



**RANCANG BANGUN APLIKASI PENCARIAN DATA  
GANDENGAN TRUK DENGAN MENGGUNAKAN  
ALGORITMA BOYER MOORE**

Disusun dan Diajukan Untuk Melengkapi Dan Memenuhi Persyaratan Ujian Akhir  
Memperoleh Gelar Sarjana Pada Fakultas Sains Dan Teknologi  
Universitas Pembangunan Panca Budi |  
Medan

---

**SKRIPSI**

---

**OLEH**

**NAMA : SITI SYAHRANI**  
**N.P.M : 1414370155**  
**PROGRAM STUDI : SISTEM KOMPUTER**

**FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS PEMBANGUNAN PANCA BUDI  
MEDAN  
2019**

## ABSTRAK

Siti Syahrani

**Rancang Bangun Aplikasi Pencarian Data Gandengan Truk Dengan  
Menggunakan Algoritma Boyer Moore  
2019**

Proses pencarian data merupakan kegiatan penting dalam pemrosesan data. Dalam masalah Pencarian Data Gandengan Truk, diperlukan ketelitian untuk mencari data gandengan truk sesuai dengan nomor *container* yang dibawanya. Tingginya denda atas kesalahan dan ketidaktepatan waktu dalam proses pengembalian *container* dapat menyebabkan kerugian besar bagi perusahaan, sehingga diperlukan suatu teknik pencarian yang efisien. Algoritma *Boyer Moore* merupakan suatu solusi pencarian yang efisien yang dapat melakukan perbandingan *pattern* mulai dari kanan ke kiri. Jika terjadi ketidakcocokan *string* dari kanan *pattern* maka ketidakcocokan akan membantu kita untuk menggerakkan *pattern* tersebut dengan jarak yang lebih jauh. Gerakan melompat ini akan memberikan informasi berapa banyak *pattern* harus digeser untuk mencocokkan karakter terakhir yang cocok dengan kemunculan awal *pattern*. Artinya, akan lebih signifikan dalam mengurangi proses perbandingan sehingga lebih efisien dalam mencari data yang dibutuhkan.

**Kata kunci : Boyer Moore, Container, Gandengan Truk, Pencarian data.**

## DAFTAR ISI

Halaman

<b>COVER</b>	
<b>LEMBAR PENGESAHAN</b>	
<b>ABSTRAK</b>	
<b>KATA PENGANTAR</b> .....	<b>i</b>
<b>DAFTAR ISI</b> .....	<b>iii</b>
<b>DAFTAR GAMBAR</b> .....	<b>vii</b>
<b>DAFTAR TABEL</b> .....	<b>ix</b>
<b>BAB I PENDAHULUAN</b> .....	<b>1</b>
1.1 Latar Belakang Masalah .....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Masalah .....	3
1.4 Tujuan Penulisan .....	3
1.5 Metode penelitian .....	3
1.6 Sistematika Penulisan.....	5
<b>BAB II LANDASAN TEORI</b> .....	<b>7</b>
2.1 Algoritma .....	7
2.2 Algoritma Pencarian (String Matching).....	8
2.2.1 Cara Kerja Algoritma String Matching.....	9
2.3 Algoritma Boyer Moore .....	10

2.3.1	Langkah-langkah Algoritma Boyer Moore.....	11
	A. Menghitung Tabel Occurance Heuristic (OH)	12
	B. Menghitung Tabel Match Heuristic (MH)	13
	C. Pencarian Dengan Boyer Moore	15
2.4	Truk Peti Kemas (Truk Container).....	18
2.5	Gandengan Truk (Chassis Semi Trailer) .....	19
2.6	Peti Kemas (Container).....	19
2.7	Ukuran Peti Kemas (Container) .....	20
2.8	Flowchart .....	22
2.9	Data Flow Diagram (DFD).....	24
2.10	Entity Relationship Diagram (ERD).....	26
2.11	Microsoft Visual Studio 2010.....	29
2.12	Microsoft Access .....	30
<b>BAB III</b>	<b>METODE PENELITIAN.....</b>	<b>32</b>
3.1	Analisa Kebutuhan Sistem .....	32
3.2	Analisa Sistem Terdahulu.....	33
3.3	Perhitungan Menggunakan Algoritma Boyer Moore .....	34
	A. Menghitung Nilai Occurrence Heuristic (OH)	34
	B. Menghitung Nilai Match Heuristic (MH)	36
	C. Mencocokkan Teks Dengan Pattern	37
3.4	Flowchart.....	40

3.5	Diagram Context.....	41
3.6	Diagram Alir Data.....	42
3.7	Entity Relationship Diagram.....	43
3.8	Perancangan Database.....	44
3.9	Perancangan Sistem.....	45
	A. Perancangan Form Login	45
	B. Perancangan Halaman Awal	46
	C. Perancangan Halaman Pencarian Data	47
	D. Perancangan Halaman Input Data	48
	E. Perancangan Halaman Menu tentang	49
<b>BAB IV</b>	<b>HASIL PENELITIAN DAN PEMBAHASAN .....</b>	<b>50</b>
4.1	Implementasi Sistem.....	50
4.2	Tujuan Implementasi Sitem.....	50
4.3	Komponen Utama Dalam Sistem.....	51
4.4	Tampilan Aplikasi .....	52
	A. Tampilan Form Login	52
	B. Tampilan Halaman Awal	52
	C. Tampilan Pencarian Data	53
	D. Tampilan Input Data	53
	E. Tampilan Menu Edit Data	54
	F. Tampilan Menu tentang	54

4.5	Pengujian Sistem .....	55
<b>BAB V PENUTUP .....</b>		<b>57</b>
5.1	Kesimpulan.....	57
5.2	Saran.....	57

**DAFTAR PUSTAKA**

**BIOGRAFI PENULIS**

**LAMPIRAN-LAMPIRAN**

## DAFTAR GAMBAR

<b>No</b>	<b>Judul</b>	<b>Hal</b>
1.	Truk Peti Kemas (Truk Container).....	19
2.	Kepala Truk dan Gandengan (Chassis Semi Trailer).....	20
3.	Peti Kemas 20 Kaki dan 40 Kaki.....	22
4.	Contoh Entity Relationship Diagram.....	28
5.	Halaman Awal Microsoft Visual Studio 2010.....	31
6.	Microsoft Access 2013.....	32
7.	Flowchart Sistem Sebelumnya.....	33
8.	Flowchart Algoritma Boyer Moore.....	42
9.	Diagram Context.....	43
10.	Diagram Alir Data.....	44
11.	Entity Relationship Diagram.....	45
12.	Tampilan Form Login.....	47
13.	Rancangan Tampilan Halaman Awal.....	48
14.	Rancangan Halaman Pencarian Data Gandengan.....	49
15.	Rancangan Tampilan Input Data.....	50
16.	Tampilan Menu Tentang.....	51
17.	Form Menu Login.....	54
18.	Tampilan Halaman Utama .....	55
19.	Tampilan Menu Pencarian Data.....	55

<b>No</b>	<b>Judul</b>	<b>Hal</b>
20.	Tampilan Menu Input Data .....	56
21.	Tampilan Menu Edit Data .....	56
22.	Tampilan Menu Tentang.....	57



## DAFTAR TABEL

<b>No</b>	<b>Judul</b>	<b>Hal</b>
1.	Menghitung Nilai Occurrence Heuristic (a) .....	12
2.	Menghitung Nilai Occurrence Heuristic (b) .....	13
3.	Menghitung Nilai Match Heuristic (a) .....	14
4.	Menghitung Nilai Match Heuristic (b) .....	15
5.	Menghitung Nilai Match Heuristic (c) .....	15
6.	Menghitung Nilai Match Heuristic (d) .....	15
7.	Perbandingan Nilai Occurrence Heuristic dan Match Heuristic .....	16
8.	Pencarian Dengan Boyer Moore (a) .....	17
9.	Pencarian Dengan Boyer Moore (b) .....	17
10.	Pencarian Dengan Boyer Moore (c) .....	18
11.	Pencarian Dengan Boyer Moore (d) .....	18
12.	Ukuran Peti Kemas .....	22
13.	Simbol-simbol Flowchart .....	23
14.	Simbol-simbol Data Flow Diagram .....	27
15.	Tabel Data Gandengan Truk .....	33
16.	Contoh Kemiripan Data .....	35
17.	Menghitung Nilai OH (a) .....	36
18.	Menghitung Nilai OH (b) .....	36

<b>No</b>	<b>Judul</b>	<b>Hal</b>
19.	Menghitung Nilai MH .....	37
20.	Menghitung Nilai MH (a).....	38
21.	Menghitung Nilai MH (b).....	38
22.	Perbandingan Nilai OH dan MH .....	39
23.	Pencarian Menggunakan Boyer Moore (a).....	39
24.	Pencarian Menggunakan Boyer Moore (b) .....	40
25.	Pencarian Menggunakan Boyer Moore (c).....	40
26.	Pencarian Menggunakan Boyer Moore (d) .....	40
27.	Pencarian Menggunakan Boyer Moore (e).....	41
28.	Pencarian Menggunakan Boyer Moore (f) .....	41
29.	Tabel Login.....	46
30.	Tabel Data Truk.....	46
31.	Pengujian Sistem.....	57

## KATA PENGANTAR

Puji Syukur penulis panjatkan kepada Tuhan Yang Maha Esa, yang telah memberikan rahmat-Nya kepada peneliti, sehingga Skripsi ini dapat diselesaikan oleh peneliti tepat pada waktunya dengan judul Rancang Bangun Aplikasi Pencarian Data Gandengan Truk Dengan Menggunakan Algoritma Boyer Moore.

Skripsi ini dilakukan guna memenuhi salah satu syarat pemenuhan kurikulum dalam menyelesaikan pendidikan pada Program Studi S1 Sistem Komputer Fakultas Sains Dan Teknologi Universitas Pembangunan Panca Budi Medan. Pada kesempatan ini, penulis menyampaikan rasa terima kasih dan penghargaan yang sebesar-besarnya kepada :

1. Kepada Kedua Orang Tua saya yang telah banyak memberikan dukungan dan bantuan baik moril maupun material selama penulis mengikuti pendidikan hingga selesainya Skripsi ini.
2. Bapak Dr. H. Muhammad Isa Indrawan, SE, MM, selaku Rektor Universitas Pembangunan Panca Budi Medan.
3. Ibu Sri Shindi Indira, ST.,M.Sc , selaku Dekan Fakultas Sains dan Teknologi Universitas Pembangunan Panca Budi Medan.
4. Bapak Dr. Muhammad Iqbal S.Kom., M.Kom, Selaku kepala program studi Sistem Komputer Universitas Pembangunan Panca Budi Medan.
5. Bapak Suherman, S.Kom., M Kom, selaku Dosen Pembimbing I yang telah memberikan pengarahan dan petunjuk dalam Skripsi ini.



## BAB I

### PENDAHULUAN

#### 1.1. Latar Belakang Masalah

Kebutuhan sarana transportasi khususnya dalam pengiriman barang semakin meningkat seiring bertambahnya jumlah permintaan barang. PT. Setia Kawan Transporindo merupakan perusahaan yang bergerak dibidang jasa pengangkutan atau pengiriman barang. Tergabung didalam Angsuspel (Angkutan Khusus Pelayaran), PT. Setia Kawan Transporindo bertanggung jawab mengantarkan barang dalam bentuk peti kemas (*Container*) dari Pelabuhan hingga ke tempat tujuan maupun sebaliknya.

Lama proses bongkar-muat barang dipengaruhi oleh batas *demurrage*. *Demurrage* adalah batas waktu pemakaian peti kemas di dalam pelabuhan (*container yard*). (<http://www.beacukai.go.id/berita/mengenal-biaya-demurrage-saat-impor-dan-ekspor-.html>). Besarnya denda yang harus dibayarkan jika melebihi batas *free time demurrage* menjadi bahan pertimbangan pihak jasa pengangkutan agar lebih tepat waktu dalam pengiriman dan penarikan *container* dengan gandengan yang tepat.

Untuk meningkatkan kinerja dan efisiensi waktu dalam mencari data gandengan truk berdasarkan nomor *container* yang tepat, maka dirancanglah sebuah aplikasi yang dapat membantu mencari data gandengan truk *container* dengan menggunakan algoritma *Boyer-Moore*. Algoritma *Boyer-Moore*

dipublikasikan oleh Robert S. Boyer, dan J. Strother Moore pada tahun 1977. Ide utama dari algoritma ini adalah dengan melakukan pencocokan dari paling kanan string yang dicari (karakter terakhir pada *pattern*). Dengan penerapan Algoritma *Boyer-Moore* pada pencarian data gandengan truk diharapkan dapat meminimalisasi kesalahan dan mengurangi kerugian yang mungkin akan ditimbulkan akibat kesalahan tersebut.

Berdasarkan latar belakang yang telah diuraikan diatas, maka dalam penulisan skripsi ini penulis memilih judul “**Rancang Bangun Aplikasi Pencarian Data Gandengan Truk Dengan Menggunakan Algoritma Boyer Moore.**”

## **1.2. Rumusan Masalah**

Dalam Pelaksanaan Penelitian Skripsi ini terdapat beberapa permasalahan yang menjadi dasar utama pembahasan, diantaranya sebagai berikut :

- a. Bagaimana aplikasi yang dirancang dapat meningkatkan efisiensi waktu yang diperlukan dalam pencarian data gandengan truk pada PT. Setia Kawan Transporindo.
- b. Bagaimana penerapan algoritma *Boyer Moore* untuk mencari data gandengan truk yang tepat berdasarkan nomor *container* yang telah diinput didalam aplikasi yang dirancang.

### 1.3. Batasan Masalah

Batasan masalah dalam laporan skripsi ini adalah :

- a. Aplikasi yang dirancang hanya untuk penyimpanan data serta pencarian data gandengan truk di PT. Setia Kawan Transporindo.
- b. Proses pencarian data gandengan truk dibatasi pada data *Container* yang telah tersimpan di aplikasi.
- c. Proses pencarian data gandengan truk dibatasi pada data bulan Juli-September 2018.
- d. Aplikasi tidak membahas harga dan laba perusahaan.

### 1.4. Tujuan Penulisan

Tujuan dari penulisan skripsi ini adalah :

- a. Untuk memaksimalkan proses pencarian data gandengan truk sehingga mengurangi kerugian yang mungkin muncul.
- b. Mengimplementasikan algoritma *Boyer Moore* dalam mengatasi permasalahan pencarian data gandengan truk.

### 1.5. Metode Penelitian

Metode penelitian yang digunakan dalam pembuatan aplikasi yaitu metode *waterfall*. Menurut (Pressman, 2010) model *waterfall* adalah model klasik yang bersifat sistematis, berurutan dalam membangun *software*. Pemilihan metode ini

Karena prosesnya lebih terstruktur, hal ini membuat kualitas *software* baik dan tetap terjaga.

Berikut adalah tahapan-tahapan yang digunakan dalam metode *waterfall*:

**a. Analisis Kebutuhan**

Informasi dikumpulkan untuk memahami perangkat lunak dan batasan perangkat yang diharapkan. Informasi ini biasanya dapat diperoleh melalui wawancara, diskusi atau survei langsung.

**b. Perancangan Sistem**

Pada tahap ini perangkat lunak dirancang untuk menentukan bagaimana desain dan arsitektur sistem dari awal hingga akhir pembuatan. Perancangan sistem menggunakan *Flowchart*, *Data Flow Diagram* (DFD), dan sebagainya. Perancangan aplikasi dibuat menggunakan *software* Visual Studio 2010.

**c. Implementasi Dan Pengujian**

Perancangan perangkat lunak dikembangkan kemudian dilakukan pengujian untuk menjamin kesesuaian program dengan tujuan yang diharapkan. Sehingga mampu menjalankan perintah sesuai dengan algoritma *Boyer Moore* yang digunakan.

**d. Tahap Pemeliharaan (*Maintenance*)**

Tahap akhir dalam model *waterfall*. Pada tahap ini Perangkat lunak yang sudah jadi dijalankan serta dilakukan pemeliharaan mencakup koreksi dari berbagai



*error* yg tdk ditemukan pada tahap-tahap sebelumnya untuk menjaga terjadinya kerusakan-kerusakan yang tidak diinginkan.

## **1.6. Sistematika Penulisan**

Penulisan Skripsi ini akan dibahas dan disusun bab demi bab dengan sistematika penulisan sebagai berikut :

### **BAB I : PENDAHULUAN**

Menjelaskan tentang Latar belakang, Rumusan Masalah, Batasan Masalah, Tujuan Penulisan, Metodologi Penelitian, dan Sistematika Penulisan

### **BAB II : LANDASAN TEORI**

Meliputi teori-teori yang relevan dengan masalah pokok yang akan dikaji. Seperti pembahasan mengenai Aplikasi, Sistem Operasi, *Flowchart*, dan piranti lunak pendukung yang digunakan dalam perancangan program.

### **BAB III : METODE PENELITIAN**

Membuat analisa Permasalahan dimana terdiri dari kelemahan sistem yang berjalan serta rancangan yang akan dibangun, DFD, Struktur Program, Perancangan Antarmuka serta *Flowchart*.

### **BAB IV : HASIL PENELITIAN DAN PEMBAHASAN**

Memaparkan perangkat lunak hasil rancangan. Selain itu, bab ini juga menyajikan hasil analisis dari perangkat lunak yang telah dibuat.

## **BAB V : PENUTUP**

Bagian Penutup berisi kesimpulan hasil analisa bab-bab sebelumnya serta saran untuk kemajuan perancangan sistem maupun program.

## BAB II

### LANDASAN TEORI

#### 2.1 Algoritma

Menurut, Munir (2001:4) Algoritma adalah urutan langkah-langkah logis penyelesaian masalah yang disusun secara sistematis, sedangkan menurut Kamus Besar Bahasa Indonesia (1976:30) Algoritma adalah urutan logis pengambilan putusan untuk pemecahan masalah.

Adapun definisi Algoritma meliputi (Suarga, 2004 : 1) :

- a. Teknik penyusunan langkah-langkah penyelesaian masalah dalam bentuk kalimat dengan jumlah kata terbatas, tetapi tersusun secara logis dan sistematis.
- b. Suatu prosedur yang jelas untuk menyelesaikan suatu persoalan dengan menggunakan langkah-langkah tertentu dan terbatas jumlahnya

Berdasarkan cirinya sifat utama dari Algoritma, yaitu (Suarga, 2004 :2):

- a. *Input* : suatu Algoritma memiliki *input* atau kondisi awal sebelum Algoritma dilaksanakan dan bisa berupa nilai-nilai pengubah yang diambil dari himpunan khusus.
- b. *Output* : suatu Algoritma akan menghasilkan *output* setelah dilaksanakan, atau Algoritma akan mengubah kondisi awal menjadi kondisi akhir, dimana nilai *output* diperoleh dari nilai *input* yang telah diproses melalui Algoritma.
- c. *Definiteness* : langkah-langkah yang dituliskan dalam Algoritma terdefinisi dengan jelas sehingga mudah dilaksanakan oleh pengguna Algoritma.

- d. *Finiteness* : suatu Algoritma harus memberi kondisi akhir atau *output* setelah melakukan sejumlah langkah yang terbatas jumlahnya untuk setiap kondisi awal atau *input* yang diberikan.
- e. *Effectiveness* : setiap langkah dalam Algoritma bisa dilaksanakan dalam suatu selang waktu tertentu sehingga pada akhirnya memberi solusi sesuai yang diharapkan.
- f. *Generality* : langkah-langkah Algoritma berlaku untuk setiap himpunan *input* yang sesuai dengan persoalan yang akan diberikan, tidak hanya untuk himpunan tertentu.

## 2.2 Algoritma Pencarian (*String Matching*)

Pencarian adalah suatu proses menemukan nilai tertentu didalam sekumpulan data dengan tipe data yang sama. Algoritma *string matching* adalah proses pencarian semua kemunculan *query* yang selanjutnya disebut *pattern* ke dalam *string* yang lebih panjang (teks). Sebagian besar algoritma yang dipelajari oleh ilmuwan komputer adalah algoritma pencarian (Nievergelt, 2000).

Algoritma *string matching* mempunyai tiga komponen utama, yaitu:

- a. *Pattern*, yaitu deretan karakter yang akan dicocokkan dengan teks, dinyatakan dengan  $x[0..m-1]$ , panjang *pattern* dinyatakan dengan  $m$ .
- b. Teks, yaitu tempat pencocokan *pattern* dilakukan, dinyatakan dengan  $y[0..n-1]$ , panjang teks dinyatakan dengan  $n$ .
- c. Alfabet, yang berisi semua simbol yang digunakan oleh bahasa pada teks dan *pattern*, dinyatakan dengan  $\Sigma$  dengan ukuran dinyatakan dengan  $A_{size}$ .

### 2.2.1 Cara Kerja Algoritma String Matching

Persoalan pencarian string dirumuskan sebagai berikut (Munir, 2004 : 1)

Diberikan :

- a. Sebuah teks (*text*), yaitu sebuah (*long*) string yang panjangnya  $n$  karakter.
- b. *Pattern*, yaitu sebuah string dengan panjang  $m$ .

Dengan sebuah nilai karakter ( $m < n$ ) yang akan dicari dalam teks. Dalam Algoritma pencocokan *string*, teks diasumsikan berada di dalam memori, sehingga bila kita mencari string di dalam sebuah arsip, maka semua isi arsip perlu dibaca terlebih dahulu kemudian disimpan di dalam memori. Jika *pattern* muncul lebih dari sekali di dalam teks, maka pencarian hanya akan memberikan keluaran berupa lokasi *pattern* ditemukan pertama kali.

Algoritma *string matching* dapat diklasifikasikan menjadi 3 bagian menurut arah pencariannya, yakni (Charras, 1997 : 12).

- a. *From left to right* Dari arah yang paling alami, dari kiri ke kanan, yang merupakan arah untuk membaca. Algoritma yang termasuk kategori ini adalah *Algoritma brute force*, *Algoritma knuth moris Pratt*.
- b. *From right to left* Dari arah kanan ke kiri, arah yang biasanya menghasilkan hasil terbaik secara partikal. Algoritma yang termasuk kategori ini adalah *Algoritma boyer- moore*.
- c. *In a specific order* Dari arah yang ditentukan secara spesifik oleh Algoritma tersebut, arah ini menghasilkan hasil terbaik secara teoritis. Algoritma yang

termasuk kategori ini adalah *Algoritma colossi* dan *Algoritma crochemore-perrin*.

### 2.3 Algoritma Boyer Moore

Algoritma *Boyer-Moore* adalah salah satu algoritma pencarian string yang dipublikasikan oleh Robert S. Boyer, dan J. Strother Moore pada tahun 1977. Tidak seperti algoritma pencarian string yang ditemukan sebelumnya, algoritma *Boyer-Moore* mulai mencocokkan karakter dari sebelah kanan *pattern*.

. Pada algoritma ini pencocokan kata dimulai dari karakter terakhir kata kunci menuju karakter awalnya. Jika terjadi perbedaan Antara karakter terakhir kata kunci dengan kata yang dicocokkan, maka karakter-karakter dalam potongan kata yang dicocokkan tadi akan diperiksa satu per satu. Hal ini dimaksudkan untuk mendeteksi apakah ada karakter dalam potongan kata tersebut yang sama dengan karakter yang ada pada kata kunci. Apabila terdapat kesamaan, maka kata kunci akan digeser sedemikian rupa sehingga posisi karakter yang sama terletak sejajar, dan kemudian dilakukan kembali pencocokan karakter terakhir dari kata kunci. Sebaliknya jika tidak terdapat kesamaan karakter, maka seluruh karakter kata kunci akan bergeser ke kanan sebanyak  $m$  karakter, di mana  $m$  adalah panjang karakter dari kata kunci.

#### 2.3.1 Langkah-Langkah Pada Algoritma Boyer Moore

Berikut adalah langkah-langkah dalam pencarian kata menggunakan algoritma Boyer Moore:

- a. Buat tabel pergeseran *pattern* yang dicari (P) dengan pendekatan *Match Heuristic* (MH) dan *Occurence Heuristic* (OH), untuk menentukan jumlah pergeseran yang akan dilakukan jika mendapat karakter tidak cocok pada proses pencocokan dengan teks (T).
- b. Jika dalam proses perbandingan terjadi ketidakcocokan antara pasangan karakter pada *pattern* dan karakter teks, pergeseran dilakukan dengan memilih salah satu nilai pergeseran dari dua tabel, dan memiliki nilai pergeseran paling besar dari tabel *Match Heuristic* dan *Occurence Heuristic* .
- c. Dua kemungkinan penyelesaian dalam melakukan pergeseran *pattern*, Jika karakter yang tidak cocok, tidak ada pada *pattern* maka pergeseran adalah sebanyak jumlah karakter pada *pattern*. dan jika karakter yang tidak cocok, ada pada *pattern*, maka banyaknya pergeseran bergantung dari nilai pada tabel *Match Heuristic* dan *Occurence Heuristic*.
- d. Jika karakter pada teks yang sedang dibandingkan cocok dengan karakter pada *pattern*, maka posisi karakter pada *pattern* dan teks diturunkan sebanyak 1 posisi, kemudian lanjutkan dengan pencocokan pada posisi tersebut dan seterusnya. Jika kemudian terjadi ketidakcocokan karakter *pattern* dan teks, maka pilih nilai pergeseran terbesar dari tabel *match heuristic* dan nilai tabel *occurence heuristic*.

#### A. Menghitung Tabel *Occurence Heuristic* (OH)

Cara menghitung tabel *occurence heuristic* (OH) :

Contoh Pattern : TOPI

Panjang Karakter (L) : 4

**Tabel 1. Menghitung Nilai *Occurrence Heuristic* (a)**

<b>Index</b>	0	1	2	3
<b>Pattern</b>	T	O	P	I
<b>Occurrence Heuristic</b>				

Langkah-langkah pemberian nilainya adalah sebagai berikut :

- a. Mencari nilai OH dengan menggunakan rumus :

$$\text{OH} = (\text{Length} - 1 - \text{Index})$$

Length = panjang karakter = 4

- a) Karakter pertama adalah “T” dengan index = 0

$$\text{OH} = 4 - 1 - 0 = 3$$

- b) Karakter kedua adalah “O” dengan index = 1

$$\text{OH} = 4 - 1 - 1 = 2$$

- c) Karakter ketiga adalah “P” dengan index = 2

$$\text{OH} = 4 - 1 - 2 = 1$$

- d) Karakter keempat adalah “I” dengan index = 3

$$\text{OH} = 4 - 1 - 3 = 0$$

- b. Setelah didapat nilai OH dari setiap karakter maka masukkan nilai pada tabel Occurance Heuristic (OH), sehingga hasilnya :



**Tabel 2. Menghitung Nilai *Occurrence Heuristic* (b)**

<b>Index</b>	0	1	2	3
<b>Pattern</b>	T	O	P	I
<b>Occurrence Heuristic</b>	3	2	1	0

- c. Jika terdapat karakter yang berulang pada pattern, gunakan nilai OH terkecil pada karakter yang berulang. Dalam kasus ini tidak ada karakter yang berulang sehingga nilai OH tidak perlu diganti.

### **B. Menghitung Tabel *Match Heuristic* (MH)**

Cara menghitung tabel *Match Heuristic* (MH)

Contoh Pattern : TOPI

Panjang Karakter (L) : 4

**Tabel 3. Menghitung Nilai *Match Heuristic* (a)**

<b>Index</b>	0	1	2	3
<b>Pattern</b>	T	O	P	I
<b>Match Heuristic</b>				

Langkah-langkah pemberian nilainya adalah sebagai berikut :

- a. Berikan nilai 1 ( nilai default MH ) pada index terbesar, dalam kasus ini index terbesar adalah “I”.



- d. Karena tidak ada karakter yang cocok lagi maka nilai pergeseran untuk karakter berikutnya adalah sebanyak panjang karakter ( $L$ ). sehingga didapat nilai untuk tabel MH sebagai berikut :

**Tabel 6. Menghitung Nilai Match Heuristic (d)**

<b>Index</b>	0	1	2	3
<b>Pattern</b>	T	O	P	I
<b>Match Heuristic</b>	4	4	4	1

### C. Pencarian Dengan Boyer Moore

Syarat-syarat pencarian dengan algoritma *Boyer Moore* :

- Jika karakter yang dibanding tidak ada pada *pattern* maka nilai pergeserannya sepanjang jumlah *pattern*.
- Untuk pergeseran bandingkan nilai OH dan MH ambil nilai terbesar sebagai keputusan pergeseran.
- Jika semua karakter telah cocok, artinya *pattern* telah ditemukan di dalam teks.

**Tabel 7. Perbandingan Nilai Occurrence Heuristic dan Match Heuristic**

<b>Index</b>	0	1	2	3
<b>Pattern</b>	T	O	P	I
<b>Occurrence Heuristic</b>	3	2	1	0
<b>Match Heuristic</b>	4	4	4	1

Contoh implementasi pencarian pada teks:

**Tabel 8. Pencarian Dengan Boyer Moore (a)**

<b>Teks</b>	B	E	L	I	E	M	P	A	T	T	O	P	I
<b>Pattern</b>	T	O	P	I									
<b>index</b>	0	1	2	3									

- a) Bandingkan karakter terakhir pada pattern dengan karakter sejajar di atasnya, apabila karakter terakhir sudah cocok periksa kembali karakter selanjutnya disebelah kiri.
- b) Jika karakter setelahnya tidak cocok maka geser karakter sepanjang nilai perbandingan pada OH dan MH. Karena nilai perbandingan terbesar adalah 4 maka begeser sejauh 4 langkah.

Nilai OH (Teks) :  $L = 4$

Nilai MH (Pattern) :  $P = 4$

**Tabel 9. Pencarian Dengan Boyer Moore (b)**

<b>Teks</b>	B	E	L	I	E	M	P	A	T	T	O	P	I
<b>Pattern</b>					T	O	P	I					
<b>index</b>					0	1	2	3					

- c) Bandingkan kembali karakter terakhir pada pattern dan teks sejajar di atasnya, Karakter "A" memiliki nilai OH = 4 dan "T" memiliki nilai MH = 1. Karena  $4 > 1$  maka nilai pergeserannya = 4

**Tabel 10. Pencarian Dengan Boyer Moore (c)**

<b>Teks</b>	B	E	L	I	E	M	P	A	T	T	O	P	I
<b>Pattern</b>									T	O	P	I	
<b>index</b>									0	1	2	3	

- d) Ulangi perbandingan seperti cara sebelumnya. Karakter akhir pada pattern “I” memiliki nilai MH = 1, dan “P” pada teks memiliki nilai = 1. Karena nilai OH dan MH sama maka bergeser sepanjang 1 karakter

**Tabel 11. Pencarian Dengan Boyer Moore (d)**

<b>Teks</b>	B	E	L	I	E	M	P	A	T	T	O	P	I
<b>Pattern</b>										T	O	P	I
<b>index</b>										0	1	2	3

- e) Bandingkan karakter dari kanan pattern pada teks yang sejajar, apabila sudah cocok artinya pattern telah ditemukan pada teks.

#### 2.4 Truk Peti Kemas (Truk Container)

Truk peti kemas disebut juga truk kontainer adalah kendaraan pengangkut peti kemas terdiri yang dari kendaraan penarik (*tractor head*) dan kereta tempelan di mana peti kemas ditempatkan.

Tren angkutan barang dengan peti kemas meningkat dengan cepat karena intermodalitinya yang tinggi sehingga mempermudah bongkar-muat/*handling* dari

barang yang mengakibatkan biaya angkutan secara keseluruhan menurun dengan drastis. Disamping itu keamanan dari barang juga lebih tinggi.



**Gambar 1. Truk Peti Kemas (Truk Container)**

Sumber : PT. Setia Kawan Transporindo

## 2.5 Gandengan truk (*Chassis Semitrailer*)

Gandengan truk atau *Chassis Semitrailer* adalah pasangan antara kepala truk yang menarik pengangkut agar dapat mengangkut muatan sesuai dengan fungsinya. Pengangkut yang ditarik inilah yang disebut gandengan truk.



**Gambar 2. Kepala Truk Dan Gandengan (*Chassis Semi Trailer*)**

Sumber : <https://www.kompasiana.com/jiminandri//hebatnya-sopir-truk-kontainer>

## 2.6 Peti Kemas (*Container*)

Peti kemas adalah peti atau kotak yang memenuhi persyaratan teknis sesuai dengan *International Organization for Standardization* (ISO) sebagai alat atau perangkat pengangkutan barang yang bisa digunakan diberbagai moda, mulai dari moda jalan dengan truk peti kemas, kereta api dan kapal peti kemas laut. (Wikipedia).

Berbagai variasi bentuk peti kemas digunakan untuk barang-barang yang spesifik namun menggunakan ukuran yang standar untuk mempermudah handling dan perpindahan moda angkutan. Berikut adalah jenis-jenis peti kemas :

- a) Peti kemas barang umum untuk diisi kotak-kotak, karung, drum, palet dls, jenis yang paling banyak digunakan
- b) Peti kemas tangki yaitu tangki baja yang dibangun di dalam kerangka container digunakan untuk mengangkut Tanki yang di dalamnya diisi barang-barang yang berbahaya, misalnya gas, minyak, bahan kimia yang mudah meledak.
- c) Peti kemas berventilasi untuk barang organik yang membutuhkan ventilasi
- d) Peti kemas Generator
- e) Peti kemas berpendingin digunakan untuk mengangkut barang – barang yang memerlukan suhu pendingin, misalnya untuk jenis sayur-sayuran, daging dll.
- f) Peti kemas curah, digunakan untuk mengangkut muatan curah, misalnya beras, gandum, dll.
- g) Peti kemas yang diperlengkapi dengan isolasi

- h) Peti kemas dengan pintu disamping digunakan untuk mengangkut muatan yang ukurannya tidak memungkinkan dimasukan dari pintu belakang Petikemas. Jadi semua sisi Peti kemas harus dibuka. Misalnya alat – alat berat.
- i) Collapsible ISO, Jenis peti kemas Tabung gas, tangki, generator biasanya tidak dilengkapi dengan dinding samping, depan belakang dan atas.

## 2.7 Ukuran Peti Kemas (*Container*)

Ukuran peti kemas standar yang digunakan ditampilkan dalam tabel berikut:

**Tabel 12. Ukuran Peti Kemas (*Container*)**

		Peti kemas 20 kaki		Peti kemas 40 kaki		Peti kemas 45 kaki	
		inch	metrik	inch	metrik	inch	metrik
dimensi luar	panjang	20' 0"	6,058 m	40' 0"	12,192 m	45' 0"	13,716 m
	lebar	8' 0"	2,438 m	8' 0"	2,438 m	8' 0"	2,438 m
	tinggi	8' 6"	2,591 m	8' 6"	2,591 m	9' 6"	2,896 m
dimensi dalam	panjang	18' 10 <sup>5</sup> / <sub>16</sub> "	5,758 m	39' 5 <sup>45</sup> / <sub>64</sub> "	12,032 m	44' 4"	13,556 m
	lebar	7' 8 <sup>19</sup> / <sub>32</sub> "	2,352 m	7' 8 <sup>19</sup> / <sub>32</sub> "	2,352 m	7' 8 <sup>19</sup> / <sub>32</sub> "	2,352 m
	tinggi	7' 9 <sup>57</sup> / <sub>64</sub> "	2,385 m	7' 9 <sup>57</sup> / <sub>64</sub> "	2,385 m	8' 9 <sup>15</sup> / <sub>16</sub> "	2,698 m
bukaan pintu	width	7' 8 <sup>1</sup> / <sub>4</sub> "	2,343 m	7' 8 <sup>1</sup> / <sub>4</sub> "	2,343 m	7' 8 <sup>1</sup> / <sub>4</sub> "	2,343 m
	tinggi	7' 5 <sup>3</sup> / <sub>4</sub> "	2,280 m	7' 5 <sup>3</sup> / <sub>4</sub> "	2,280 m	8' 5 <sup>45</sup> / <sub>64</sub> "	2,585 m
volume		1,169 ft <sup>3</sup>	33,1 m <sup>3</sup>	2,385 ft <sup>3</sup>	67,5 m <sup>3</sup>	3,040 ft <sup>3</sup>	86,1 m <sup>3</sup>
berat kotor		52.910 pon	24.000 kg	67.200 pon	30.480 kg	67.200 pon	30.480 kg
berat kosong		4.850 pon	2.200 kg	8.380 pon	3.800 kg	10.580 pon	4.800 kg
muatan bersih		48.060 pon	21.800 kg	58.820 pon	26.680 kg	56.620 pon	25.680 kg

sumber : [http://id.wikipedia.org/wiki/Peti\\_kemas](http://id.wikipedia.org/wiki/Peti_kemas)





**Gambar 3. Peti Kemas (*Container*) 20 Kaki Dan 40 Kaki**

Sumber : <http://bayuputraperkasa.com/jenis-ukuran-kontainer>







## 2.8 Flowchart


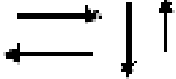
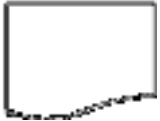
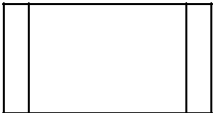
Menurut Wikipedia *Flowchart* adalah sebuah diagram dengan simbol-simbol grafis yang menyatakan aliran algoritma atau proses yang menampilkan langkah-langkah yang disimbolkan dalam bentuk kotak, beserta urutannya dengan menghubungkan masing masing langkah tersebut menggunakan tanda panah. Diagram ini bisa memberi solusi selangkah demi selangkah untuk penyelesaian masalah yang ada di dalam proses atau algoritma tersebut.

Berikut adalah Simbol-simbol yang terdapat didalam *flowchart* :

**Tabel 13. Simbol – simbol *Flowchart***

Sumber : Kristanto (2008: 75)

No.	Simbol	Fungsi
1		Terminal, menunjukkan awal dan akhir dari suatu alur program <i>flowchart</i> .
2		Proses, suatu simbol yang menunjukkan setiap pengolahan yang dilakukan oleh computer.
3		Input – output, untuk memasukkan data maupun menunjukkan hasil dari suatu proses
4		<i>Decision</i> , suatu kondisi yang akan menghasilkan beberapa kemungkinan jawaban atau pilihan.
5		<i>Preparation</i> , menunjukkan deklarasi atau pemesanan variabel atau konstanta.
6		<i>Connector</i> , suatu prosedur akan masuk dan keluar melalui symbol ini dalam lembar yang sama.

7		<p><i>Offline Connector</i>, merupakan simbol untuk masuk dan keluarnya suatu prosedur pada lembar kertas yang lain.</p>
8		<p>Arus atau <i>Flow</i>, prosedur yang dapat dilakukan dari atas ke bawah, bawah ke atas, dari ke kanan, atau dari kanan ke kiri.</p>
9		<p>Dokumen, merupakan simbol untuk data yang berbentuk informasi</p>
10		<p><i>Predefined process</i>, untuk menyatakan sekumpulan langkah proses yang ditulis sebagai prosedur</p>
11		<p><i>Direct access storage</i>, media penyimpanan data yang dapat dibaca/disimpan secara acak.</p>

## 2.9 *Data Flow Diagram (DFD)*

*Data Flow Diagram (DFD)* adalah alat pembuatan model yang memungkinkan profesional sistem untuk menggambarkan sistem sebagai suatu jaringan proses fungsional yang dihubungkan satu sama lain dengan alur data, baik secara manual maupun komputerisasi. DFD ini sering disebut juga dengan nama *Bubble chart*, *Bubble diagram*, model proses, diagram alur kerja, atau model fungsi.

Menurut jogiyanto *Data Flow Diagram* adalah Diagram yang menggunakan notasi simbol untuk menggambarkan arus data sistem". (Jogiyanto Hartono, 2005, 701).

DFD ini adalah salah satu alat pembuatan model yang sering digunakan, khususnya bila fungsi-fungsi sistem merupakan bagian yang lebih penting dan kompleks dari pada data yang dimanipulasi oleh sistem. Dengan kata lain, DFD adalah alat pembuatan model yang memberikan penekanan hanya pada fungsi sistem. DFD ini merupakan alat perancangan sistem yang berorientasi pada alur data dengan konsep dekomposisi dapat digunakan untuk penggambaran analisa maupun rancangan sistem yang mudah dikomunikasikan oleh profesional sistem kepada pemakai maupun pembuat program.

Didalam DFD terdapat 3 level, yaitu :

- a. Diagram Konteks : menggambarkan satu lingkaran besar yang dapat mewakili seluruh proses yang terdapat di dalam suatu sistem. Merupakan tingkatan

tertinggi dalam DFD dan biasanya diberi nomor 0 (nol). Semua entitas eksternal yang ditunjukkan pada diagram konteks berikut aliran-aliran data utama menuju dan dari sistem. Diagram ini sama sekali tidak memuat penyimpanan data dan tampak sederhana untuk diciptakan.

- b. Diagram Nol (diagram level-1) : merupakan satu lingkaran besar yang mewakili lingkaran-lingkaran kecil yang ada di dalamnya. Merupakan pemecahan dari diagram Konteks ke diagram Nol. di dalam diagram ini memuat penyimpanan data.
- c. Diagram Rinci : merupakan diagram yang menguraikan proses apa yang ada dalam diagram Nol.

Berikut ini merupakan simbol-simbol yang biasanya digunakan pada Data Flow Diagram (DFD) :

**Tabel 14. Simbol – simbol *Data Flow Diagram***

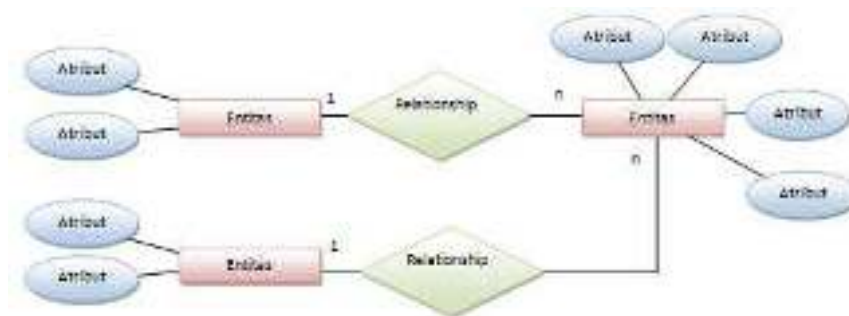
<b>Gane/Sarson</b>	<b>Yourdon/De Marco</b>	<b>Keterangan</b>
		Entitas eksternal, dapat berupa orang/unit terkait yang berinteraksi dengan sistem tetapi diluar sistem
		Orang, unit yang mempergunakan atau melakukan transformasi data. Komponen fisik tidak diidentifikasi.
		Aliran data dengan arah khusus dari sumber ke tujuan
		Penyimpanan data atau tempat data direfer oleh proses.

Sumber : <http://darmansyah.weblog.esaunggul.ac.id/2012/11/10/data-flow-diagram/>

## 2.10 Entity Relationship Diagram (ERD)

ERD merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. ERD untuk memodelkan struktur data dan hubungan antar data, untuk menggambarannya digunakan beberapa notasi dan simbol.

Entity Relationship diagram (ERD) merupakan teknik yang digunakan untuk memodelkan kebutuhan data dari suatu organisasi, biasanya oleh sistem analisis dalam tahap analisis persyaratan proyek pengembangan sistem. Sementara seolah-olah teknik diagram atau alat peraga memberikan dasar untuk desain *database* relasional yang mendasari sistem informasi yang dikembangkan. ERD bersama-sama dengan detail pendukung merupakan model data yang pada gilirannya digunakan sebagai spesifikasi untuk *database*.



**Gambar 4. Contoh Entity Relationship Diagram**

Sumber : <https://fairuzelsaid.wordpress.com/sistem-basis-data-erd/>

Pada dasarnya ada tiga komponen yang digunakan, yaitu :

a. Entitas

Entitas merupakan objek yang mewakili sesuatu yang nyata dan dapat dibedakan dari sesuatu yang lain. Simbol dari entiti ini biasanya digambarkan dengan persegi panjang.

b. Atribut

Setiap entitas pasti mempunyai elemen yang disebut atribut yang berfungsi untuk mendeskripsikan karakteristik dari entitas tersebut. Isi dari atribut mempunyai sesuatu yang dapat mengidentifikasi isi elemen satu dengan yang lain. Gambar atribut diwakili oleh simbol *elips*. Beberapa jenis Atribut antara lain:

a) *Atribut key*

*Atribut Key* adalah satu atau gabungan dari beberapa atribut yang dapat membedakan semua baris data (*Row/Record*) dalam tabel secara unik. Dikatakan unik jika pada atribut yang dijadikan *key* tidak boleh ada baris data dengan nilai yang sama

b) *Atribut Composite*

*Atribut composite* adalah suatu atribut yang terdiri dari beberapa atribut yang lebih kecil yang mempunyai arti tertentu yang masih bisah dipecah lagi atau mempunyai *sub attribute*.

c) Hubungan / Relasi

Hubungan antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda.

Derajat relasi atau kardinalitas rasio menjelaskan jumlah maksimum hubungan antara satu entitas dengan entitas lainnya.

*a. One to One (1:1)*

Setiap anggota entitas A hanya boleh berhubungan dengan satu anggota entitas B, begitu pula sebaliknya.

*b. One to many (1:M / Many)*

Setiap anggota entitas A dapat berhubungan dengan lebih dari satu anggota entitas B tetapi tidak sebaliknya.

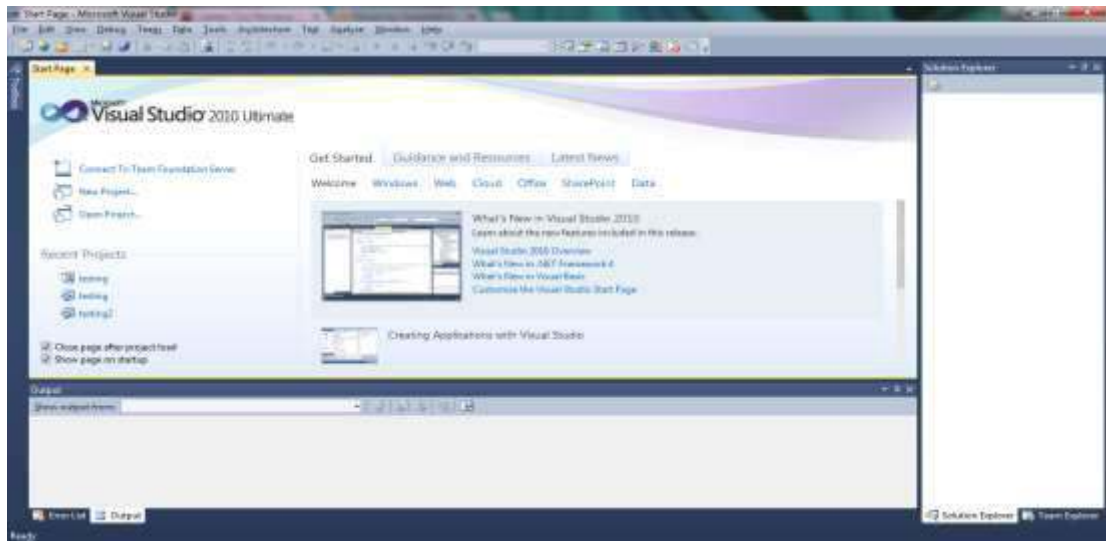
*c. Many to Many (M:M)*

Setiap entitas A dapat berhubungan dengan banyak entitas himpunan entitas B dan demikian pula sebaliknya.

## **2.11 Microsoft Visual Studio 2010**

Microsoft Visual Studio 2010 merupakan sebuah perangkat lunak lengkap (*suite*) yang dapat digunakan untuk melakukan pengembangan aplikasi, baik itu aplikasi bisnis, aplikasi personal, ataupun komponen aplikasinya, dalam bentuk *console* aplikasi windows, ataupun aplikasi web. Visual studio mencakup kompiler, SDK, *Integrated Development Environment* (IDE), dan dokumentasi (umumnya berupa MSDN library). Kompiler yang dimasukkan kedalam paket Visual Studio antara lain Visual C++, Visual C#, Visual Basic, Visual Basic.NET, Visual Interdev, Visual J++, Visual J#, Visual FoxPro, dan Visual sourcesafe.





**Gambar 5. Halaman Awal Microsoft Visual Studio 2010**

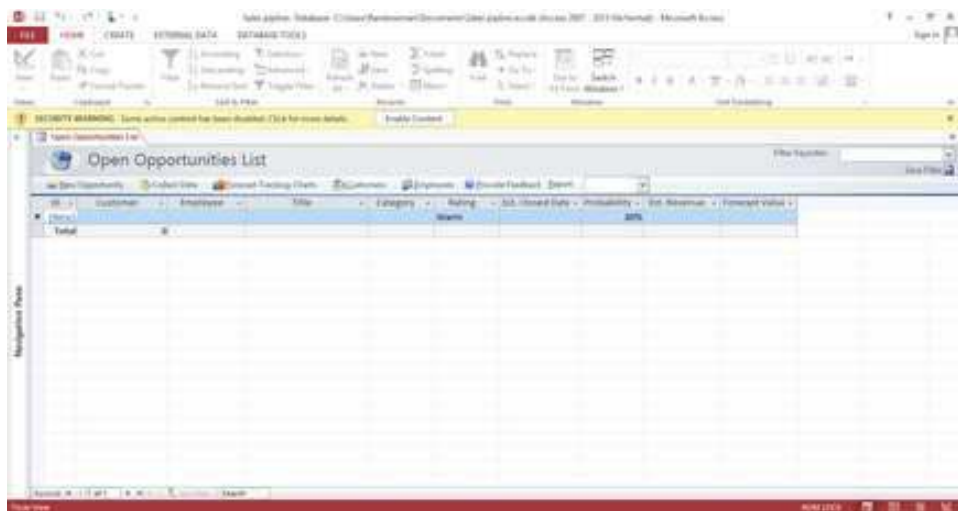
Sumber : Rolly Yesputa (2017 : 11)

## 2.12 Microsoft Access

Microsoft Access adalah sebuah program aplikasi basis data komputer relasional yang ditunjukkan untuk kalangan rumahan dan perusahaan kecil hingga menengah. Aplikasi ini merupakan anggota dari beberapa aplikasi Microsoft Office, selain tentunya Microsoft Word, Microsoft Excel, dan Microsoft Power Point. Aplikasi ini menggunakan mesin basis data Microsoft Jet Database Engine, dan juga menggunakan tampilan grafis yang intuitif sehingga memudahkan pengguna.

Microsoft Access dapat menggunakan data yang disimpan di dalam format Microsoft Access, Microsoft Jet Database Engine, Microsoft SQL Server, Oracle Database, atau semua kontainer basis data yang mendukung standar ODBC. Para pengguna/programmer yang mahir dapat menggunakannya untuk

mengembangkan perangkat lunak aplikasi yang kompleks, sementara para programmer yang kurang mahir dapat menggunakannya untuk mengembangkan perangkat lunak aplikasi yang sederhana. Access juga mendukung teknik-teknik pemrograman berorientasi objek, tetapi tidak dapat digolongkan ke dalam perangkat bantu pemrograman berorientasi objek.



**Gambar 6. Microsoft Access 2013**

Sumber : [https://id.wikipedia.org/wiki/Microsoft\\_Access](https://id.wikipedia.org/wiki/Microsoft_Access)

### BAB III

#### METODE PENELITIAN

##### 3.1 Analisa Kebutuhan Sistem

Pencarian kata adalah masalah yang akan diselesaikan dengan menggunakan sistem. Masalah utama yang menjadi dasar perancangan sistem adalah masalah efisiensi waktu untuk mencari data gandengan dan *container* setiap harinya. Data gandengan dicari berdasarkan kode nomor yang terdapat dicontainer, miripnya kode yang terdapat disetiap container juga menjadi alasan utama perancangan sistem. Berikut adalah tabel data gandengan beserta nomor containernya :

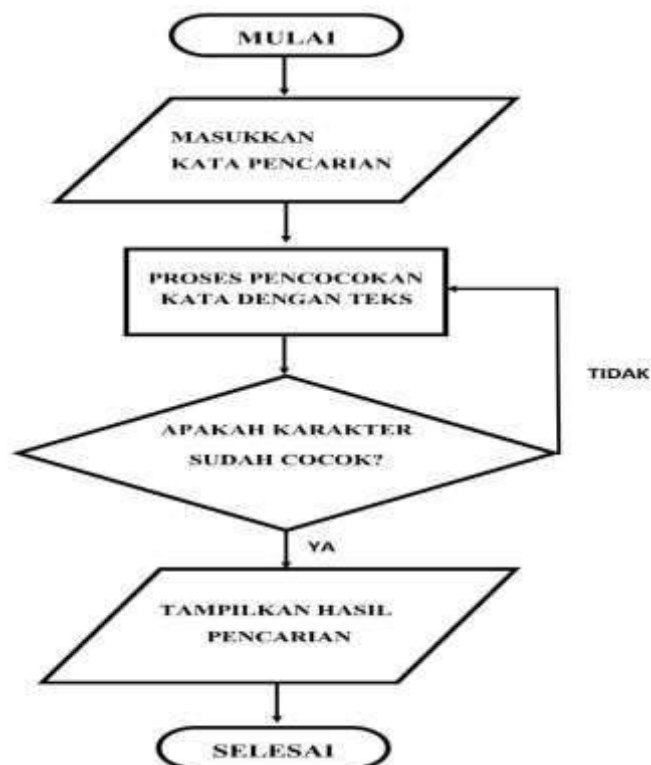
TANGGAL	KODE GANDENGAN	STATUS	PLAT	NOMOR CONTAINER	DEMMURAGE/ CLOSING	TUJUAN
19-Jul-18	SKT14	FULL	BH 8509 MU	BEAU 4602852	25-Jul-18	MUSIM MAS
9-Jul-18	SKT36	EMPTY	BK 8738 BG	BEAU 4822862	12-Jul-18	MNA K.TANJUNG
30-Jul-18	SKT33	EMPTY	BH 8572 MU	CLHU 3404899	31-Jul-18	SIANTAR TOP
4-Jul-18	SKT23	FULL	B 9553 UIY	CLHU 3404279	8-Jul-18	WILMAR PELINTUNG
29-Jul-18	SKT37	EMPTY	B 9552 UIY	FCIU 3502635	1-Aug-18	JAMPALAN TANJUNG BALAI
28-Jul-18	SKT12	EMPTY	B 9438 UZ	FCIU 4015375	31-Jul-18	INDOFOOD
25-Jul-18	SKT22	EMPTY	BK 8738 BG	FCIU 4950685	2-Aug-19	PT. SUKABUMI
31-Jul-18	SKT29	FULL	L 9598 UU	OOCU 6984077	2-Aug-18	MEGA JAYA K.TANJUNG
30-Jul-18	SKT02	EMPTY	N 9321 UR	OOCU 7363077	3-Aug-18	MEGA JAYA K.TANJUNG
17-Jul-18	SKT12	FULL	B 9438 UZ	SPNU 4610681	30-Jul-18	SEIA TEBING
13-Jul-18	SKT26	EMPTY	BK 8738 BG	SPNU 4624181	17-Jul-18	SEIA TEBING
12-Jul-18	SKT12	FULL	B 9553 UIY	SPNU 4635798	18-Jul-18	JL. PULAU SOLOR
14-Jul-18	SKT01	FULL	N 9569 US	SPNU 4637928	24-Jul-18	SEIA TEBING
28-Jul-18	SKT32	FULL	B 9213 UOK	TAKU 2457624	17-Jul-18	MDR P.SIANTAR
18-Jul-18	SKT01	EMPTY	N 9980 UR	TAKU 2475274	23-Jul-18	MDR P.SIANTAR
4-Jul-18	SKT29	EMPTY	L 9598 UU	APZU 2665274	12-Jul-18	AROMA/DALU 10

**Tabel 15. Tabel Data Gandengan Truk**  
Sumber : PT. Setia Kawan Transporindo

Berdasarkan tabel diatas, dapat kita lihat bahwa kode nomor *container* memiliki karakter yang cukup sulit untuk diingat, juga tanggal demurrage dan *closing* kapal yang berbeda-beda sehingga memungkinkan kekeliruan yang tinggi dalam pencarian datanya.

### 3.2 Analisa Sistem Terdahulu

Proses pencarian data pada sistem terdahulu memiliki estimasi waktu yang kurang efisien ditambah lagi banyaknya data yang tersimpan menyebabkan waktu tunggu untuk melakukan pencarian data lebih lama. Berikut flowchart sistem terdahulu:



Gambar 7. Flowchart Sistem Sebelumnya



**Tabel 17. Menghitung Nilai OH (a)**

Index	0	1	2	3
Pattern	7	9	2	8
Occurrence Heuristic				

Langkah-langkah pemberian nilainya adalah sebagai berikut :

- a. Mencari nilai OH dengan menggunakan rumus :

$$\text{OH} = (\text{Length} - 1 - \text{Index})$$

Length = panjang karakter = 4

- a. Karakter pertama adalah

- b. “7” dengan index = 0

$$\text{OH} = 4 - 1 - 0 = 3$$

- c. Karakter kedua adalah “9” dengan index = 1

$$\text{OH} = 4 - 1 - 1 = 2$$

- d. Karakter ketiga adalah “2” dengan index = 2

$$\text{OH} = 4 - 1 - 2 = 1$$

- e. Karakter keempat adalah “8” dengan index = 3

$$\text{OH} = 4 - 1 - 3 = 0$$

**Tabel 18. Menghitung Nilai OH (b)**

Index	0	1	2	3
Pattern	7	9	2	8
Occurrence Heuristic	3	2	1	0

- b. Jika terdapat karakter yang berulang pada pattern, gunakan nilai OH terkecil pada karakter yang berulang. Pada contoh ini tidak terdapat karakter berulang sehingga nilai OH tidak perlu diganti.

## B. Menghitung Nilai *Match Heuristic*

Cara menghitung tabel *Match Heuristic* (MH)

Data pencarian : Data 1 : SPNU  
4635768 pada Gandengan SKT09

Data 2 : SPNU 4637928 pada Gandengan SKT01

Pattern Yang Dicari : 7928

Panjang Pattern (L) : 4

**Tabel 19. Menghitung Nilai MH**

Teks	S	P	N	U
Pattern	7	9	2	8
Match Heuristic				

Langkah-langkah pemberian nilainya adalah sebagai berikut :

- Berikan nilai 1 ( nilai default MH ) pada index terbesar, dalam kasus ini index terbesar adalah “8”.

**Tabel 20. Menghitung Nilai MH (a)**

Teks	S	P	N	U
Pattern	7	9	2	8
Match Heuristic				1

- Bandingkan karakter terakhir pada pattern dan teks, jika karakter terakhir berbeda maka cek kembali apakah ada karakter berikutnya pada teks yang sama dengan karakter terakhir pada pattern. Jika tidak maka beri nilai karakter kedua dari akhir sampai karakter awal pattern sebesar nilai panjang pattern yaitu “4”.





Tabel 23. Pencarian Menggunakan Boyer Moore(a)

Teks	S	P	N	U	4	6	3	5	7	9	8	-	S	P	N	U	4	6	3	7	9	2	8
Pattern	7	9	2	8																			
Index	0	1	2	3																			

- a. Bandingkan karakter terakhir pada pattern dengan karakter sejajar di atasnya, apabila karakter terakhir belum cocok periksa nilai OH dan MH terbesar dari karakter terakhir pada pattern dan teks di atasnya. Nilai terbesar untuk U = 4 dan nilai terbesar untuk 8 = 1, maka pattern bergeser 4 langkah.

Tabel 24. Pencarian Menggunakan Boyer Moore(b)

Teks	S	P	N	U	4	6	3	5	7	9	8	-	S	P	N	U	4	6	3	7	9	2	8
Pattern					7	9	2	8															
Index					0	1	2	3															

- b. Setelah melakukan pergeseran, cek kembali karakter terakhir pada pattern dengan karakter di atasnya, jika tidak cocok geser pattern kembali sejauh nilai OH dan MH terbesar. Karena nilai perbandingan terbesar adalah 4 maka bergeser sejauh 4 langkah.

Tabel 25. Pencarian Menggunakan Boyer Moore(c)

Teks	S	P	N	U	4	6	3	5	7	9	8	-	S	P	N	U	4	6	3	7	9	2	8
Pattern									7	9	2	8											
Index									0	1	2	3											

- c. Setelah melakukan pergeseran, cek kembali karakter terakhir pada pattern dengan karakter di atasnya, jika tidak cocok geser pattern kembali sejauh nilai

OH dan MH terbesar. Karena nilai perbandingan terbesar adalah 4 maka begeser sejauh 4 langkah.

**Tabel 26. Pencarian Menggunakan Boyer Moore (d)**

<b>Teks</b>	S	P	N	U	4	6	3	5	7	9	8	-	S	P	N	U	4	6	3	7	9	2	8
<b>Pattern</b>												7	9	2	8								
<b>Index</b>												0	1	2	3								

- d. Setelah melakukan pergeseran, cek kembali karakter terakhir pada pattern dengan karakter di atasnya, jika tidak cocok geser pattern kembali sejauh nilai OH dan MH terbesar. Karena nilai perbandingan terbesar adalah 4 maka begeser sejauh 4 langkah.

**Tabel 27. Pencarian Menggunakan Boyer Moore (e)**

<b>Teks</b>	S	P	N	U	4	6	3	5	7	9	8	-	S	P	N	U	4	6	3	7	9	2	8
<b>Pattern</b>																	7	9	2	8			
<b>Index</b>																	0	1	2	3			

- e. Setelah melakukan pergeseran, cek kembali karakter terakhir pada pattern dengan karakter di atasnya, jika tidak cocok geser pattern kembali sejauh nilai OH dan MH terbesar. Karena nilai perbandingan terbesar adalah 3 maka begeser sejauh 3 langkah.

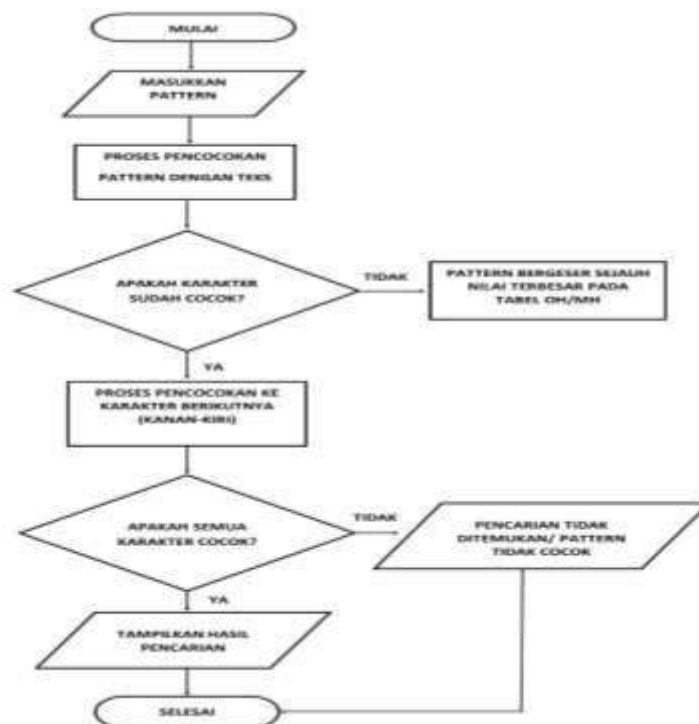
**Tabel 28. Pencarian Menggunakan Boyer Moore (f)**

<b>Teks</b>	S	P	N	U	4	6	3	5	7	9	8	-	S	P	N	U	4	6	3	7	9	2	8		
<b>Pattern</b>																					7	9	2	8	
<b>Index</b>																						0	1	2	3

- f. Bandingkan karakter dari kanan pattern pada teks yang sejajar, apabila sudah cocok artinya pattern telah ditemukan pada teks.
- g. Pattern ditemukan pada nomor *container* SPNU4637928, lihat kembali pada data pencarian : Data 1 : SPNU 4635768 pada Gandengan SKT09  
 Data 2 : SPNU 4637928 pada Gandengan SKT01
- h. Data dengan nomor *container* SPNU4637928 berada pada Gandengan SKT01.

### 3.4 Flowchart

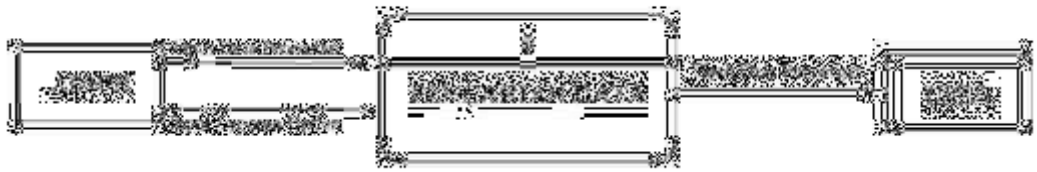
*Flowchart* Sistem merupakan bagian yang menunjukkan alur kerja atau apa yang sedang dikerjakan didalam sistem secara keseluruhan dan menjelaskan urutan dari prosedur – prosedur yang ada didalam sistem. Adapun *flowchat* sistem yang menggambarkan langkah kerja sistem yaitu :



**Gambar 8. Flowchart Algoritma Boyer Moore**

### 3.5 Diagram Context

Diagram Context menggambarkan satu lingkaran besar yang dapat mewakili seluruh proses yang terdapat di dalam suatu sistem. Merupakan tingkatan tertinggi dalam DFD dan biasanya diberi nomor 0 (nol). Adapun diagram *context* pada sistem adalah :

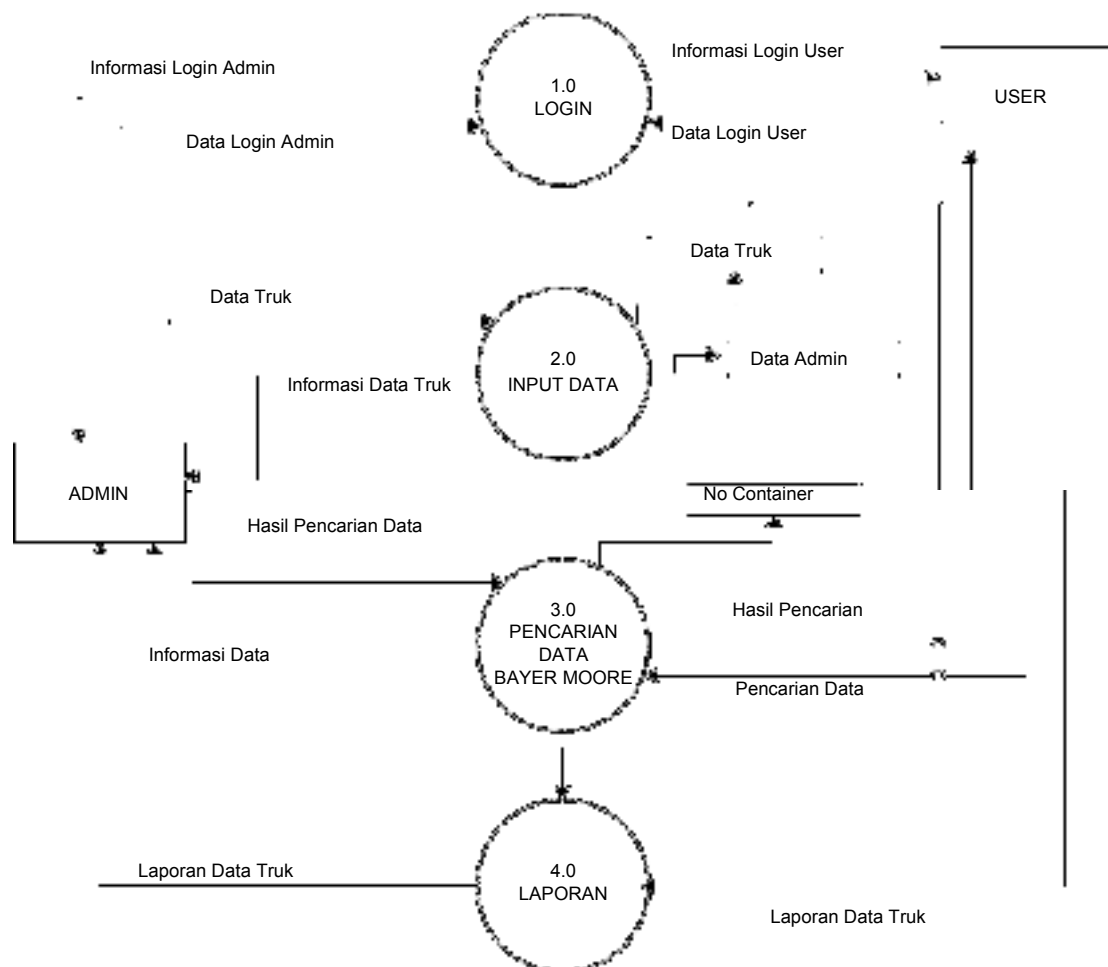


**Gambar 9. Diagram Context**

### 3.6 Diagram Alir Data

Diagram alir data merupakan alat perancangan sistem yang berorientasi pada alur data dengan konsep dekomposisi dapat digunakan untuk penggambaran analisa maupun rancangan sistem yang mudah dikomunikasikan oleh profesional sistem kepada pemakai maupun pembuat program.

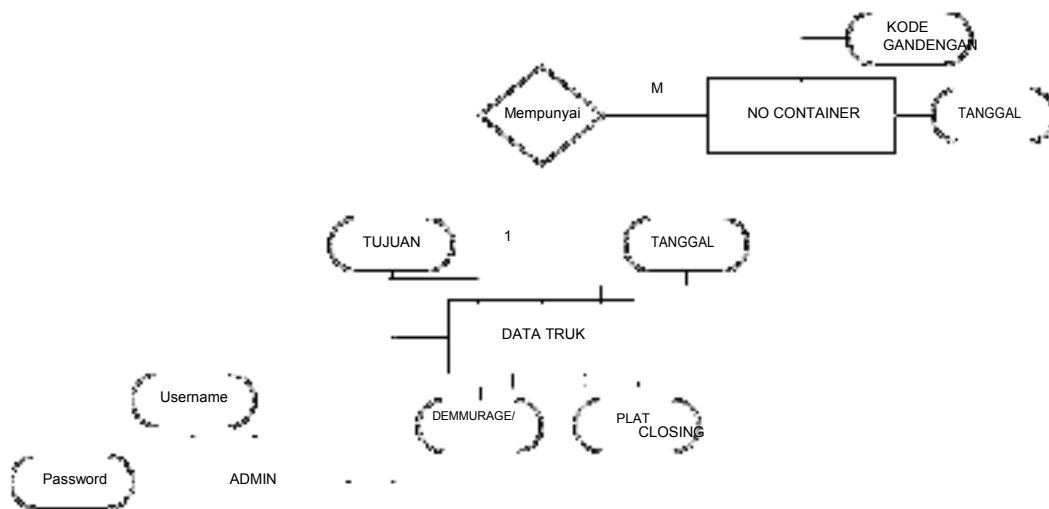
Diagram Alir data yang akan Dirancang pada sistem adalah :



**Gambar 10. Diagram Alir Data Pada Sistem**

### 3.7 Entity Relationship Diagram (ERD)

*Entity Relationship Diagram* (ERD) berfungsi untuk mempermudah menganalisis pada suatu basis data atau suatu sistem dengan cara yang cepat, *Entity Relationship Diagram* (ERD) juga berfungsi untuk menjelaskan hubungan-hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang dihubungkan oleh suatu relasi. Berikut *Relationship Diagram* (ERD) yang dirancang pada sistem :



**Gambar 11. Entity Relationship Diagram (ERD) Pada Sistem**

### 3.8 Perancangan Database

Struktur Tabel adalah penggambaran tentang *file-file* dalam tabel sehingga dapat dilihat bentuk-bentuk *file* tersebut baik *field-fieldnya*, tipe datanya serta ukuran dari data tersebut. Adapun struktur tabel yang ada pada *database* Ms. Access dari sistem yang akan dibuat dapat digambarkan sebagai berikut:

**Tabel 29. Tabel Login**

No	Field Name	Type	Length	Keterangan
1	username	Text	11	Primary Key
2	password	Text	225	

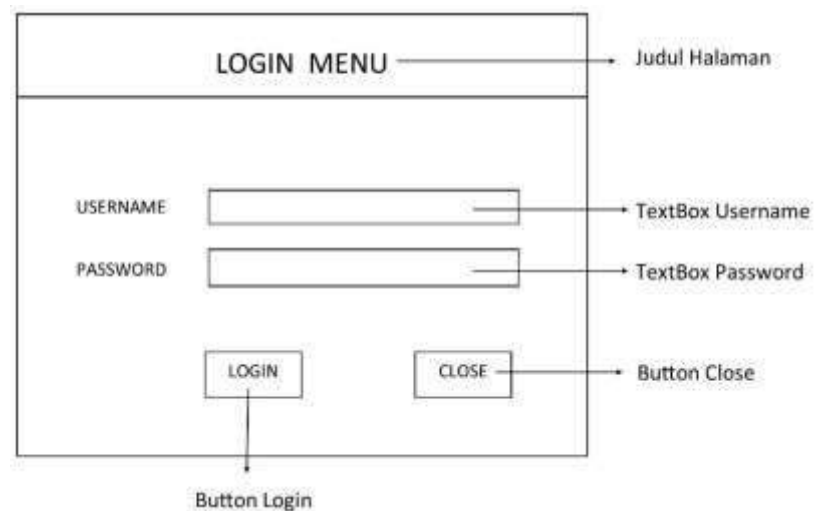
**Tabel 30. Tabel Data Truk**

No	Field Name	Type	Length	Keterangan
1	Tanggal	Text	11	
2	Kodegandengan	Text	225	
3	Full	Text	255	
4	Plat	Text	255	
5	NomorContainer	Text	255	Primary Key
6	Demurage/Closing	Text	255	
7	Tujuan	Text	255	

### 3.9 Perancangan Sistem

#### A. Perancangan Form Login

Form login berfungsi untuk mengamankan aplikasi dengan membatasi hak akses bagi user untuk menggunakan data yang tersedia di aplikasi, hanya user yang telah terdaftar yang dapat mengakses data didalam aplikasi. Berikut adalah rancangan gambar form login yang akan dirancang :

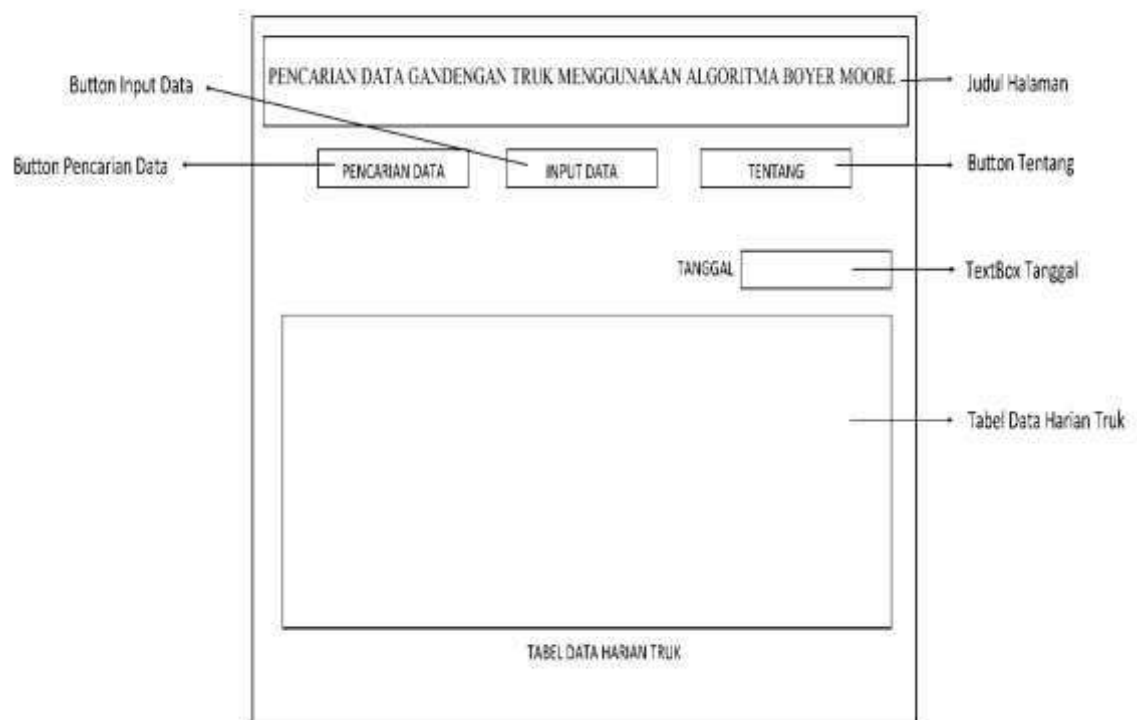
**Gambar 12. Tampilan Form Login**

**Keterangan :**

- a. Judul Halaman : Berisi keterangan nama halaman yang akan dirancang.
- b. *TextBox Username* : Berfungsi memasukkan data Username yang telah terdaftar.
- c. *TextBox Password* : Berfungsi untuk memasukkan Password.
- d. *Button Close* : Berfungsi untuk mengeluarkan user dari aplikasi.
- e. *Button Login* : Berfungsi untuk memproses data user untuk masuk ke aplikasi.

**B. Rancangan Halaman Awal**

Halaman awal berisi Menu-menu yang dapat dipilih sesuai kebutuhan pengguna, berikut rancangan halaman awal user yang akan dirancang :



**Gambar 13. Rancangan Tampilan Halaman Awal**

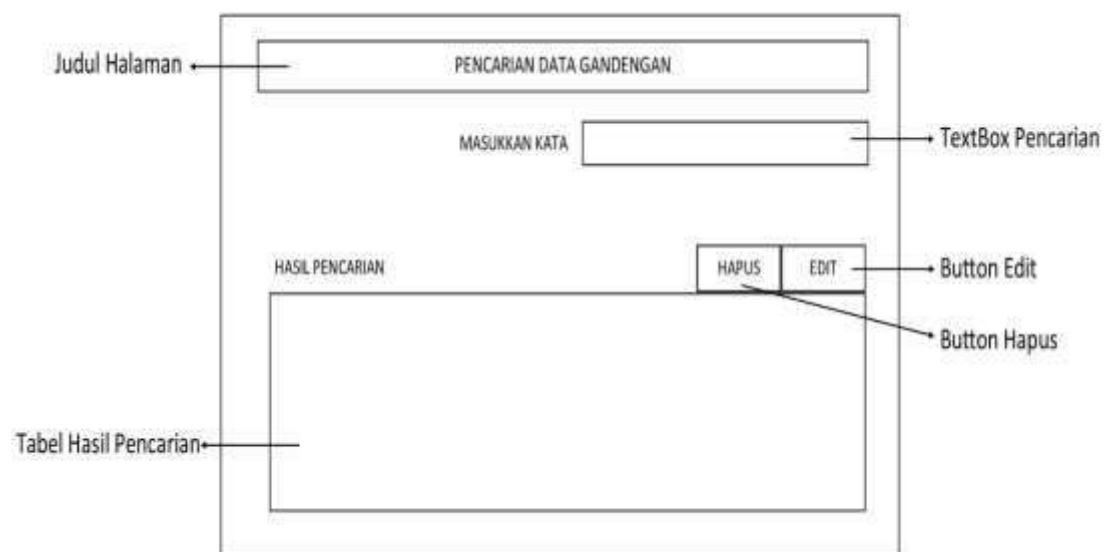


**Keterangan :**

- a. Judul Halaman : Berisi keterangan nama halaman yang akan dirancang.
- b. *Button* Tentang : Berfungsi menampilkan halaman tentang.
- c. *TextBox* Tanggal : Berisi pilihan tanggal yang akan diproses datanya.
- d. Tabel Data Harian Truk : Berfungsi menampilkan data harian truk berdasarkan tanggal yang telah ditentukan.
- e. *Button Input* Data : Berfungsi menampilkan halaman Input Data.
- f. *Button* Pencarian Data : Berfungsi menampilkan halaman Pencarian Data.

**C. Rancangan Tampilan Halaman Pencarian Data Gandengan**

Halaman pencarian data gandengan berisi informasi mengenai data yang akan dicari menggunakan Algoritma *Boyer Moore*. Berikut rancangan halaman pencarian data yang akan dirancang :



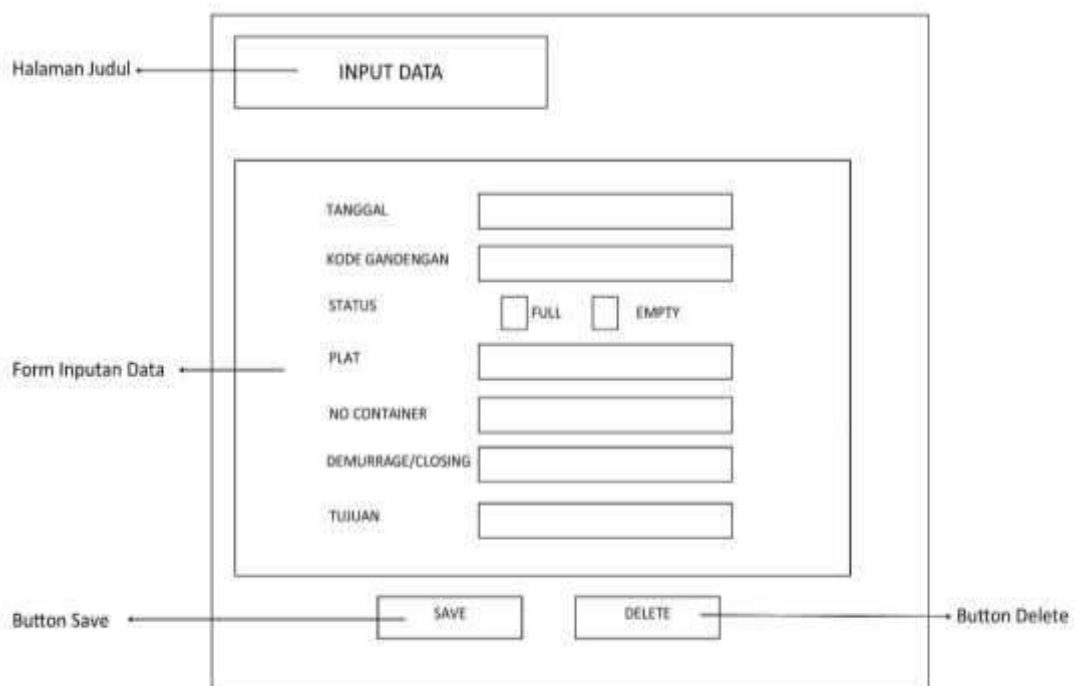
**Gambar 14. Rancangan Halaman Pencarian Data Gandengan**

**Keterangan :**

- Judul Halaman : Berisi keterangan nama halaman yang akan dirancang.
- TextBox* Pencarian : Berfungsi untuk memasukkan data yang akan dicari.
- Button Edit* : Berfungsi untuk mengedit data yang dipilih.
- Button Hapus* : Berfungsi untuk menghapus data yang dipilih.
- Tabel Hasil Pencarian : Menampilkan hasil pencarian data yang telah diproses dengan algoritma *Boyer Moore*.

**D. Rancangan Tampilan Halaman Input Data**

Tampilan input data berisi form inputan yang berfungsi untuk memasukkan informasi data baru kedalam database program. Berikut rancangan tampilan *input* data yang akan dirancang :



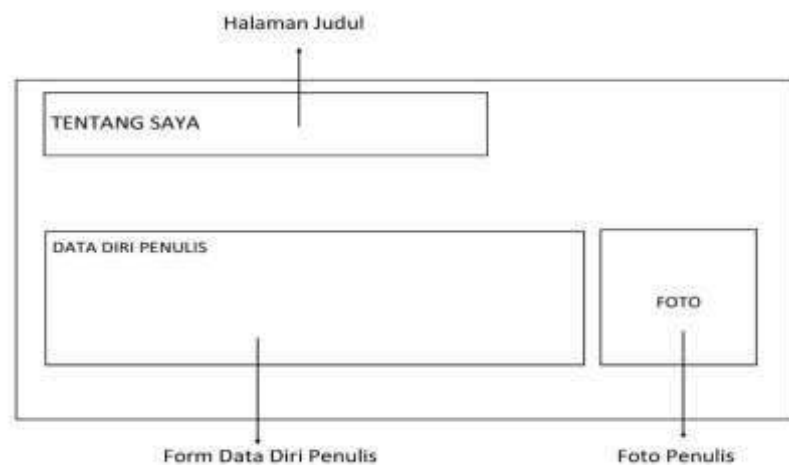
**Gambar 15. Rancangan Tampilan Menu Input Data**

**Keterangan :**

- a. Halaman Judul : Berisi keterangan nama halaman yang akan dirancang.
- b. *Form* Inputan Data : Berfungsi untuk menginput data.
- c. *Button Save* : Berfungsi untuk menyimpan data yang telah diinput.
- d. *Button Delete* : Berfungsi menghapus data inputan/membatalkan penginputan.

**E. Rancangan Tampilan Menu Tentang**

Menu tentang berisi biodata penulis, keterangan mengenai data penulis serta foto penulis.



**Gambar 16. Tampilan Menu Tentang**

**Keterangan :**

- a. Halaman Judul : Berisi keterangan nama halaman yang akan dirancang.
- b. *Form* Data Diri Penulis : Berfungsi menampilkan informasi tentang penulis.
- c. Foto Penulis : Berfungsi menampilkan foto penulis.

## **BAB IV**

### **HASIL PENELITIAN DAN PEMBAHASAN**

#### **4.1 Implementasi Sistem**

Implementasi sistem adalah langkah-langkah atau prosedur-prosedur yang dilakukan dalam menyelesaikan desain sistem yang telah disetujui, untuk menguji, menginstal dan memulai sistem baru atau yang diperbaiki untuk menggantikan sistem yang lama. Adapun langkah-langkah yang dibutuhkan dalam implementasi sistem adalah:

- a. Mendapatkan software dan hardware yang tepat/sesuai untuk merancang .
- b. Menyelesaikan rancangan sistem.
- c. Menulis, menguji, mengontrol dan mendokumentasikan .
- d. Mendapatkan persetujuan.

#### **4.2 Tujuan Implementasi Sistem**

Tujuan Implementasi Sistem adalah sebagai berikut:

- a. Menyelesaikan desain sistem yang telah disetujui sebelumnya.
- b. Memastikan bahwa pemakai (*user*) dapat mengoperasikan sistem baru.
- c. Menguji apakah sistem baru tersebut sesuai dengan pemakai.
- d. Memastikan bahwa konversi ke sistem baru berjalan yaitu dengan membuat rencana, mengontrol dan melakukan instalasi baru secara benar.

### **4.3 Komponen Utama Dalam Sistem**

Untuk menjalankan sistem yang telah dirancang, dibutuhkan beberapa komponen, antara lain:

#### **A. Hardware**

Merupakan suatu komponen yang sangat dibutuhkan dalam mewujudkan sistem yang diusulkan. Dalam hal ini, dapat dirincikan spesifikasi komponen hardware yaitu:

- a. PC dengan processor minimal dual core 1.7 GHz.
- b. Hard disk 20 GB.
- c. Monitor Super VGA.
- d. Memory minimal 1 GB.
- e. Keyboard.
- f. Mouse.

#### **B. Software**

Hardware tidak akan memecahkan suatu masalah tanpa adanya komponen software. Adapun software yang sering digunakan dalam pembuatan ini adalah: a.

Sistem operasi Ms. Windows 7.

b. Microsoft Visual Studio 2010 dan Microsoft Access sebagai database server.

#### **C. Brainware**

Brainware adalah semua pihak yang bertanggung jawab dalam pengembangan informasi, pemrosesan dan penggunaan keluaran informasi.

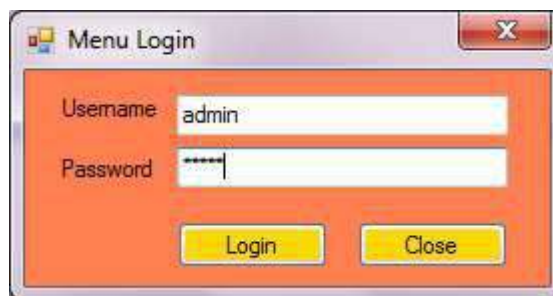
Brainware dalam sistem ini terbagi atas:

- a. Sistem analis: orang yang menganalisa sistem dengan mempelajari masalah-masalah yang timbul dan menentukan kebutuhan-kebutuhan pemakai dan mengidentifikasi pemecahan yang beralasan.
- b. Programmer: orang yang membuat sistem dengan menggunakan salah satu bahasa pemrograman yang dikuasainya.
- c. Operator: orang yang menggunakan dan memanfaatkan sistem.

#### 4.4 Tampilan Aplikasi

##### A. Form Menu Log In

Merupakan tampilan awal untuk mulai masuk kedalam program, hanya *user* yang terdaftar yang bisa masuk kedalam program.



**Gambar 17. Form Menu Log In**

##### B. Tampilan Halaman Awal

Merupakan Form Tampilan utama yang dapat diakses oleh user, untuk memulai pencarian data atau penginputan data.

Tampilan Halaman Awal dapat ditampilkan dengan listing program sebagai berikut:

tanggal	kodegandengan	full	plat	nomor	des	tujuan
07/07/2018	SKT04	FULL	B 9438 UZ	TAKU 2283590	10/07/2018	RIMBUN SE
12/07/2018	SKT22	EMPTY	BK 8152 BT	WHSU 2254011	14/07/2018	WHSU 2254
07/07/2018	SKT02	FULL	N 9321 UR	TAKU 2308873	11/07/2018	SAUDARA J
07/07/2018	SKT02	EMPTY	N 9321 UR	TAKU 2308873	11/07/2018	SAUDARA J
07/07/2018	SKT04	EMPTY	B 9438 UZ	TAKU 2283590	10/07/2018	RIMBUN SE
09/07/2018	SKT08	FULL	B 9552 UY	TAKU 2290265	11/07/2018	BINTANG M
09/07/2018	SKT08	EMPTY	B 9552 UY	TAKU 2290265	11/07/2018	BINTANG M

Gambar 18. Tampilan Halaman Awal

### C. Tampilan Menu Pencarian Data

Halaman pencarian data gandengan berisi informasi mengenai data yang akan dicari menggunakan Algoritma *Boyer Moore*.

tanggal	kodegandengan	full	plat	nomor	des	tujuan
11/07/2018	SKT12	EMPTY	B 9438 UZ	SPNU 4635798	18/07/2018	JL. PULAU S
13/07/2018	SKT26	FULL	BK 8738 BG	SPNU 4624181	17/07/2018	SEIA TEBIM
13/07/2018	SKT26	EMPTY	BK 8738 BG	SPNU 4624181	17/07/2018	SEIA TEBIM
14/07/2018	SKT01	FULL	N 9565 US	SPNU 4637928	24/07/2018	SEIA TEBIM
14/07/2018	SKT12	FULL	B 9438 UZ	SPNU 4633373	26/07/2018	SEIA TEBIM
14/07/2018	SKT12	EMPTY	B 9438 UZ	SPNU 4633373	26/07/2018	SEIA TEBIM
14/07/2018	SKT01	FULL	N 9565 US	SPNU 4637928	24/07/2018	SEIA TEBIM

Gambar 19. Tampilan Pencarian Data

#### D. Tampilan Menu Input Data

Merupakan Form Menu Input Data yang dapat diakses oleh admin, untuk menginputkan data-data harian truk.



**Gambar 20. Tampilan Menu Input Data**

#### E. Form Menu Edit Data

Merupakan Form Menu Edit Data yang dapat diakses oleh admin, untuk mengubah data-data truk.

Tampilan Menu Edit Data dapat ditampilkan dengan listing program sebagai berikut:

```
FromEdit.Dhow()  
FormEdit.Panggildata()
```



tanggal	kodegandengan	isi	alat	nomor	dan	tahun
12/07/2018	SKT22	EMPTY	BK 5152 BT	WHSU 2254011	14/07/2018	WHSU 2254038
12/07/2018	SKT26	EMPTY	BK 6738 BC	TRHU 1413184	14/07/2018	MANDALA
12/07/2018	SKT22	EMPTY	BK 5152 BT	WHSU 2254011	14/07/2018	WHSU 2254038
26/07/2018	SKT24	EMPTY	R 9327 BK	DAU 3640540	31/07/2018	INTAN CI
12/07/2018	SKT22	EMPTY	BK 5152 BT	WHSU 2254011	14/07/2018	WHSU 2254038
13/07/2018	SKT12	EMPTY	B 5438 UZ	SPNU 4835798	18/07/2018	JL. PULAU SOL
13/07/2018	SKT22	EMPTY	B 5213 UOK	SPNU 2560758	22/07/2018	ATC AMPLAS
25/07/2018	SKT22	FULL	BK 6738 BC	SPNU 2722125	25/08/2018	MAJAH FEED
26/07/2018	SKT01	EMPTY	B 5213 UOK	SHU 3003175	31/07/2018	MOR TO BALAJ
26/07/2018	SKT01	FULL	B 5213 UOK	SHU 3003175	31/07/2018	MOR TO BALAJ

**Gambar 21. Tampilan Menu Edit Data**

#### F. Tampilan Menu Tentang

Menu tentang berisi biodata penulis, keterangan mengenai data penulis serta foto penulis.

**DATA DIRI PENULIS**

NAMA	: SITI SYAHRANI
NPM	: 1414370155
TEMPAT TANGGAL LAHIR	: MEDAN, 24 JANUARI 1997
PRODI	: SISTEM KOMPUTER
ALAMAT	: JL. ISLAMIYAH NO.289

Log Out

**Gambar 22. Tampilan Menu Tentang**

#### 4.5 Pengujian Sistem

Berikut tabel pengujian sistem untuk memastikan apakah sistem yang dirancang telah sesuai dengan rancangan yang diharapkan.

Tabel 31. Pengujian Sistem

No.	Butir Pengujian	Output Yang Diharapkan	Output Yang Keluar	Keterangan
1	<i>Login</i>	Keluar <i>Form Login</i>	Keluar <i>Form Login</i>	Sesuai
2	Tampilan Utama	Keluar Tampilan Utama	Keluar Tampilan Utama	Sesuai
3	<i>Icon</i> Menu Pencarian Data	Keluar Tampilan Menu Pencarian Data	Keluar Tampilan Menu Pencarian Data	Sesuai
4	<i>Icon</i> Menu Input Data	Keluar Tampilan Menu <i>Input</i> Data	Keluar Tampilan Menu <i>Input</i> Data	Sesuai
5	<i>Icon</i> Menu Edit/Hapus Data	Keluar Tampilan Menu Edit/Hapus Data	Keluar Tampilan Menu Edit/Hapus Data	Sesuai
6	<i>Icon</i> Menu Tentang	Keluar Tampilan Menu Tentang	Keluar Tampilan Menu Tentang	Sesuai
7	<i>Icon</i> Menu Cari Pada Tampilan Utama	Dapat Menampilkan Data Sesuai Hasil Pencarian	Dapat Menampilkan Data Sesuai Hasil Pencarian	Sesuai
No.	Butir Pengujian	Output Yang Diharapkan	Output Yang Keluar	Keterangan
8	<i>Icon</i> Menu <i>Log Out</i>	Dapat Menutup Aplikasi	Dapat Menutup Aplikasi	Sesuai
9	<i>Form</i> Inputan Data Pada Menu Input Data	Dapat Menambahkan Data Truk	Dapat Menambahkan Data Truk	Sesuai
10	<i>Icon</i> <i>Save</i> Pada Menu Input Data	Dapat Menyimpan Data Yang Telah Ditambahkan	Dapat Menyimpan Data Yang Telah Ditambahkan	Sesuai
11	<i>Icon</i> <i>Clear</i> Pada Menu Input Data	Dapat Menghapus Data Yang Telah Ditambahkan	Dapat Menghapus Data Yang Telah Ditambahkan	Sesuai
12	<i>Icon</i> Cari Pada Menu Pencarian Data	Dapat Menampilkan Data Hasil Pencarian Menggunakan Algoritma <i>Boyer Moore</i>	Dapat Menampilkan Data Hasil Pencarian Menggunakan Algoritma <i>Boyer Moore</i>	Sesuai
13	<i>Icon</i> Cari Pada	Dapat Menampilkan Hasil Pencarian data	Dapat Menampilkan Hasil Pencarian data	Sesuai

	Menu Edit	yang akan diedit	yang akan diedit	
--	-----------	------------------	------------------	--

<b>NO.</b>	<b>Butir Pengujian</b>	<b>Output Yang Diharapkan</b>	<b>Output Yang Keluar</b>	<b>Keterangan</b>
14	<i>Icon Edit &amp; Delete</i> Pada Menu Edit	Dapat Meng-edit dan Menghapus data yang dipilih	Dapat Meng-edit dan Menghapus data yang dipilih	Sesuai
15	<i>Icon Log Out</i> Pada Menu Tentang	Dapat Menutup Aplikasi	Dapat Menutup Aplikasi	Sesuai

## **BAB V**

### **PENUTUP**

#### **5.1 Kesimpulan**

Dari penjelasan bab-bab sebelumnya, penulis merangkum beberapa kesimpulan, yaitu:

- a. Dalam proses pencarian data gandengan truk berdasarkan nomor container menjadi lebih cepat, dan tepat dikarenakan proses sebelumnya yang manual masih sering terjadi kesalahan dalam pendataan data gandengan truk tersebut..
- b. Algoritma *Boyer Moore* menghitung kesamaan dalam proses pencarian data dengan perbandingan pattern mulai dari kanan ke kiri.
- c. Semakin banyak jumlah karakter pada kata kunci pencarian maka semakin akurat hasil pencarian yang dilakukan. Jumlah karakter pada kata kunci pencarian dapat mempengaruhi keakuratan hasil pencarian.

#### **5.2 Saran**

Adapun saran dapat diberikan pada Laporan Skripsi ini adalah :

- a. Untuk pengembangan selanjutnya diharapkan pencarian data gandengan truk tidak hanya berdasarkan nomor *container* saja, tetapi dapat menggunakan data lainnya.
- b. Aplikasi ini diharapkan dapat menggunakan algoritma pencarian lainnya untuk lebih memudahkan pencarian data.

- c. Proses pengolahan data masih menggunakan *Microsoft Access*, diharapkan untuk pengembangan selanjutnya menggunakan *software database* yang lebih baik sehingga dapat menyimpan data yang lebih banyak.

## DAFTAR PUSTAKA

- Adisantoso, Julio., Rambe, A., dan Kaliana, Indra. Juli 2013. Pencarian Nama Yang Memiliki Kesamaan Fonetik Menggunakan Algoritma Kesamaan *String Boyer Moore*.
- Akbar, A. (2018). Pembangunan Model Electronic Government Pemerintahan Desa Menuju Smart Desa. *Jurnal Teknik dan Informatika*, 5(1), 1-5.
- Argakusumah, Kencana Wulan. 2014. Implementasi Algoritma *Boyer-Moore* pada Aplikasi Kamus Kedokteran Berbasis Android.
- Badawi, A. (2018). Evaluasi Pengaruh Modifikasi Three Pass Protocol Terhadap Transmisi Kunci Enkripsi.
- Batubara, S., Wahyuni, S., & Hariyanto, E. (2018, September). Penerapan Metode Certainty Factor Pada Sistem Pakar Diagnosa Penyakit Dalam. In Seminar Nasional Royal (SENAR) (Vol. 1, No. 1, pp. 81-86).
- Boyer, RS and Moore, JS. 1977. A fast string searching algorithm. *Communications of the ACM* .
- Chiquita, Christabella. 2012. Penerapan Algoritma Boyer-Moore Dynamic Programming untuk Layanan Auto-Complete dan Auto Correct.
- Hariyanto, E., & Rahim, R. (2016). Arnold's cat map algorithm in digital image encryption. *International Journal of Science and Research (IJSR)*, 5(10), 1363-1365.
- Havena, M., & Marlina, L. (2018). The Technology of Corn Processing as an Effort to Increase The Income of Kelambir V Village. *Journal of Saintech Transfer*, 1(1), 27-32.
- Hendrawan, J. (2018). Rancang Bangun Aplikasi Mobile Learning Tuntunan Shalat. *INTECOMS: Journal of Information Technology and Computer Science*, 1(1), 44-59.
- Khairul, K., Haryati, S., & Yusman, Y. (2018). Aplikasi Kamus Bahasa Jawa Indonesia dengan Algoritma Raita Berbasis Android. *Jurnal Teknologi Informasi dan Pendidikan*, 11(1), 1-6.
- Kumara, Gozali Harda. 2009. Visualisasi Beberapa Algoritma Pencocokan String dengan Java. *Jurnal: Jurusan Teknik Informatika, Fakultas Elektro Informatika, Institut Teknologi Bandung*.
- Kurnia, D. (2017). Analisis QoS Pada Pembagian Bandwidth Dengan Metode Layer 7 Protocol, PCQ, HTB Dan Hotspot Di SMK Swasta Al-Washliyah Pasar Senen. *CESS (Journal of Computer Engineering, System and Science)*, 2(2), 102-111.

- Kurnia, D., Dafitri, H., & Siahaan, A. P. U. (2017). RSA 32-bit Implementation Technique. *Int. J. Recent Trends Eng. Res*, 3(7), 279-284.
- Mariance, U. C. (2018). Analisa dan Perancangan Media Promosi dan Pemasaran Berbasis Web Menggunakan Work System Framework (Studi Kasus di Toko Mandiri Prabot Kota Medan). *Jurnal Ilmiah Core IT: Community Research Information Technology*, 6(1).
- Munir, Rinaldi. *Algoritma dan pemrograman : Dalam bahasa pascal dan C*, Informatika, Bandung, 2001
- Muntaha, Amir. Juli 2013. Pengenalan Sidik Jari Dengan Menggunakan Algoritma Pencocokan *String Boyer Moore*.
- Nievergelt, Jurg., Hinrichs, Klaus H. 2000 . *Algorithms & Data Structures with Application to Graphs and Geometry*, Prentice-Hall,.
- Putri, N. A. (2018). Sistem Pakar untuk Mengidentifikasi Kepribadian Siswa Menggunakan Metode Certainty Factor dalam Mendukung Pendekatan Guru. *INTECOMS: Journal of Information Technology and Computer Science*, 1(1), 78-90.
- Rahim, R., Aryza, S., Wibowo, P., Harahap, A. K. Z., Suleman, A. R., Sihombing, E. E., ... & Agustina, I. (2018). Prototype file transfer protocol application for LAN and Wi-Fi communication. *Int. J. Eng. Technol.*, 7(2.13), 345-347.
- Rompah, Edward. 2009. Contoh Kasus Sederhana Penerapan Algoritma *Boyer-Moore*.
- Ruwaida, D., & Kurnia, D. (2018). Rancang Bangun File Transfer Protocol (FTP) dengan Pengamanan Open SSL pada Jaringan VPN Mikrotik di SMK Dwiwarna. *CESS (Journal of Computer Engineering, System and Science)*, 3(1), 45-49.
- Sarif, M. I. (2017). Penemuan Aturan yang Berkaitan dengan Pola dalam Deret Berkala (Time Series).
- Suarga, 2004. *Algoritma Pemrograman*. Yogyakarta: Andi.
- Sulistyo, Imam, Adie Pratipto, dkk. 2006. *Algoritma Boyer- Moore dalam Pencarian String*.
- Suranto, SE, 2011, *Manajemen Operasional Angkutan Laut dan Terminal Peti Kemas*, Gema Indonesia, Medan.
- Wahana, K., 2013, *Visual Basic 2010 Source Code*, Andi Offset, Yogyakarta.



Wijaya, R. F., Utomo, R. B., Niska, D. Y., & Khairul, K. (2019). Aplikasi Petani Pintar Dalam Monitoring Dan Pembelajaran Budidaya Padi Berbasis Android. Rang Teknik Journal, 2(1).

Yesputra, Rolly. 2017. Belajar Visual Basic.NET Dengan Visual Studio 2010, Royal Asahan Press, Kisaran.

<http://www.beacukai.go.id/berita/mengenal-biaya-demurrage-saat-impor-dan-ekspor-.html> diakses pada 20 Februari 2018.

<https://bayuputraperkasa.com/jenis-ukuran-kontainer> diakses pada 9 Maret 2018.

<https://darmansyah.weblog.esaunggul.ac.id/2012/11/10/data-flow-diagram> diakses pada 19 Maret 2018.

<https://fairuzelsaid.wordpress.com/sistem-basis-data-erd/> diakses pada 5 April 2018.

<https://www.kompasiana.com/jiminandri//hebatnya-sopir-truk-kontainer> diakses pada 20 Februari 2018.

[https://id.wikipedia.org/wiki/Microsoft\\_Access](https://id.wikipedia.org/wiki/Microsoft_Access) diakses pada 13 Maret 2018.

[https://id.wikipedia.org/wiki/Peti\\_kemas](https://id.wikipedia.org/wiki/Peti_kemas) diakses pada 22 Februari 2018.

