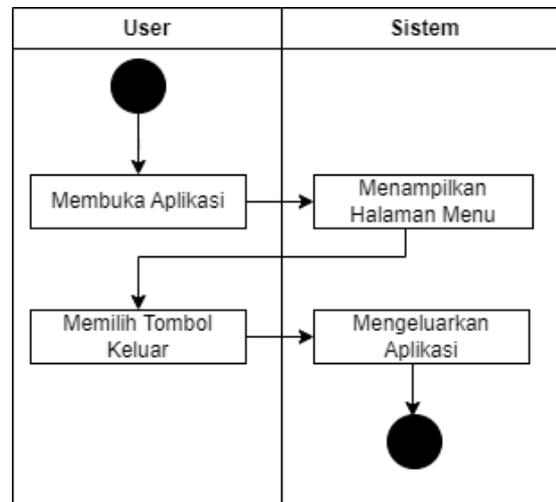
h. *Activity diagram Menu Tentang Aplikasi*



Gambar 4.9 *Activity diagram Menu Tentang Aplikasi*

Pada Gambar 4.9 menjelaskan proses menampilkan menu tentang aplikasi pada aplikasi kamus latin hewan, yang dimana ketika user menekan tombol tentang aplikasi maka sistem menampilkan halaman tentang aplikasi kamus latin hewan .

i. *Activity diagram Menu Keluar*



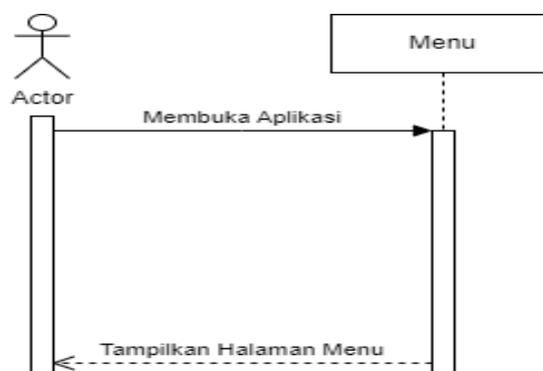
Gambar 4.10 *Activity diagram Menu Keluar*

Pada Gambar 4.10 menjelaskan proses menu keluar pada aplikasi kamus latin hewan, yang dimana ketika user menekan tombol keluar maka sistem akan mengeluarkan aplikasi kamus latin hewan .

3. *Sequence Diagram*

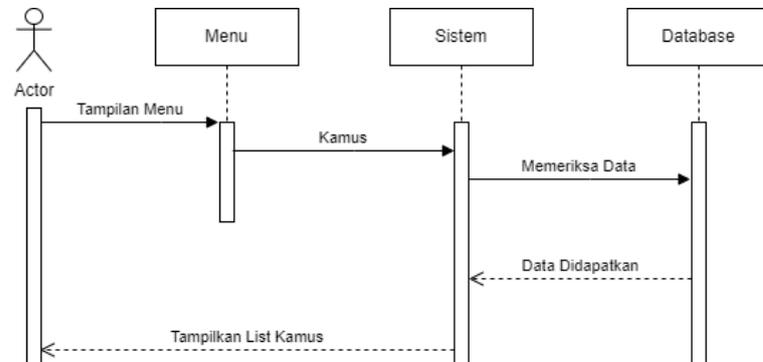
Sequence diagram adalah sebuah diagram yang digunakan untuk menjelaskan dan menampilkan interaksi antar objek-objek dalam sebuah sistem secara terperinci.

a. *Sequence diagram home*



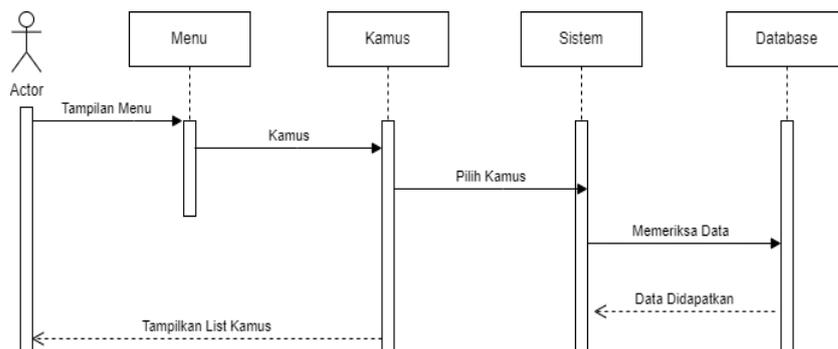
Gambar 4.11 *Sequence diagram Halaman Menu*

b. *Sequence diagram Menu Kamus*



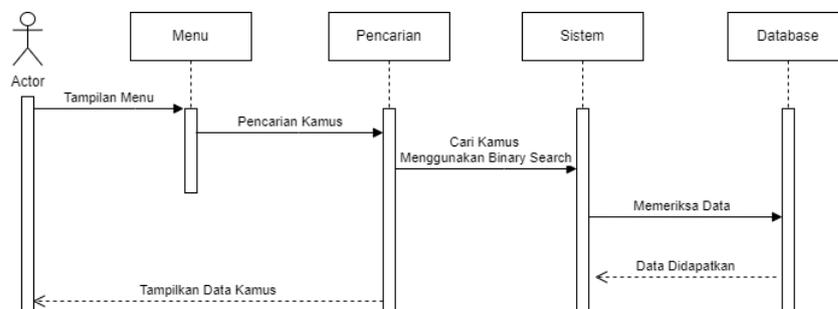
Gambar 4.12 *Sequence diagram Menu Kamus*

c. *Sequence diagram Detail Kamus*



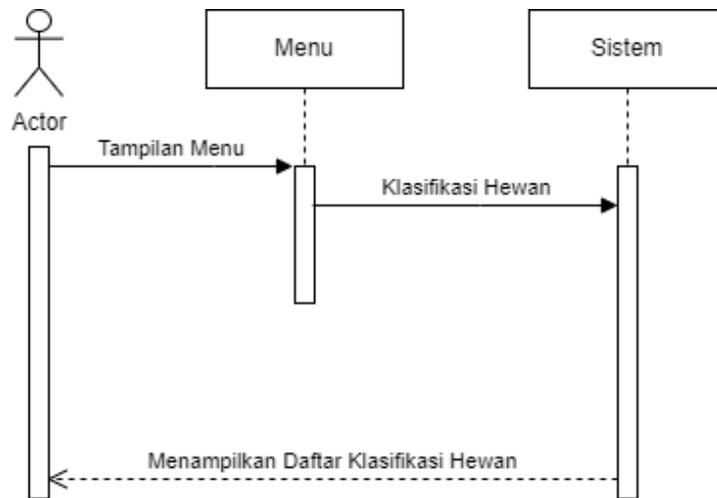
Gambar 4.13 *Sequence diagram Detail Kamus*

d. *Sequence diagram Menu Pencarian*



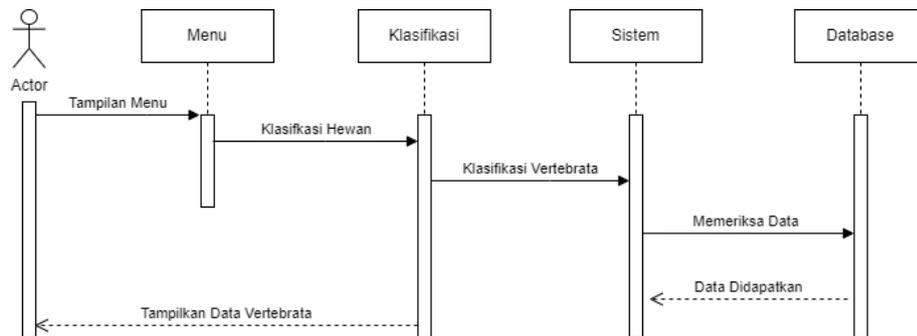
Gambar 4.14 *Sequence diagram Menu Pencarian*

e. *Sequence diagram Menu Klasifikasi*



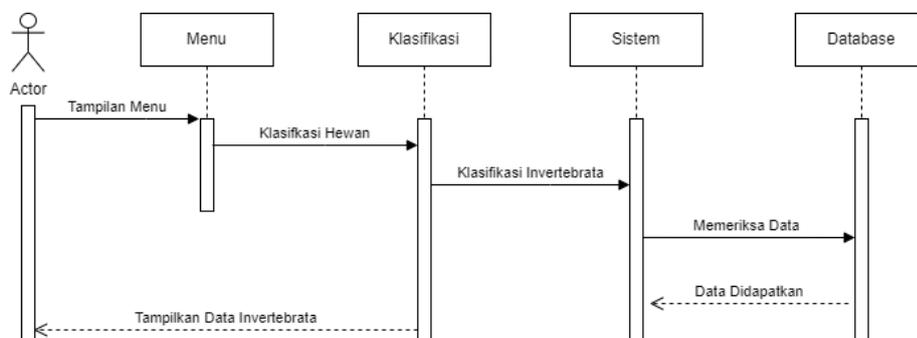
Gambar 4.15 *Sequence diagram Menu Klasifikasi*

f. *Sequence diagram Halaman Vertebrata*



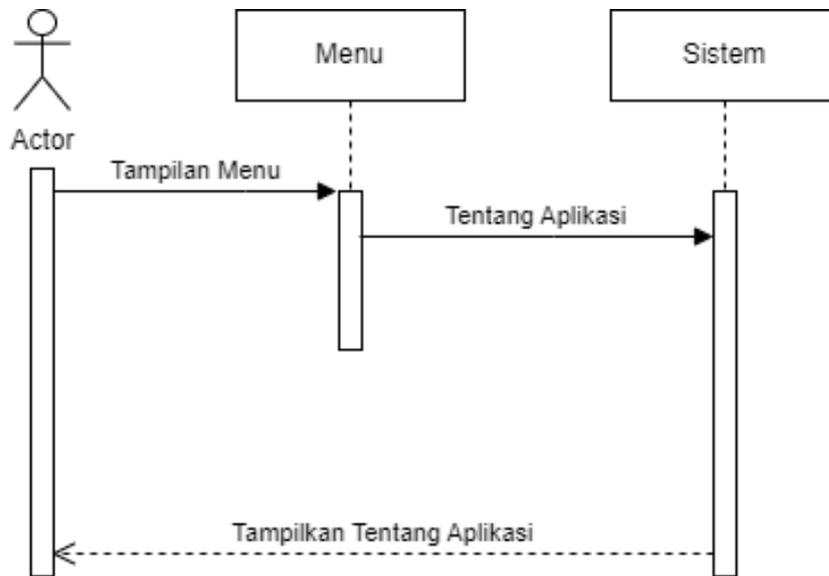
Gambar 4.16 *Sequence diagram Halaman Vertebrata*

g. *Sequence diagram Halaman Invertebrata*



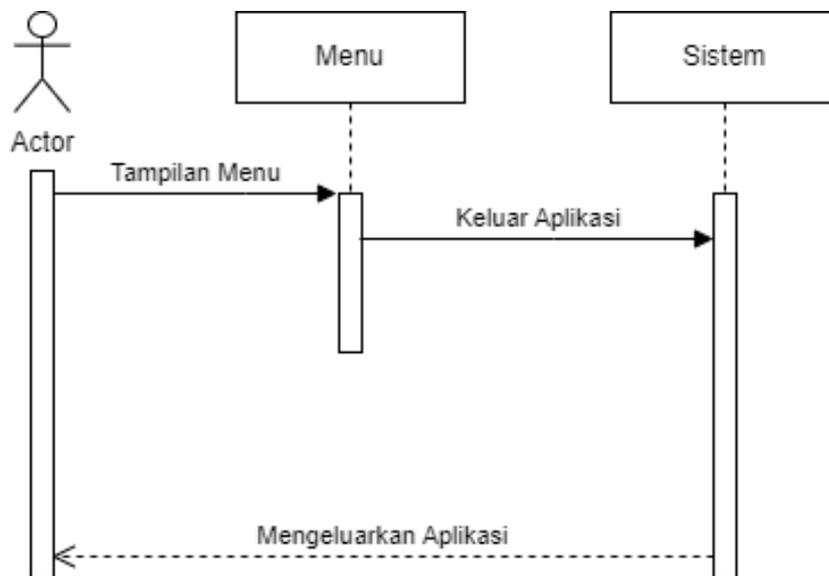
Gambar 4.17 *Sequence diagram Halaman Invertebrata*

h. *Sequence diagram Menu Tentang Aplikasi*



Gambar 4.18 *Sequence diagram Menu Tentang Aplikasi*

i. *Sequence diagram Menu Keluar*



Gambar 4.19 *Sequence diagram Menu Keluar*

B. Detail Sistem

1. Tampilan Halaman Menu

Merupakan halaman awal saat user membuka aplikasi yang berisi berbagai menu-menu pada aplikasi kamus nama latin hewan. Halaman ini dibuat dengan menggunakan Java.



Gambar 4.20 Tampilan Halaman *Menu*

2. Tampilan Halaman Kamus

Merupakan halaman daftar kamus yang tampil saat user mengklik tombol kamus, halaman ini berisi daftar kamus yang ada dalam database yang dimulai dari abjad A-Z.



Gambar 4.21 Tampilan Halaman *Kamus*

3. Tampilan Halaman Detail

Merupakan halaman detail kamus yang tampil saat user mengklik salah satu daftar kamus, halaman detail kamus ini berisi informasi tentang nama hewan, nama latin hewan, klasifikasi hewan, deskripsi hewan serta menampilkan gambar hewan.



Gambar 4.22 Tampilan Halaman Detail Kamus

Berikut merupakan codingan java dan xml untuk mengelolah tampilan dari halaman :

KamusDetailActivity Java

// Identifikasi Variabel

```
public class KamusDetailActivity extends AppCompatActivity {
```

```
    ImageView imgHolder;
```

// Identifikasi Komponen Form

```
imgHolder = (ImageView) findViewById(R.id.imgHolder);
```

```
// Menghubungkan di server API

mApiInterface = ApiClient.getClient().create(ApiInterface.class);

activity_kamus_detail XML

<ImageView

    android:id="@+id/imgHolder"

    android:layout_width="match_parent"

    android:layout_height="350dp"

    android:layout_marginBottom="4dp"

    android:scaleType="centerCrop"

    app:layout_constraintEnd_toStartOf="@+id/guideline"

    app:layout_constraintTop_toTopOf="parent"

    app:layout_constraintEnd_toEndOf="parent"

    app:layout_constraintBottom_toTopOf="@id/edt_name"

    tools:src="@color/colorAccent" />
```

4. Tampilan Menu Pencarian

Merupakan halaman pencarian kamus yang tampil saat user mengklik tombol pencarian kamus, halaman pencarian kamus untuk mempermudah pencarian data kamus yang ingin dicari dengan cara menginput nama hewan yang ingini dicari.



Gambar 4.23 Tampilan Menu Pencarian

Berikut merupakan codingan binary search untuk mengelolah halaman pencarian :

```
// Penerapan binary search

private List<Kamus> binarySearch(String query) {

    List<Kamus> result = new ArrayList<>();

    int low = 0;

    int high = mHerosListFull.size() - 1;

    while (low <= high) {
```

```
int mid = (low + high) / 2;

Kamus midItem = mHerosListFull.get(mid);

int comparison =
midItem.getName().compareToIgnoreCase(query);

if (comparison == 0) {
    result.add(midItem);

    // Add items around the found item

    int left = mid - 1;

    while (left >= 0 &&
mHerosListFull.get(left).getName().equalsIgnoreCase(query)) {

        result.add(mHerosListFull.get(left));

        left--;

    }

    int right = mid + 1;

    while (right < mHerosListFull.size() &&
mHerosListFull.get(right).getName().equalsIgnoreCase(query)) {

        result.add(mHerosListFull.get(right));

        right++;

    }

    break;

} else if (comparison < 0) {
```

```
        low = mid + 1;
    } else {
        high = mid - 1;
    }
}
return result;
}
```

5. Tampilan Menu Klasifikasi

Merupakan halaman menu klasifikasi hewan yang tampil saat user mengklik tombol klasifikasi hewan, menu klasifikasi hewan ini berisi tentang hewan vertebrata dan hewan invertebrata.



Gambar 4.24 Tampilan Menu Klasifikasi

6. Tampilan Halaman Vertebrata

Merupakan halaman hewan vertebrata yang tampil saat user mengklik tombol vertebrata, halaman vertebrata ini berisi tentang informasi hewan vertebrata.



Gambar 4 25 Tampilan Halaman Vertebrata

7. Tampilan Halaman Invertebrata

Merupakan halaman hewan invertebrata yang tampil saat user mengklik tombol invertebrata, halaman invertebrata ini berisi tentang informasi hewan invertebrata.



Gambar 4.26 Tampilan Halaman Invertebrata

8. Tampilan Halaman Tentang Aplikasi

Merupakan halaman menu tentang aplikasi yang tampil saat user mengklik tombol tentang aplikasi, halaman ini berisi tentang informasi perancang aplikasi dan tentang aplikasi.



Gambar 4.27 Tampilan Menu Tentang Aplikasi

C. Pengujian Sistem

Pengujian Sistem pada penelitian ini menggunakan metode pengujian algoritma, black box testing, dan white box testing. Berikut dibawah ini hasil pengujian menggunakan metode tersebut.

1. Pengujian Algoritma

a. Pengujian Perhitungan Algoritma Binary Search

Dalam perancangan sistem aplikasi kamus ilmoah hewan digunakan algoritma binary search untuk memudahkan pencarian data yang dibutuhkan. Berikut adalah pencocokan dan perhitungan dari algoritma Binary search

ketika ingin mencari kata pada sekumpulan data berikut :

- a. Pertama-tama sebelum melakukan perhitungan siapkan daftar nama latin hewan terlebih dahulu.
- b. Tentukan data yang ingin di cari yaitu GAJAH

Tabel 4.2 Data

1	Zebra
2	Angsa
3	Flamingo
4	Belut
5	Walet
6	Capung
7	Harimau
8	Landak
9	Gajah
10	Ikan Lele
11	Kucing
12	Ular Hijau
13	Macan Tutul
14	Singa
15	Jerapah
16	Nuri Merah
17	Rubah
18	Panda
19	Trenggeling
20	Elang Jawa

Setelah data tersedia maka fungsi Binary Search baru mulai bekerja. Berikut adalah cara algoritma Binary Search mencari kata GAJAH pada data tersebut.

Tabel 4.3 Proses Pencarian 1

1	2	3	4	5	12	13	14	15	16	...20
Awal					Tengah	Akhir				

Jumlah data ada (N)20, mula-mula dicari data tengah dengan rumus $(1+21) / 2 = 12$. Berarti data tengah adalah data ke-12 yaitu ULAR HIJAU. Data yang dicari yaitu GAJAH, karena kata GAJAH urutannya di bawah dari kata ULAR HIJAU, maka $GAJAH < ULAR HIJAU$ (data tengah). Jika yang dicari ternyata lebih kecil maka (-) dikarenakan pencarian belum diketemukan maka nilai tengah dikurang 1 kolom array akan didapatkan rumus $(12-1) / 2 = 5$. Berarti data tengah yang baru adalah kolom array ke-5 yaitu WALET. Data yang dicari dibandingkan dengan data tengah ternyata masih berbeda.

Tabel 4.4 Proses Pencarian 2

1	2	3	4	5	6	7	8	9	10	...12
Awal					Tengah	Akhir				

Dari pencarian kedua didapat kata yang dicari lebih besar dari data tengah, dikarenakan pencarian belum juga ditemukan maka nilai tengah ditambah 1 $(5+1) = 6$ akan didapatkan rumus $(6+12) / 2 = 9$. Berarti data

tengah yang baru adalah kolom array ke 9 yaitu GAJAH.

Tabel 4.5 Proses Pencarian 3

1	2	3	4	5	9.....	10	11	12	13	...20
Awal					Tengah	Akhir				

Data yang dicari dibandingkan dengan data tengah ternyata sama yaitu GAJAH. Jadi data telah ditemukan dan berada pada indeks ke-9.

b. Pengujian Kecepatan Algoritma Binary Search

Pada pengujian ini dilakukan untuk mengetahui kecepatan penacri binary search menggunakan microtime, sebagai berikut :

Tabel 4.6 Proses Pengujian Kecepatan *Binary Search*

NO	Nama Kamus	Kecpatan Binary Search
1.	Ayam	0,197 ms
2.	Beo	0,286 ms
3.	Cheetah	0,201 ms
4.	Domba Salju	0,218 ms
5.	Gajah	0,244 ms
6.	Serigala	0,275 ms
7.	Harimau	0,201 ms
8.	Udang	0,245 ms
9.	Walet	0,187 ms
10.	Zebra	0,259 ms
RATA-RATA KECEPATAN		0,2313 ms

Jadi rata-rata kecepatan pencarian binary search yaitu 0,2313ms

$$E (\text{edge}) = 17$$

$$N (\text{node}) = 14$$

$$P (\text{Predikat node}) = 4$$

Penyelesaian :

$$\begin{aligned} V(G) &= E - N + 2 \\ &= 17 - 14 + 2 \\ &= 5 \end{aligned}$$

$$\begin{aligned} \text{Predikat (P)} &= P + 1 \\ &= 4 + 1 \\ &= 5 \end{aligned}$$

b) Berdasarkan perhitungan *Cyclomatic Complexity* dari *flowgraph* di atas memiliki *Region* = 5

c) *Independent path* pada *flowgraph* tersebut yakni:

$$\text{Path 1} = 1 - 2 - 3 - 4 - 9 - 13 - 14$$

$$\text{Path 2} = 1 - 2 - 3 - 4 - 5 - 10 - 13 - 14$$

$$\text{Path 3} = 1 - 2 - 3 - 4 - 5 - 6 - 11 - 13 - 14$$

$$\text{Path 4} = 1 - 2 - 3 - 4 - 5 - 6 - 7 - 12 - 13 - 14$$

$$\text{Path 5} = 1 - 2 - 3 - 5 - 6 - 7 - 8 - 13 - 14$$

d) Grafik matriks

Tabel 4.7 Grafik Matriks

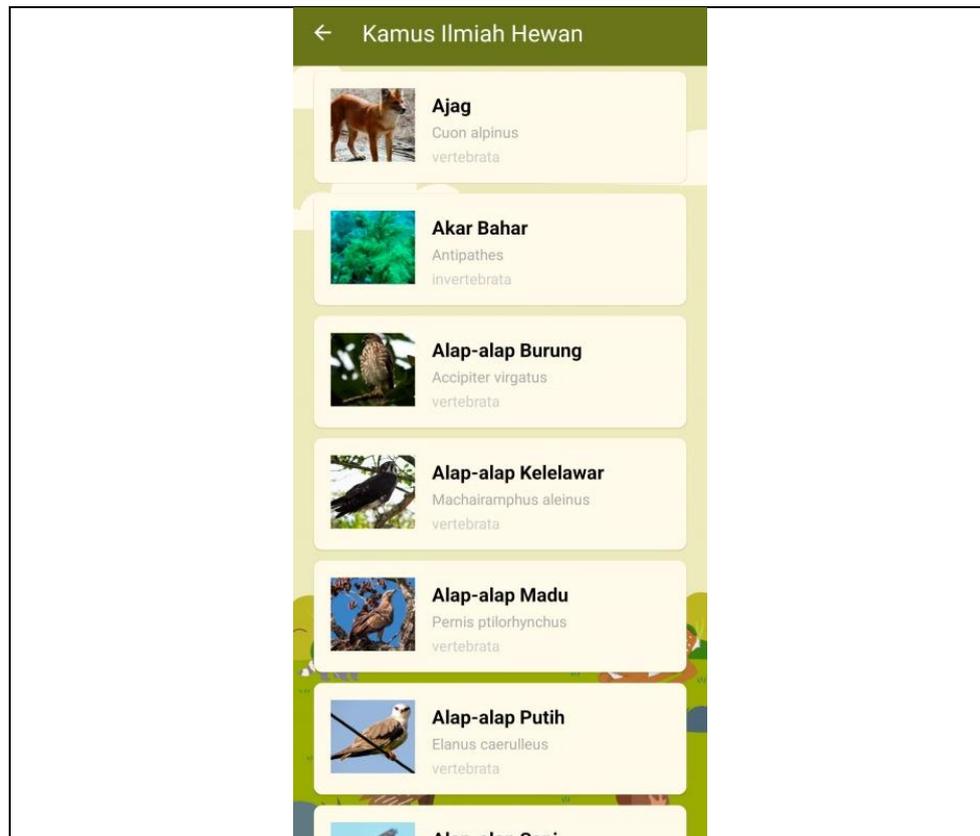
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	E-1
1		1													$1 - 1 = 0$
2			1												$1 - 1 = 0$



b. Pengujian *Black box Halaman Kamus*

Tabel 4.9 Pengujian *black box Halaman Kamus*

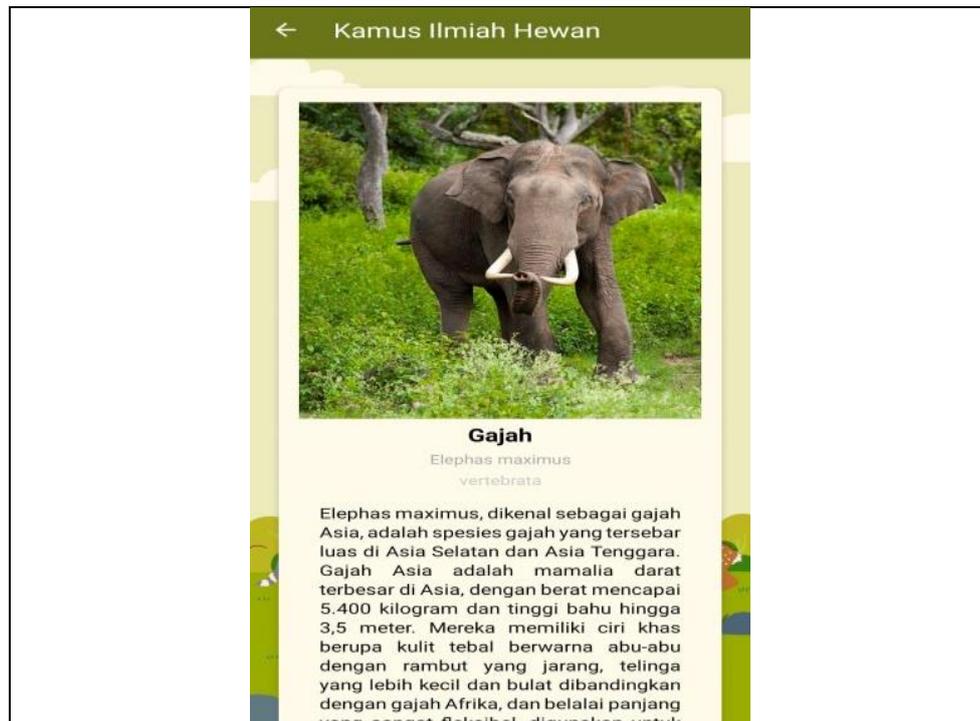
Tes Faktor	Hasil	Keterangan
User mengklik tombol kamus pada halaman <i>menu</i>	✓	Berhasil, ketika <i>user</i> mengklik tombol kamus maka tampil halaman <i>kamus</i> seperti pada gambar dibawah.
<i>Screenshot</i>		



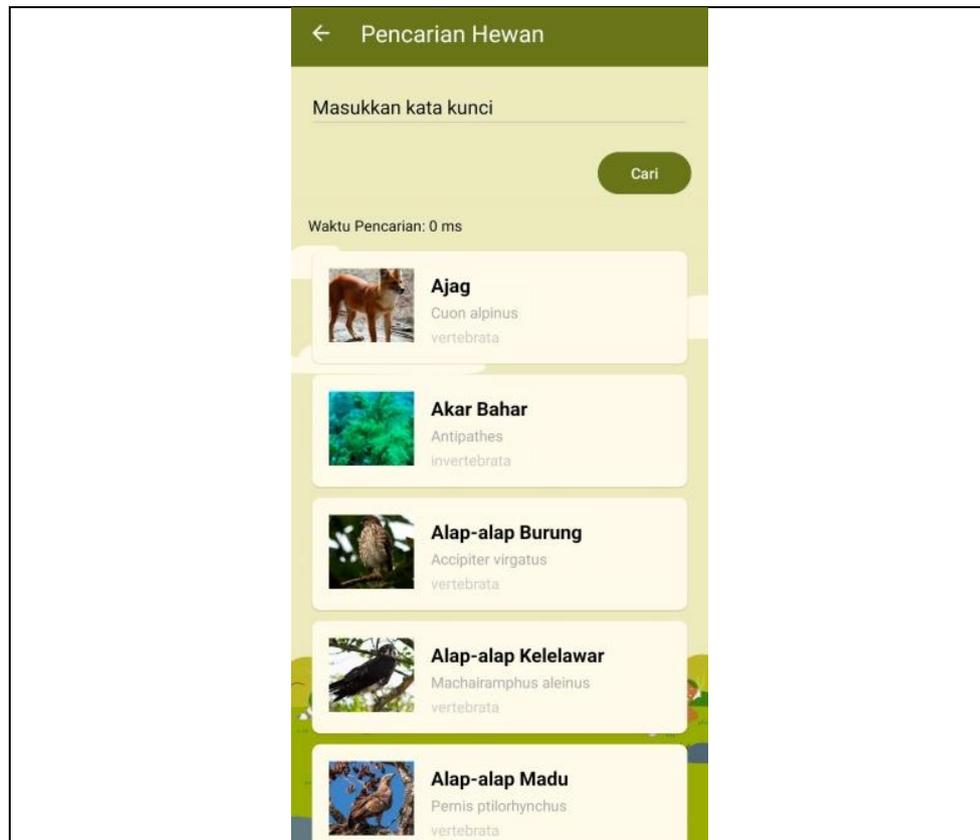
c. Pengujian *Black box* Halaman Detail

Tabel 4.10 Pengujian *black box* Halaman Detail

Tes Faktor	Hasil	Keterangan
User mengklik tombol lihat pada salah satu daftar wayang.	✓	Berhasil, ketika <i>user</i> mengklik tombol lihat maka tampil halaman detail wayang seperti pada gambar dibawah.
<i>Screenshot</i>		

d. Pengujian *black box* Menu Pencarian**Tabel 4.11** Pengujian *black box* Menu Pencarian

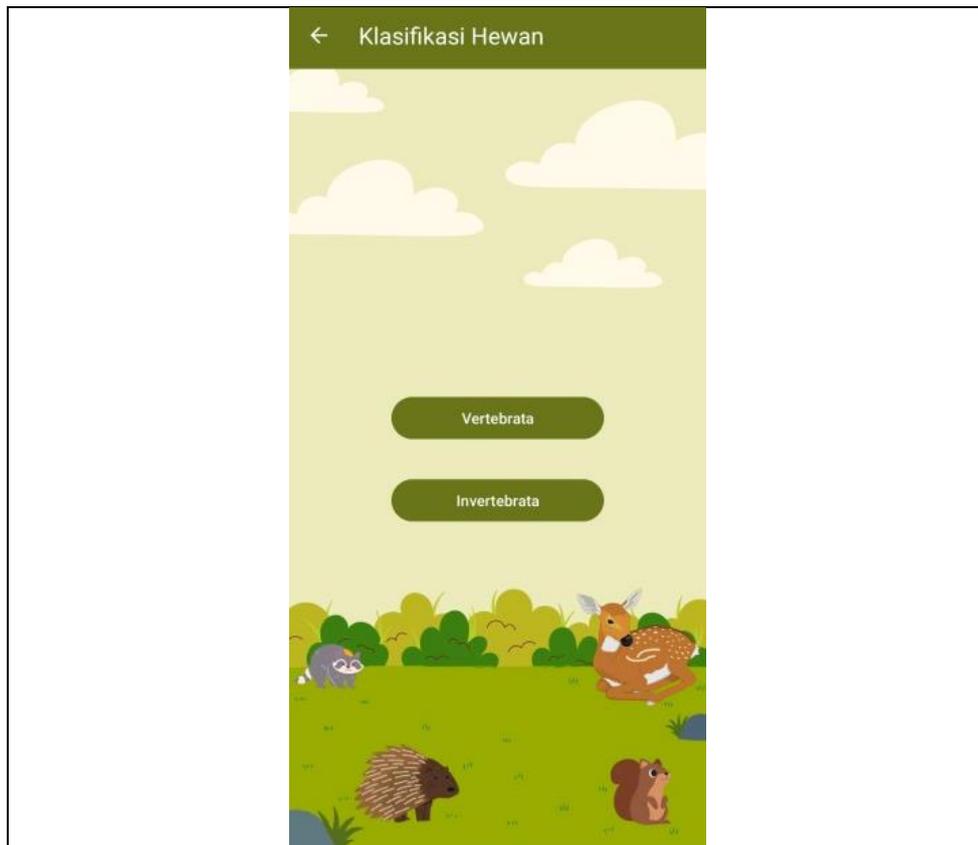
Tes Faktor	Hasil	Keterangan
User mengklik tombol Pencarian Kamus.	✓	Berhasil, ketika <i>user</i> mengklik tombol <i>cari kamus</i> maka akan muncul data yang ingin dicari dan jika data tidak ditemukan maka tampil “Data Tidak Ditemukan”
<i>Screenshot</i>		



e. Pengujian *black box Menu Spesifikasi*

Tabel 4.12 Pengujian *black box Menu Spesifikasi*

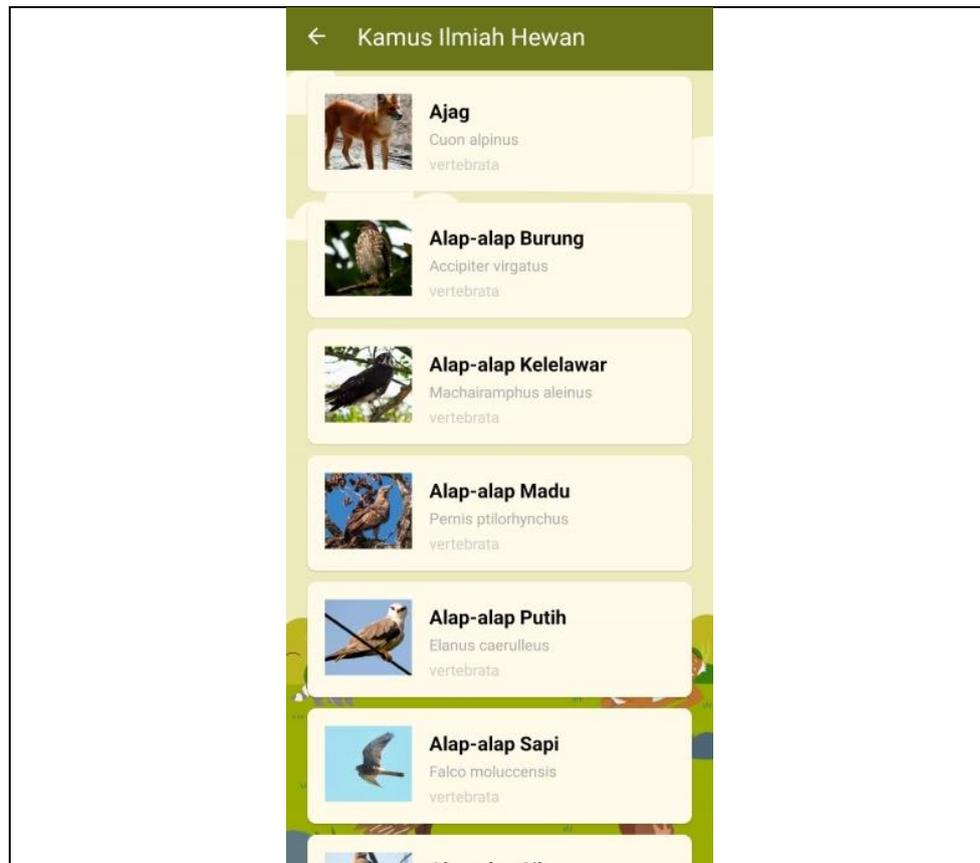
Tes Faktor	Hasil	Keterangan
User mengklik tombol Spesifikasi hewan	✓	Berhasil, ketika <i>user</i> mengklik tombol <i>spesifikasi</i> maka akan muncul vertebrata dan invertebrata
<i>Screenshot</i>		



f. Pengujian *black box* Halaman Vertebrata

Tabel 4.13 Pengujian *black box* Halaman Vertebrata

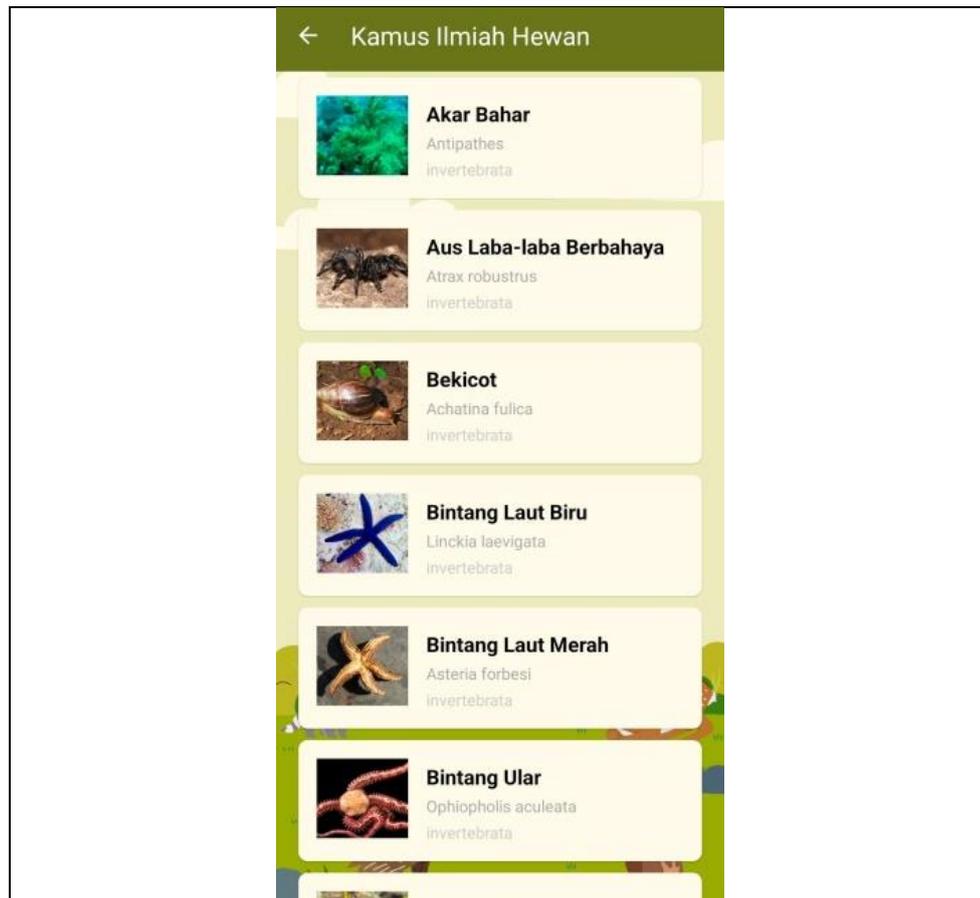
Tes Faktor	Hasil	Keterangan
User mengklik tombol vertebrata.	✓	Berhasil, ketika <i>user</i> mengklik tombol <i>vertebrata</i> maka akan muncul list data hewan vertebrata.
Screenshot		



g. Pengujian *black box* Halaman *Invertebrata*

Tabel 4.14 Pengujian *black box* Halaman *Invertebrata*

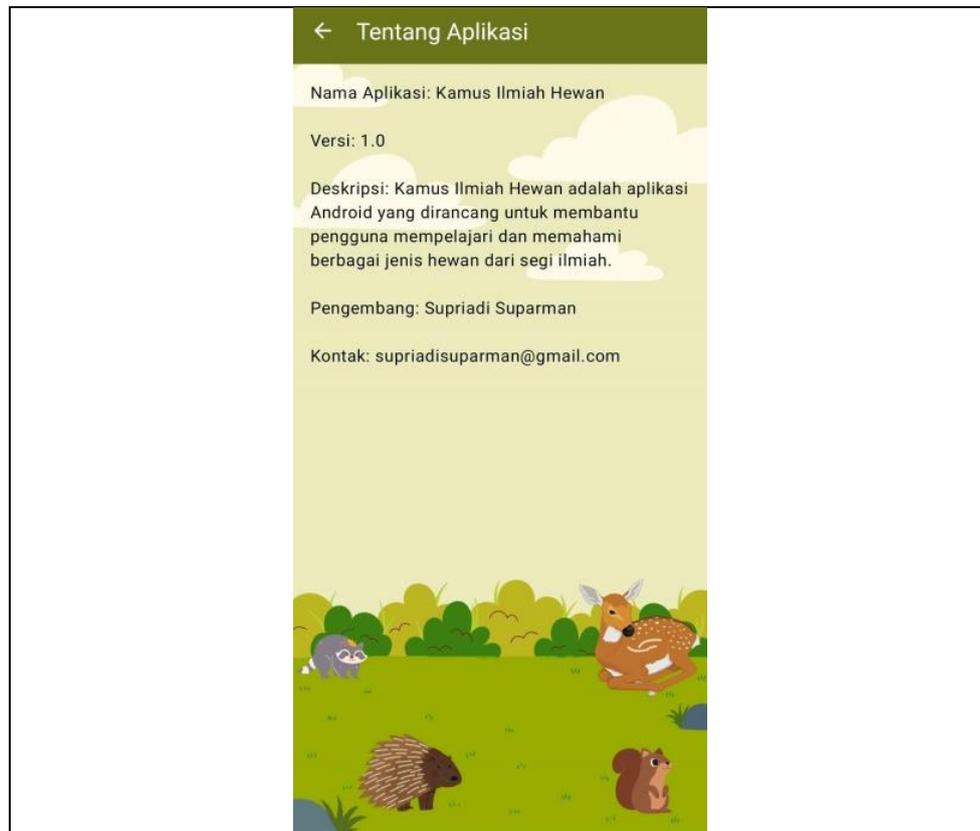
Tes Faktor	Hasil	Keterangan
User mengklik tombol <i>invertebrata</i> .	✓	Berhasil, ketika <i>user</i> mengklik tombol <i>invertebrata</i> maka akan muncul list data hewan <i>invertebrata</i> .
<i>Screenshot</i>		



h. Pengujian *black box* Menu Tentang Aplikasi

Tabel 4.15 Pengujian *black box* Menu Tentang Aplikasi

Tes Faktor	Hasil	Keterangan
User mengklik tombol tentang aplikasi.	✓	Berhasil, ketika <i>user</i> mengklik tombol <i>tentang aplikasi</i> maka akan muncul info tentang aplikasi.
<i>Screenshot</i>		



i. Pengujian *black box* Menu keluar

Tabel 4.16 Pengujian *black box* Menu Keluar

Tes Faktor	Hasil	Keterangan
User mengklik tombol keluar.	✓	Berhasil, ketika <i>user</i> mengklik tombol <i>keluar</i> maka aplikasi akan keluar.
Screenshot		
		

BAB V

KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan penelitian yang telah dilakukan maka dapat ditarik kesimpulan sebagai berikut, membuat aplikasi pencarian nama latin hewan berbasis android menggunakan bahasa pemrograman java di android studio untuk menerapkan algoritma binary search.

B. Saran

Pada penelitian ini penulis menyadari bahwa masih ada beberapa kekurangan yang sangat di perlu perbaikan dan pengembangan di penelitian selanjutnya. Oleh karena itu, penulis memiliki beberapa saran untuk pengembangan selanjutnya, sebagai berikut:

1. Pengembangan fitur tambahan, aplikasi ini dapat dikembangkan lebih lanjut untuk memberikan pengalaman belajar yang lebih interaktif dan menarik.
2. Pengoptimalan tampilan yang lebih menarik dan lebih interaktif.
3. Meskipun algoritma binary search sudah efisien, pengoptimalan lebih lanjut pada struktur data dan penyimpanan dapat dilakukan untuk mempercepat proses pencarian dan mengurangi penggunaan memori.

DAFTAR PUSTAKA

- Ade Hastuty Hasyim. (2021). *Dasar Pemrograman*
- Andriani, A., Faizah, A., Lazulfa, I., & Istiqomah, I. (2021). *Aplikasi belajar tajwid menggunakan binary search*. *Jurnal Eltek*, 19(2), 1-8.
- Alfian, J. (2021). *Analisis Dan Perancangan Kamus Bahasa Ilmiah Tumbuhan Berbasis Android*. *Jurnal Informatika Dan Rekayasa Perangkat Lunak*, 2(3), 399-406.
- Deva, A. R. S. (2020). *Penerapan Algoritma Binary Search Pada Aplikasi E-Order (Studi Kasus Paris Van Java Kota Bengkulu)*. *JUKOMIKA (Jurnal Ilmu Komputer dan Informatika)*, 3(6), 571-580.
- Khairul, K., Haryati, S., & Yusman, Y. (2018). *Aplikasi Kamus Nama Jawa Indonesia dengan Algoritma Raita Berbasis Android*. *Jurnal Teknologi Informasi dan Pendidikan*, 11(1), 1-6.
- Kasema, L. O., Sentinuwo, S. R., & Sambul, A. M. (2018). *Aplikasi Kamus Nama Daerah Pasan Berbasis Android*. *Jurnal Teknik Informatika*, 13(2).
- Masnur, M., & Difla, D. I. (2021). *Sistem Informasi Penyedia Lowongan Kerja Berbasis Web*. *Jurnal Sintaks Logika*, 1(2), 82-88.
- Markuci, D., & Prianto, C. (2022). *Analisis perbandingan penggunaan algoritma Sequential Search dan Binary Search pada aplikasi surat perjalanan dinas*. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 6(1), 110-119.
- Maya, S. dan Nur, R. A. (2020), *Zoologi Invertebrata*, CV Widina Media Utama.
- Maya, S. dan Nur, R. A. (2021), *Zoologi Vertebrata*, CV Widina Media Utama.
- Mahfudh, A. A., Nur'aini, S., Wibowo, N. C. H., & Kusnanto, C. (2022). *Aplikasi Media Pembelajaran Klasifikasi Hewan Vertebrata Menggunakan Augmented Reality Dengan Marker Based*. *Walisongo Journal of Information Technology*, 4(2), 95-103.
- Mide, B., & Masnur, M. (2021). *Aplikasi Virtual Tour Fakultas Teknik Berbasis Android Mobile*. *Jurnal Sintaks Logika*, 1(2), 113-119.
- Pebriyanti, R., & Ardian, Z. (2018). *Rancang Bangun Aplikasi Kamusnamaindonesia-nama Aceh Menggunakan Metode Rule Based Berbasis Android*. *Journal of Informatics and Computer Science*, 4(1), 91-104.
- Sulistio, B., Lutfi, S., & Ridwan, R. (2019). *Aplikasi Kamus Nama Taliabu Berbasis Android Dengan Menggunakan Metode Binary Search*. *JIKO (Jurnal Informatika dan Komputer)*, 2(2), 67-72.
- Silalahi, M. (2018). *Perbandingan performansi database mongodb dan mysql dalam aplikasi file multimedia berbasis web*. *Computer Based Information System Journal*, 6(1), 63-63.
- Rahmanto, Y., Alfian, J., Damayanti, D., & Borman, R. I. (2021). *Penerapan Algoritma Sequential Search pada Aplikasi Kamus Bahasa Ilmiah Tumbuhan*. *Jurnal Buana Informatika*, 12(1), 21-30.

- Ramadhani, R. Z., & Rusdianto, H. (2019). *Rancang Bangun Aplikasi Pusat Informasi Sekolah Dengan Penerapan Chatbot Menggunakan Aiml Berbasis Android Pada Smk Otomotif Al Husna Tangerang. JIKA (Jurnal Informatika)*, 3(2), 110-116.
- Tobing, F. A. T., & Nainggolan, R. (2020). *Analisis Perbandingan Penggunaan Metode Binary Search Dengan Regular Search Expression. METHOMIKA: Jurnal Manajemen Informatika & Komputisasi Akuntansi*, 4(2), 168-172.
- Wijaya, A., Maulana, M. F., & Danil, M. (2021). *Aplikasi Pencarian Resep Masakan Khas Bengkulu Menggunakan Algoritma Binary Search Berbasis Android. Jurnal Media Infotama*, 17(1).