

## Penerapan Algoritma *Sequential Search* pada Aplikasi Kamus Bahasa Ilmiah Tumbuhan

Yuri Rahmanto<sup>1</sup>, Joni Alfian<sup>2</sup>, Damayanti<sup>3</sup>, Rohmat Indra Borman<sup>4</sup>

Program Studi Teknik Komputer, Fakultas Teknik dan Ilmu Komputer,  
Universitas Teknokrat Indonesia

Jl. ZA. Pagar Alam No.9-11, Labuhan Ratu, Bandar Lampung 35142, Lampung, Indonesia  
Email: <sup>1</sup>yurirahmanto@teknokrat.ac.id, <sup>2</sup>jonialfian@gmail.com, <sup>3</sup>damayanti@teknokrat.ac.id,  
<sup>4</sup>rohmat\_indra@teknokrat.ac.id

**Abstract.** *Application of Sequential Search Algorithms in the Plant Scientific Language Dictionary Application.* Scientific dictionaries such as plants dictionaries are thick and less attractive. Therefore, it is necessary to develop a scientific language dictionary application for plant which is attractive and can be accessed anytime and anywhere. The dictionary application is closely related to the search feature and the sequential search algorithm is used in the search feature at the application development. This algorithm performs the search by matching data in a row from all existing data. The application built had features such as a list of plants with scientific names equipped with A-Z alphabet menus, plant categories, and a search feature to make it easier for users to find the plant names. The results of the search speed performance test based on page load time with the microtime function show that sequential search algorithms can perform the searches quickly for relatively limited and not too much data.

**Keywords:** *Scientific Language Dictionary, Plant Dictionary, Sequential Search*

**Abstrak.** Kamus ilmiah seperti kamus ilmiah tumbuhan berbentuk tebal dan kurang menarik. Oleh karena itu, diperlukan pengembangan aplikasi kamus Bahasa ilmiah tumbuhan yang menarik dan dapat diakses kapanpun dan dimanapun. Aplikasi kamus akan berhubungan erat dengan pencarian dan algoritma sequential search digunakan sebagai fitur pencarian pada pengembangan aplikasi kamus bahasa ilmiah tumbuhan ini. Algoritma ini melakukan pencarian dengan mencocokkan data secara beruntun dari seluruh data yang ada. Aplikasi yang dibangun memiliki fitur-fitur seperti daftar tumbuhan dengan nama ilmiah yang dilengkapi menu abjad A-Z, kategori tumbuhan, dan fitur pencarian untuk memudahkan pengguna dalam menemukan nama tumbuhan. Hasil pengujian kecepatan pencarian berdasarkan page load time dengan fungsi microtime menunjukkan bahwa algoritma sequential search dapat melakukan pencarian dengan cepat untuk data yang relatif terbatas atau tidak terlalu banyak.

**Kata Kunci:** *Kamus Bahasa Ilmiah, Kamus Tumbuhan, Sequential Search*

### 1. Pendahuluan

Teknologi informasi saat ini telah menjadi keunggulan kompetitif bagi organisasi atau perusahaan. Hal ini disebabkan dengan adanya teknologi informasi mampu mengelola dan menyebarkan informasi cepat, akurat dan terpercaya, sehingga dapat mendukung organisasi untuk berkembang [1]. Penggunaan teknologi informasi kini tidak lagi berfokus pada *Personal Computer* (PC) tetapi sudah merambah ke telepon seluler atau *handphone*. Hal ini terlihat berdasarkan penelitian dari SimilarWeb menunjukkan bahwa akses web menggunakan telepon seluler pada tahun 2017 sampai dengan 2019 terjadi peningkatan hingga 30,6%, hal ini sangat berbeda jauh dengan penggunaan *Personal Computer* (PC) yang hanya 3,3% [2]. Perkembangan teknologi telepon seluler menjadi *smartphone* diimbangi dengan perkembangan sistem operasi yang ada pada *smartphone*. Sistem operasi pada *smartphone* yang populer saat ini adalah Andorid. Android merupakan sistem operasi yang *open source* berbasis Linux yang dirilis oleh google pada tahun 2007 dengan mesin yang diberi nama Dalvik [3]. Penggunaan *smartphone* sangat membantu dalam segala bidang, salah satunya dibidang pendidikan, karena teknologi *mobile* ini memiliki kelebihan dapat digunakan dimana saja dan kapan saja. Munculnya

teknologi *mobile* ini menjadi paradigma baru dalam membantu proses belajar mengajar dalam dunia pendidikan [4].

Saat ini sumber yang digunakan dalam dunia pendidikan sebagian besar masih menggunakan buku. Hal ini bisa menjadi sebuah peluang untuk pengembangan aplikasi yang mengubah buku menjadi aplikasi digital. Jenis buku yang sering digunakan untuk mengetahui istilah-istilah tertentu adalah kamus. Kamus merupakan sebuah buku yang didalamnya terdapat kata berikutan dengan arti dan maknanya yang tersusun berdasarkan abjad yang bermanfaat untuk mengetahui kata-kata baru [5]. Fungsi kamus adalah sebagai alat bantu untuk mengenali kata-kata yang sebelumnya belum diketahui. Kamus berfungsi untuk membantu seseorang mengenal perkataan baru [6]. Selain itu, kamus juga digunakan untuk bidang-bidang tertentu untuk mempermudah mengenali objek tertentu yang jumlahnya sangat banyak. Dalam struktur keilmuan *Biological Science Curriculum Study (BSCS)*, terdapat ribuan istilah serta objek di dunia ini, sehingga objek-objek makhluk hidup perlu dikelompokkan atau pengelompokan atau diklasifikasikan [7]. Pengelompokan atau taksonomi serta pemberian nama ilmiah tersebut disebut dengan *Binomial Nomenklatur*. Satu dari beberapa bahasa ilmiah yang sering digunakan dalam dunia pendidikan yaitu nama ilmiah pada tumbuhan (*plantae*).

Bahasa ilmiah sudah mulai dipelajari pada mata pelajaran Biologi di sekolah. Istilah-istilah ilmiah dari makhluk hidup menjadi menarik untuk dipelajari karena melalui nama ilmiah ini dapat diketahui ciri-cirinya, hubungan kekerabatannya, dan interaksi terhadap lingkungannya [8]. Untuk mempelajari istilah-istilah ilmiah maka membutuhkan kamus bahasa ilmiah. Akan tetapi kamus biasanya berbentuk buku yang tebal sehingga membuat para siswa yang akan belajar bahasa ilmiah menjadi tidak tertarik untuk mempelajarinya. Selain itu kurangnya minat tersebut juga dikarenakan sumber informasi yang belum tersedia secara luas. Maka dibutuhkan digitalisasi kamus untuk mempermudah mempelajari istilah-istilah ilmiah, khususnya bahasa ilmiah pada tumbuhan.

Untuk membangun aplikasi kamus digital maka diperlukan algoritma pencarian yang tepat agar pencarian lebih efektif dan membutuhkan waktu yang cepat. Pencarian kata akan melibatkan algoritma *string matching*. *String matching* merupakan porses pencarian kata dengan cara mencocokkan dari dokumen yang ada [9]. Terdapat beberapa algoritma pencarian yang dapat digunakan, salah satunya adalah *sequential search*. Algoritma *sequential search* biasanya disebut sebagai metode pencarian beruntun, merupakan algoritma yang melakukan pencarian dengan pencocokan dari data yang akan dicari dengan seluruh data yang ada pada sekelompok data dengan cara beruntun mulai dari data pertama sampai dengan data dapat ditemukan [10]. Penggunaan metode *sequential search* digunakan untuk kamus dinilai sangat efektif, ini terbukti dari beberapa penelitian yang menggunakan algoritma ini. Penelitian sebelumnya, tentang implementasi *sequential search* yang digunakan untuk *Jakabaring dictionary* [11], menunjukkan bahwa *sequential search* mampu melakukan pencarian dengan baik dan dapat menemukan kata yang dicari dari *database* yang disimpan. Penelitian lain, tentang pengembangan aplikasi kamus istilah *Neurology* dengan metode pencarian menggunakan *sequential search* [12], yang menghasilkan nilai sebesar 87,75% dari hasil pengujian berdasarkan kegunaan, fungsi, tampilan, kalimat yang digunakan serta kemudahan. Penelitian selanjutnya tentang penerapan algoritma *sequential search* dalam kamus bahasa inawatan [13]. Penelitian ini menunjukkan bahwa algoritma *sequential search* dapat melakukan pencarian dengan efektif, hal ini disebabkan karena metode ini sederhana dan melakukan pencarian secara beruntun sehingga pencarian membutuhkan waktu yang singkat.

Pada penelitian ini, akan membangun aplikasi kamus bahasa ilmiah untuk tumbuhan berbasis Android agar menarik dan memudahkan pengguna dalam melakukan pencarian istilah bahasa ilmiah pada tumbuhan. Penelitian ini menggunakan algoritma *sequential search*, dimana pencarian dilakukan dengan cara beruntun mulai dari data pertama sampai dengan data ditemukan agar pencarian akan lebih tepat dan efektif.

## 2. Tinjauan Pustaka

### 2.1. Kamus Bahasa Ilmiah

Kamus merupakan buku acuan yang di dalamnya terdapat kata-kata dan ungkapan tersusun berdasarkan abjad dengan disertai keterangan dan makna yang digunakan untuk mengenali perkataan baru [5]. Di samping menjelaskan mengenai makna dari istilah tertentu kamus juga berisi mengenai pedoman sebutan, asal mula suatu cara dan bagaimana penggunaan kata tersebut. Dengan kamus seseorang dapat melakukan pencarian kata yang bisa mempermudah dalam mengartikan kata dan mendapatkan informasi mengenai kata tersebut [14]. Kamus telah digunakan di berbagai bidang ilmu, salah satunya bidang ilmu biologi. Pada *Biological Science Curriculum Study* (BSCS) menguak bahwa terdapat ribuan istilah serta objek di muka bumi, untuk itu dilakukan pengelompokan atau pengklasifikasian makhluk hidup [7]. Pengelompokan atau taksonomi serta pemberian nama ilmiah tersebut disebut dengan *Binomial Nomenklatur*. Istilah-istilah pada bidang Biologi yang dipergunakan pada proses belajar mengajar dalam pendidikan adalah klasifikasi (*taksonomi*) serta tata nama ilmiah untuk tumbuhan (*plantae*). Bahasa ilmiah dari makhluk hidup menjadi menarik untuk dipelajari karena melalui nama ilmiah ini dapat diketahui ciri-cirinya, hubungan kekerabatannya, dan interaksi terhadap lingkungannya [8]. Kamus bahasa ilmiah tumbuhan berisi daftar istilah nama-nama tumbuhan dengan bahasa latin atau bahasa ilmiah dari berbagai spesies tanaman dan bunga [15].

### 2.2. Sequential Search (Pencarian Beruntun)

Algoritma *sequential searching* yang biasanya disebut sebagai pendekatan pencarian secara beruntun merupakan pencarian yang diawali dari elemen awal sampai dengan elemen akhir secara beruntun hingga bisa menemukan elemen yang akan dikunjungi [16]. Algoritma ini mampu melakukan pencarian dengan *array* yang data-datanya tidak perlu melalui pengurutan data sebelumnya [17]. Cara kerja algoritma ini adalah dengan melakukan perbandingan antara data yang ada dengan data yang akan dicari satu per satu secara berurutan sampai dapat menemukan datanya. Berikut konsep penggunaan *sequential searching*:

Terdapat A yang terdapat enam data yang disimpan pada *array*. Maka, didapatkan persamaan 1 seperti berikut ini:

$$A[1], A[2], A[3], \dots, A[n-1] \quad (1)$$

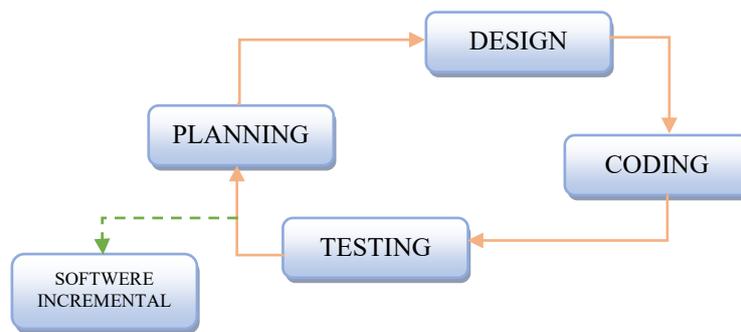
Untuk data yang akan dilakukan pencarian disimbolkan dengan b. Maka, untuk menemukan data tersebut dapat dinotasikan dengan  $A[i]=b$ . Dimana, i merupakan bilangan index yang paling kecil yang dapat terpenuhi dengan kondisi  $0 \leq k \leq n-1$ . Tetap, dapat dimungkinkan data yang akan dilakukan pencarian tidak ditemukan.

Pendekatan *sequential searching* dapat melakukan pengembalian nilai *record* atau *pointer* ke *record*. *Record* merupakan jenis tipe data yang berisi atas variable-varibel atau kumpulanya yang diberi nama *field*. Tahapan dari *Sequential Search* melakukan kunjungan dari satu pohon ke pohon yang lain dengan cara pada tiap-tiap simpulnya dikunjungi sekali saja yang biasanya disebut dengan *tree transversal*/kunjungan pohon. Tahapan pada pendekatan pencarian secara berurutan melalui beberapa proses [16], yaitu melakukan identifikasi array, menentukan data yang akan dilakukan pencarian, mencocokkan data diawali dari data pertama hingga data terakhir, data yang dilakukan pencarian kemudian membandingkan dengan setiap datanya pada array. Pencarian dilakukan dengan membandingkan seluruh data pada array sampai selesai. Apabila data yang akan dilakukan pencarian telah didapatkan maka proses membandingkan elemen *array* akan dihentikan, dan apabila data yang dicari pada seluruh array tidak ditemukan maka proses akan berhenti.

## 3. Metodologi Penelitian

Untuk membangun aplikasi kamus bahasa ilmiah tumbuhan maka digunakan metodologi dalam pengembangan sistemnya. Metode pengembangan sistem yang digunakan adalah pendekatan *Extreme Programming* (XP). Pendekatan XP merupakan metode rekayasa

sistem yang mampu membangun sistem berdasarkan *requirement* yang berubah-ubah [18]. Metode ini menitikberatkan pada proses pengembangan yang *responsive* terhadap apa yang dibutuhkan oleh pengguna [19]. Dengan XP aplikasi dapat dengan cepat dibangun, karena dengan XP memungkinkan dilakukannya *pair programming*, dimana kegiatan pengembangan perangkat lunak dikerjakan oleh tim dalam waktu yang bersamaan. Selain itu XP memiliki kelebihan yaitu mampu menyederhanakan tiap-tiap tahapannya, sehingga pembangunan *software* lebih fleksibel dan adaptif [20]. Gambar 1 menggambarkan tahapan XP yang akan dilakukan:



Gambar 1. Tahapan Penelitian Dengan Pendekatan XP

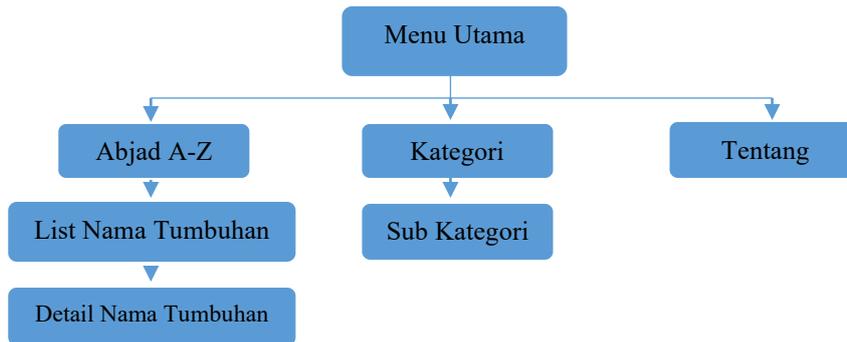
### 3.1. Planning (Perencanaan)

Tahapan dalam XP diawali dengan *planning* (perencanaan) dengan memahami proses dan mendeskripsikan fasilitas serta fungsi setiap fasilitasnya. Pada tahapan ini pada intinya berfokus pada penentuan fungsionalitas dari sistem yang akan dikembangkan. Kebutuhan fungsional merupakan pernyataan tentang fasilitas atau fitur terhadap masukan tertentu dan di situasi tertentu [21]. Fungsionalitas didapatkan berdasarkan identifikasi masalah dengan tujuan agar dapat diketahui permasalahan dan kendala yang dialami pengguna [22]. Untuk itu, sebelum menyusun kebutuhan fungsional, peneliti melakukan analisa terhadap permasalahan mengenai penggunaan kamus Bahasa ilmiah dengan melakukan wawancara di SMA Tri Sukses Natar Lampung, dengan sampel responden lima orang Guru dan sepuluh Siswa. Dari permasalahan yang telah didapatkan maka, kebutuhan fungsional dari kamus Bahasa ilmiah untuk tumbuhan yang akan dikembangkan diantaranya:

1. Sistem dapat menampilkan daftar tumbuhan berdasarkan abjad dari A sampai dengan Z.
2. Sistem dapat menampilkan bahasa ilmiah dari nama ilmiahnya, kategori dan jenisnya.
3. Sistem dapat menampilkan kategori yang berisi: perkebunan, tanaman pangan dan hortikultura.
4. Sistem dapat melakukan pencarian tumbuhan yang akan dicari oleh pengguna.

### 3.2. Design (Perancangan)

Berikutnya tahapan *design* (perancangan), dimana pada tahap ini akan dilakukan pemodelan guna memvisualkan sistem agar mudah untuk dipahami. Perancangan yg dilakukan dengan merancang struktur menu aplikasi. Struktur menu aplikasi merupakan rancangan dari bagian-bagian dari menu aplikasi yang akan dibangun dan untuk mengetahui bagian mana yang terlebih dahulu nantinya yang akan diakses setelah aplikasi tersebut selesai dibuat. Gambar 2 berikut ini merupakan perancangan arsitektur menu dari aplikasi yang akan dibangun.



Gambar 2. Arsitektur Menu Kamus Bahasa Ilmiah Tumbuhan

### 3.3. Coding (Pengkodean)

Tahapan selanjutnya adalah *coding* (pengkodean), dimana pada tahapan ini rancangan yang telah dibuat diimplementasikan kedalam bahasa pemrograman dengan menggunakan *compiler* untuk membuat aplikasi [21]. Aplikasi kamus Bahasa ilmiah tumbuhan yang dikembangkan dibangun dengan *text editor* Adobe Dreamweaver CS6, notepad++ dan XAMPP yang digunakan untuk membuat server lokal dan MySQL sebagai databasenya melalui fasilitas phpMyAdmin. Sedangkan bahasa pemrograman menggunakan PHP, CSS dan JQuery.

### 3.4. Test (Pengujian)

Tahapan terakhir adalah tahapan *testing* atau pengujian. Sebelum sistem digunakan maka alangkah baiknya sistem terlebih dahulu diuji agar sistem dapat bekerja sesuai dengan keinginan pengguna [23]. Pada penelitian pengujian dilakukan dengan pengujian kecepatan algoritma *sequential search* untuk melakukan pencarian. Pengujian kecepatan menggunakan *page load time*, yaitu lamanya waktu yang diperlukan untuk menampilkan semua konten dari aplikasi [24]. Untuk menghitung kecepatan *page load time* pada penelitian ini menggunakan fungsi *microtime*, yaitu fungsi PHP yang digunakan untuk mendapatkan nilai *timestamp* saat ini dengan menampilkan waktu hingga mikro detik.

## 4. Hasil dan Diskusi

Aplikasi kamus Bahasa ilmiah tumbuhan yang dikembangkan untuk algoritma pencariannya menggunakan *sequential searching*, dimana pada algoritma ini akan melakukan pencarian secara beruntun yang diawali dari elemen awal sampai dengan elemen akhir. Misalkan, terdapat data pencarian tumbuhan sebagai berikut:

$A \leftarrow [\text{anggrek, padi, cempaka, alpukat, semangka}] \rightarrow i=1 \text{ s/d } n$

Apabila dicari pada array tersebut dengan data alpukat.

Sehingga proses dalam pencarian data tersebut adalah:

$B = \text{alpukat}$

$\text{ditemukan} \leftarrow \text{false}$

$i = 1$

$A[1] \diamond x \{ \text{ditemukan} \leftarrow \text{false} \}$

$i = 1 + 1 = 2$

$A[2] \diamond x \{ \text{ditemukan} \leftarrow \text{false} \}$

$i = 2 + 1 = 3$

$A[3] \diamond x \{ \text{ditemukan} \leftarrow \text{false} \}$

$i = 3 + 1 = 4$

$A[4] \diamond x \{ \text{ditemukan} \leftarrow \text{false} \}$

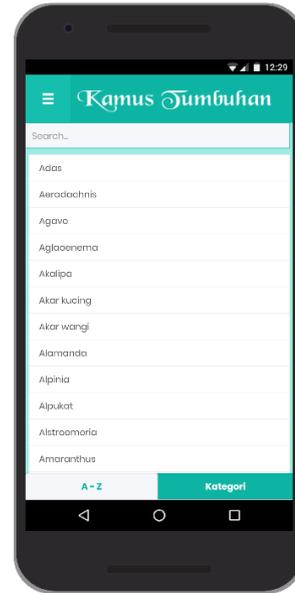
$A[4] = x \{ \text{ditemukan} \leftarrow \text{true} \}$

Berdasarkan contoh tersebut ditemukan kata alpukat pada pencarian data pada posisi data ke-4.

Aplikasi Kamus Bahasa Ilmiah Tumbuhan ini di implementasikan dengan *text editor* Adobe Dreamweaver CS6, notepad++ dan XAMPP yang digunakan untuk membuat server lokal dan MySQL sebagai databasenya melalui fasilitas phpMyAdmin. Sedangkan bahasa pemrograman menggunakan PHP, CSS dan JQuery. Aplikasi ini diawali dengan menu utama, di mana menu ini fitur awal yang berisi pilihan fitur-fitur berikutnya. Halaman menu utama terdapat menu abjad a-z, kategori dan tentang. Visual hasil implementasi menu utama terlihat pada Gambar 3. Pengguna dapat memilih kamus berdasarkan abjad a-z terdapat urutan index kamus yang dapat dilihat dan dipilih oleh pengguna. Pengguna dapat mencari nama tumbuhan yang diinginkan. Hasil implementasi halaman abad a-z terlihat pada Gambar 4.

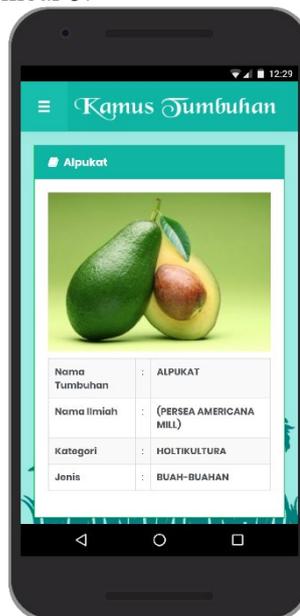


Gambar 3. Tampilan Menu Utama



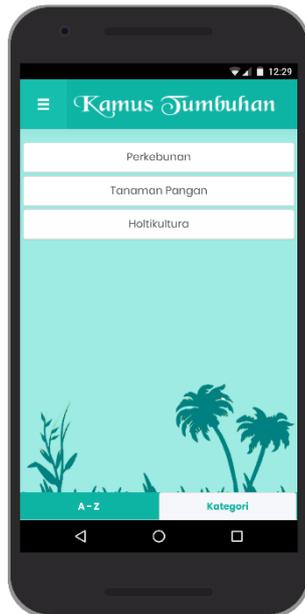
Gambar 4. Tampilan Halaman Abjad a-z

Setelah memilih tumbuhan yang akan diketahui artinya dalam bahasa ilmiah maka akan tampil halaman nama tumbuhan yang terdapat keterangan gambar tumbuhan, nama tumbuhan, nama ilmiah tumbuhan, kategori tumbuhan dan jenis tumbuhan. Hasil implementasi halaman nama tumbuhan dapat dilihat pada Gambar 5.

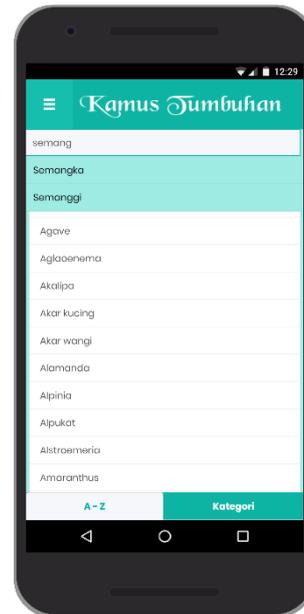


Gambar 5. Tampilan Halaman Nama Tumbuhan

Berikutnya pengguna dapat membuka kategori, dimana pada menu ini terdapat menu-menu perkebunan, tanaman pangan dan hortikultura. Hasil implementasi menu kategori ditunjukkan pada Gambar 6. Halaman menu kategori tanaman pangan terdapat menu padi, palawija, kacang dan umbi. Pengguna dapat dengan mudah mencari tanaman sesuai dengan sub kategori yang ada pada kategori tanaman pangan. Sedangkan halaman menu kategori hortikultura terdapat menu buah-buahan, sayuran, biofarmaka (tanaman obat), dan tanaman hias. Pengguna dapat dengan mudah mencari tanaman sesuai dengan sub kategori yang ada pada kategori hortikultura. Kemudian untuk melakukan pencarian, pengguna dapat menggunakan halaman pencarian yang terdapat kolom untuk menginputkan nama tumbuhan yang akan dicari (Gambar 7).



Gambar 6. Tampilan Menu Kategori



Gambar 7. Tampilan Pencarian

Pencarian ini menggunakan algoritma atau metode *sequential search* yang bertujuan agar pengguna dapat menemukan nama tumbuhan yang diinginkan melalui pencarian yang ada pada aplikasi kamus. Algoritma *sequential searching* pendekatan pencarian secara beruntun, dimana melakukan pencarian dengan melakukan perbandingan antara data yang ada dengan data yang akan dicari satu per satu secara berurutan sampai dapat menemukan datanya. Gambar 8 menunjukkan *source code* yang digunakan pada *sequential search*.

```

$array = array();
while ($row=mysqli_fetch_assoc($result))
{
    $array[] = $row;
}
$x = $_POST['ind'];
$data = explode(" ", $x);
$jumlah_kata = (integer)count($data);
for ($j=0; $j < count($data); $j++) {
    for ($i = 0; $i <= count($array); $i++) {
        if ($array[$i]['cari'] == $data[$j]) {
            echo $h = $array[$i]['kata_dicari']. "(Position : ". $i . ")";
        }
    }
}
}

```

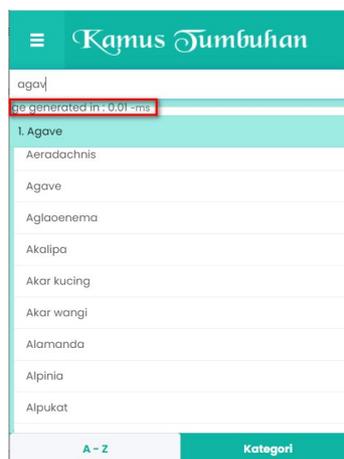
Gambar 8. Source Code yang Digunakan pada Sequential Search

Source code tersebut memperlihatkan bahwa pencocokkan data diawali dari data pertama hingga data terakhir, data yang dilakukan pencarian kemudian dibandingkan dengan setiap datanya pada array. Pencarian dilakukan dengan membandingkan seluruh data pada array sampai selesai. Apabila data yang akan dilakukan pencarian telah didapatkan maka proses membandingkan elemen *array* akan dihentikan.

Selanjutnya dilakukan pengujian kecepatan dari hasil pencarian yang telah dilakukan oleh algoritma *sequential search*. Pengujian kecepatan dilakukan dengan melihat *page load time* dengan melihat pada fungsi *microtime*. Fungsi *microtime* berfungsi untuk menghitung waktu yang dibutuhkan ketika *page* tersebut memproses pencarian. Penggunaan fungsi *microtime* ditunjukkan pada Gambar 9. Sementara, Gambar 10 menunjukkan hasil pengujian kecepatan *page load* untuk pencarian dengan fungsi *microtime*.

```
$start = microtime(true);
$finish = microtime(true);
print 'Page generated in :
'. round(($finish - $start) * 1000, 2) .' <small>-ms</small>';
```

Gambar 9. Fungsi *Microtime*



Gambar 10. Pengujian Kecepatan Dengan Fungsi *Microtime* pada *Load Page* Pencarian

Berdasarkan pengujian tersebut, *sequential search* dapat menemukan kata yang dicari secara *real time*, karena waktu yang dibutuhkan rata-rata 0,01 ms. Akan tetapi, pada kasus ini kata yang akan dicari sudah berurutan secara abjad sehingga mempermudah algoritma bekerja. Selain itu data Bahasa ilmiah tumbuhan yang ada pada *database* sejumlah 484 kata, jumlah ini menunjukkan data relatif terbatas atau tidak terlalu banyak. Dari hasil pengujian, terlihat bahwa algoritma *sequential search* memiliki kelebihan yaitu: 1) Pencarian dapat dilakukan dengan cepat untuk jumlah data yang terbatas atau tidak terlalu banyak; 2) *Sequential search* merupakan algoritma yang sederhana dan tidak rumit. Jika diperhatikan, algoritma *sequential search* melakukan pencarian dimana akan dicari satu per satu secara berurutan, hal ini menyebabkan jika data yang dicari terletak dibelakang atau paling akhir, maka akan membutuhkan waktu yang lama dalam proses pencariannya. Selain itu perlu dilakukan uji coba penggunaan algoritma *sequential search* pada sejumlah data yang besar.

## 5. Kesimpulan dan Saran

### 5.1. Kesimpulan

Berdasarkan hasil penelitian dan pembahasan, maka dapat disimpulkan bahwa penelitian ini menghasilkan aplikasi kamus bahasa ilmiah tumbuhan berbasis Android yang dapat digunakan oleh siswa dan mahasiswa serta masyarakat, sebagai sarana belajar dan alternatif untuk memperluas pengetahuan pengguna dalam mengenal nama-nama ilmiah yang terdapat pada setiap tumbuhan. Aplikasi kamus ini terdapat menu abjad a-z sehingga dapat

menampilkan nama-nama tumbuhan mulai dari abjad a sampai dengan abjad z, serta dilengkapi fitur pencarian yang dapat mempermudah pengguna dalam mencari nama-nama tumbuhan. Selain itu terdapat fitur kategori mulai dari perkebunan, tanaman pangan dan hortikultura. Algoritma pencarian yang digunakan di dalam aplikasi Kamus Bahasa Ilmiah Tumbuhan ini adalah algoritma *sequential search* dimana pencarian dilakukan secara beruntun yang digunakan dalam pencarian data, baik pada *array* yang berurutan ataupun tidak berurutan. Pencarian ini dapat digunakan dengan memasukkan nama tumbuhan yang akan dicari, setelah pengguna menemukan nama tumbuhan yang dicari maka pengguna dapat melihat gambar, nama ilmiah, kategori dan jenis pada tumbuhan tersebut. Berdasarkan hasil pengujian kecepatan pencarian berdasarkan *page load time* dengan fungsi *microtime* menunjukkan bahwa algoritma *sequential search* dapat melakukan pencarian dengan cepat untuk data yang relatif terbatas atau tidak terlalu banyak.

## 5.2. Saran

Tentu saja penelitian yang dilakukan tidak sempurna, ada beberapa hal yang perlu diperbaiki untuk penelitian yang akan datang. Maka, saran yang perlu diperhatikan dalam mengembangkan aplikasi selanjutnya adalah (1) Pada pengembangan selanjutnya perlu dikembangkan dengan menambahkan nama-nama tumbuhan beserta nama ilmiah, gambar, kategori dan jenis tumbuhan tersebut, (2) Penambahan fitur pencarian menggunakan suara, (3) Penambahan fitur atau menu bagi pengguna untuk menambahkan nama tumbuhan ke dalam aplikasi kamus bahasa ilmiah tumbuhan, (4) Perlu dilakukan pengujian terhadap algoritma *sequential search* dengan data yang besar dan efektifitas komputasinya.

## Referensi

- [1] M. Akbar and Y. Rahmanto, "Desain data warehouse penjualan menggunakan Nine Step Methodology untuk business intelligence pada PT Bangun Mitra Makmur," *J. Inform. dan Rekayasa Perangkat Lunak*, vol. 1, no. 2, pp. 137–146, Apr. 2021, doi:10.33365/jatika.v1i2.331
- [2] Kompas.com, "Trafik Internet di 2019 Meningkat, Perangkat Mobile Jadi Penyumbang Terbesar," *kompas.com*, 2020. [Online]. Available: <https://tekno.kompas.com/read/2020/02/19/15070077/trafik-internet-di-2019-meningkat-perangkat-mobile-jadi-penyumbang-terbesar?page=all>.
- [3] S. Surahman and E. B. Setiawan, "Aplikasi mobile driver online berbasis Android untuk perusahaan rental kendaraan," *J. Ultim. InfoSys*, vol. 8, no. 1, pp. 35–42, Aug. 2017, doi:10.31937/si.v8i1.554
- [4] Y. Rahmanto, M. F. Randhika, F. Ulum, and B. Priyopradono, "Aplikasi pembelajaran audit sistem informasi dan tata kelola teknologi informasi berbasis Mobile," *J. TEKNOKOMPAK*, vol. 14, no. 2, pp. 62–67, Aug. 2020, doi:10.33365/jtk.v14i2.723
- [5] A. Amri, S. Sarkum, and I. Purnama, "Media pembelajaran kamus bahasa latin berbasis Android," *InfoTekJar (Jurnal Nas. Inform. dan Teknol. Jaringan)*, vol. 3, no. 1, pp. 60–64, Sept. 2018, doi:10.30743/infotekjar.v3i1.560
- [6] A. S. Sugiharto, "Pengembangan aplikasi kamus nama latin biologi sebagai media belajar siswa SMA Negeri 01 Ngaglik berbasis Android," *J. Elektron. Pendidik. Tek. Inform.*, vol. 7, no. 1, pp. 32–39, 2018.
- [7] A. Anofrizen, "Perancangan sistem pencarian nama latin tumbuhan berbasis Website menggunakan Algoritma Knuth Morris Pratt (KMP)," *J. Sains, Teknol. dan Ind.*, vol. 17, no. 2, pp. 59–64, Jun. 2020.
- [8] N. D. Yohana and D. Purnomo, "Aplikasi kamus hewan dan tumbuhan berbasis Android," *JOINTECS (Journal Inf. Technol. Comput. Sci.)*, vol. 2, no. 1, pp. 1–4, Jan. 2017.
- [9] R. I. Borman and A. Pratama, "Penerapan String Matching dengan Algoritma Boyer Moore pada aplikasi Font Italic untuk deteksi kata asing," *J. Teknoinfo*, vol. 10, no. 2, pp. 1–5, Jul. 2016.

- [10] A. A. Rismayadi and L. Jamaliah, "Implementasi Algoritma Sequential Searching pada aplikasi E-Office," *Naratif(Jurnal Nasional, Riset, Apl. dan Tek. Inform.*, vol. 1, no. 1, pp. 29–34, Feb. 2019.
- [11] S. Suhartini, M. Muchlis, and R. P. Lestari, "Implementation of Sequential Search method on Android-based Jakabaring dictionary," *J. Transform.*, vol. 16, no. 1, p. 74, Jul. 2018, doi: <http://dx.doi.org/10.26623/transformatika.v16i1.830>
- [12] H. Salim, F. Marisa, and I. D. Wijaya, "Aplikasi kamus istilah neurology berbasis Mobile menggunakan metode Sequential Search," *J. Inform. Merdeka Pasuruan*, vol. 3, no. 1, pp. 62–70, Mar. 2018, doi: <http://dx.doi.org/10.37438/jimp.v3i1.90>
- [13] K. Yuliawan, "Algoritma Sequential Search dalam kamus bahasa inawatan berbasis Android," *J. Mediat. J. Media Pendidik. Tek. Inform. dan Komput.*, vol. 4, no. 1, pp. 35–37, Jan. 2021.
- [14] G. Gunawan, "Aplikasi kamus istilah ekonomi (inggris-indonesia) menggunakan metode Sequential Searching," *Pseudocode*, vol. III, no. 2, pp. 122–128, Sept. 2016, doi:10.33369/pseudocode.3.2.122-128
- [15] I. P. G. P. Damayanto, F. S. Fastanti, and S. H. Dalimunthe, "Pemanfaatan portal basis data daring dalam validasi nama ilmiah jenis dan suku tumbuhan," *Berk. Ilmu Perpust. dan Inf.*, vol. 16, no. 2, pp. 170–183, Dec. 2020, doi: <https://doi.org/10.22146/bip.v16i2.770>
- [16] P. Wahyuningsih and N. Mustika, "Sistem informasi lokasi sejarah pahlawan Sulawesi Selatan menggunakan Algoritma Sequential Search berbasis Android," *Electro Luceat*, vol. 6, no. 2, pp. 266–273, Nov. 2020.
- [17] N. Chafid and S. Alfian, "Penerapan Algoritma Pencarian Sequential pada aplikasi kamus tiga Bahasa Indonesia-Jawa-Jawa Banten," *J. Satya Inform. Vol. 4 No. 1, Mei 2019 Hal.*, vol. 4, no. 1, pp. 53–62, May 2019.
- [18] R. I. Borman, A. T. Priandika, and A. R. Edison, "Implementasi metode pengembangan sistem Extreme Programming (XP) pada aplikasi investasi peternakan," *JUSTIN (Jurnal Sist. dan Teknol. Informasi)*, vol. 8, no. 3, pp. 272–277, Jul. 2020.
- [19] I. Ahmad, R. I. Borman, J. Fakhurozi, and G. G. Caksana, "Software development dengan Extreme Programming (XP) pada aplikasi deteksi kemiripan judul skripsi berbasis Android," *J. Invotek Polbeng - Seri Inform.*, vol. 5, no. 2, pp. 297–307, Nov. 2020, doi: <https://doi.org/10.35314/isi.v5i2.1654>
- [20] N. Nugroho, Y. Rahmanto, R. Rusliyawati, D. Alita, and H. Handika, "Software development sistem informasi kursus mengemudi (kasus : kursus mengemudi Widi Mandiri)," *J. Sains Komput. Inform.*, vol. 5, no. 1, pp. 328–336, Mar. 2021.
- [21] M. Melinda, R. I. Borman, and E. R. Susanto, "Rancang bangun sistem informasi publik berbasis Web (studi kasus : Desa Durian Kecamatan Padang Cermin Kabupaten Pesawaran)," *J. Tekno Kompak*, vol. 11, no. 1, pp. 1-4, Feb. 2018.
- [22] R. I. Borman, I. Yasin, M. A. P. Darma, I. Ahmad, Y. Fernando, and A. Ambarwari, "Pengembangan dan pendampingan sistem informasi pengolahan pendapatan jasa pada PT. DMS Konsultan Bandar Lampung," *J. Soc. Sci. Technol. Community Serv.*, vol. 1, no. 2, pp. 24–31, Sept. 2020, doi: <https://doi.org/10.33365/jsstcs.v1i2.849>
- [23] A. E. Kumala, R. I. Borman, and P. Prasetyawan, "Sistem informasi monitoring perkembangan sapi di lokasi uji performance (studi kasus : Dinas Peternakan Dan Kesehatan Hewan Provinsi Lampung)," *J. Tekno Kompak*, vol. 12, no. 1, pp. 1-9, Feb. 2018, doi: <https://doi.org/10.33365/jtk.v12i1.52>
- [24] S. Suliman, "Analisis performa Website Universitas Teuku Umar dan Universitas Samudera menggunakan Pingdom Tools dan Gtmetrix," *Simkom*, vol. 5, no. 1, pp. 24–32, Jan. 2020, doi: <https://doi.org/10.51717/simkom.v5i1.47>