

APLIKASI KAMUS BAHASA LATIN HEWAN DAN TUMBUHAN MENGGUNAKAN METODE *BRUTE FORCE* DAN FITUR *AUTOCOMPLETE* BERBASIS ANDROID

Nur Imran Ruslan^{*1}, Bambang Pramono², Subardin³

^{*1,2,3}Jurusan Teknik Informatika, Fakultas Teknik, Universitas Halu Oleo, Kendari

e-mail : ^{*1}familyoflocosimbank@gmail.com, ²bambangpramono09@gmail.com,

³mail.bardin@gmail.com

Abstrak

Kamus bahasa latin hewan dan tumbuhan berbasis android adalah suatu pengembangan aplikasi *mobile* yang memungkinkan penggunanya dapat mencari kata dengan mudah dan cepat. Aplikasi ini memungkinkan pengguna dengan cepat mencari arti dari bahasa latin hewan maupun tumbuhan. Proses yang terjadi pada aplikasi ini meliputi pencarian kesamaan kata dengan menggunakan metode pencarian *Brute Force*. *Input*-an berupa nama hewan atau tumbuhan merupakan nilai *string* yang akan dijadikan parameter pencarian didalam database.

Hasil pengujian Kamus pencarian menggunakan metode *Brute Force* berhasil dengan kecepatan kurang dari satu detik waktu pencarian. Dalam pengujian ini, digunakan enam kali pencarian dimana tiga dari enam pencarian menggunakan kata hewan dan sisanya menggunakan kata tumbuhan. Kecepatan proses pencarian menggunakan metode *Brute Force* bergantung dengan panjang kata atau (*String*) yang dicari.

Kata kunci— Kamus Android, Kamus Hewan dan Tumbuhan, *Brute Force*

Abstract

Latin dictionary of animals and plants is a development of Android-based mobile application that allows users to search for words easily and quickly. This application allows users to quickly search for the meaning of the Latin animals and plants. The process that occurs in these applications include word similarity search using the search Brute Force. Input form Animal or Plant name is a string value that will be used as search parameters in the database.

The test results Dictionary search using the Brute Force successfully at speeds of less than one second search time. In this test, is used six times a search in which three of the six search using the Animal and Plant remainder using the word. Speed search process using Brute Force methods depend on the length of the word or (String) sought.

Keywords— *Android Dictionary, Dictionary of Animals and Plants, Brute Force*

1. PENDAHULUAN

Kehidupan manusia tidak lepas dengan adanya berbagai macam makhluk hidup lainnya yang berada di alam semesta seperti hewan dan tumbuhan dimana itu semua terdapat dan bisa kita pelajari dalam dunia biologi, selain kita bisa mempelajari seputar siklus hidup, jenis serta ciri-ciri dari makhluk hidup tersebut kita juga bisa mempelajari nama latin makhluk hidup

tersebut untuk mengetahui penggolongan ciri dan jenisnya berdasarkan nama latinnya, dalam bahasa ilmiah atau bisa disebut dengan bahasa latin. Tetapi dalam beberapa kasus para siswa sering kali mengalami kesulitan ketika diharuskan untuk mengenal bahkan menghafal nama ilmiah dari hewan dan tumbuhan tersebut.

Teknologi *mobile* merupakan bagian yang familiar dalam kehidupan masyarakat

Indonesia. Hal itu disebabkan ukurannya yang cukup kecil sehingga mudah dibawa dan fungsinya yang memudahkan untuk berkomunikasi setiap saat dan dimana saja. Bahkan hanya komunikasi *verbal* yang dimungkinkan dengan teknologi ini, teknologi pengaksesan informasi juga mengalami dampak yang positif, pembangunan teknologi telepon seluler beberapa tahun belakangan juga menawarkan potensi untuk pengalaman multimedia yang berkualitas. Ukuran yang kecil dan fungsi yang memudahkan komunikasi dan pengaksesan data merupakan salah satu sebab menariknya dunia *mobile* untuk terus dieksplorasi dan dikembangkan, termasuk untuk kepentingan *edukasi* yang bersifat *ubiquitous* (dimana saja dan kapan saja). *Handphone* merupakan sebuah sarana telekomunikasi yang efektif. Seiring dengan berjalannya waktu serta berkembangnya dunia informasi dan teknologi, *Handphone* dengan tidak menghilangkan fungsi utamanya bermetamorfosis menjadi perangkat yang multifungsi.

Kemajuan Teknologi Informasi juga menyebabkan berkembangnya sistem operasi ponsel. Salah satu sistem operasi ponsel yang saat ini sedang banyak digunakan adalah android. Android adalah salah satu *platform* yang diidam-idamkan masyarakat, selain *platform* Blackberry yang akhir-akhir ini menjadi trend masa kini, yang hampir digunakan oleh semua usia dari yang muda hingga tua. Android adalah kumpulan perangkat lunak yang ditujukan bagi perangkat bergerak mencakup sistem operasi, *middleware*, dan aplikasi kunci. Dengan adanya aplikasi ini diharapkan masyarakat umum khususnya pelajar akan mempermudah dalam mempelajari bahasa latin hewan dan tumbuhan guna mendukung proses belajar dari pelajaran yang terkait seperti halnya pelajaran ilmu pengetahuan alam dan biologi.

Nama ilmiah dari makhluk hidup terutama hewan dan tumbuhan merupakan salah satu hal yang pada dasarnya sangat menarik untuk dipelajari dan diketahui, karena nama ilmiah makhluk hidup memberikan peran penting, antara lain, dengan nama ilmiah maka akan dengan mudah mengetahui ciri-ciri, hubungan kekerabatan, dan interaksi makhluk hidup di lingkungan. Tetapi pada umumnya para siswa terutama bagi mereka yang sedang mempelajari nama ilmiah hewan dan

tumbuhan mengalami kesulitan dalam mempelajari nama ilmiah tersebut karena semua menggunakan bahasa latin atau bahasa biologi. Dalam mempelajari tentang nama ilmiah hampir semua masyarakat dan para pelajar pada umumnya tidak menggunakan media bantu, karena memang media yang tersedia kebanyakan hanya berupa buku dan itu tidak semua orang mempunyai buku tersebut. Dan ketidakterediaan media teknologi informasi yang dapat dengan cepat memberikan informasi tentang nama ilmiah hewan atau tumbuhan membuat para siswa semakin sulit dalam mempelajari nama ilmiah tersebut. Kekurangan lainnya dalam mempelajari nama ilmiah hewan ataupun tumbuhan adalah tidak adanya gambar-gambar hewan dan tumbuhan beserta tata namanya. Tidak adanya media yang mendukung inilah semakin membuat para siswa kesulitan untuk mempelajari nama ilmiah hewan dan tumbuhan dan memperoleh informasi tentang bahasa ilmiah dari hewan dan tumbuhan secara cepat dan akurat.

Dari beberapa masalah yang telah dikemukakan tersebut, maka perlu adanya sebuah pembangunan aplikasi yang dapat membantu mengatasi hal tersebut. Ada beberapa metode yang dapat di implementasikan dalam pembangunan aplikasi tersebut salah satunya adalah metode *Brute Force*.

Berdasarkan latar belakang yang telah diuraikan maka penulis mengambil penelitian “Aplikasi Kamus Bahasa Latin Hewan Dan Tumbuhan Menggunakan Metode *Brute Force* Dan Fitur *Autocomplete* Berbasis Android”. Data-data yang diperoleh untuk mendukung aplikasi bersumber dari buku “Kamus *nomenklatur zoologi dan botani*” [1] dan [2].

2. METODE PENELITIAN

2.1 Kamus

Menurut [3], Kamus yaitu buku acuan yang memuat kata dan ungkapan yang biasanya disusun menurut abjad berikut keterangan tentang maknanya. Kamus adalah sejenis buku rujukan yang menerangkan makna kata-kata. Kamus berfungsi untuk membantu seseorang mengenal perkataan baru. Selain menerangkan maksud kata kamus juga mungkin mempunyai pedoman sebutan, asal usul (etimologi) sesuatu perkataan dan

juga contoh penggunaan bagi sesuatu perkataan. Untuk memperjelas kadang kala terdapat juga ilustrasi didalam kamus. Kata kamus diserap dari bahasa Arab *qamus*, dengan bentuk jamaknya *qawamis*. Kata Arab itu sendiri berasal dari kata Yunani *okeanos* yang berarti lautan. Sejarah kata itu jelas memperlihatkan mana dasar yang terkandung dalam kata kamus, yaitu wadah pengetahuan, khususnya pengetahuan bahasa, yang tidak terhingga dalam dan luasnya. dalam pengertian lain, Kamus adalah buku acuan yang memuat kata dan ungkapan, biasanya disusun menurut abjad beserta penjelasan tentang makna dan pemakainya (Kamus Besar Bahasa Indonesia).

Kamus disusun sesuai dengan abjad dari A-Z dengan tujuan untuk memudahkan pengguna kamus dalam mencari istilah yang diinginkannya dengan cepat dan mudah. Kamus memiliki kegunaan untuk memudahkan penggunaannya dalam mencari istilah-istilah yang belum dipahami maknanya.

2.2 Bahasa Latin

Bahasa Latin adalah sebuah bahasa Italik yang berasal dari Latium, sebuah daerah di Italia sekeliling kota Roma yang termasuk dalam rumpun Indo Eropa. Bahasa ini menjadi penting karena munculnya kekaisaran Romawi dimana bahasa Latin adalah bahasa resminya. Pada puncak kejayaan kerajaan ini, bahasa Latin dituturkan dari pulau Britania di barat laut sampai Palestina di ujung tenggara dengan kata lain menjadi bahasa Internasional.

Setelah runtuhnya kekaisaran Romawi, sekitar akhir abad ke-5, bahasa Latin tidak ikut runtuh melainkan semakin berkembang. Bahasa ini dipakai sebagai *lingua franca*, bahasa *liturgis* gereja dan bahasa ilmu pengetahuan, bahkan juga daerah-daerah yang tidak pernah ditaklukkan oleh Roma.

Bahasa Latin juga mempunyai turunan yang disebut sebagai *sermo vulgaris* (bahasa Roman/bahasa Latin Rakyat) yang dituturkan oleh antara lain bala tentara Romawi menjadi pengantar di seluruh daerah kerajaan. Di beberapa tempat bahasa ini bahkan menggantikan bahasa setempat.

Bahkan bahasa Inggris pun sebagian besar diambil dari bahasa latin ini (selain bahasa Yunani Kuno tentunya) [4].

2.3 Hewan

Hewan atau binatang atau margasatwa atau satwa saja adalah kelompok organisme yang diklasifikasikan dalam kerajaan *Animalia* atau *Metazoa*, adalah salah satu dari berbagai makhluk hidup yang terdapat di alam semesta. Hewan dapat terdiri dari satu sel (*uniselular*) atau pun banyak sel (*multiselular*). Semua hewan merupakan *organisme heterotrof* [5].

2.4 Tumbuhan

Tumbuhan adalah organisme benda hidup yang terkandung dalam alam *Plantae*. Biasanya, organisme yang menjalankan proses *fotosintesis* adalah diklasifikasikan sebagai tumbuhan. Tumbuhan memerlukan cahaya matahari untuk menjalani proses *fotosintesis*.

Tumbuhan merangkumi semua benda hidup yang mampu menghasilkan makanan dengan menggunakan *klorofil* untuk menjalani proses *fotosintesis* dan menghasilkan kanji. Sel tumbuhan berbeda dengan sel hewan, dalam beberapa segi sel tumbuhan mempunyai dinding sel [5].

2.5 Brute-Force

Algoritma *Brute Force* adalah salah satu cara penyelesaian masalah khususnya di dalam dunia komputer. *Brute Force* adalah sebuah pendekatan yang lempang (*straightforward*) untuk memecahkan suatu masalah, biasanya didasarkan pada pernyataan masalah (*problem statement*) dan definisi konsep yang dilibatkan.

Algoritma *Brute Force* memecahkan masalah dengan sangat sederhana, langsung dan dengan cara yang jelas (*obvious way*). Algoritma yang secara jelas langsung ke pusat permasalahan. Algoritma ini biasanya tidak memerlukan teori khusus untuk mengimplementasikannya. Algoritma ini sering juga disebut Algoritma sapu jagad karena hampir semua persoalan pemrograman bisa diselesaikan dengan algoritma ini.

Algoritma *Brute Force* sering disebut Algoritma tidak cerdas karena tidak memerlukan pemikiran secara kompleks, hanya dengan beberapa baris semua permasalahan dapat dipecahkan dengan mudahnya [6].

Adapun simbol dari pencarian *String* ini adalah sebagai berikut :

t = teks (*text*), yaitu (*long string*) yang panjangnya n karakter

p = *pattern*, yaitu *string* dengan panjang m karakter ($m < n$) yang akan dicari di dalam teks.

c = *character*/karakter

$$\left. \begin{array}{l} t : t_{c1}, t_{c2}, t_{c3}, t_{c4}, \dots, t_{cn} \\ p : p_{c1}, p_{c2}, p_{c3}, \dots, p_{cm} \end{array} \right\} (p_{cm} < t_{cn})$$

Adapun langkah-langkah yang dilakukan algoritma *Brute Force* pada saat mencocokkan *string* adalah:

1. Algoritma *Brute Force* mulai mencocokkan *pattern* pada awal teks.
2. Dari kiri ke kanan, algoritma ini akan mencocokkan karakter per karakter *pattern* dengan karakter di teks yang bersesuaian, sampai salah satu kondisi berikut dipenuhi:
 - a) Karakter di *pattern* dan di teks yang dibandingkan tidak cocok (*mismatch*).
 - b) Semua karakter di *pattern* cocok. Kemudian algoritma akan memberitahukan penemuan di posisi ini.

Algoritma kemudian terus menggeser *pattern* sebesar satu ke kanan, dan mengulangi langkah ke-2 sampai *pattern* berada di ujung teks [7]

2.6 Autocomplete

Autocomplete adalah fungsi yang menampilkan perkiraan kata yang akan dimasukkan tanpa harus mengetikkan keseluruhan kata. Fitur ini juga merupakan salah satu bentuk *search* yang sering kita jumpai pada saat melakukan pencarian di Google.

Autocomplete atau *autocomplete* atau *word completion* adalah fasilitas yang disediakan oleh berbagai *web browser*, *email-programs*, *search engine interface*, *source code editors*, *database query tools*, *word processor*, dan *command line interpreters*. *Autocomplete* juga tersedia untuk atau sudah terintegrasi di dalam *text editor*.

Autocomplete bekerja ketika penulis menulis huruf pertama atau beberapa huruf/karakter dari sebuah kata. Program yang melakukan prediksi akan mencari satu atau lebih kemungkinan kata sebagai pilihan. Jika kata yang dimaksud ada dalam pilihan itu, maka penulis dapat memilih itu. Jika kata yang dimaksud tidak ada dalam pilihan prediksi

maka penulis harus menulis huruf/karakter selanjutnya. Ketika penulis memilih pilihan kata yang ada dalam daftar pilihan kata prediksi maka kata yang dipilih tersebut akan disisipkan pada teks.

Dalam *source code editor*, *autocomplete* menyederhanakan struktur regular dari sebuah bahasa pemrograman. Biasanya hanya ada sejumlah kata yang berarti dalam konteks saat ini atau *namespace*, seperti nama dari sebuah *variable* atau nama dari sebuah fungsi. Contoh dari *code completion* adalah desain *IntelliSense* dari *Microsoft*. Dia menampilkan sebuah *pop-up list* yang berisi prediksi yang mungkin dari karakter yang dimasukkan oleh pengguna untuk dipilih pengguna yang sesuai dengan apa yang diinginkan pengguna.

Fitur ini sangat berguna dalam pemrograman berorientasi objek karena seringkali *programmer* tidak tahu secara benar *member* apa saja yang sebuah *class* punya [8].

2.7 Teknik Pengumpulan Data dan Informasi

Dalam penelitian ini, penulis membutuhkan data atau informasi dan referensi yang relatif lengkap. Oleh karena itu penulis melakukan riset atau penelitian untuk mendapatkan data dan referensi yang diperlukan. Adapun metode pengumpulan data-data dan informasi yang diperlukan adalah studi pustaka, yaitu teknik pengumpulan data dengan menghimpun dan menganalisis dokumen. Dokumen yang termasuk didalamnya yaitu penelitian terdahulu, buku, artikel, dan jurnal yang berkaitan dengan objek penelitian yang didapatkan melalui internet. Secara rinci yang penulis lakukan pada studi pustaka ini adalah:

1. Mempelajari buku-buku atau dokumen-dokumen, jurnal dan artikel-artikel yang berhubungan dengan bahasa latin hewan dan tumbuhan.
2. Mempelajari penelitian sebelumnya yang berhubungan dengan kamus *mobile android* dan metode *Brute Force*.

2.8 Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan adalah metode *Rational Unified Process* (RUP) karena pengembangan aplikasi ini dalam ruang lingkup kecil/ menengah. Terdapat beberapa tahapan dalam metode ini, yaitu:

1. Inception

Pada tahap ini penulis menentukan ruang lingkup proyek dari hasil observasi yang penulis lakukan, meliputi dari hasil penelitian skripsi-skripsi terdahulu.

2. Elaboration

Pada tahap ini setelah penulis mengadakan observasi/studi pustaka, kemudian penulis melakukan identifikasi masalah, dan menentukan alur bisnis dan aplikasi serta wilayah persoalan data yang akan didukung oleh sistem yang akan dikembangkan serta ditentukan pula jangkauan atau batasan sistem.

Di dalam *elaboration* terdapat dua tahapan, yaitu:

- a. Analisis
- b. *Modelling* (Perancangan)

3. Construction

Dalam tahapan *construction* ini meliputi bagaimana suatu aplikasi itu bisa diimplementasikan dan diuji coba yaitu :

- a. Implementasi.
- b. *Coding*
- c. *Testing*

4. Transition

Pada tahap ini penulis membuat apa yang sudah dimodelkan di tahapan-tahapan sebelumnya menjadi suatu barang yang layak jadi.

2.9 Analisis Sistem

Analisis sistem merupakan suatu tahapan yang bertujuan untuk mengetahui dan mengamati apa saja yang terlibat dalam suatu sistem. Pembahasan yang ada pada analisis sistem ini yaitu analisis masalah, analisis algoritma, analisis kebutuhan nonfungsional, dan analisis kebutuhan fungsional.

2.10 Analisis Kebutuhan Perangkat Keras

Perangkat keras adalah alat yang dipakai untuk mengolah data serta menyajikan laporan. Spesifikasi *hardware* yang digunakan untuk membuat aplikasi ini yaitu:

- 1. Processor : Intel (R) Core i3 (R) (CPU 560 @ 2.53GHz 2.5 Ghz)
- 2. Monitor LCD 14" dengan resolusi 1280 x 800 pixels
- 3. *Smartphone* android Advan S5, *ice cream sandwich* 4.0.4

2.11 Analisis Kebutuhan Perangkat Lunak

Perangkat lunak yang digunakan pada pembangunan aplikasi kamus bahasa latin

hewan dan tumbuhan ini dengan spesifikasi *Software* yang digunakan untuk membuat aplikasi ini yaitu:

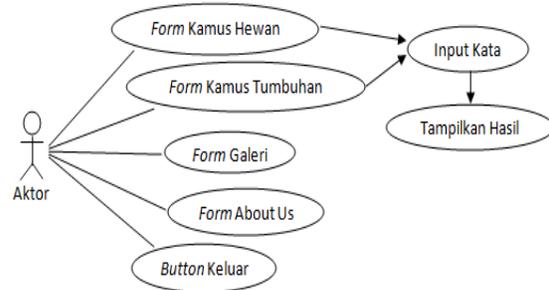
- 1. *Operating System* : *Windows Ultimate*
- 2. *Java Development Kit* (JDK) : JDK 1.8.0
- 3. *Software Development Tools* : SDK
- 4. *Android Developer Tools* : ADT
- 5. *Database Tools* : Sqlite

JDK dipilih sebagai perangkat lunak untuk bahasa pemrograman java, SDK dipilih sebagai perangkat lunak untuk memilih package dari android yang akan digunakan, ADT dipilih sebagai perangkat lunak untuk editor pembuatan program.

2.12 Perancangan Sistem

a. Use Case Diagram

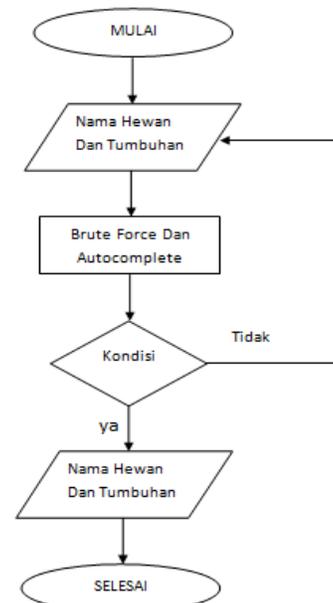
Gambar 1 menunjukkan *Use Case Diagram* sistem.



Gambar 1 *Use case diagram* sistem

b. Flowchart Sistem

Gambar 2 menunjukkan *flowchart* sistem.



Gambar 2 *Flowchart* sistem

Gambar 2 menyatakan bahwa untuk melakukan pencarian terlebih dahulu ditentukan kata atau pola hewan dan tumbuhan yang akan dicari, kata tersebut akan diproses dengan metode *Brute Force*, dimana algoritma *Brute Force* bergeser dari posisi kiri ke kanan untuk mengecek satu persatu kata yang diinputkan dan menyamakan kata tersebut dengan teks yang ada didatabase lalu fitur *autocomplete* melengkapi kata yang ada dan menampilkan kata secara lengkap yang sesuai dari hasil pencarian. Apabila kata atau pola ditemukan maka hasil akan ditampilkan, proses akan selesai dan apabila hasil tidak ditemukan maka kita diisyaratkan untuk menginput kata atau pola hewan dan tumbuhan baru yang akan dicari.

2.13 Gambaran Umum Sistem

Dalam pencarian arti dari bahasa latin hewan atau tumbuhan dibagi menjadi 3 tahap, tahap yang pertama adalah proses penginputan teks yang akan dicari, tahap kedua adalah proses pencocokan data *inputan* dengan data yang ada didalam database menggunakan algoritma *Brute Force*, tahap ketiga menampilkan hasil pencarian. Secara umum alur program ditunjukkan oleh Gambar 3.



Gambar 3 Alur sistem aplikasi Kamus Latin Hewan dan Tumbuhan

3. HASIL DAN PEMBAHASAN

3.1 Implementasi Desain Antarmuka

Tahap ini bertujuan untuk mengetahui apakah kebutuhan sistem yang telah diintegrasikan telah terpenuhi. Implementasi perancangan antarmuka aplikasi yaitu :

1. Implementasi Splash Screen

Tampilan *splash screen* merupakan tampilan pembuka/proses *loading* yang muncul ketika aplikasi dijalankan sebelum ke menu utama, yang ditunjukkan oleh Gambar 4.

2. Form Menu Utama

Dalam *form* menu utama ini tampil beberapa pilihan menu, disini *user* bisa

memilih menu yang disediakan, yang ditunjukkan oleh Gambar 5.



Gambar 4 Tampilan *Splash Screen*



Gambar 5 *Form* Menu Utama

3. Form Pencarian Data Kamus Hewan

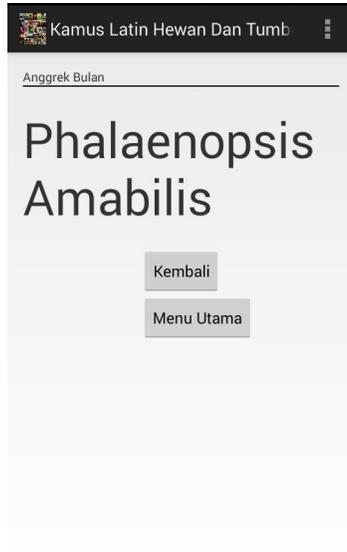
Dalam *form* pencarian data Kamus Hewan *user* dapat menginputkan kata hewan yang akan dicari dan melihat terjemahannya dengan cara mengklik salah satu daftar nama hewan. Gambar 6 menunjukkan *form* Pencarian Data Kamus Hewan.



Gambar 6 Form Kamus Hewan

4. Form Pencarian Data Kamus Tumbuhan

Dalam *form pencarian* Kamus Hewan *user* dapat menginputkan kata hewan yang akan dicari dan melihat terjemahannya dengan cara mengklik salah satu daftar nama hewan, yang ditunjukkan oleh Gambar 7.



Gambar 7 Form Kamus Tumbuhan

5. Form Gallery Hewan dan Tumbuhan

Pada *form gallery*, pengguna bisa melihat *list image* dan mendengar penjelasan keterangan hewan dan tumbuhan melalui rekaman suara penerjemah dari daftar hewan

dan tumbuhan, yang ditunjukkan oleh Gambar 8.



Gambar 8 Gallery hewan (kiri) dan tumbuhan (kanan)

6. Form Detail Gallery

Dalam *form detail* galeri *user* dapat melihat tampilan gambar hewan dan tumbuhan serta mendengar keterangan melalui rekaman suara yang ada pada daftar *gallery*, ditunjukkan oleh Gambar 9.



Gambar 9 Detail gallery Tumbuhan

3.2 Pengujian Sistem

Pengujian sistem merupakan tahap utama dalam pembuatan suatu aplikasi, hasil pengujian yang didapatkan dijadikan sebagai

tolak ukur dalam proses pengembangan selanjutnya.

Pengujian fungsionalitas mengacu pada analisis kebutuhan perangkat lunak yang telah dijabarkan sebelumnya. Pada intinya yang dilakukan pada tahap ini adalah menguji apakah sistem dapat bekerja seperti yang diharapkan pada analisis kebutuhan.

Pengujian performansi dilakukan untuk membandingkan hasil sebelum dan sesudah data ditampilkan. Tabel 1 menunjukkan hasil Pengujian Fungsionalitas.

Tabel 1 Hasil Pengujian Fungsionalitas

No.	Pengujian	Kriteria Keberhasilan	Status
1.	Menerima <i>inputan</i> berupa kata hewan pada menu kamus hewan	Perangkat lunak dapat menerima masukan berupa kata atau teks dengan panjang teks yang berbeda-beda.	Berhasil
2.	Menerima <i>inputan</i> berupa kata tumbuhan pada menu kamus tumbuhan	Perangkat lunak dapat menerima masukan berupa kata atau teks dengan panjang teks yang berbeda-beda.	Berhasil
3.	Mencari data kata hewan maupun tumbuhan menggunakan pencarian <i>Brute Force</i>	Perangkat lunak dapat mencari data yang <i>diinputkan</i> .	Berhasil
4.	Menampilkan data <i>gallery</i> menggunakan menu <i>gallery</i>	Perangkat lunak dapat menampilkan data <i>gallery</i>	Berhasil
5.	Menguji tombol keluar dari sistem	Perangkat melakukan <i>dispose system</i>	Berhasil

Tabel 1 menunjukkan bahwa pengujian dari aspek fungsionalitas sudah sesuai dengan analisis kebutuhan sistem. Perangkat lunak dapat menerima masukan berupa kata atau teks yang mempunyai panjang berbeda-beda

dan mencari pola *string* yang cocok menggunakan metode *Brute Force*.

Tabel 2 menunjukkan hasil Pengujian Performansi.

Tabel 2 Hasil Pengujian Performansi

No.	Input	Hasil	Waktu Pencarian
1.	Anjing	<i>Canis Familiaris</i>	Time < 1 second
2.	Babirusa	<i>Babyrousa Babyrussa</i>	Time < 1 second
3.	Banteng Jawa	<i>Bos Javanicus Javanicus</i>	Time < 1 second
4.	Anggur	<i>Vitis Vinifera</i>	Time < 1 second
5.	Bawang Putih	<i>Allium Sativum</i>	Time < 1 second
6.	Anggrek	<i>Orcidaceae Sp</i>	Time < 1 second

Tabel 2 menunjukkan bahwa pengujian dari aspek performansi perangkat lunak ini membutuhkan waktu pencarian yang relatif sama dan cepat untuk ukuran *string* yang *input*-kan.

4 KESIMPULAN

Informasi adalah sesuatu yang sangat penting dan berguna bagi banyak orang. Salah satu informasi yang dapat berguna yaitu bahasa latin hewan dan tumbuhan sebagai media informasi dan pembelajaran khususnya pada mata pelajaran biologi. Ditambah lagi dengan media pembelajaran seperti buku-buku dan kamus yang ketersediaannya sangat kurang sehingga penulis dalam tugas akhir ini merancang aplikasi kamus bahasa latin hewan dan tumbuhan berbasis android dengan menggunakan akses *smartphone* yang dapat diakses kapan saja dan dimana saja tanpa harus membawa buku yang tebal dan bersifat manual saja.

Dari penelitian dan pengujian aplikasi maka kesimpulan yang dapat diambil dari penelitian, pengujian aplikasi dan penulisan tugas akhir ini adalah sebagai berikut:

1. Kamus dengan pencarian data menggunakan metode *Brute Force*

- memungkinkan pencarian data tidak ada yang terlewatkan.
2. Proses pencarian menggunakan metode *Brute Force* pada aplikasi ini berjalan normal dan efektif berdasarkan data pengujian yang telah dilakukan yaitu tingkat kecepatan yang terhitung cepat dimana proses pencarian memerlukan waktu kurang dari satu detik.
 3. Aplikasi kamus bahasa latin hewan dan tumbuhan berbasis android lebih mempermudah mengakses kamus bahasa latin hewan dan tumbuhan melalui teknologi khususnya *smartphone* android.
 4. Memberi fasilitas aplikasi yang berguna kepada para pengguna aplikasi nantinya untuk mengoptimalkan penggunaan *smartphone* sebagai sarana informasi dan edukasi.
 5. Mempermudah pengguna untuk mencari arti dari bahasa latin hewan dan tumbuhan dengan berpatokan akan kurangnya sumber berupa buku dan media informasi lain mengenai bahasa latin hewan dan tumbuhan.

5. SARAN

Aplikasi yang dibangun tentunya masih memiliki beberapa kekurangan, sehingga perlu adalah beberapa saran untuk proses pengembangan aplikasi kedepannya sehingga bisa menjadi lebih baik dan berkembang. Saran yang perlu diperhatikan untuk penelitian lebih lanjut adalah:

1. Membuat *update* data secara *online* agar pengetahuan pengguna lebih luas lagi.
2. Aplikasi ini hanya melakukan pencarian dalam bentuk kata atau teks. untuk pengembangan selanjutnya diharapkan aplikasi ini sudah bisa dikembangkan dengan pencarian dalam bentuk *voice search* atau pencarian menggunakan suara.
3. Fitur pada aplikasi diharapkan dapat di kembangkan lebih sehingga lebih efisien dalam hal penggunaan memori dan penggunaannya.
4. Agar data pada sistem tidak terlalu berat, maka perlu dilakukan kompresi untuk setiap file gambar dan suara ke ukuran terkecil untuk mengefisienkan penggunaan memori.

5. Perlunya penambahan fitur pada aplikasi sehingga lebih menarik user untuk menggunakannya.

DAFTAR PUSTAKA

- [1] Martinus. S. dan Hartono, R.S., 2008, *Kamus Nomenklature Zoologi Dan Botani*, edisi : 1, Penerbit Andi, Yogyakarta.
- [2] Adiwijaya, S. dan Ilyas, A.M., 2008, *Kamus Biologi*, edisi : 1, Penerbit Pustaka Karya, Surabaya.
- [3] Hoetomo, M.A., 2005, *Kamus Lengkap Bahasa Indonesia*, Mitra pelajar : Surabaya.
- [4] Susanto, A. E., Sejarah Bahasa Latin, www.scribd.com/doc/127313880/Sejarah-Bahasa-Latin#scribd, Diakses Tanggal 16 Oktober 2015.
- [5] Suroso, A. dan Permatasari, A., 2003, *Ensiklopedia Sains dan Kehidupan: Referensi dan Petunjuk Lengkap untuk ilmu Biologi, Fisika, dan Kimia*, CV Tarity Samudra Berlian, Jakarta.
- [6] Munir, R., 2004, *Algoritma Brute Force Bagian 2 – Algoritma Brute Force (Lanjutan).ppt*, ITB, Bandung.
- [7] Abdeen, R.A., 2011, An Algorithm for String Searching Based on Brute-Force Algorithm, *International Journal of Computer Science and Network Security*, 11,7, 24-27.
- [8] Kusuma, M.W., 2013, *Pencocokan String dalam Fitur Autocompletion pada Text Editor atau Integrated Development Environment (IDE)*, Institut Teknologi Bandung, Bandung.

