



Penerapan Algoritma Smith Pada Aplikasi Pencarian Anime Dalam Kumpulan Anime

Imam Fansuri

Prodi Teknik Informatika, Universitas Budi Darma, Medan, Indonesia

Email: fanzuri02@gmail.com

Abstrak—Perkembangan web sangat maju dari tahun ketahun. Web lebih diminati penggunaannya oleh masyarakat karena beragam fitur dapat ditampilkan untuk memenuhi kebutuhan dan daya tarik tersendiri bagi pengguna. Media dengan jejaring paling luas di dunia (world wide web), tidak terbatas oleh waktu dan tak tersekat oleh batas wilayah. Media yang benar-benar efektif karena tidak pernah berhenti beroperasi dan selalu terbuka untuk dikunjungi setiap orang di seluruh dunia. Untuk sebagian orang anime merupakan hiburan untuk dilihat. Namun terkadang mereka tidak mengetahui semua jenis cerita atau pengarang dari cerita tersebut. Adapun solusi yang dapat dilakukan adalah dengan membuat suatu sistem pencarian anime berbasis web yang dapat memberikan informasi lengkap mengenai anime. Dengan menerapkan algoritma pencarian Smith, dimana cara kerja algoritma Smith adalah dengan melakukan 2 kali proses pencarian yaitu Fase Preprocessing dan Fase Pencarian. Dengan di terapkannya algoritma Smith pada sistem pencarian Anime berbasis web ini, maka diharapkan dapat melakukan proses pencarian anime dengan lebih mudah dan cepat. Sistem yang dibangun tersebut juga sangat praktis.

Kata Kunci: Website, Anime, Algoritma Smith.

Abstract—The development of the web is very advanced from year to year. The public is more interested in using the web because a variety of features can be displayed to meet user needs and appeal. The media with the most extensive network in the world (world wide web), is not limited by time and is not bound by regional boundaries. Media that is truly effective because it never stops operating and is always open to everyone around the world. For some people, anime is an entertainment to be seen. But sometimes they don't know all kinds of stories or the author of the stories. The solution that can be done is to create a web-based anime search system that can provide complete information about anime. By implementing the Smith search algorithm, where the Smith algorithm works is by doing 2 searches, namely the Preprocessing Phase and the Search Phase. By implementing the Smith algorithm in this web-based Anime search system, it is hoped that the search process for anime will be easier and faster. The system built is also very practical.

Keywords: Website, Anime, Smith's Algorithm.

1. PENDAHULUAN

Anime adalah istilah untuk menyebut film animasi atau kartun dari Jepang. Asal katanya dari *animation*, dalam bahasa Jepangnya *animeshon* yang disingkat menjadi Anime. Dan sampai sekarang istilah *Anime* sendiri dipakai untuk membedakan film kartun buatan Jepang dengan yang lain. Kata anime tampil dalam bentuk tulisan dalam tiga karakter katakana a, ni, me (アニメ) yang merupakan bahasa serapan dari bahasa Inggris "*Animation*" dan diucapkan sebagai "*Anime-shon*". Anime (アニメ) (dibaca : a-ni-me, bukan a-nim) adalah animasi khas Jepang yang biasanya dicirikan melalui gambar-gambar berwarna-warni yang menampilkan tokoh-tokoh dalam berbagai macam lokasi dan cerita yang ditujukan pada beragam jenis penonton. Anime dipengaruhi gaya gambar manga, komik khas Jepang.

Adapun masalah yang sering muncul saat melakukan pencarian anime adalah pada waktu mengetikkan sebuah judul tidak menampilkan gambar sampul anime, sehingga user kesulitan untuk mengenali anime yang dicari karena banyak judul anime yang memiliki kesamaan judul. Contohnya akagami karena ada banyak judul yang berawal dari kata akagami sehingga user kesulitan untuk mengenali judul dari anime yang akan dicari.

Untuk mengatasi permasalahan diatas penulis mencoba membuat aplikasi pencarian anime berbasis web yang dapat menampilkan informasi mengenai anime secara detail yang meliputi judul dari anime beserta gambar sampul untuk mengetahui karakter-karakter yang terdapat dalam anime, genre, informasi, sinopsis dan penulis akan memasukkan informasi tentang pengarang dari anime-anime yang ada di dalam aplikasi pencarian anime ini.

2. METODOLOGI PENELITIAN

2.1 Anime

Sejarah Anime di Jepang awalnya sejumlah film animasi oleh *First Experiments in Animation* oleh Shimokawa Bokoten, Koichi Junichi, dan Kitayama Seitaro pada tahun 1913. Lanjut adanya film pendek (cuma durasi sekitar 5 menit) karya Oten Shimokawa yang judulnya *Imokawa Mukuzo Genkanban no Maki* tahun 1917.

Tidak sedikit orang yang pergi ke Jepang untuk belajar mengenai pembuatan anime (dan manga tentunya) karena tertarik setelah melihat berbagai anime yang telah menyebar ke berbagai pelosok dunia di berbagai benua. Adapun pihak yang membuat hasil karya yang serupa atau bahkan mungkin meniru ciri anime, misalnya Korea dan beberapa negara Asia lainnya.

Lama-kelamaan teknologi Anime makin maju dengan adanya anime yang memakai *optic track* (seperti yang digunakan pada masa sekarang) adalah Chikara To Onna No Yononaka (1932) karya Kenzo Masaoka yang memakai



teknologi itu. Di tahun 1943 Masaoka bersama muridnya, Senoo Kosei, bikin lima episode anime berjudul *Momotaro no Umiwashii* (Momotaro, the Sea Eagle). Dimana anime Jepang pertama dengan durasi lebih dari 30 menit.

Teknologi CG (Computer Graphics) dan Teknologi Visual, Komputer dsb telah mempermudah pembuatan anime sekarang ini, karena itu ada yang menganggap bahwa kualitas artistiknya lebih rendah dibandingkan dengan anime masa lalu. Hanya saja perlu diperhatikan bahwa kualitas gambarnya pun sekarang ini lebih nikmat dilihat dan lebih mudah dimengerti karena gambarnya lebih proporsional dan warnanya lebih bagus, ditambah keberadaan teknologi HD [12].

2.2 Algoritma Smith

Algoritma Smith diperkenalkan oleh P. D. Smith pada tahun 1991. Smith memperhatikan bahwa dengan menghitung pergeseran dengan karakter teks persis di samping karakter jendela teks paling kanan terkadang memberikan pergeseran yang lebih pendek daripada menggunakan karakter jendela teks paling kanan. Lalu ia menganjurkan untuk mengambil maksimum antara kedua nilai [9].

Kebanyakan algoritma pencocokan string terdiri dari fase preprocessing dan fase pencarian. Tahap preprocessing menganalisis karakter dalam pola untuk menggunakan informasi ini untuk menentukan pergeseran pola dalam kasus ketidakcocokan atau seluruh pencocokan, dengan tujuan mengurangi jumlah perbandingan karakter, sedangkan fase pencarian mendefinisikan urutan perbandingan karakter dalam setiap upaya antara pola dan teks.

Pada algoritma Smith terbagi atas 2 fase, yaitu :

1. Fase Preprocessing

Fase preprocessing pada algoritma Smith adalah perpaduandari perhitungan fungsi pergeseran bad-character dari algoritma Quick Search dengan fungsi pergeseran bad-character dari algoritma Boyer Moore. Bad-character merupakan kumpulan karakter yang mewakili pola(pattern). Fungsi pergeseran bad-character dari algoritma Boyer Moore disimpan pada tabel $bmBc$, sementara fungsi pergeseran bad-character dari algoritma Quick Search disimpan pada tabel $qsBc$. Kedua fungsi tersebut digunakan pada fase preprocessing algoritma Smith.

2. Fase Pencarian

Pada fase pencarian, jumlah pergeseran berdasarkan perbandingan terbesar nilai $bmBc$ dari karakter yang paling kanan dari jendela teks yang bersesuaian dengan panjang pattern dengan nilai $qsBc$ dari karakter yang $qsBc[x[i]] = m - i$ disebelahnya. Algoritma ini akan tetap mencari dengan bergerak kekanan hingga karakter pada teks berakhir. Tetapi, jika sebelum teks berakhir pattern yang dicari sudah ditemukan, maka jumlah pergeseran pencarian selanjutnya berubah menjadi nilai $qsBc$ dari karakter yang disebelah kanan karakter yang paling kanan dari jendela teks yang bersesuaian dengan panjang pattern. Hingga teks berakhir [9].

Algoritma adalah urutan langkah-langkah logis penyelesaian masalah yang di susun secara sistematis. Berikut ini adalah algoritma aplikasi pencarian anime dalam melakukan proses pencarian anime:

Input : T = Teks

P = *Pattern* atau Pola

Proses : Fase Preprocessing

1. $bmBc$

Jika $P[0] = P[1]$ maka bernilai sama

Jika $P[0] \neq P[1]$ maka diberi nilai $P + 1$

2. $qsBc$

Hitung jumlah karakter $T[0-n]$ disusun sesuai abjad

Jika terdapat kesamaan karakter ditulis 1 kali saja

Lakukan penjumlahan jumlah karakter $m + 1$ untuk nilai semua karakter

Lakukan perhitungan pada karakter P seperti pada rumus di bawah ini

$$qsBc[x[i]] = m - i$$

$$qsBc[x[0]] = 4 - 0 = 4$$

$$qsBc[x[1]] = 4 - 1 = 3$$

$$qsBc[x[2]] = 4 - 2 = 2$$

$$qsBc[x[3]] = 4 - 3 = 1$$

Fase Pencarian

Jika $P[0] = T[0]$ maka dilanjutkan ke karakter selanjutnya dari P dan T sampai tidak memiliki kesamaan dan pergeseran dilakukan sesuai dengan nilai $bmBc$ dan $qsBc$ pada karakter tersebut

Jika $P[0] \neq T[0]$ maka dilakukan pergeseran sesuai dengan nilai $bmBc$ dari karakter P yg memiliki kesamaan dengan nilai $qsBc$ pada karakter P

Jika teks sudah ditemukan maka dilakukan pergeseran menggunakan nilai $qsBc$ sampai karakter pada Teks habis

Output : Hasil Pencarian = $P[0] = T[s]$ dan $P[1] = T[s]$

3. HASIL DAN PEMBAHASAN

3.1 Penerapan Algoritma *Smith* Dalam Pencarian

Aplikasi pencarian anime berbasis web yang dirancang akan menggunakan algoritma *Smith*. Sebagai solusi dalam penyelesaian masalah pencarian anime. Algoritma *Smith* memiliki 2 fase, yaitu fase *preprocessing* dan fase pencarian.

Pada algoritma *Smith* adalah perpaduan dari perhitungan fungsi pergeseran bad-character dari algoritma Quick Search dengan fungsi pergeseran bad-character dari algoritma Boyer Moore. Bad-character merupakan kumpulan karakter yang mewakili pola (pattern). Fungsi pergeseran bad-character dari algoritma Boyer Moore disimpan pada tabel *bmBc*, sementara fungsi pergeseran bad-character dari algoritma Quick Search disimpan pada tabel *qsBc*. Kedua fungsi tersebut digunakan pada fase *preprocessing* algoritma *Smith*.

Contoh perhitungan tabel pergeseran *bmBc* untuk *pattern* PIECE pada teks ONE PIECE.

Pattern:

Tabel 1. Proses Pencarian *bmBc*

I	0	1	2	3	4
x[i]	P	I	E	C	E
<i>bmBc</i>					

Langkah-langkah untuk menghitung nilai *bmBc* adalah sebagai berikut :

1. Isi kolom *bmBc* paling kanan dengan nilai 1.

Tabel 2. Langkah Pertama Proses Pencarian *bmBc*

I	0	1	2	3	4
x[i]	P	I	E	C	E
<i>bmBc</i>					1

2. Membandingkan karakter E dengan C. Karena tidak cocok maka diberi nilai 2

Tabel 3. Langkah Kedua Proses Pencarian *bmBc*

I	0	1	2	3	4
x[i]	P	I	E	C	E
<i>bmBc</i>				2	1

3. Membandingkan karakter E dan E karena memiliki kesamaan maka beri nilai 1

Tabel 4. Langkah Ketiga Proses Pencarian *bmBc*

I	0	1	2	3	4
x[i]	P	I	E	C	E
<i>bmBc</i>			1	2	1

4. Membandingkan karakter E dengan I. Karena tidak memiliki kesamaan maka diberi nilai 3

Tabel 5. Langkah Keempat Proses Pencarian *bmBc*

I	0	1	2	3	4
x[i]	P	I	E	C	E
<i>bmBc</i>		3	1	2	1

5. Membandingkan nilai I dengan P karena tidak memiliki kesamaan maka diberi nilai

Tabel 6. Langkah Kelima Proses Pencarian *bmBc*

I	0	1	2	3	4
x[i]	P	I	E	C	E
<i>bmBc</i>	4	3	1	2	1

Tabel 7. Hasil Perhitungan *bmBc*

I	0	1	2	3	4
x[i]	P	I	E	C	E
<i>bmBc</i>	4	3	1	2	1

Langkah-langkah perhitungan tabel pergeseran *qsBc* untuk *pattern* PIECE pada teks ONE PIECE.

1. karakter yang memuat teks diurutkan sesuai abjad untuk membuat tabel Quick Search bad-character shif. Teks tersebut memiliki karakter sebanyak 6 huruf, yaitu C, E, I, N, O, dan P. Untuk proses awal, nilai pergeseran bernilai $m + 1$ untuk semua karakter $[c]$ dari indeks $i = 0$ sampai $i = ASIZE - 1$ sehingga akan didapatkan nilai pergeseran yang ditunjukkan pada Tabel 8 berikut.

Tabel 8. Nilai pergeseran karakter $[c]$ bernilai $m + 1$

Char	C	E	I	N	O	P
qsBc	6	6	6	6	6	6

2. Pada tahap kedua, nilai pada tabel diganti menjadi $m - 1$ untuk semua karakter pattern dari $i = 0$ sampai $i = m - 1$, dimana i adalah indeks karakter pattern yang besarnya dari 0 sampai $m - 1$. Dengan demikian, akan didapat nilai perhitungannya sebagai berikut.

$$\begin{aligned}
 qsBc [x[i]] &= m - 1 \\
 qsBc [x[0]] &= 4 - 0 = 4 \\
 qsBc [x[1]] &= 4 - 1 = 3 \\
 qsBc [x[2]] &= 4 - 2 = 2 \\
 qsBc [x[3]] &= 4 - 3 = 1
 \end{aligned}$$

Kemudian, nilai pergeseran awal diganti dengan nilai pergeseran baru sesuai dengan indeks pada karakter pattern. Pattern PIECE memiliki nilai indeks dari 0 sampai 3 sehingga untuk $[x[0]]$ diisi karakter P akan memiliki nilai pergeseran baru, yaitu 4. Untuk $[x[1]]$ diisi karakter I akan memiliki nilai pergeseran baru, yaitu 3, begitu seterusnya sampai pada pattern terakhir. Setelah semua perhitungan selesai, maka nilai pergeseran akan berubah menjadi seperti pada Tabel 9.

Tabel 9. Nilai pergeseran karakter $[c]$ bernilai $m - 1$

Char	C	E	I	N	O	P
qsBc	1	2	3	6	6	4

Contoh proses pencarian pattern PIECE pada teks ONE PIECE. Dengan nilai $bmBc[c]$ dan $qsBc[c]$ untuk pattern PIECE dapat dilihat dari tabel 10 dibawah ini.

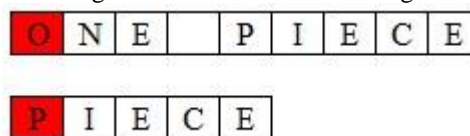
Tabel 10. Nilai $bmBc[c]$ dan $qsBc[c]$ untuk pattern PIECE

Pattern	P	I	E	C	E
$bmBc[c]$	4	3	1	2	1
$qsBc[c]$	4	3	2	1	2

Pada fase pencarian, jumlah pergeseran berdasarkan perbandingan terbesar nilai $bmBc$ dari karakter yang paling kanan dari jendela teks yang bersesuaian dengan panjang *pattern* dengan nilai $qsBc$ dari karakter yang $qsBc [x[i]] = m - i$. Algoritma ini akan tetap mencari dengan bergerak kekanan hingga karakter pada teks berakhir. Tetapi, Jika sebelum teks berakhir pattern yang dicari sudah ditemukan, maka jumlah pergeseran pencarian selanjutnya berubah menjadi nilai $qsBc$ dari karakter yang disebelah kanan karakter yang paling kanan dari jendela teks yang bersesuaian dengan panjang *pattern*. hingga teks berakhir.

Tahap 1

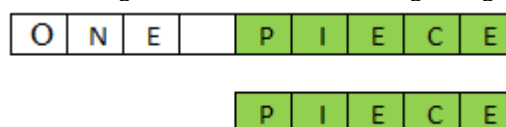
Tabel 11. Langkah Pertama Pencarian Dengan Algoritma Smith



Bergeser sejauh 4 ($bmBc[P] = qsBc[P]$)

Tahap 2

Tabel 12. Langkah Kedua Pencarian Dengan Algoritma Smith



Tabel 13. Hasil Pencarian Dengan Algoritma Smith

O N E P I E C E

P I E C E

Pada percobaan keempat, pattern sudah sesuai dengan karakter pada teks. Pada tahap keempat pencarian selesai, dikarenakan teks sudah berakhir.

3.2 Implementasi Program

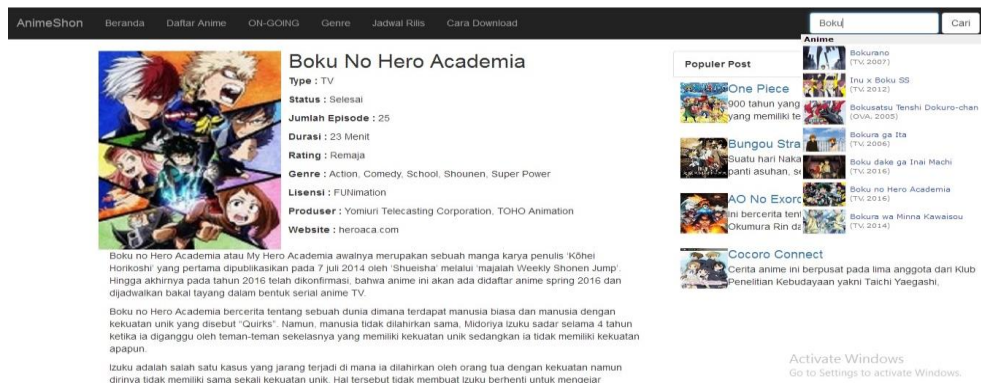
Menu ini berfungsi untuk menampilkan menu pilihan yang akan digunakan. Pilih menu dengan cara menekan *Options Menu* dibagian pojok kanan atas layar. Menu-menu pilihan tersebut terdiri dari menu Pencarian, Bantuan, Tentang dan Keluar. Adapun tampilan menu utama tersebut dapat dilihat pada gambar 1 berikut:



Gambar 1. Tampilan Menu Utama

a. Tampilan Menu Pencarian

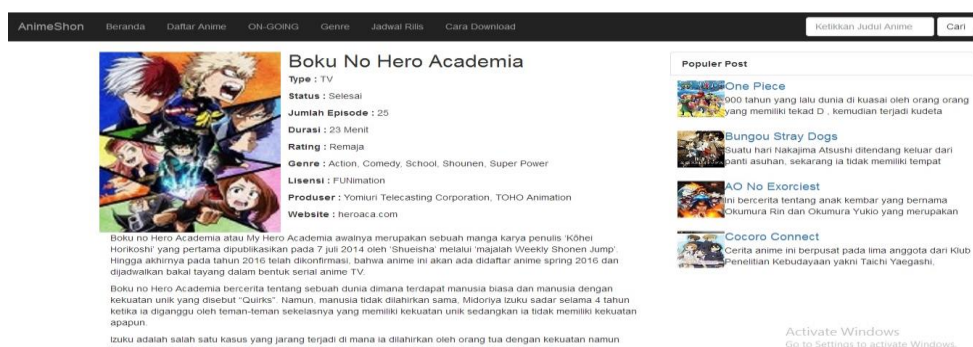
Pada menu pencarian akan ditampilkan halaman untuk melakukan pencarian. Untuk melakukan pencarian ketikkan judul anime yang ingin dicari pada kolom pencarian atau pada sistem ini kolom pencarian ditandai dengan teks cari, maka jika kata yang di *input* mengalami kecocokkan dengan data yang tersimpan didalam *database*, anime akan muncul di list menu yang tertera. Jika yang diketik tidak terdaftar di dalam *database* maka akan muncul pesan “Data yang anda cari tidak ada”. Adapun tampilan menu pencarian tersebut dapat dilihat pada gambar 2 berikut:



Gambar 2. Tampilan Menu Pencarian

b. Tampilan Menu Hasil Pencarian

Pada menu hasil pencarian sistem akan menampilkan judul anime dan gambar sampul yang di pilih oleh *user* setelah melakukan proses *input* teks pada menu pencarian. Hasil yang tampil pada menu ini berupa judul anime, gambar sampul, informasi dan sinopsis. Misalnya pada contoh kali ini sistem menampilkan anime yang berjudul “Boku No Hero Academia”. Adapun tampilan menu hasil pencarian tersebut dapat dilihat pada gambar 3 berikut:



Gambar 3. Tampilan Menu Hasil Pencarian



4.3 Hasil Pengujian Program

Pengujian sistem dilakukan untuk memastikan apakah sistem setelah dibangun dapat berjalan dengan baik sesuai dengan analisa dan perancangan yang telah dilakukan pada bab sebelumnya. Fokus utama pada pengujian program ini adalah untuk mengetahui bagaimana kinerja sistem dalam proses pencocokkan atau pencarian *string*, dengan menerapkan algoritma *Smith* pada aplikasi pencarian anime berbasis web. Hasil pengujian dapat dilihat pada tabel 1 berikut:

Tabel 14. Hasil Pencarian Program

No	Teks yang di input	Teks yang ditemukan	Keterangan
1	Boku	Boku dake ga inai Machi Boku No Hero Academia Boku No Hero Academia S2	Valid
2	Fa	Fairy Tail Fate/Apocrypha Fate/Stay Night Fate/Stay Night Ultimate Blade Work	Valid
3	One	One Piece One Piece Movie One Piece OVA One Piece Episode Spesial One Puch Man	Valid
4	Hai	Haikyuu Haikyuu S2 Haikyuu S3 Hai to Gensou no Grimgar	Valid
5	Norami	“Data yang anda cari tidak ada”	Tidak Valid

4. KESIMPULAN

Berdasarkan hasil penelitian analisa dan perancangan aplikasi pencarian anime berbasis web maka dapat di tarik beberapa kesimpulan sebagai berikut:

1. Berdasarkan penelitian yang di lakukan oleh penulis, algoritma Smith dapat bekerja di dalam aplikasi pencarian anime mampu memberikan kemudahan kepada *user* yang ingin melakukan pencarian anime dalam melakukan pencarian anime yang memiliki kesamaan dengan kata yang dinput dan dengan kata yang telah di simpan sebelumnya di dalam *database* anime.
2. Perancangan aplikasi pencarian anime menggunakan *software Sublime Text* sebagai *editor*. UC Browser sebagai *emulator* sebagai proses atau mesin dari pembuatan aplikasi, serta *MySQL* sebagai wadah perancangan *database*. Berdasarkan penelitian penulis *software-software* pendukung pembuatan aplikasi pencarian anime di atas telah bekerja dengan baik sehingga tercipta suatu aplikasi pencarian anime.

REFERENCES

- [1] R. Fitri, Sarno R, and Anistyasari Y, Semantic Search. Yogyakarta, Indonesia: ANDI, 2012.
- [2] L. Sitorus, Algoritma dan Pemrograman. Yogyakarta, Indonesia: ANDI, 2015.
- [3] M. Shalahudin and A. S. Rosa, Rekayasa Perangkat Lunak. Bandung, Indonesia, 2011.
- [4] Abdul Kadir. Pengenalan Sistem Informasi. Yogyakarta: Andi, 2003.
- [5] Huda Miftakhul dan Komputer Bunafit. (2010). Membuat Aplikasi Database. Jakarta. PT Elex Media Komputindo.
- [6] Wiwit Siswoutomo. Panduan Karir di Dunia Teknologi Informasi. Jakarta: PT Elex Media Komputindo, 2005.
- [7] Sulistyawan, Rubianto, Rahmad Saleh. Modifikasi Blog Multiplay dengan CSS. Jakarta: PT Elex Media Komputindo, 2008.
- [8] Andi Sunyoto M.Kom. AJAX Membangun Web dengan Teknologi Asynchronouse JavaScript dan XML. Yogyakarta: ANDI, 2007.
- [9] C.Charras & T. Lecroq, Handbook of Exact String-Matching Algorithms. London: Kings"s College Publications, 1997.
- [10] PENGERTIANMU.COM (2016, November) www.pengertianmu.com [Online]. Available: <http://www.pengertianmu.com/2016/11/pengertian-penerapan-menurut-para-ahli.html>.
- [11] Burung Internet. (2015, Maret) www.burung-net.com [Online] Available: <http://www.burung-net.com/2015/03/pengertian-dan-fungsi-adobe-dreamweaver.html>.
- [12] Ayu Sumarni. (2011, Maret) www.ayusumarni93.blogspot.co.id [Online]. Available: <http://ayusumarni93.blogspot.co.id/2011/03/pengertian-anime.html>.