

## **BAB III**

### **METODOLOGI**

#### **3.1 Metode Pengembangan Perangkat Lunak**

Berdasarkan permasalahan yang telah diuraikan di atas, pendekatan untuk melakukan penelitian ini adalah dengan menggunakan model proses *waterfall* dengan tujuan untuk memecahkan masalah dengan mengembangkan aplikasi dengan kinerja yang lebih terstruktur dan sistematis.

Tahapan *waterfall* yang dilakukan untuk membuat sistem informasi ini akan dijabarkan sebagai berikut:

##### **3.1.1 Analisis Kebutuhan**

###### **a. Studi Literatur**

Metode ini merupakan metode pengumpulan data dengan cara mencari, membaca, dan mempelajari buku-buku yang tersedia di perpustakaan, toko buku, dan melalui internet yang digunakan sebagai literatur yang dapat mendukung di dalam penyusunan dan penulisan skripsi. Tujuan dari Kepustakaan yaitu untuk menemukan teori pendukung yang telah berhasil melakukan pengembangan sistem yang dijadikan referensi dalam penelitian. Ada pun literatur yang digunakan adalah sebagai berikut: Buku-buku yang berkaitan dengan ilmu komputer dan tata bahasa serta jurnal-jurnal yang terkait dengan judul yang akan dilakukan penelitian.

###### **b. Wawancara**

Observasi (Pengamatan Langsung) merupakan salah satu teknik pengumpulan data. Adapun observasi yang dilakukan adalah sebagai berikut: Dengan melakukan wawancara dalam rangka memperoleh informasi yang berkaitan dengan kata - kata kasar, kepada Ketua Prodi SI Pendidikan dan Sastra Bahasa Indonesia Universitas Lampung yaitu, Bapak Bambang Riadi, S.pd, M.pd.

c. Dokumentasi

Metode ini juga berfungsi sebagai pengumpulan data-data pendukung seperti gambar-gambar atau dokumen yang lainnya sebagai penjelasan tentang data data yang sudah diperoleh dari hasil pengamatan dan penelitian.

### 3.1.1.1 Analisis Kebutuhan Perangkat

Penjelasan dari analisis kebutuhan perangkat keras, perangkat lunak dan analisa kebutuhan sumber daya manusia:

1) Analisis Kebutuhan Perangkat Lunak.

*Software* untuk pembuatan aplikasi ini adalah menggunakan dua perangkat lunak, yang pertama perangkat lunak untuk proses pembuatan aplikasi dan kedua perangkat lunak untuk penerapan aplikasi. Perangkat lunak pembuatan Software yang digunakan pada pembuatan aplikasi adalah :

2) Analisa kebutuhan perangkat keras

Perangkat keras yang dibutuhkan dalam pembuatan sistem ini :

- a. Processor Intel CORE i3-370M.
- b. RAM 6GB.
- c. Harddisk 320 GB.
- d. SSD 240 GB.

3) Perangkat Lunak (*Software*)

Perangkat lunak yang dibutuhkan dalam pembuatan sistem ini adalah:

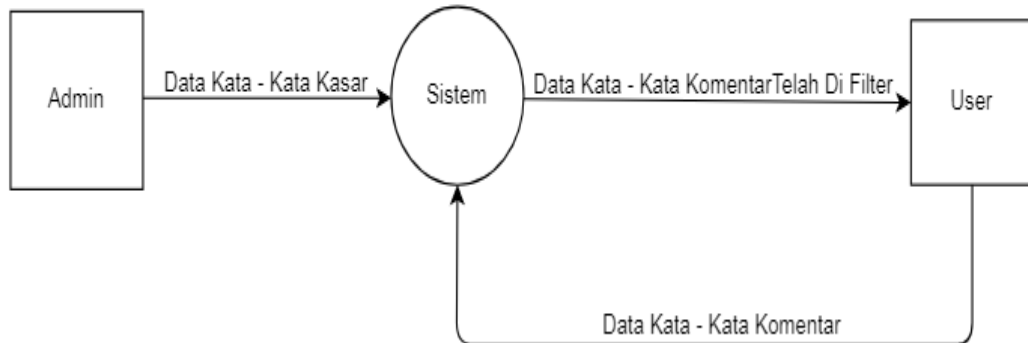
- a. Sistem Operasi Microsoft Windows.
- b. PHP *Strom*.

4) Analisa kebutuhan sumber daya manusia.

*User* atau pengguna untuk aplikasi ini adalah semua kalangan. Pengguna disini tidak dituntut untuk mengerti bagaimana program berjalan tetapi pengguna dituntut hanya mengerti menggunakan aplikasi ini.

### 3.1.2 Desain sistem.

Ini adalah desain sistem *diagram context* dalam sistem.

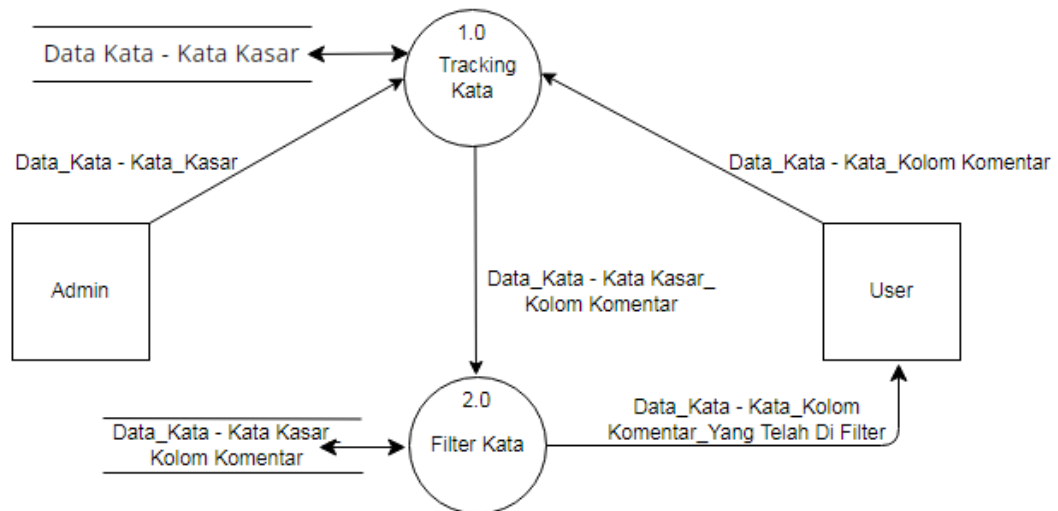


Gambar 3.1 Diagram Context.

Berikut adalah penjabaran dari Data Flow Diagram yang akan mendeskripsikan sistem informasi ini:

#### 1. DFD Level 0

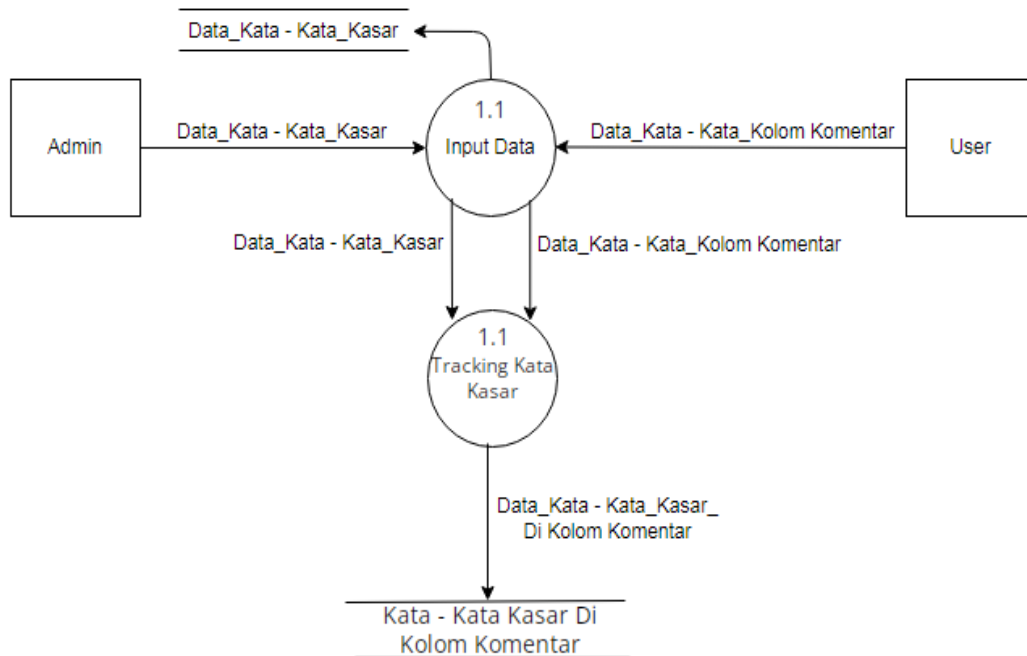
Ini adalah DFD Level 0 dalam sistem ini::



Gambar 3.2 DFD Level 0

## 2. DFD *Level 1* Proses 1

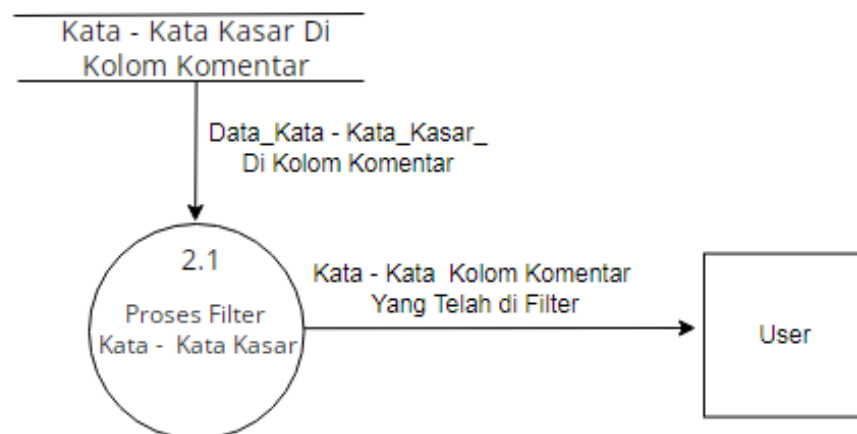
Ini adalah DFD *Level 1* proses 1 dalam sistem ini:



Gambar 3.3 DFD *Level 1* Proses 1

## 3. DFD *Level 1* Proses 2

Ini adalah DFD *Level 1* proses 2 dalam sistem ini:



Gambar 3.4 DFD *Level 1* Proses 2

### 3.1.2.1 Rancangan Sistem Interface.

Tahap selanjutnya adalah membuat desain sistem. Pada sistem informasi ini terdapat 2 pengguna yakni admin dan *user*. Ada 3 fitur dalam akun *administrator*, yakni *login*, *entri data*, dan *logout*. Sedangkan di akun *user* terdapat 4 fitur, yakni *login*, menu *dashboard*, kolom komentar, dan *logout*.

Ketika admin akan masuk dashboard maka admin perlu menuju ke halaman login terlebih dahulu sebelum masuk dashboard.

Ketika *user* akan memberikan komentar pada halaman kolom komentar *user* akan diarahkan terlebih dahulu pada halaman login *user*.

Ketika admin *login* ke dalam sistem informasi ini, akan langsung diarahkan ke menu input data yang bertujuan untuk menginputkan seluruh data yang diperlukan dalam sistem informasi.

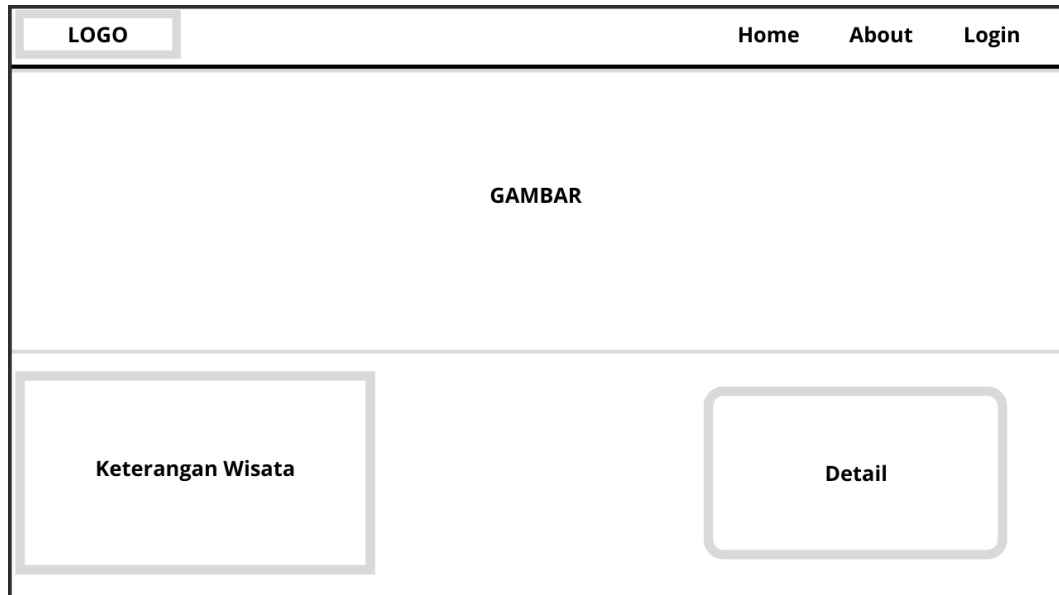
Ketika *user login*, akan langsung diarahkan ke menu *dashboard*. Di halaman *dashboard*, *user* dapat melihat informasi wisata yang ada di dalam sistem informasi ini. Di halaman wisata, *user* akan di kenalkan tentang detail tempat wisata tersebut dan pada halamanan kolom komentar *user* dapat memberikan komentar.

Di halaman keterangan wisata, *user* dapat melihat dan akan di kenalkan tentang detail tempat wisata tersebut dan dapat memberikan komentar.

Di dalam halaman kata - kata kasar, *admin* dapat menginputkan pada *field* yang ada yang akan digunakan dalam melakukan *Tracking* Kata - Kata Kasar.

a. Desain Implementasi halaman wisata.

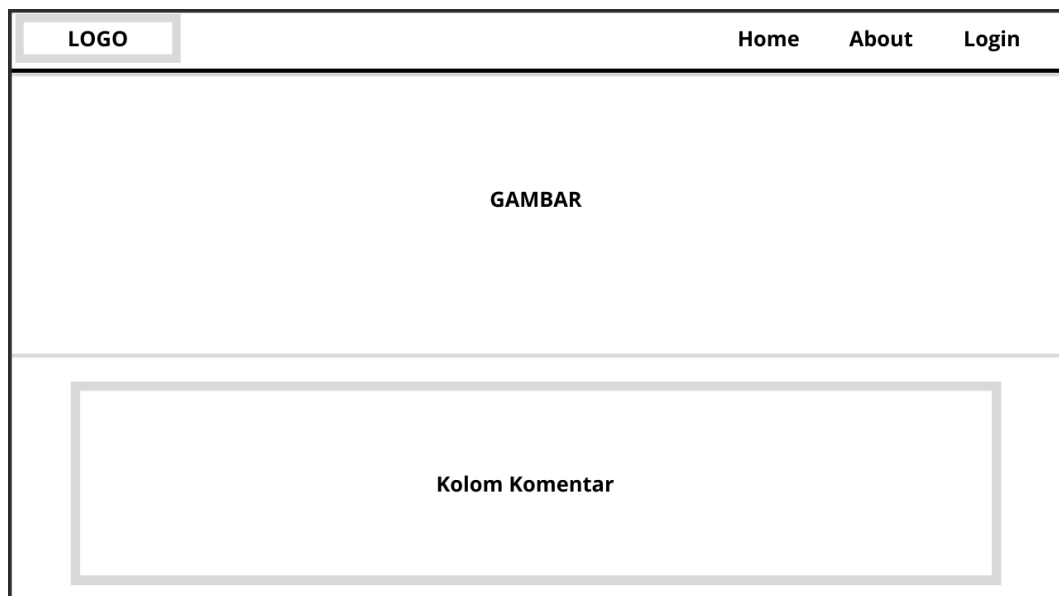
Berikut ini adalah rancangan desain implementasi input yang ada di dalam sistem informasi ini pada gambar;



Gambar 3.5 Halaman 1 Implementasi.

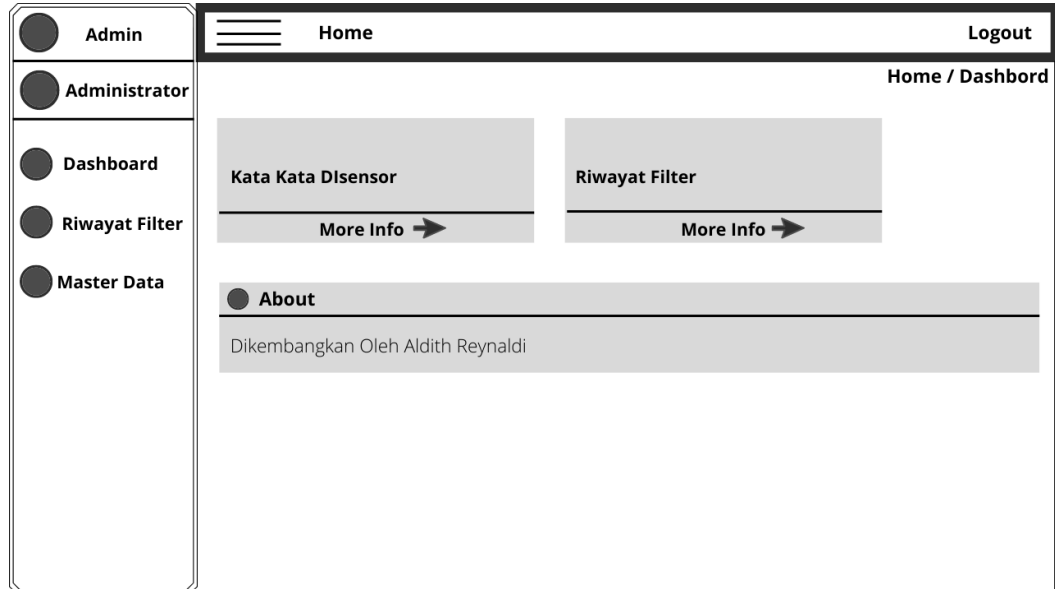
b. Desain Implementasi halaman wisata.

Berikut ini adalah rancangan desain implementasi yang ada di dalam sistem informasi ini pada gambar ;



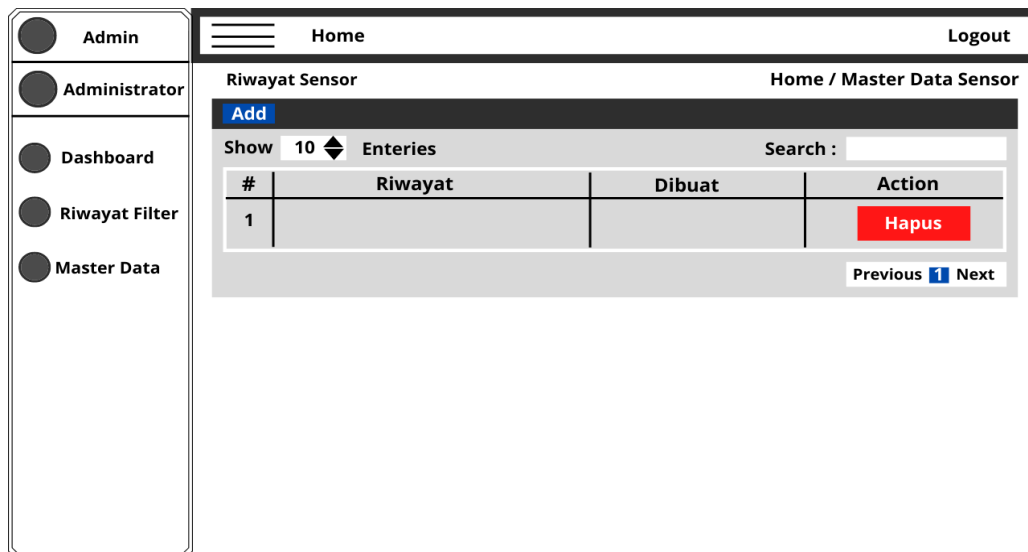
Gambar 3.6 Halaman 2 Implementasi.

- c. Desain Implementasi halaman *Dashboard tracking* kata kata kasar.  
Berikut ini tadalah rancangan desain implementasi halaman *dashboard* yang ada di dalam sistem informasi ini pada gambar;



Gambar 3.7 Halaman *Dashboard* Kata- Kata Kasar.

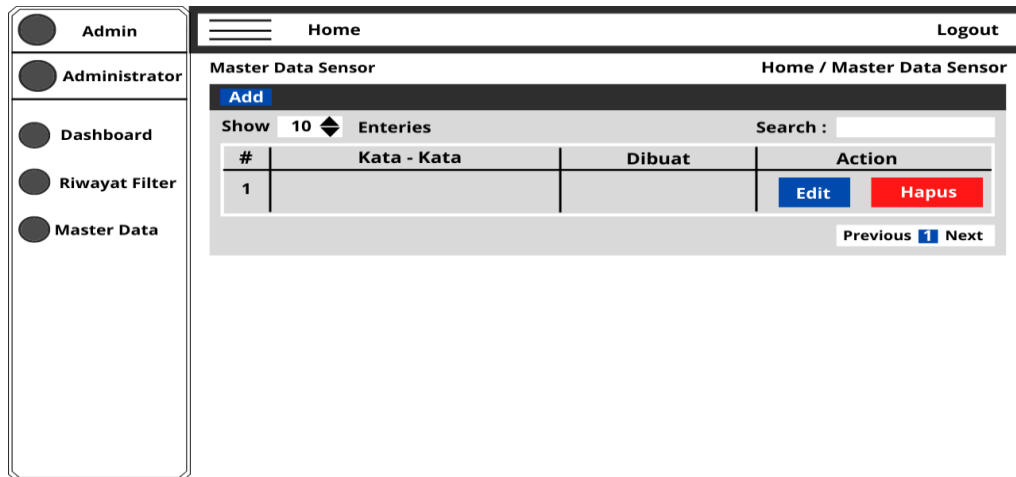
- d. Desain Implementasi halaman riwayat *filter*  
Berikut ini adalah rancangan desain implementasi halaman riwayat *filter* yang ada di dalam sistem informasi ini pada gambar.



Gambar 3.8 Halaman Riwayat *Filter* Kata- Kata Kasar.

e. Desain Implementasi halaman data *tracking*

Berikut ini adalah rancangan desain implementasi halaman data *tracking* yang ada di dalam sistem informasi ini pada gambar.



Gambar 3.9 Halaman Data Sensor Kata- Kata Kasar.

### 3.1.3 Implementasi (Pengkodingan)

Setelah melakukan perancangan *visualisasi* sistem, tahap selanjutnya adalah melakukan tahap pengkodean / koding aplikasi dengan maksud untuk membuat sistem informasi dari desain yang telah dibuat sebelumnya.

### 3.1.4 Testing (Pengujian Program)

Pada tahap ini akan dilakukan pengujian terhadap fitur yang ada di dalam sistem informasi ini. Jika terdapat kekurangan maka peneliti akan memperbaiki kekurangan tersebut. Tahap pengujian ini akan menggunakan pengujian *black box testing* yang akan di bahas pada BAB selanjutnya.

### 3.1.5 Perawatan.

Setelah program melalui tahap pengujian dan dinyatakan lolos uji fungsional, langkah selanjutnya adalah membuat program tersedia untuk digunakan pengguna dan melakukan pemeliharaan pada aplikasi dengan harapan dapat terus berfungsi dengan baik sehubungan dengan fitur yang telah di tetapkan.





## 1. Langkah ke Pertama.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
P	E	N	J	A	G	A	_	I	T	U	_	B	R	E	N	G	S	E	K
B	R	E	N	G	S	E	K												
0	1	2	3	4	5	6	7												

Langkah 1 di dalam algoritma *brute force* tidak ditemukan kecocokan pada karakter ke-0 pada teks dan karakter ke-0 pada *pattern* maka akan dilakukan sebuah pergeseran, *pattern* bergerak sebanyak satu kali ke karakter berikutnya menuju arah kanan.

## 2. Langkah ke dua

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
P	E	N	J	A	G	A	_	I	T	U	_	B	R	E	N	G	S	E	K
	B	R	E	N	G	S	E	K											
	0	1	2	3	4	5	6	7											

Langkah 2 di dalam algoritma *brute force* tidak ditemukan kecocokan pada karakter ke-1 pada teks dan karakter ke-1 pada *pattern* maka akan dilakukan sebuah pergeseran, *pattern* bergerak sebanyak satu kali ke karakter berikutnya menuju arah kanan.

## 3. Langkah ke tiga

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
P	E	N	J	A	G	A	_	I	T	U	_	B	R	E	N	G	S	E	K
		B	R	E	N	G	S	E	K										
		0	1	2	3	4	5	6	7										

Langkah 3 di dalam algoritma *brute force* tidak ditemukan kecocokan pada karakter ke-2 pada teks dan karakter ke-2 pada *pattern* maka akan dilakukan sebuah pergeseran, *pattern* bergerak sebanyak satu kali ke karakter berikutnya menuju arah kanan.

## 4. Langkah ke empat

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
P	E	N	J	A	G	A	_	I	T	U	_	B	R	E	N	G	S	E	K
			B	R	E	N	G	S	E	K									
			0	1	2	3	4	5	6	7									

Langkah 4 di dalam algoritma *brute force* tidak ditemukan kecocokan pada karakter ke-3 pada teks dan karakter ke-3 pada *pattern* maka akan dilakukan sebuah pergeseran, *pattern* bergerak sebanyak satu kali ke karakter berikutnya menuju arah kanan.

## 5. Langkah ke lima

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
P	E	N	J	A	G	A	_	I	T	U	_	B	R	E	N	G	S	E	K
				B	R	E	N	G	S	E	K								
				0	1	2	3	4	5	6	7								

Langkah 5 di dalam algoritma *brute force* tidak ditemukan kecocokan pada karakter ke-4 pada teks dan karakter ke-4 pada *pattern* maka akan dilakukan sebuah pergeseran, *pattern* bergerak sebanyak satu kali ke karakter berikutnya menuju arah kanan.

## 6. Langkah ke enam

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
P	E	N	J	A	G	A	_	I	T	U	_	B	R	E	N	G	S	E	K
					B	R	E	N	G	S	E	K							
					0	1	2	3	4	5	6	7							

Langkah 5 di dalam algoritma *brute force* tidak ditemukan kecocokan pada karakter ke-5 pada teks dan karakter ke-5 pada *pattern* maka akan dilakukan sebuah pergeseran, *pattern* bergerak sebanyak satu kali ke karakter berikutnya menuju arah kanan.

## 7. Langkah ke tujuh

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
P	E	N	J	A	G	A	_	I	T	U	_	B	R	E	N	G	S	E	K
						B	R	E	N	G	S	E	K						
						0	1	2	3	4	5	6	7						

Langkah 7 di dalam algoritma *brute force* tidak ditemukan kecocokan pada karakter ke-6 pada teks dan karakter ke-6 pada *pattern* maka akan dilakukan sebuah pergeseran, *pattern* bergerak sebanyak satu kali ke karakter berikutnya menuju arah kanan.

## 8. Langkah ke delapan

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
P	E	N	J	A	G	A	_	I	T	U	_	B	R	E	N	G	S	E	K
							B	R	E	N	G	S	E	K					
							0	1	2	3	4	5	6	7					

Langkah 8 di dalam algoritma *brute force* tidak ditemukan kecocokan pada karakter ke-7 pada teks dan karakter ke-7 pada *pattern* maka akan dilakukan sebuah pergeseran, *pattern* bergerak sebanyak satu kali ke karakter berikutnya menuju arah kanan.

## 9. Langkah ke sembilan

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
P	E	N	J	A	G	A	_	I	T	U	_	B	R	E	N	G	S	E	K
								B	R	E	N	G	S	E	K				
								0	1	2	3	4	5	6	7				

Langkah 9 di dalam algoritma *brute force* tidak ditemukan kecocokan pada karakter ke-8 pada teks dan karakter ke-8 pada *pattern* maka akan dilakukan sebuah pergeseran, *pattern* bergerak sebanyak satu kali ke karakter berikutnya menuju arah kanan.

## 10. Langkah ke sepuluh

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
P	E	N	J	A	G	A	_	I	T	U	_	B	R	E	N	G	S	E	K
									B	R	E	N	G	S	E	K			
								0	1	2	3	4	5	6	7				

Langkah 10 di dalam algoritma *brute force* tidak ditemukan kecocokan pada karakter ke-9 pada teks dan karakter ke-9 pada *pattern* maka akan dilakukan sebuah pergeseran, *pattern* bergerak sebanyak satu kali ke karakter berikutnya menuju arah kanan.

## 11. Langkah ke sebelas

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
P	E	N	J	A	G	A	_	I	T	U	_	B	R	E	N	G	S	E	K
										B	R	E	N	G	S	E	K		
									0	1	2	3	4	5	6	7			

Langkah 11 di dalam algoritma *brute force* tidak ditemukan kecocokan pada karakter ke-10 pada teks dan karakter ke-10 pada *pattern* maka akan dilakukan sebuah pergeseran, *pattern* bergerak sebanyak satu kali ke karakter berikutnya menuju arah kanan.

## 12. Langkah ke duabelas

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
P	E	N	J	A	G	A	_	I	T	U	_	B	R	E	N	G	S	E	K
											B	R	E	N	G	S	E	K	
										0	1	2	3	4	5	6	7		

Langkah 12 di Dalam *brute force* tidak ditemukan kecocokan pada karakter ke-11 pada teks dan karakter ke-11 pada *pattern* maka akan dilakukan sebuah pergeseran, *pattern* bergerak sebanyak satu kali ke karakter berikutnya menuju arah kanan.

## 13. Langkah ke tigabelas

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
P	E	N	J	A	G	A	_	I	T	U	_	B	R	E	N	G	S	E	K
												B	R	E	N	G	S	E	K
												0	1	2	3	4	5	6	7

Pada langkah 13 dari algoritma *brute force*, telah ditemukan karakter yang cocok dengan karakter ke-12 dalam teks yang memiliki panjang karakter yang sama dengan karakter dalam *Pattern*. Kemudian proses berakhir saat ketika karakter terakhir dari teks telah dilakukan pencocokan dengan karakter terakhir pada *pattern* yang telah ditemukan.

### 3.4 Kata Kasar Dalam Bahasa Indonesia.

Kata kata kasar adalah ungkapan bahasa yang secara sosial bersifat ofensif, menghina, menistakan dan merendahkan orang lain. Kategori kata kata kasar adalah kata yang melanggar norma dan kesopanan masyarakat umum dalam tata bahasa, dalam kegiatan penelitian ini terdapat 64 kosa kata - kata kasar yang digunakan. terdiri dari kondisi, hewan, makhluk astral, benda, bagian tubuh, anggota keluarga, aktivitas, dan profesi, 64 kosa kata - kata kasar tersebut didapatkan melalui wawancara Ketua Prodi SI Pendidikan dan Sastra Bahasa Indonesia Universitas Lampung yaitu, Bapak Bambang Riadi, S.pd, M.pd.

- Kondisi.

Kata-kata yang menggambarkan kondisi percakapan yang tidak menyenangkan biasanya digunakan sebagai kata-kata kasar. Secara umum, ada tiga hal yang dapat atau mungkin berhubungan dengan kondisi tidak menyenangkan ini, yaitu gangguan mental (Contoh: gila, bego, goblok, idiot, sinting, bodoh, tolol, sontoloyo, geblek, sarap), penyimpangan seksual (misalnya: lesbi, homo, banci, waria), kurangnya modernisasi (misalnya: kampung, udik, alay), disabilitas (misalnya: buta, budek, bolot, bisu), kondisi di mana seseorang tidak memiliki etika (misalnya: brengsek, bejat, bajingan) kondisi yang tidak disetujui oleh Tuhan atau agama (misalnya: keparat, jahanam, terkutuk, kafir, najis), dan kondisi

yang terkait dengan keadaan yang tidak menguntungkan (misalnya: celaka, mati, modar, sialan, pantek, mampus).

- Hewan.

Tidak semua hewan bisa digunakan sebagai kata kasar. satwa Kata-kata yang digunakan sebagai kata-kata ofensif biasanya mengacu pada kualitas Hal-hal buruk tertentu yang dibenci beberapa orang (misalnya anjing, Bajingan, ,kampret), jijik dan larangan dalam agama tertentu (Contoh: babi), menyebarkan (misalnya: bangsat, kucing, kunyuk) parasit (contoh: lintah), sehat (contoh: bandot, buaya), berisik (contoh: burung beo).

- Makhluk Astral.

Contoh makhluk astral yang biasanya digunakan Kata-kata kasarnya adalah Setan, Iblis, Tuyul, dan Kunti. Mereka semua Keberadaan makhluk asrtal yang sering mengintervensi atau mengganggu kehidupan manusia.

- Sebuah Objek.

Seperti keberadaan binatang dan makhluk astral , benda biasanya dibuang karena kualitas buruk seperti bau (misalnya tai dan bangkai).

- Bagian dari tubuh.

Bagian tubuh yang digunakan sebagai kata kasar biasanya berkaitan erat dengan aktivitas seksual, Misal,. Serta ada Bagian tubuh yang sering digunakan adalah mata dalam berbentuk matamu (Mata dalam bahasan Indonesia). Artinya, yang satu mengutuk yang lain karena dia tidak menggunakan mata nya dengan benar dan akibatnya membuat kesalahan. Ekspresi lain adalah mata elang yang digunakan secara kiasan untuk mengutuk seorang pria Seorang cabul dan mata duitan untuk orang yang memilih uang dari apa pun. Dan makian bagian tubuh “mulut” digunakan untuk mewakili orang yang banyak bicara atau sering berbohong.

- Anggota keluarga.

Orang Indonesia biasanya menambahkan akhiran-mu pada kata yang mengacu ke hubungan sebagai kutukan, seperti ibumu, bapakmu, kakekmu, dan nenekmu.

- Profesi.

Pekerjaan seseorang, terutama pekerjaan kelas rendah yang dilarang oleh agama, sering digunakan oleh orang Indonesia sebagai kata-kata kasar. Pekerjaan-pekerjaan itu termasuk maling, sundel, copet, lonte, cecenguk, kacung, pelacur, pecun, jablay dan perek.

- Menggunakan bentuk informal dari bahasa kasar.

Sebuah kata kasar biasanya dibuat dari sebuah kata kasar yang diperpanjang. Bentuk informal dibuat dengan membuat kosakata baru yang pengucapannya mirip dengan kata kasar yang asli. Sebagai contoh, banyak netizen mengetik anjay atau anjirt untuk mengatakan anjing.

- Menggunakan bahasa asing dan lokal.

Karena banyak juga pengguna yang mengucapkan kata-kata kasar dalam bahasa asing (misalnya bitch, cyka blyat , shit, dll). Atau bahasa daerah (misalnya asu dll). Tidak hanya secara formal, tetapi paling sering dengan cara informal. Beberapa pengguna biasanya mengetik kata-kata sulit baik dalam bahasa Indonesia murni, bahasa asing murni, atau bahasa lokal murni. Namun, ada juga pengguna yang memasukkan bahasa yang sulit dalam bahasa campuran.

- Pengulangan huruf.

Dalam keadaan yang sangat marah, netizen terkadang mengetik kata-kata kasar dengan mengulangi beberapa huruf untuk menunjukkan kemarahan mereka. Contoh untuk pola ini adalah baaaanggsaaaattt (bangsat), taaiiii (tai), annjiiiiingg (anjing), dan lain-lain.

- Menghapus bagian vokalnya.

Di banyak jejaring sosial terutama di Twitter yang membatasi jumlah huruf dalam sebuah postingan, sering ditemukan netizen yang mengetik kata kasar dengan menghapus vokal. Sebagai contoh, mereka mengetik *bgst* untuk mengucapkan *bangsat* dan mengetik *ajg* untuk mengatakan *anjing*.



- Pergantian huruf.

Banyak netizen Indonesia mengganti beberapa huruf dalam kata-kata kasar saat menggunakan bahasa kasar karena beberapa alasan. Untuk menunjukkan kemarahan mereka, beberapa netizen misalnya mengganti t dengan d, misalnya: bangsad (bangsat), bejad (bejat), dan lain-lain. Dalam konteks lelucon, netizen biasanya mengubah s dengan c, misalnya : bangcat (bangsat), acu (asu), dan dan mengganti vokal dengan angka (mis: b4ngs4t (bangsat), b3g0 (bego)).

### 3.5. Patren Kata Kata Kasar.

#### 3.5.1 Kata - Kata Kasar Kondisi.

Kata-kata yang menggambarkan kondisi percakapan yang tidak menyenangkan biasanya digunakan sebagai kata-kata kasar

Tabel 3.1 Kata - Kata Kasar Kondisi.

Jenis Kata - Kata Kasar	<i>Pattren Kata</i>
Kondisi Gangguan Mental	Gila
	Bego
	Idiot
	Sinting
	Bodoh
	Tolol
	Sontoloyo
	Goblok
	Sarap
Penyimpangan Seksual	Lesbi
	Homo
	Banci
	Waria
Kurangnya Modernisasi	Kampungan
	Udik
	Alay
Disabilitas	Buta
	Budek
	Bolot
	Bisu
	Cacat
Tidak Memiliki Etika	Brengsek
	Bejat
	Bajingan

Yang Tidak Disetujui Oleh Tuhan atau Agama	Keparat
	Jahanam
	Haram
	Terkutuk
	Najis
Tidak menguntungkan	Celaka
	Modar
	Pantek
	Mampus
	Sialan

### 3.5.2 Kata - Kata Kasar Hewan.

Tidak semua hewan bisa digunakan sebagai kata kasar. satwa Kata-kata yang digunakan sebagai kata-kata ofensif biasanya mengacu pada kualitas Hal-hal buruk tertentu yang dibenci beberapa orang.

Tabel 3.2 Kata - Kata Kasar Hewan.

Jenis Kata - Kata Kasar	<i>Pattren</i> Kata
Hal-hal buruk tertentu yang dibenci beberapa orang	Anjing
	Bajingan
larangan dalam agama tertentu	Babi
Menyebalkan	Bangsat
	Kampret
Parasit	Lintah
Berisik	Burung beo

### 3.5.3 Kata - Kata Kasar Bagian Tubuh.

Bagian tubuh yang digunakan sebagai kata kasar biasanya berkaitan erat dengan kutukan atau bagian tubuh terutama mata

Tabel 3.3 Kata - Kata Kasar Bagian Tubuh

Jenis Kata - Kata Kasar	<i>Pattren</i> Kata
Yang berhubungan dengan mata	Matamu
	Mata duitan
	Mata elang
	Mata keranjang

### 3.5.4 Kata - Kata Kasar Objek.

Seperti keberadaan binatang dan makhluk astral , benda biasanya dibuang karena kualitas buruk seperti bau

Tabel 3.4 Kata - Kata Kasar Objek

Jenis Kata - Kata Kasar	<i>Pattren</i> Kata
Objek yang menjijikan / bau	Tai
	Bangkai
	Sampah
	Ampas

### 3.5.5 Kata - Kata Kasar Anggota keluarga.

Orang Indonesia biasanya menambahkan akhiran-mu pada kata yang mengacu ke hubungan sebagai kutukan.

Tabel 3.5 Kata - Kata Kasar Anggota keluarga

Jenis Kata - Kata Kasar	<i>Pattren</i> Kata
Menggunakan akhiran-mu yang mengacu pada kutukan	Ibumu
	Bapakmu
	Kakekmu
	Nenekmu

### 3.5.6 Kata - Kata Kasar Profesi.

Pekerjaan seseorang, terutama pekerjaan kelas rendah yang dilarang oleh agama, sering digunakan oleh orang Indonesia sebagai kata-kata kasar contoh perkerjaan itu sebagai berikut.

Tabel 3.6 Kata - Kata Profesi

Jenis Kata - Kata Kasar	<i>Pattren</i> Kata
Profesi terutama perkerjaan kelas rendah yang dilarang agama	Maling
	Sundel
	Copet
	Lonte
	Cecenguk
	Kacung
	Pecun
	Pelacur
	Jablay
	Perek
	Jalang

### 3.6 Pseudocode algoritma brute force string matching.

Berdasarkan analisis di atas tentang cara kerja algoritma *brute force*, di bawah ini adalah *pseudocode* cara kerja algoritma *brute force* untuk pencarian / pencocokan kata - kata kasar di dalam komentar *review* dalam aplikasi . di mana  $m$  adalah panjang *Pattern*, dan  $n$  didefinisikan sebagai panjang teks.

Tabel 3.7 Algoritma *brute force string matching* pada Program.

Procedure Algoritma

Input :

$m$ ,  $n$  sebagai length dari pattern dan teks  
 $x$  sebagai pattern  
 $y$  sebagai teks

proses

perulangan  $i= 0$  ke  $m-n$  maka  
 $j=0$   
 jika  $j$  lebih kecil dari  $n$  dan  $x [ i \text{ ditambah } 1 ] = y [j]$   
 maka  
 $j = j \text{ ditambah } 1$   
 akhir jika  
 jika  $j$  lebih besar sama dengan  $n$   
 maka  
 pattern= benar (ditemukan)  
 akhir jika  
 akhir perulangan

output :

pattern ditemukan