

## ANALISIS KINERJA STRUKTUR DATA TRIE UNTUK KAMUS INDONESIA – JAWA

### *The Performance Analysis Of Trie Data Structure For Indonesia – Java Dictionary*

Jesira Enjeliva Br. Girsang<sup>1</sup>, Eko Darwiyanto, S.T., M.T.<sup>2</sup>, Yanuar Firdaus Arie Wibowo, S.T., M.T.<sup>3</sup>  
Program Studi Teknik Informatika, Fakultas Informatika, Universitas Telkom, Bandung  
<sup>1</sup>jejegirsang@student.telkomuniversity.ac.id, <sup>2</sup>ekodarwiyanto@telkomuniversity.ac.id,  
<sup>3</sup>yanuar@telkomuniversity.ac.id

---

#### Abstrak

Pada saat ini, kamus di pasaran sudah banyak bentuknya. Salah satunya adalah kamus elektronik. Kamus elektronik dibagi menjadi tiga jenis, yaitu *software* kamus, kamus *website* atau kamus *online*, dan kamus *mobile*. Seringkali orang memiliki *laptop* atau *PC*, tetapi belum tentu memiliki telepon seluler yang mumpuni. Atau mungkin saja orang memiliki telepon seluler, tetapi, tidak ingin memberatkan telepon seluler tersebut dengan aplikasi kamus berbasis *android* yang akan membuat performa telepon seluler tersebut menjadi menurun. Maka dari itu, diciptakanlah *software* kamus berbasis *desktop* yang dapat digunakan *offline* atau tanpa membutuhkan internet dalam penggunaannya. Ada dua struktur data yang biasanya digunakan dalam pembuatan *software* kamus berbasis *desktop* yang menggunakan *python* dalam pembuatannya, yaitu struktur data trie dan struktur data dafsa. Dengan membandingkan kinerja kedua struktur tersebut, didapatkanlah bahwa struktur data trie seringkali lambat dalam mencari arti kata dan memakan *memory* yang lebih banyak daripada dengan menggunakan struktur data dafsa.

**Kata kunci** : Kamus Berbasis *Desktop*, Struktur Data Trie, Struktur Data Dafsa, *Python*.

---

#### Abstract

*At this time, the dictionary on the market has many forms. One of them is an electronic dictionaries. Electronic dictionaries are divided into three types, namely software dictionaries, website dictionaries or online dictionaries, and mobile dictionaries. People often have a laptop or PC, but they don't necessarily have a capable cell phone. Or maybe people have cell phones, but, they don't want to load their cellphones with an android-based dictionary application that will reduce the performance of these cellphones. Therefore, a desktop-based dictionary software was created that can be used offline or without the need for internet usage. There are two data structures that are usually used in making desktop-based dictionary software that uses python in its creation, namely the trie data structure and the dafsa data structure. By comparing the performance of these two structures, it is found that the trie data structure is often slow in finding the meaning of words and consuming more memory than using the dafsa data structure.*

**Key words** : *Desktop-based Dictionary, Trie Data Structure, Dafsa Data Structure, Python.*

---

#### 1. Pendahuluan

Kamus elektronik merupakan salah satu kamus yang banyak digunakan pada saat ini. Ada 3 jenis kamus elektronik yang dikenal masyarakat, yaitu kamus dalam bentuk *software*, kamus *website*, dan kamus *mobile*. Kamus dalam bentuk *software* biasanya harus *install* terlebih dahulu pada *PC* atau *laptop* untuk menggunakannya. Seiring perkembangannya, kamus ini telah dapat digunakan tanpa proses *install* terlebih dahulu. Dengan menggunakan *flashdisk* yang tertancap ke *laptop* atau *PC*, *software* ini dapat digunakan. Sementara kamus *website* memerlukan koneksi dengan internet dalam penggunaannya. Sehingga kamus *website* sering juga disebut kamus *online*. Sedangkan kamus *mobile* adalah kamus yang penggunaannya terbatas hanya pada telepon seluler. Dan beberapa kamus *mobile* seringkali menggunakan internet dengan kuota yang tidak sedikit, bahkan memakan *memory* telepon seluler sehingga sering membuat performa telepon seluler menjadi lambat.

Maka dari itu, topik dari jurnal ini menggunakan kamus dalam bentuk *software*. Karena kamus dalam bentuk ini tidak memerlukan koneksi internet dalam penggunaannya. Dan *memory* yang digunakan juga sedikit. Kamus *software* juga diharapkan dapat membantu masyarakat yang tidak memiliki telepon seluler yang mumpuni. Pembuatan kamus menggabungkan struktur data trie dan struktur data dafsa. Fungsinya adalah untuk membandingkan kinerja keduanya.

Aplikasi yang dibuat dan dianalisis disini adalah aplikasi yang dibuat menggunakan bahasa pemrograman *python*. Memiliki kosakata yang terbatas, aplikasi ini diharapkan dapat dikembangkan nantinya. Aplikasi ini hanya akan menerjemahkan kata, bukan kalimat. Aplikasi ini hanya menggunakan struktur data trie dan struktur data dafsa.

Adapun tujuan dan manfaat dari penelitian ini adalah merancang dan mengimplementasikan serta menganalisis struktur data tertentu dalam suatu kamus elektronik dalam bentuk *software* berbasis *desktop*. Diharapkan dapat digunakan untuk mempermudah mencari kosakata hasil terjemahan bahasa Indonesia ke bahasa Jawa agar bisa memberikan kemudahan bagi segala usia pemakainya. Dan proses analisis disini berfungsi untuk melihat kinerja kedua struktur data dan didapatkan bahwa struktur data tertentu memiliki beberapa keunggulan dan kelemahan.

## 2. Studi Terkait

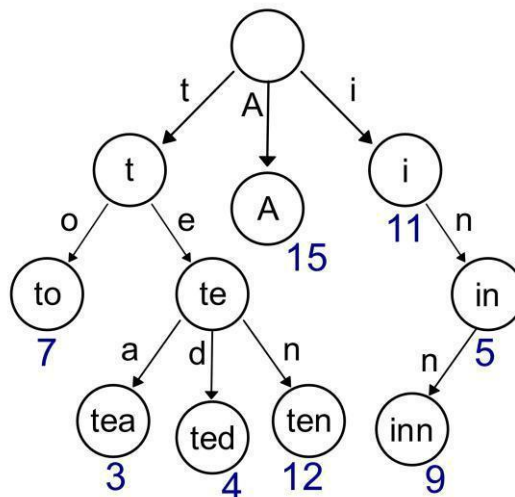
Penelitian oleh Kurniawan Dwi Ely, Reinhard Simon dan Irwansyah pada tahun 2013 dengan judul Aplikasi Kamus Aneka Bahasa Daerah Berbasis Smartphone Android. Melibatkan lima bahasa daerah di Indonesia diantaranya bahasa Melayu, Minang, Jawa, Sunda dan Batak. Jhoni Pranata Sembiring tahun 2013 melakukan penelitian berjudul Perancangan Aplikasi Kamus Bahasa Indonesia – Karo Online Berbasis Web Dengan Metode Sequential Search. Penelitian ini menerapkan metode Sequential Search untuk proses pencarian yang lebih cepat. Pada tahun 2012, Ibnu Priharyanto melakukan penelitian berjudul Aplikasi Kamus Bahasa Indonesia – Jawa – Jawa Krama Berbasis Android. Aplikasi ini menampilkan arti dalam bahasa Jawa, Jawa Krama beserta Aksara Jawa[1].

Kamus adalah sejenis buku rujukan yang menerangkan arti kata. Selain menerangkan definisi kata, kamus juga mempunyai pedoman pelafalan, asal usul kata dan juga contoh penggunaan kata. Kata kamus sendiri diserap dari bahasa Arab *qamus* ( قاموس ), dengan bentuk jamaknya adalah *qawamis*. Kata Arab itu sendirinya berasal dari kata dalam bahasa Yunani *Ωκεανός* (*oceanos*) yang berarti samudra[2]. Ada 3 bentuk dasar yang paling utama dalam bahasa Jawa menurut undhakundhuk basa : Basa Ngoko (kasar), Basa Madya (Biasa), Basa Krama (Halus)[3].

Python adalah sejenis bahasa pemrograman sama seperti C, C++, Pascal, atau bahkan java. Python memiliki ciri tersendiri yang membuatnya beda dari bahasa pemrograman lainnya. Tadinya, motivasi dalam membuat python adalah untuk digunakan dalam bahasa skrip tingkat tinggi. Python termasuk freeware yang lengkap dengan source code, debugger, dan profiler, antarmuka yang mendukung fungsi sistem, GUI ( antarmuka pengguna grafis) dan basis datanya[4].

Ada banyak algoritma dan struktur data yang sering digunakan untuk merepresentasikan dan mengolah data dalam bentuk string. Beberapa dari struktur data tersebut adalah table hash, pohon pencarian biner, dan trie. Kata trie berasal dari kata retrieval yang artinya adalah pengambilan. Istilah ini diciptakan oleh Edward Fredkin, setelah percobaan pertama dimulai oleh René de la Briandais pada tahun 1959. Yang membuat trie unggul dari struktur data lain adalah dalam hal memori dan kecepatan yang digunakan trie dalam memecahkan kasus yang sama[5].

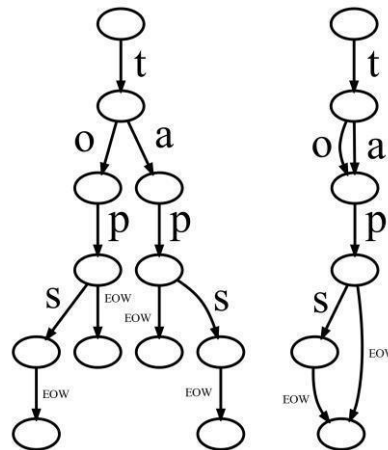
Trie adalah struktur data berupa pohon terurut untuk menyimpan suatu himpunan string yang mengandung awalan yang sama. Awalan yang dimaksud disini adalah prefix. Itu alasan dari trie juga disebut sebagai pohon prefix. Pada trie, tidak ada simpul yang menyimpan kunci yang terkait dengan simpul itu sendiri. Yang ada adalah posisinya pada pohon yang menentukan kunci yang terkait dengannya. Setiap keturunan dari sebuah simpul mempunyai prefix yang sama dengan string yang diwakilkannya dan string kosong adalah akar yang menandai simpul tersebut.



Gambar 2.1. Penggunaan trie

Dari gambar di atas, seharusnya bisa didapatkan gambaran tentang kompleksitas waktu yang diperlukan dalam mencari sebuah kata dalam trie. Jika kata terpanjang dimisalkan dengan  $J$ , maka memerlukan waktu terburuk  $O(J)$  untuk mencari sebuah kata dalam trie.

Dalam ilmu computer, dikenal juga struktur data DAFSA ( deterministic acyclic finite state automata ) yang berfungsi untuk mengatasi suffix. Jika dibandingkan dengan cara pencarian kata dalam trie, DAFSA memakan waktu yang cukup lama. Misalnya, kata dalam bahasa inggris “tap”, “taps”, “top”, dan “tops”. Trie membutuhkan 11 simpul untuk keempat kata tersebut. Sedangkan DAFSA hanya membutuhkan 6 simpul. Trie sendiri memang memiliki lebih banyak simpul. Tetapi, kemampuan trie dalam menghilangkan awalan dari setiap kata akan memangkas memory yang digunakan dalam proses yang dilakukan. Sementara DAFSA membutuhkan tambahan memory yang digunakan[6].



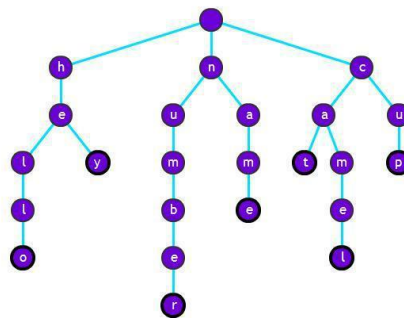
Gambar 2.2. Perbandingan trie dan dafsa

Implementasi kamus pada struktur data trie sendiri dapat dilihat pada menu *find* [7].  
 Find (node, key)

```

    for each char in key
        if char not in node.children return None
        else
            node = node.children[char] return node.value
    
```

Berikut struktur kamusnya dari implementasi coding tersebut.

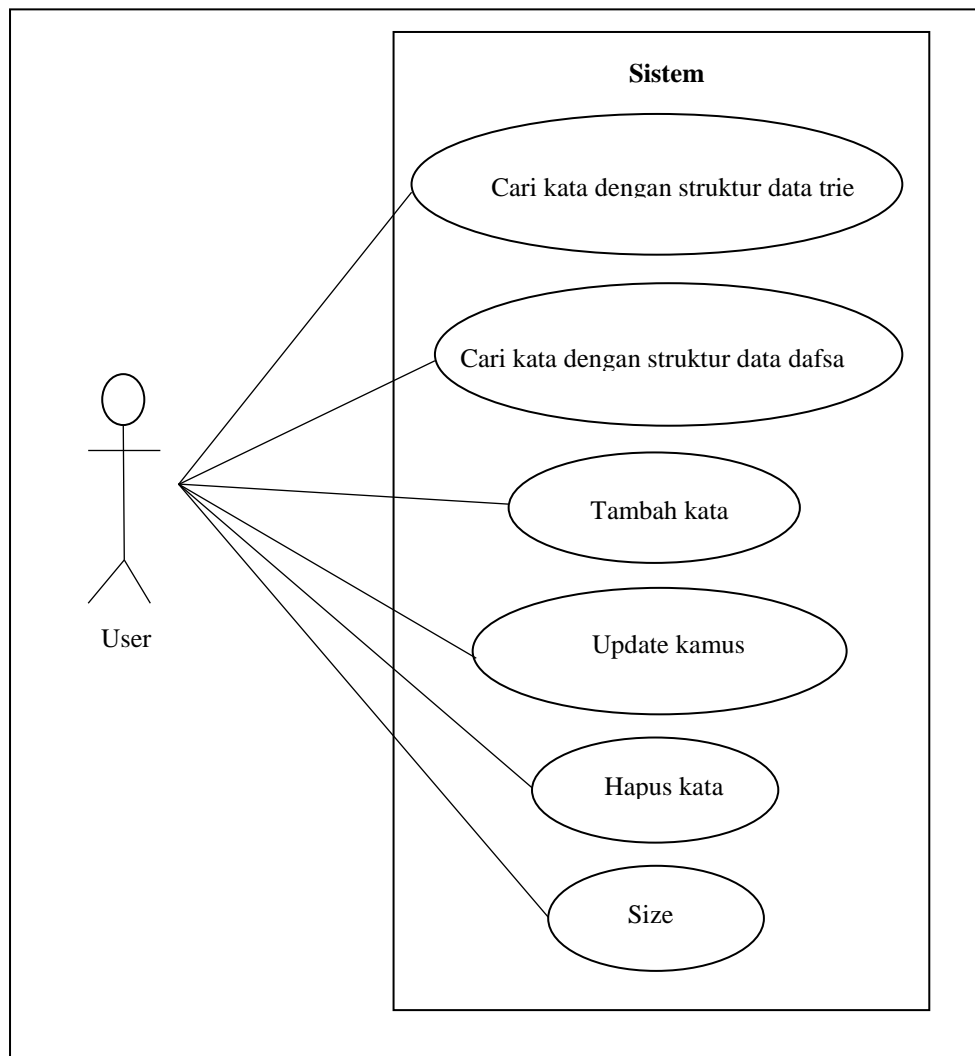


Gambar 2.3. Contoh implementasi coding struktur data trie

### 3. Sistem yang Dibangun

#### 3.1 Use Case

Sistem yang dibangun pada penelitian ini adalah sistem aplikasi kamus *software* yang dapat digunakan untuk mencari kata dengan struktur data trie dan struktur data dafsa. User maupun admin juga dapat menambahkan kata supaya bisa dimasukkan ke dalam kumpulan kata. Untuk mengganti terjemahan kata yang salah, peneliti juga memasukkan fungsi *update*, sehingga data yang salah bisa diganti langsung oleh pengguna. Setiap hasil waktu yang digunakan dalam memasukkan, menghapus, bahkan mengupdate data adalah yang menjadi parameter yang diukur dan dianalisis dalam penelitian.



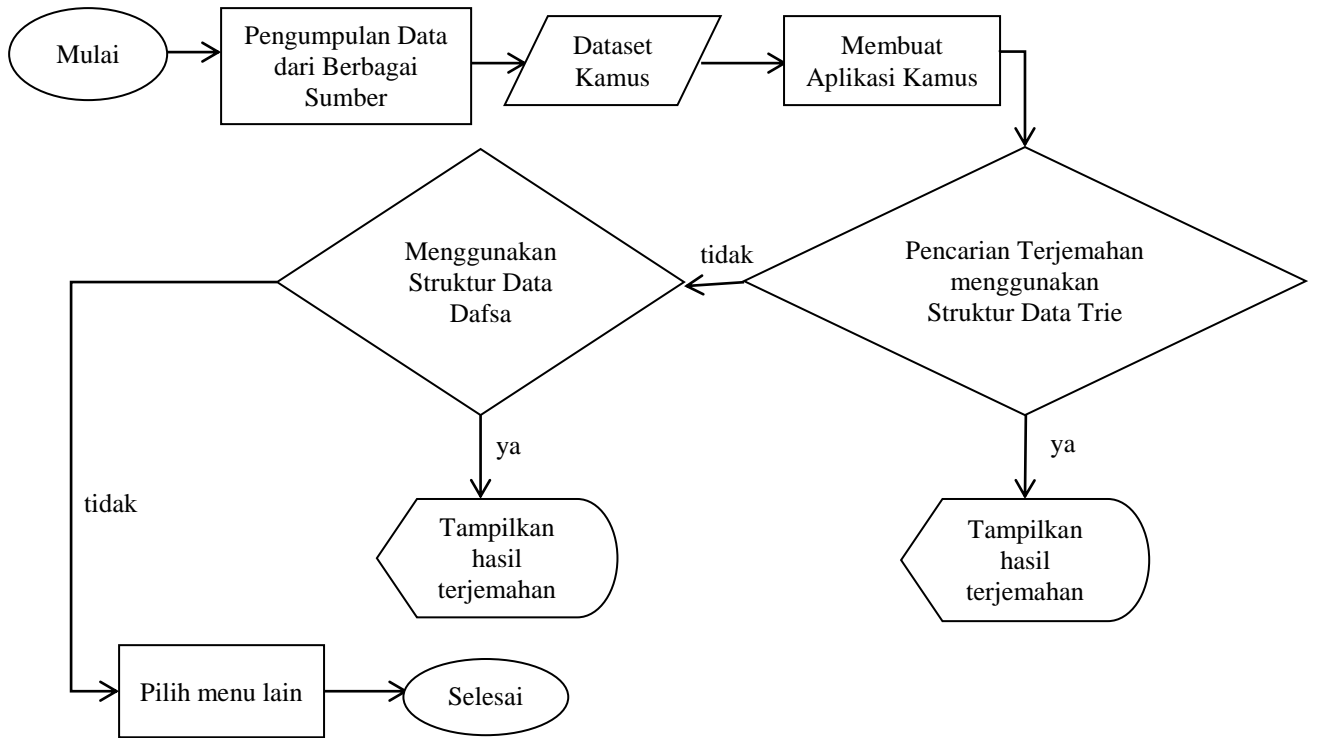
**Gambar 3.1. Use Case Diagram Sistem**

### 3.2 Tahapan Penelitian

Adapun tahap pengerjaan dari awal sampai akhir peneliti dapat dijelaskan sebagai berikut.

1. Peneliti mengumpulkan data awal yang masih berupa kosakata dari beberapa kamus
2. Kata-kata tersebut dipilih dan diolah untuk dibentuk sebagai *dataset* kamus.
3. Setelah *dataset* kamus jadi, dibentuklah aplikasi kamus.
4. Di dalam aplikasi kamus terdapat menu pencarian, tambah, update, dan hapus kata, serta terdapat menu size memory untuk melihat jumlah memory yang digunakan.
5. Di dalam menu pencarian sendiri disusun dengan 2 struktur data, yakni pencarian menggunakan struktur data trie dan pencarian menggunakan struktur data dafsa.
6. Jika tidak ingin mencari terjemahan kata bahasa Indonesia ke Jawa, kamus ini memiliki menu lain yang bisa digunakan juga untuk menambah terjemahan dalam kamus, mengupdate/mengubah arti terjemahan, menghapus kata, melihat jumlah memory yang digunakan dalam 2 struktur data.

Sementara diagram alir penelitian sendiri dapat dilihat dalam gambar di bawah ini.



Gambar 3.2. Flowchart Penelitian

4. Evaluasi

4.1 Hasil Pengujian

Data dari kamus, tidak stabil, yakni terus ditambah dan akan terus dapat ditambah dengan fungsi tambah kata. Data dari kamus yang saat ini diuji masih sekitar 50 kata dari 150 kata. Hasil analisis dan pengujian yang telah dilakukan menggunakan data sebanyak 50 kata, maka didapat hasil sebagai berikut.

Tabel 1. Hasil pengujian terhadap 50 kata

Daftar Kata	Menggunakan struktur data trie	Menggunakan struktur data dafsa
abaaba	0,000594	0,025266
abad	1,183197	0,002327
abadi	0,002362	0,002680
abah	0,002224	0,002541
abai	0,002276	0,002809
abang	0,002439	0,001321
abdi	0,002413	0,002972
abjad	0,000236	0,025489
abnormal	0,000701	0,001832
absen	0,004239	0,003026
abu	0,002351	0,002716
abuabu	0,002906	0,002184
acak	0,002362	0,002312
ada	0,002168	0,002248
adegan	0,002565	0,002133
adem	0,002813	0,002813
adik	0,006461	0,002891
adil	0,001802	0,002537
adinda	0,002532	0,002630
advokat	0,002171	0,001992
agak	0,083457	0,003027
agama	0,002905	0,001335

agar	0,002339	0,002673
agung	0,000282	0,002135
ahli	0,016158	0,002450
ahmak	0,005207	0,002513
aib	0,000989	0,002699
air	0,003257	0,002184
ajal	0,002310	0,002424
bab	0,002864	0,002197
babi	0,002957	0,002584
babon	0,002590	0,002217
babu	0,002415	0,002753
baca	0,002198	0,002690
bacaan	0,003461	0,002445
bacin	0,003100	0,002213
bacok	0,001021	0,002393
badai	0,002884	0,000810
badak	0,002447	0,003050
badan	0,002625	0,002956
badut	0,002606	0,001741
bagai	0,002633	0,002327
bagaimana	0,002276	0,002797
bagi	0,002211	0,002458
bagus	0,002686	0,002367
bahagia	0,002186	0,002895
bahan	0,002611	0,002495
bahasa	0,002620	0,002650
bahaya	0,002341	0,002292
bahenol	0,002919	0,001542

#### 4.2 Analisis Hasil Pengujian

Pengujian ini dilakukan secara manual. Parameter yang diukur dalam pengujian ini adalah ukuran memory yang digunakan untuk menyimpan data, kecepatan dalam mencari data dalam kamus, kecepatan menambah data baru, kecepatan memperbaharui data kamus, dan kecepatan menghapus sebuah data dalam kamus. Dan parameter yang diukur tersebut dapat diwujudkan dalam tabel di bawah.

**Tabel 2. Tabel analisis hasil pengujian**

Parameter yang diukur	Struktur data	
	Trie	Dafsa
Ukuran memory yang digunakan	16534 bytes	2052 bytes
Kecepatan mencari data	0,27987 s	0,003321 s
Kecepatan menambah data		0,04156 s
Kecepatan memperbaharui data		0,00301 s
Kecepatan menghapus data		0,00166 s

Pengujian ini membandingkan kinerja kedua struktur data berdasarkan ukuran memory yang digunakan dan kecepatan mencari data. Sedangkan kecepatan menambah data, memperbaharui dan kecepatan menghapus data tidak dibandingkan karena fokus utama dari penelitian ini adalah struktur data trie. Berdasarkan hasil pengujian pada 50 data kamus, diperoleh hasil ukuran memory dan kecepatan mencari data seperti yang ada dalam Tabel 2 tersebut.

#### 5. Kesimpulan

Penelitian ini menghasilkan sebuah kesimpulan dimana ternyata struktur data trie dan struktur data dafsa memiliki keunggulan dan kelemahan tersendiri. Struktur data trie unggul dalam mencari kata yang memiliki awalan, sementara struktur data dafsa unggul dalam mencari kata dasar tanpa awalan dan dengan akhiran. Struktur data trie sendiri memakan memory yang banyak dalam penggunaannya. Sementara struktur data dafsa lebih hemat karena data yang dimiliki tidak bersifat permanen.

Saran untuk penelitian ini adalah data dalam penelitian ini masih 150 kata. Mungkin bisa ditambah lagi jika ingin dikembangkan nantinya. Penyebab dari keterbatasan data sendiri adalah peneliti kesulitan mencari data

yang pasti. Dan kalau harus membuat manual data kamus, akan memakan waktu yang cukup lama. Dan analisis kinerja struktur data trie ini hanya bisa dilaksanakan dengan memakai *python* 3. Maka dari itu diharapkan, jika ingin melanjutkan topik tentang penelitian ini, diharapkan pengembang selanjutnya mahir dalam menggunakan bahasa pemrograman *python*.

#### Daftar Pustaka

- [1] Heru Nurwarsito, Sutrisno. 2010. Aplikasi Kamus Bahasa Madura-Indonesia-Jawa Online dengan XHTML MP dan WCSS. Jurnal.
- [2] Syahrul Noer Rachmad, Program Pembelajaran Aksara Jepang Hiragana dan Katakana menggunakan Java, UNISBANK Semarang
- [3] Yadi, S. 2009. Kawruh Pepak Basa Jawa Anyar. Surakarta : Pelangi Ilmu
- [4] Sianipar, R.H & Hamzan Wadi. 2015. Pemrograman Python. Bandung : Penerbit Informatika
- [5] <https://en.wikipedia.org/wiki/Trie> [diakses tanggal 12 Desember 2018]
- [6] [https://en.wikipedia.org/wiki/Deterministic\\_acyclic\\_finite\\_state\\_automaton](https://en.wikipedia.org/wiki/Deterministic_acyclic_finite_state_automaton) [diakses tanggal 13 Desember 2018]
- [7] Najogie, Reinhard D. Jurnal. Pengenalan Trie dan Aplikasinya. Bandung : STEI ITB
- [8] Raharjo, Budi. 2016. Kumpulan Solusi Pemrograman Python. Bandung : Penerbit Informatika
- [9] Susanto, Rani & Anna Dara Adriana. Jurnal. Perbandingan Model Waterfall dan Prototyping untuk pengembangan Sistem Informasi. Bandung : UNIKOM
- [10] Raharjo, Budi. 2015. Mudah belajar Python. Bandung : Penerbit Informatika
- [11] Safaat H, Nazruddin. 2014. Android. Revisi Kedua. Bandung : Penerbit Informatika
- [12] Nugroho, Adi. 2010. Rekayasa Perangkat Lunak Berorientasi Objek dengan metode USDP. Yogyakarta : Andi Offset
- [13] Yudhita D.R.K. 2013. Analisis dan Perancangan Aplikasi Mobile Kamus Bahasa Indonesia Berbasis Android. Skripsi. Jogjakarta: STIMIK AMIKOM