

**PEMBUATAN LIBRARY SALTIKJS DALAM MENDETEKSI SALAH
KATA DAN PENENTUAN REKOMENDASI KATA YANG TEPAT
MENGUNAKAN ALGORITMA LEVENSHTAIN DISTANCE
DAN METODE REGULAR SEARCH EXPRESSION**

SKRIPSI



disusun oleh

Mu'alif Lihawa

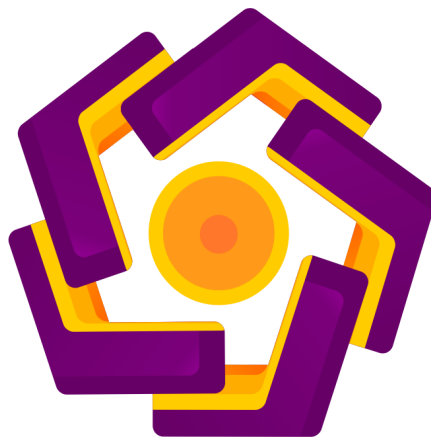
17.11.0981

**PROGRAM SARJANA
PROGRAM STUDI INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA
2021**

**PEMBUATAN LIBRARY SALTIKJS DALAM MENDETEKSI SALAH
KATA DAN PENENTUAN REKOMENDASI KATA YANG TEPAT
MENGUNAKAN ALGORITMA LEVENSHTTEIN DISTANCE
DAN METODE REGULAR SEARCH EXPRESSION**

SKRIPSI

untuk memenuhi sebagian persyaratan
Mencapai gelar Sarjana
Pada Program Studi Informatika



disusun oleh

Mu'alif Lihawa

17.11.0981

**PROGRAM SARJANA
PROGRAM STUDI INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA
2021**

PERSETUJUAN

SKRIPSI

**PEMBUATAN LIBRARY SALTIKJS DALAM MENDETEKSI SALAH KATA DAN
PENENTUAN REKOMENDASI KATA YANG TEPAT MENGGUNAKAN
ALGORITMA LEVENSHTAIN DISTANCE
DAN METODE REGULAR SEARCH EXPRESSION**

yang dipersiapkan dan disusun oleh

Mu'alif Lihawa

17.11.0981

telah disetujui oleh Dosen Pembimbing Skripsi
pada tanggal 17 Maret 2021

Dosen Pembimbing,

Anggit Dwi Hartanto, M.Kom.
NIK. 190302163

PENGESAHAN

SKRIPSI

**PEMBUATAN LIBRARY SALTIKJS DALAM MENDETEKSI SALAH KATA DAN
PENENTUAN REKOMENDASI KATA YANG TEPAT MENGGUNAKAN
ALGORITMA LEVENSHTAIN DISTANCE
DAN METODE REGULAR SEARCH EXPRESSION**

yang dipersiapkan dan disusun oleh

Mu'alif Lihawa

17.11.0981

telah dipertahankan di depan Dewan Penguji
pada tanggal 17 Maret 2021

Susunan Dewan Penguji

Nama Penguji

Anggit Dwi Hartanto, M.Kom.
NIK. 190302163

Yoga Pristyanto, S.Kom, M.Eng.
NIK. 190302412

Erni Seniwati, S.Kom, M.Cs.
NIK. 190302231

Tanda Tangan

Skripsi ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Sarjana Komputer
Tanggal 17 Maret 2021

DEKAN FAKULTAS ILMU KOMPUTER

Krisnawati, S.Si, M.T.
NIK. 190302038

PERNYATAAN

Saya yang bertandatangan dibawah ini menyatakan bahwa, skripsi ini merupakan karya saya sendiri (ASLI), dan isi dalam skripsi ini tidak terdapat karya yang pernah diajukan oleh orang lain untuk memperoleh gelar akademis di suatu institusi pendidikan tinggi manapun, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis dan/atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Segala sesuatu yang terkait dengan naskah dan karya yang telah dibuat adalah menjadi tanggungjawab saya pribadi.

Yogyakarta, 25 Maret 2021



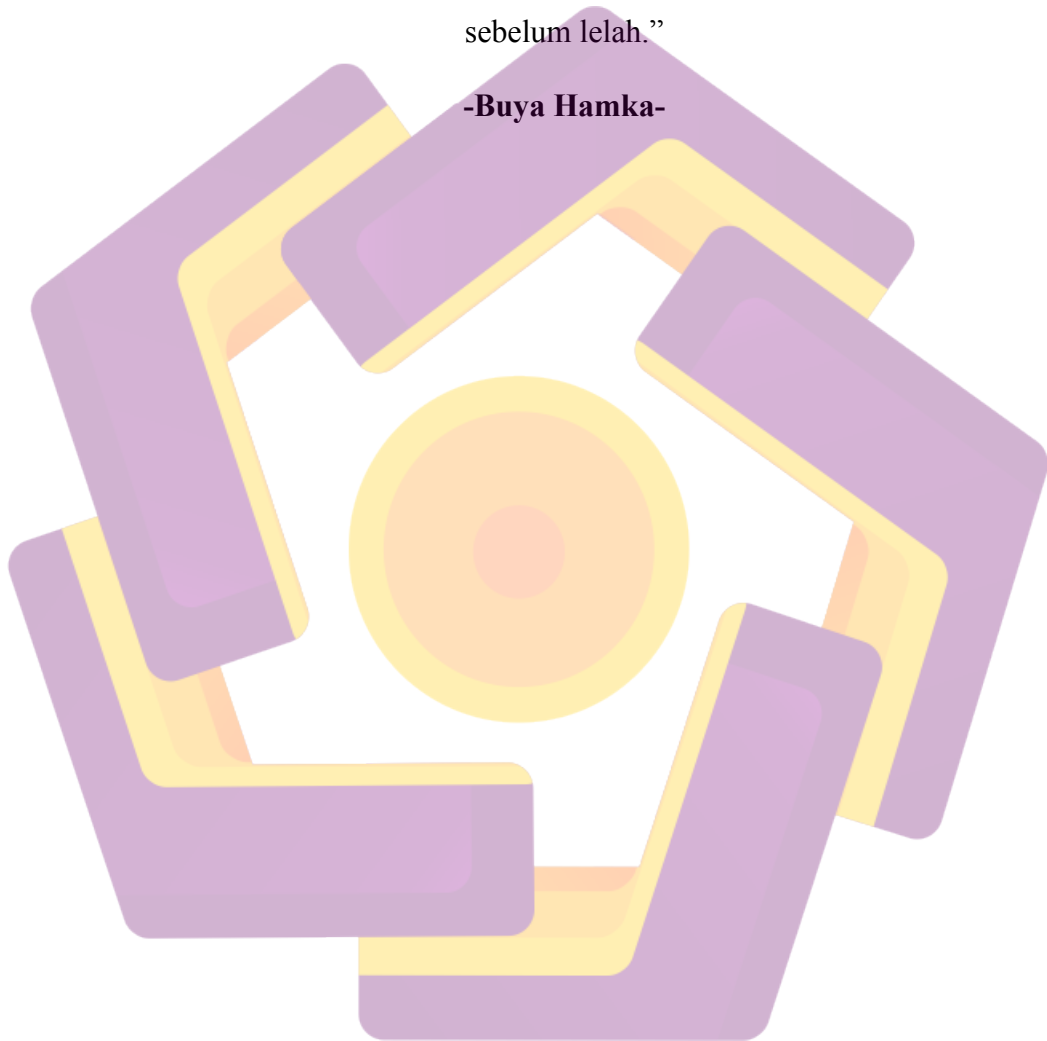
Mu'arif Libawa

NIM. 17.11.0981

MOTTO

“Salah satu pengkerdilan terkejam dalam hidup adalah membiarkan pikiran yang cemerlang menjadi budak bagi tubuh yang malas, yang mendahulukan istirahat sebelum lelah.”

-Buya Hamka-



PERSEMBAHAN

Puji syukur kepada Allah subhanahu wa ta'ala atas segala rahmat dan karunianya sehingga skripsi ini bisa selesai tepat waktu. Saya ucapkan banyak terima kasih kepada seluruh pihak yang telah memberikan dukungan dan doa selama ini.

Skripsi ini saya persembahkan Kepada :

“**Kedua orang tua** tercinta yang tanpa lelah memberikan dukungan dan doa agar anaknya menjadi orang yang sukses dan berilmu serta mempunyai akhlak yang baik. Tanpa didikan dari beliau saya tidak bisa seperti ini. Terima kasih banyak Bapak dan Ibuku”

“**Bapak Anggit Dwi Hartanto M.Kom** selaku dosen pembimbing yang telah memberikan bimbingan, pengarahan dan semangat dalam penyusunan tugas akhir saya”

“**Teman – teman komunitas komputer** yang telah membantu dan menemani saya dalam berproses”

KATA PENGANTAR

Assalamualaikum Warahmatullahi Wabarakatuh

Puji syukur kepada Allah subhanahu wa ta'ala atas segala rahmat dan karunianya, sehingga penulis dapat menyelesaikan skripsi yang berjudul “Pembuatan *Library Saltikjs* Dalam Mendeteksi Salah Kata Dan Penentuan Rekomendasi Kata Yang Tepat Menggunakan Algoritma *Levenshtein Distance* Dan Metode *Regular Search Expression*” sebagai salah satu syarat untuk menyelesaikan studi jenjang Strata Satu (S-1) program Studi Informatika di Universitas Amikom Yogyakarta.

Dalam penulisan skripsi ini tentunya tidak lepas dari bantuan berbagai pihak, oleh karena itu pada kesempatan ini penulis ingin menyampaikan rasa terima kasih kepada semua pihak yang telah membantu dalam penulisan tugas akhir ini, antara lain :

1. Ibu Windha Mega P.D, M.Kom, selaku Ketua Program Studi Informatika Universitas Amikom Yogyakarta.
2. Bapak Arif Akbarul Huda, S.i, M.Eng, selaku Sekretaris Program Studi Informatika Universitas Amikom Yogyakarta.
3. Bapak Anggit Dwi Hartanto M.Kom, selaku dosen pembimbing atas kritik, saran, motivasi dan bimbingan yang diberikan kepada penulis dalam menyelesaikan skripsi ini.
4. Seluruh staf dan karyawan Universitas Amikom Yogyakarta.

5. Secara khusus penulis ingin mengucapkan terima kasih kepada Ayahanda yang penulis banggakan Kasmat Lihawa dan Ibunda penulis tercinta Nella Amin yang telah banyak memberikan senyum manisnya, dukungan dan pengorbanan baik secara moril maupun materil sehingga penulis dapat menyelesaikan studi dengan baik, lancar dan tepat waktu.
6. Teman-teman semua di Informatika Universitas Amikom Yogyakarta.
7. Semua pihak yang telah memberikan bantuan baik tenaga maupun pikiran dalam penyelesaian skripsi ini yang tidak bisa disebutkan satu persatu. Terima kasih banyak.

Sebagai manusia biasa tentunya tidak luput dari kesalahan, penulis menyadari bahwa dalam penyusunan karya tulis ini masih jauh dari kata sempurna. Oleh karena itu, kritik dan saran yang membangun sangatlah diharapkan guna menambah wawasan dan pengembangan ilmu yang telah penulis peroleh selama ini. Semoga karya tulis ini dapat bermanfaat bagi semua pihak.

Yogyakarta, 2021

Mu'alif Lihawa


DAFTAR ISI

PERSETUJUAN	iii
PENGESAHAN	iv
PERNYATAAN	v
MOTTO	vi
PERSEMBAHAN	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR TABEL	xv
DAFTAR GAMBAR	xvii
INTISARI	xix
ABSTRACT	xx
BAB I	1
PENDAHULUAN	1
1.1. Latar Belakang Masalah	1
1.2. Rumusan Masalah	3
1.3. Batasan Masalah	4
1.4. Maksud Dan Tujuan Penelitian	4
1.5. Manfaat Penelitian	4
1.6. Metode Penelitian	5
1.6.1. Metode Pengumpulan Data	5
1.6.2. Metode Pengembangan Saltikjs	5

1.7.	Sistematika Penulisan	6
BAB II		8
LANDASAN TEORI		8
2.1.	Tinjauan Pustaka	8
2.2.	Landasan Teori	13
2.2.1.	Javascript	13
2.2.2.	Library	13
2.2.3.	Object Oriented Programming	13
2.2.4.	Algoritma	14
2.2.5.	Levenshtein Distance	14
a.	Cara Kerja Algoritma Levenshtein Distance	15
2.2.6.	Regular Search Expression	17
a.	Contoh Implementasi Regular Search Expression	20
BAB III		22
ANALISIS DAN PERANCANGAN SISTEM		22
3.1.	Analisis Sistem	22
3.1.1.	Analisis Kebutuhan	22
a.	Analisis Kebutuhan Fungsional	22
b.	Analisis Kebutuhan Non Fungsional	23
1.	Kebutuhan Perangkat Keras	23
2.	Kebutuhan Perangkat Lunak	24
3.1.2.	Analisis Alur Kerja Sistem	25
3.2.	Perancangan Sistem	25

3.2.1.	UML	25
a.	Use Case Diagram	26
1.	Use Case Diagram Sistem	26
b.	Activity Diagram	27
1.	Activity Diagram Input Teks	27
2.	Activity Diagram Mengolah Teks	28
3.	Activity Diagram Data Kamus	29
4.	Activity Diagram Membandingkan Kata Pada Teks Dengan Kata Pada Kamus	30
5.	Activity Diagram Mark Kata Yang Salah Pada Editor	31
6.	Activity Diagram Rekomendasi Kata Yang Benar	32
c.	Sequence Diagram	33
1.	Sequence Diagram Olah Teks	33
2.	Sequence Diagram Data Kamus	34
3.	Sequence Diagram Perbandingan Kata Teks Pada Kamus	35
4.	Sequence Diagram Mark Kata Yang Salah Pada Editor	35
5.	Sequence Diagram Rekomendasi Kata Yang Benar	36
d.	Desain Antarmuka	37
1.	Tampilan Utama Antarmuka Text Editor	37
2.	Tampilan Antarmuka Jika Pada Text Editor Ada Kata Yang Salah	38
3.	Tampilan Rekomendasi Kata Yang Benar	38
4.	Tampilan Rekomendasi Kata Lainnya	39

5.	Tampilan Jika Kata Yang Salah Tidak Ada Di Kamus	40
3.2.2.	Perancangan Sistem Dengan Algoritma Levenshtein Distance Dan Metode Regular Search Expression	41
a.	Contoh Perhitungan Levenshtein Distance	42
BAB IV	49
IMPLEMENTASI DAN PEMBAHASAN	49
4.1.	Implementasi Sistem	49
4.1.1.	Desain Antarmuka	49
a.	Tampilan Utama Antarmuka Text Editor	49
b.	Tampilan Antarmuka Jika Pada Text Editor Ada Kata Yang Salah	50
c.	Tampilan Rekomendasi Kata Yang Benar	51
d.	Tampilan Rekomendasi Kata Lainnya	51
e.	Tampilan Jika Kata Yang Salah Tidak Ada Di Kamus	52
4.1.2.	Implementasi Metode	53
a.	Implementasi Struktur Class Library	53
b.	Implementasi Algoritma Levenshtein Distance Pada Javascript .	53
c.	Implementasi Proses Mengolah Teks	55
d.	Implementasi Proses Mengambil Data Kamus	57
e.	Implementasi Proses Cek Kata Teks Pada Kamus	58
f.	Implementasi Proses Mencari Kesamaan Kata	59
4.2.	Pembahasan Sistem	61
4.2.1.	Proses Mencari Kesamaan Kata	61

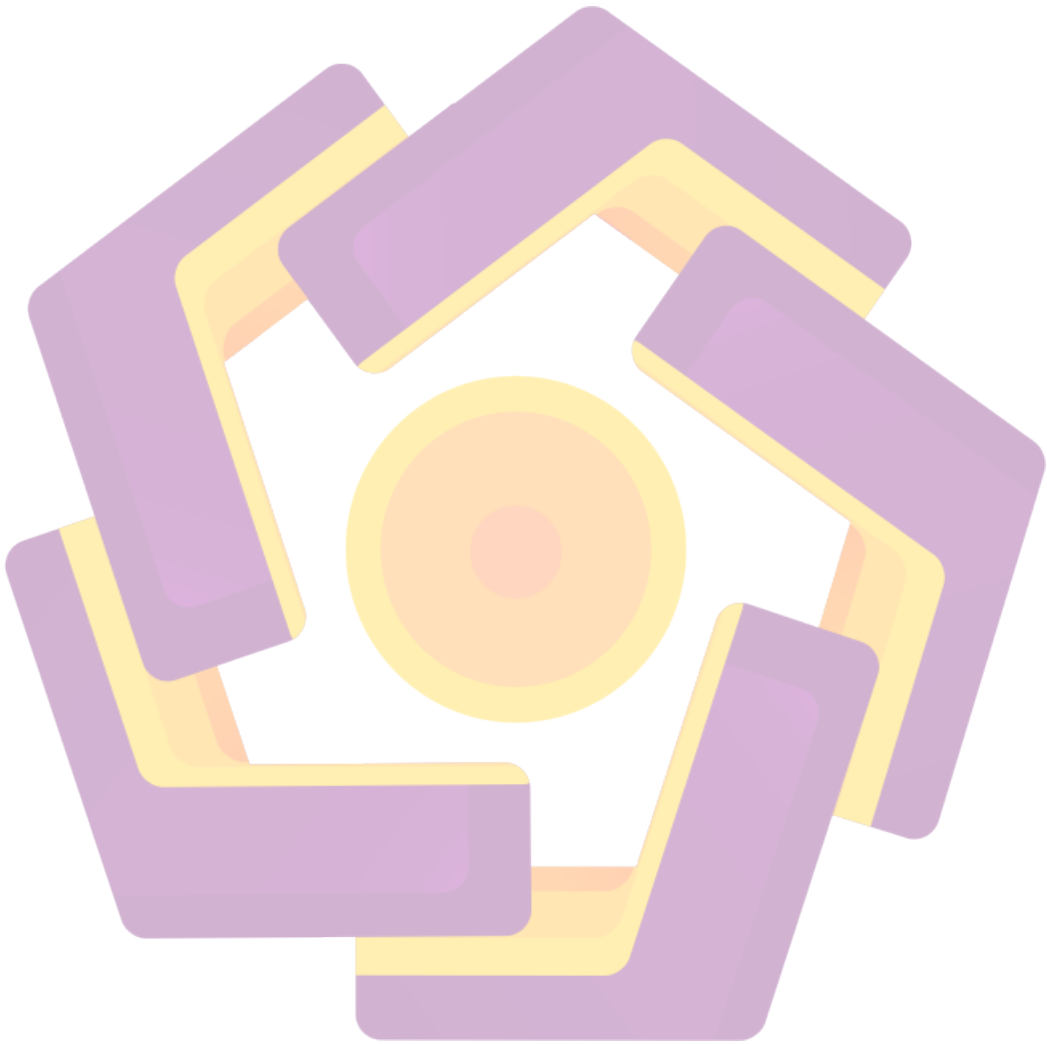


4.2.2.	Upload Library Ke NPM	64
4.2.3.	Install Package Library Saltikjs	65
4.3.	Pengujian Sistem	67
4.3.1.	Pengujian Dengan Confusion Matrix	68
4.3.2.	Pengujian Dengan Black Box	73
a.	Editor	73
b.	Pop Up Saltikbox	74
c.	Rekomendasi Kata Lainnya	75
BAB V	77
KESIMPULAN DAN SARAN	77
5.1.	Kesimpulan	77
5.2.	Saran	78
DAFTAR PUSTAKA	80

DAFTAR TABEL

Tabel 2.1.	Perbandingan Dengan Penelitian Sebelumnya	12
Tabel 2.2.	Matriks array kams x kams	15
Tabel 2.3.	Contoh Beberapa Meta Karakter Regular Expression	20
Tabel 3.1.	Kebutuhan Perangkat Keras	24
Tabel 3.2.	Kebutuhan Perangkat Lunak	24
Tabel 3.3.	Data Kata Yang Digunakan	42
Tabel 3.4.	Inisialisasi Tabel Matrix sebuah x sambal	44
Tabel 3.5.	Nilai $D[1,1]$	44
Tabel 3.6.	Nilai $D[2,1]$	45
Tabel 3.7.	Matrix sebuah x sambal	46
Tabel 3.8.	Matrix sebuah x serbarumah	47
Tabel 3.9.	Matrix sebuah x serbasusah	47
Tabel 3.10.	Matrix sebuah x serbausaha	48
Tabel 3.11.	Hasil Nilai Similarity	48
Tabel 4.1.	Data Kamus	62
Tabel 4.2.	Bentuk Confusion Matrix	68
Tabel 4.3.	Data Confusion Matrix Prediksi Kata Typo	69
Tabel 4.4.	Hasil Pengukuran Dalam Pencarian Kata Typo	71
Tabel 4.5.	Hasil Pengukuran Dalam Kemiripan Kata Typo	72
Tabel 4.6.	Uji Coba Editor	73
Tabel 4.7.	Uji Coba Pop Up Saltikbox	75

Tabel 4.8. Uji Coba Rekomendasi Kata Lainnya 76



DAFTAR GAMBAR

Gambar 2.1.	Contoh Sederhana Menggunakan Regular Expression.	17
Gambar 2.2.	Contoh Penggunaan Regex Pada Situs Regex101.Com	20
Gambar 3.1.	Alur Kerja Sistem	25
Gambar 3.2.	Use Case Diagram Sistem	26
Gambar 3.3.	Activity Diagram Input Teks	27
Gambar 3.4.	Activity Diagram Mengolah Teks	28
Gambar 3.5.	Activity Diagram Data Kamus	29
Gambar 3.6.	Activity Diagram Membandingkan Kata Pada Teks Dengan Kata Pada Kamus	30
Gambar 3.7.	Activity Diagram Mark Kata Yang Salah Pada Editor	31
Gambar 3.8.	Activity Diagram Rekomendasi Kata Yang Benar	32
Gambar 3.9.	Sequence Diagram Olah Teks	33
Gambar 3.10.	Sequence Diagram Data Kamus	34
Gambar 3.11.	Sequence Diagram Perbandingan Kata Teks Pada Kamus	35
Gambar 3.12.	Sequence Diagram Mark Kata Yang Salah Pada Editor	36
Gambar 3.13.	Sequence Diagram Rekomendasi Kata Yang Benar	36
Gambar 3.14.	Tampilan Utama Antarmuka Text Editor	37
Gambar 3.15.	Tampilan Antarmuka Jika Pada Text Editor Ada Kata Yang Salah	38
Gambar 3.16.	Tampilan Rekomendasi Kata Yang Benar	39
Gambar 3.17.	Tampilan Rekomendasi Kata Lainnya	40

Gambar 3.18. Tampilan Jika Kata Yang Salah Tidak Ada Di Kamus	40
Gambar 3.19. Desain Sistem Penggunaan Algoritma Dan Metode Yang Digunakan	41
Gambar 4.1. Tampilan Utama Antarmuka Text Editor	50
Gambar 4.2. Tampilan Antarmuka Jika Pada Text Editor Ada Kata Yang Salah	50
Gambar 4.3. Tampilan Rekomendasi Kata Yang Benar	51
Gambar 4.4. Tampilan Rekomendasi Kata Lainnya	52
Gambar 4.5. Tampilan Jika Kata Yang Salah Tidak Ada Di Kamus	52
Gambar 4.6. Ouput Kata Yang Serupa Dengan Pola “sebuah”	62
Gambar 4.7. Output Fungsi Levenshtein Distance	63
Gambar 4.8. Halaman Package Saltik Pada Situs Npm	64
Gambar 4.9. Struktur File Pada Package Library Saltikjs	65
Gambar 4.10. Menjalankan Perintah Install Package	65
Gambar 4.11. Uji Coba Editor Marking	74
Gambar 4.12. Uji Coba Editor Clear Mark	74
Gambar 4.13. Pop Up Saltikbox	75
Gambar 4.14. Rekomendasi Kata Lainnya	76

INTISARI

Mengetik adalah suatu kegiatan untuk menuliskan sebuah tulisan dalam bentuk cetakkan yang sudah dirangkai oleh mesin ketik. Dengan perkembangan zaman yang cepat mesin ketik digantikan oleh komputer karena efisien dalam membuat sebuah tulisan atau teks. sebuah teks atau tulisan yang mudah dipahami dalam penyampaian informasi tidak memiliki kesalahan kata atau *typo* yang mengakibatkan ketidakjelasan dalam informasi yang disampaikan.

Dalam aplikasi pengolah kata seperti microsoft office word, memiliki fitur *suggestions word* dan *Autocorrect word* yang dimana sangat berguna dalam mengecek sebuah tulisan yang terdapat kesalahan kata dalam penulisannya.

Penelitian ini dilakukan untuk membangun sebuah *library* javascript untuk mendeteksi kesalahan kata atau *typo* terhadap penulisan kata yang salah dan merekomendasikan kata yang tepat untuk merubah kata yang salah. saltikjs adalah nama dari *library* javascript ini, saltikjs akan dibangun menggunakan algoritma *Levenshtein Distance* dan Metode *Regular Search Expression*.

Kata kunci : *Regular Search Expression*, *Regex*, *Library*, *Javascript*, *Saltikjs*,
Levenshtein Distance.

ABSTRACT

Typing is an activity to write a piece of writing in printed form that has been assembled by a typewriter. With the rapid development of the era, typewriters are replaced by computers because they are efficient in making writing or text. a text or writing that is easy to understand in delivering information does not have a word or typo that results in unclarity in the information conveyed.

In a word processing application such as Microsoft Office Word, it features word suggestions and autocorrect words which are very useful in checking a text that has word errors in writing.

This research was conducted to build a javascript library to detect word errors or typos against writing the wrong words and recommend the right words to change the wrong words. saltikjs is the name of this javascript library, saltikjs will be built using the Levenshtein Distance algorithm and the Regular Search Expression Method.

Keyword : *Regular Search Expression, Regex, Library, Javascript, Saltikjs, Levenshtein.*