

**IMPLEMENTASI ALGORITMA BOYER MOORE PADA  
PENCARIAN DATA DI SISTEM INFORMASI  
MANAJEMEN LABORATORIUM  
FAKULTAS SAINS DAN  
TEKNOLOGI**

**SKRIPSI**

Oleh :  
**ASIFATUL MU'AWANAH**  
**NIM. 14650001**



**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2018**

**IMPLEMENTASI ALGORITMA BOYER MOORE PADA PENCARIAN  
DATA DI SISTEM INFORMASI MANAJEMEN LABORATORIUM  
FAKULTAS SAINS DAN TEKNOLOGI**

**SKRIPSI**

**Diajukan Kepada :  
Universitas Islam Negeri Maulana Malik Ibrahim Malang  
Untuk Memenuhi Salah Satu Persyaratan Dalam  
Memperoleh Gelar Sarjana Komputer (S.Kom)**

**Oleh:  
ASIFATUL MU'AWANAH  
NIM. 14650001**

**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2018**

**LEMBAR PERSETUJUAN**

**IMPLEMENTASI ALGORITMA BOYER MOORE PADA PENCARIAN  
DATA DI SISTEM INFORMASI MANAJEMEN LABORATORIUM  
FAKULTAS SAINS DAN TEKNOLOGI**

**SKRIPSI**

Oleh:  
**ASIFATUL MU'AWANAH**  
**NIM. 14650001**

Telah Diperiksa dan Disetujui untuk Diuji

Tanggal: Juni 2018

Pembimbing I

Pembimbing II

Linda Salma Angreani, M.T  
NIP. 19770803 200912 2 005

Supriyono, M.Kom  
NIDT. 19841010 20160801 1 078

Mengetahui,  
Ketua Jurusan Teknik Informatika  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Dr. Cahyo Crysdiyan  
NIP. 19740424 200901 1 008

## HALAMAN PENGESAHAN

### IMPLEMENTASI ALGORITMA BOYER MOORE PADA Pencarian DATA DI SISTEM INFORMASI MANAJEMEN LABORATORIUM FAKULTAS SAINS DAN TEKNOLOGI

#### SKRIPSI

Oleh:  
**ASIFATUL MU'AWANAH**  
NIM. 14650001

Telah Dipertahankan di Depan Dewan Penguji Skripsi  
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan  
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)  
Tanggal: 05 Juni 2018

Susunan Dewan Penguji		Tanda Tangan
Penguji Utama	: <u>M. Ainul Yaqin, M.Kom</u> NIP. 19761013 200604 1 004	
Ketua Penguji	: <u>Khadijah F.H. Holle, M.Kom</u> NIDT. 19900626 20160801 2 077	
Sekretaris Penguji	: <u>Linda Salma Angreani, M.T</u> NIP. 19770803 200912 2 005	
Anggota Penguji	: <u>Supriyono, M.Kom</u> NIDT. 19841010 20160801 1 078	

Mengetahui,  
Ketua Jurusan Teknik Informatika  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Dr. Cahyo Crysdian

NIP. 19740424 200901 1 008

## PERNYATAAN

### ORISINALITAS PENELITIAN

Saya yang bertanda tangan dibawah ini:

Nama : Asifatul Mu'awanah  
NIM : 14650001  
Fakultas/Jurusan : Sains dan Teknologi/ Teknik Informatika  
Judul Penelitian : Implementasi Algoritma *Boyer Moore* Pada Pencarian Data Di Sistem Informasi Manajemen Laboratorium Fakultas Sains dan Teknologi.

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-banar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan Skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, Juli 2018  
Yang Membuat Pernyataan,



Asifatul Mu'awanah  
NIM. 14650001

## MOTTO

---

---

### **“First Time always Be The Hardest”**

Segala sesuatu yang pertama kali engkau lakukan jika terasa berat atau susah bersabarlah, selalu ingat bahwa apa yang kamu lakukan akan terasa sulit jika kamu belum terbiasa



## PERSEMBAHAN

---

---

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

*Alhamdulillahirobbil'alamin*

Sujud syukurku kusembahkan kepada Allah SWT Tuhan yang Maha Agung nan Maha Tinggi nan Maha Adil nan Maha Penyayang, atas takdirmu telah kau jadikan aku manusia yang senantiasa berpikir, berilmu, beriman dan bersabar dalam menjalani kehidupan ini. Semoga keberhasilan ini menjadi satu langkah awal bagiku untuk meraih cita-cita besarku.

Kupersembahkan karya sederhana ini kepada orang yang sangat kukasihi dan kusayangi ibunda, ayahanda, dan Keluargaku Tercinta Sebagai tanda bakti, hormat, dan rasa terima kasih yang tiada terhingga yang telah memberikan kasih sayang, segala dukungan, dan cinta kasih. Untuk Ibu, Ayah yang selalu membuatku termotivasi yang senantiasa memberikan doa dan restunya kepadaku dalam menuntut ilmu.

Tak lupa terima kasih kepada sahabat-sahabatku yang tiada hentinya selalu memberikan dorongan, motivasi dan telah menyempatkan memberikan waktu luangnya untuk membantu dalam proses pengerjaan penelitian ini.

**Asifatul Mu'awanah**

## KATA PENGANTAR

*Assalamu'alaikum Wr. Wb.*

Syukur alhamdulillah penulis haturkan kehadiran Allah SWT yang telah melimpahkan Rahmat dan Hidayah-Nya, sehingga penulis dapat menyelesaikan studi di Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang sekaligus menyelesaikan Skripsi ini dengan baik.

Selanjutnya penulis haturkan ucapan terima kasih seiring do'a dan harapan jazakumullah ahsanal jaza' kepada semua pihak yang telah membantu terselesaikannya Skripsi ini. Ucapan terima kasih ini penulis sampaikan kepada:

1. Prof. DR. Abdul Haris, M.Ag, selaku rektor UIN Maulana Malik Ibrahim Malang, yang telah banyak memberikan pengetahuan dan pengalaman yang berharga.
2. Dr. Sri Harini, M.Si selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Dr. Cahyo Crysdiyan, selaku Ketua Jurusan Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Linda Salma Angreani, M.T dan Supriyono, M.Kom selaku dosen pembimbing Skripsi, yang telah banyak memberikan pengarahan dan pengalaman yang berharga.
5. Suhartono, M.Kom selaku dosen wali selama penulis menuntut ilmu di Universitas Islam Negeri Maulana Malik Ibrahim Malang.
6. Segenap sivitas akademika Jurusan Teknik Informatika, terutama seluruh dosen, terima kasih atas segenap ilmu dan bimbingannya.

7. Ayahanda dan Ibunda tercinta yang senantiasa memberikan doa dan restunya kepada penulis dalam menuntut ilmu.
8. Teman-teman seperjuangan bimbingan skripsi yang senantiasa saling mendukung, menguatkan, dan memberi semangat satu sama lain.
9. Teman-teman Universitas Islam Negeri Maulana Malik Ibrahim Malang khususnya jurusan Teknik Informatika angkatan 2014 yang telah memberikan dukungan, motivasi, serta berbagai ilmu dan informasi selama menimba ilmu di Universitas Islam Negeri Maulana Malik Ibrahim Malang.
10. Semua pihak yang ikut membantu dalam menyelesaikan Skripsi ini baik berupa materiil maupun moril.

Penulis menyadari bahwa dalam penyusunan Skripsi ini masih terdapat kekurangan dan penulis berharap semoga Skripsi ini bias memberikan manfaat kepada para pembaca khususnya bagi penulis secara pribadi. Amin Ya Rabbal Alamin.

*Wassalamu'alaikum Wr. Wb.*

Malang, Juli 2018

Penulis

## DAFTAR ISI

HALAMAN JUDUL.....	i
LEMBAR PERSETUJUAN .....	iii
HALAMAN PENGESAHAN.....	iv
HALAMAN PERNYATAAN .....	v
MOTTO .....	vi
HALAMAN PERSEMBAHAN.....	vii
KATA PENGANTAR .....	viii
DAFTAR ISI.....	x
DAFTAR GAMBAR .....	xii
DAFTAR TABEL.....	xiv
ABSTRAK.....	xvi
ABSTRACT.....	xvii
ملخص.....	xviii
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar Belakang Masalah.....	1
1.2 Pernyataan Masalah.....	4
1.3 Tujuan Penelitian.....	4
1.4 Manfaat Penelitian.....	5
1.5 Batasan Masalah.....	5
1.6 Sistematika Penulisan.....	5
<b>BAB II STUDI PUSTAKA.....</b>	<b>7</b>
2.1 Pencarian .....	7
2.1.1 Algoritma Pencarian String.....	8
2.1.2 Algoritma <i>Boyer Moore</i> .....	10
2.2 Sistem Informasi Manajemen Laboratorium.....	18
2.3 <i>Receiver Operating Characteristic</i> .....	19
2.4 Penelitian Terkait .....	21
<b>BAB III METODOLOGI PENELITIAN .....</b>	<b>24</b>
3.1 Prosedur Penelitian.....	24
3.2 Perancangan Sistem.....	25
3.2.1 Pemodelan UML.....	25
3.2.2 Perancangan <i>Database</i> .....	35

3.2.3.	Perancangan Sistem Dengan Algoritma <i>Boyer Moore</i> .....	38
BAB IV UJI COBA DAN PEMBAHASAN .....		50
4.1	Implementasi Sistem .....	50
4.1.1.	Implementasi Perangkat Keras .....	50
4.1.2.	Implementasi Perangkat Lunak .....	51
4.1.3	Implementasi Algoritma <i>Boyer Moore</i> .....	51
4.2	Pengujian Sistem .....	55
4.2.1	Pengujian <i>Black Box</i> .....	55
4.2.2	Pengujian <i>User</i> .....	69
4.2.3	Pengujian Algoritma <i>Boyer Moore</i> .....	81
BAB V KESIMPULAN DAN SARAN .....		94
5.1	Kesimpulan .....	94
5.2	Saran .....	94
DAFTAR PUSTAKA .....		95



## DAFTAR GAMBAR

Gambar 2.1 <i>Good-suffix shift</i> , $u$ terjadi lagi didahului karakter $c$ berbeda dari $a$ .	13
Gambar 2.2 <i>Good-suffix shift</i> , hanya <i>suffix</i> dari $u$ yang terjadi lagi di <i>pattern</i> $x$ ..	14
Gambar 2.3 <i>Bad-character shift</i> , $b$ terdapat di <i>pattern</i> $x$ .....	14
Gambar 2.4 <i>Bad-character shift</i> , $b$ tidak ada di <i>pattern</i> $x$ .....	14
Gambar 3.1 Prosedur Penelitian.....	24
Gambar 3.2 <i>Use Case Diagram</i> .....	26
Gambar 3.3 <i>Activity Diagram</i> Kelola Data .....	27
Gambar 3.4 <i>Activity Diagram</i> Pencarian Data .....	28
Gambar 3.5 <i>Activity Diagram</i> Peminjaman Alat/Bahan .....	29
Gambar 3.6 <i>Activity Diagram</i> Pengembalian Alat.....	30
Gambar 3.7 <i>Activity Diagram</i> Detail Alat/Bahan Praktikum .....	30
Gambar 3.8 <i>Sequence Diagram</i> Kelola Data .....	31
Gambar 3.9 <i>Sequence Diagram</i> Peminjaman Alat/Bahan .....	32
Gambar 3.10 <i>Sequence Diagram</i> Pengembalian Alat.....	33
Gambar 3.11 <i>Sequence Diagram</i> Pencarian Data .....	34
Gambar 3.12 <i>Class Diagram</i> Sistem .....	35
Gambar 3.13 <i>Conceptual Data Model</i> .....	36
Gambar 3.14 <i>Physical Data Model</i> .....	37
Gambar 3.15 <i>Database</i> .....	37
Gambar 3.16 Desain Sistem .....	38
Gambar 3.17 Flowchart Pencarian dengan <i>Boyer Moore</i> .....	40
Gambar 3.18 Penyelesaian contoh kasus Prosedur $preBmBc$ .....	42
Gambar 3.19 Penyelesaian contoh kasus Prosedur $preBmGs$ .....	47
Gambar 3.20 Penyelesaian contoh kasus Prosedur $BM$ .....	48
Gambar 4.1 Form Pencarian .....	51
Gambar 4.2 Contoh Kalimat Pencarian .....	51
Gambar 4.3 Hasil Pencarian.....	52
Gambar 4.4 <i>Source Code Tokenizing</i> .....	52
Gambar 4.5 Potongan <i>SourceCode</i> Pemberian Nilai $OH$ .....	53
Gambar 4.6 Potongan <i>Source Code</i> Pembentukan <i>Suffix</i> dan Nilai Pergeseran. ....	54
Gambar 4.7 Potongan <i>SourceCode</i> Pemberian Nilai $MH$ pada Karakter .....	54
Gambar 4.8 Potongan <i>Source Code</i> untuk Pencocokan String .....	55
Gambar 4.9 Form Login.....	56
Gambar 4.10 Login Gagal.....	57
Gambar 4.11 <i>Form</i> Tambah Bahan.....	58
Gambar 4.12 Data Bahan Berhasil di Tambahkan .....	58
Gambar 4.13 <i>Alert</i> Data Tidak Boleh Kosong .....	59
Gambar 4.14 Tampilan Detail Bahan.....	60
Gambar 4.15 Peringatan Penghapusan Bahan .....	61
Gambar 4.16 <i>Form</i> Tambah Alat .....	62
Gambar 4.17 Data Alat Berhasil di Tambahkan .....	62
Gambar 4.18 <i>Form</i> Edit Alat .....	63
Gambar 4.19 Detail Alat Praktikum .....	64
Gambar 4.20 Form Peminjaman Bahan <i>Step1</i> .....	66
Gambar 4.21 Form Peminjaman Bahan <i>Step2</i> .....	66

Gambar 4.22 Proses Peminjaman Berhasil .....	66
Gambar 4.23 PeringatanInput Data Peminjaman Bahan Kosong .....	67
Gambar 4.24 Form Peminjaman Alat Step 1 .....	68
Gambar 4.25 Form Peminjaman Alat Step 2 .....	68
Gambar 4.26 Data Peminnjaman Alat Berhasil di Simpan.....	69
Gambar 4.27 Peringatan Input Data Peminjaman Alat Kosong .....	69



## DAFTAR TABEL

Tabel 2.1 Tabel Kontingensi ROC.....	20
Tabel 3.1 Contoh Data Alat .....	39
Tabel 3.2 Daftar pencacahan <i>prefix</i> dan <i>suffix</i> dari <i>pattern</i> .....	45
Tabel 3.3 Contoh Hasil Pencarian.....	49
Tabel 4.1 Perangkat Keras Yang Digunakan .....	50
Tabel 4.2 Perangkat Lunak Yang Digunakan .....	51
Tabel 4.3 Uji Coba Form Login .....	56
Tabel 4.4 Uji Coba <i>Form</i> Tambah Bahan .....	57
Tabel 4.5 Uji Coba Menu Detail Bahan .....	59
Tabel 4.6 Uji Coba Menu Hapus Bahan .....	60
Tabel 4.7 Uji Coba <i>Form</i> Tambah Alat .....	61
Tabel 4.8 Uji Coba <i>Form</i> Edit Alat.....	63
Tabel 4.9 Uji Coba Menu Detail Alat .....	63
Tabel 4.10 Uji Coba Menu Hapus Alat.....	64
Tabel 4.11 Uji Coba Form Peminjaman Bahan .....	65
Tabel 4.12 Uji Coba Form Peminjaman Alat.....	67
Tabel 4.13 Bobot Nilai .....	70
Tabel 4.14 Keterangan Nilai Kuesioner .....	70
Tabel 4.15 Hasil Pengujian Kuesioner Pertanyaan Nomor 1.....	71
Tabel 4.16 Hasil Pengujian Kuesioner Pertanyaan Nomor 2.....	71
Tabel 4.17 Hasil Pengujian Kuesioner Pertanyaan Nomor 3.....	72
Tabel 4.18 Hasil Pengujian Kuesioner Pertanyaan Nomor 4.....	72
Tabel 4.19 Hasil Pengujian Kuesioner Pertanyaan Nomor 5.....	73
Tabel 4.20 Hasil Pengujian Kuesioner Pertanyaan Nomor 6.....	73
Tabel 4.21 Hasil Pengujian Kuesioner Pertanyaan Nomor 7.....	74
Tabel 4.22 Hasil Pengujian Kuesioner Pertanyaan Nomor 8.....	74
Tabel 4.23 Hasil Pengujian Kuesioner Pertanyaan Nomor 9.....	75
Tabel 4.24 Hasil Pengujian Kuesioner Pertanyaan Nomor 10.....	75
Tabel 4.25 Hasil Pengujian Kuesioner Pertanyaan Nomor 11.....	76
Tabel 4.26 Hasil Pengujian Kuesioner Pertanyaan Nomor 12.....	76
Tabel 4.27 Hasil Pengujian Kuesioner Pertanyaan Nomor 13.....	77
Tabel 4.28 Hasil Pengujian Kuesioner Pertanyaan Nomor 14.....	77
Tabel 4.29 Hasil Pengujian Kuesioner Pertanyaan Nomor 15.....	78
Tabel 4.30 Hasil Pengujian Kuesioner Pertanyaan Nomor 16.....	78
Tabel 4.31 Hasil Pengujian Kuesioner Pertanyaan Nomor 17.....	79
Tabel 4.32 Hasil Pengujian Kuesioner Pertanyaan Nomor 18.....	79
Tabel 4.33 Hasil Pengujian Kuesioner Pertanyaan Nomor 19.....	80
Tabel 4.34 Hasil Pengujian Kuesioner Pertanyaan Nomor 20.....	80
Tabel 4.35 Jumlah Data Yang Digunakan .....	81
Tabel 4.36 Hasil Uji Coba Pencarian .....	83
Tabel 4.37 Hasil Uji Coba Pencarian (Lanjutan) .....	84
Tabel 4.38 Hasil Uji Coba Pencarian (Lanjutan) .....	85
Tabel 4.39 Hasil Uji Coba Pencarian (Lanjutan) .....	86
Tabel 4.40 Hasil Uji Coba Pencarian (Lanjutan) .....	87
Tabel 4.41 Hasil Uji Coba Pencarian (Lanjutan) .....	88

Tabel 4.42 Hasil Uji Coba Pencarian (Lanjutan) .....	89
Tabel 4.43 Hasil Uji Coba Pencarian (Lanjutan) .....	90
Tabel 4.44 Hasil Uji Coba Pencarian (Lanjutan) .....	91
Tabel 4.45 Hasil Uji Coba Pencarian (Lanjutan) .....	92



## ABSTRAK

Mu'awanah, Asifatul. 2018. **Implementasi Algoritma Boyer Moore Pada Pencarian Data Di Sistem Informasi Manajemen Laboratorium Fakultas Sains Dan Teknologi**. Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Linda Salma Angreani, M.T. (II) Supriyono, M.Kom.

Kata Kunci : Pencarian, Sistem Informasi Manajemen, Laboratorium, Algoritma Boyer Moore

Proses pencarian merupakan salah satu kegiatan penting dalam pemrosesan data. Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim dalam pengolahan data laboratorium masih dilakukan secara manual. Hal ini mengakibatkan waktu yang sangat lama dalam pencarian atau pengecekan data. Pembangunan sistem informasi manajemen laboratorium dengan mengimplementasikan algoritma *Boyer Moore* (BM) pada pencarian data ditujukan untuk membantu proses pencarian menjadi lebih cepat. Pada Algoritma *boyer moore* pencocokan karakter dimulai dari kanan ke kiri dan terdapat tiga tahapan proses. Pertama mendapatkan nilai *Occurence Heuristic* (OH) yang dihasilkan dari prosedur *preBmBc*, kemudian mendapatkan nilai *Match Heuristic* (MH) dari proses *preBmGs*, selanjutnya nilai maksimal OH dan MH akan digunakan untuk loncatan pergeseran pada pencocokan karakter BM. Hasil pengujian sebanyak 50 kali untuk pencarian lima kata menghasilkan rata-rata akurasi sebesar 98% dengan durasi waktu sebesar 0.00016 detik.

## ABSTRACT

Mu'awanah, Asifatul. 2018. **Implementation Of Boyer Moore Algorithm On Search Data In Management Information Systems Laboratory Science And Technology**. Undergraduate Theses. Informatics Engineering Department. Faculty of Sains And Technology. State Islamic University of Maulana Malik Ibrahim Malang. Advisor : (I) Linda Salma Angreani, M.T. (II) Supriyono, M.Kom.

Keyword : Searching, Management Information Systems, Laboratory, Boyer Moore Algorithm

The search process is one of the important activities in processing data. At Faculty of Science and Technology State Islamic University Maulana Malik Ibrahim, in processing the data laboratory is still done manually. This made a very long time in searching or checking the data. Development of laboratory information management system by implementing Boyer Moore (BM) algorithm in search data is expected to help the search process become faster. In the Algorithm boyer moore, the character matching is began from the right to left and there are three stages of the process. First get the Occurrence Heuristic (OH) value generated from the preBmBc procedure, then get the Match Heuristic (MH) value from the preBmGs process, then the maximum value of OH and MH will be used to see the BM characters. The test results of 50 times for five word search generated 98% averaged accuracy with the duration of 0.00016 seconds.

## ملخص

المعونة ، آسفة. 2018. تنفيذ خوارزمية بوير مور على البحث عن البيانات في نظام المعلومات الإدارية المختبر لكلية العلوم والتكنولوجيا. البحث الجامعي. قسم المعلوماتية كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم الإسلامية الحكومية بمالانج. المشرفة الأولى: ليندا سلمى أنغرياني، الماجستير. المشرف الثاني: سوفريونو، الماجستير.

الكلمات الرئيسية: البحث، نظام المعلومات الإدارية، المختبر، خوارزمية بوير مور

عملية البحث من الأنشطة الهامة في عملية معالجة البيانات. كانت معالجة البيانات في كلية العلوم والتكنولوجيا لجامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج لا تزال يدوية. فتؤدي هذه الحالة إلى العملية بوقت طويل للغاية في البحث أو التحقق من البيانات. يهدف بناء نظام معلومات إدارة المعمل لتنفيذ الخوارزمية بوير مور (BM) على البحث عن البيانات إلى المساعدة في عملية البحث ليكون أسرع. كانت مطابقة الأحرف في بوير مور خوارزمية تبدأ من اليمين إلى اليسار، وفيها ثلاث مراحل. الأول الحصول على حدوث قيمة *Match Occurrence Heuristic* (OH) المحسولة عليها عن إجراء *preBmBc*، ثم الحصول على قيمة *Match Heuristic* (MH) المحسولة عليها عن إجراء *preBmGs*، وبالتالي استخدم الحد الأقصى لقيمة OH و MH لندوس الاحتكاك على مطابقة الأحرف ل BM. وأما نتائج الاختبار في نحو 50 مرة للبحث عن خمس كلمات تولد متوسط دقة 98% بمدة البحث 0.00016 ثانية.

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang Masalah

Proses pencarian merupakan salah satu kegiatan penting dalam pemrosesan data. Banyaknya data yang ada akan membutuhkan waktu yang lebih lama jika proses pencarian dilakukan secara manual. Hal tersebut dapat membuang waktu yang seharusnya dapat digunakan untuk melakukan kegiatan lain yang bermanfaat. Kecepatan dan ketepatan dalam pencarian data sangat dibutuhkan untuk efisiensi waktu dalam melakukan suatu pekerjaan.

Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim dalam pengolahan data laboratorium kebanyakan masih manual. Seperti penyimpanan data alat dan bahan praktikum yang mana masih didata dalam kertas, dan beberapa menggunakan *Microsoft excel* dalam penyimpanannya. Peminjaman dan pengembalian alat praktikum juga masih menggunakan *log book* dikertas. Hal ini akan membutuhkan waktu yang sangat lama dalam pencarian data untuk laporan atau pengecekan data yang yang dibutuhkan karena banyaknya data yang disimpan.

Adanya Sistem Informasi Manajemen Laboratorium dan pencarian data yang efektif dan efisien akan mempermudah kegiatan pengguna laboratorium Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim untuk melakukan pencarian data seperti pada laboran untuk pengecekan data alat

dan bahan praktikum yang masih tersedia, sedang dipinjam dan stok sudah habis. Pencarian juga dibutuhkan untuk mengetahui mahasiswa yang sedang meminjam alat praktikum, dan belum mengembalikan alat praktikum.

Proses pencarian ini dapat menghabiskan waktu dalam ruang pencarian yang besar sehingga diperlukan suatu teknik pencarian yang efisien. Teknik pencarian yang digunakan adalah menggunakan algoritma *boyer moore* karena pada penelitian sebelumnya telah disimpulkan bahwa algoritma *Boyer-Moore* memiliki waktu pencarian yang cepat (Hasan dan Rashid, 2012). Algoritma *Boyer-Moore* telah banyak dikenal oleh masyarakat dan dianggap paling efisien untuk pencarian string. Ide di balik algoritma ini adalah dengan memulai pencocokan karakter dari kanan dan bukan dari kiri, maka akan lebih banyak informasi yang didapat. Jika terjadi ketidakcocokan string dari kanan *pattern* maka ketidakcocokan akan membantu 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, jika kita bisa melompati atau tidak melakukan perbandingan karakter yang diprediksi akan gagal (Jon Orwant, et al, 1999).

Efisiensi waktu yang digunakan dalam pencarian sangat penting karena waktu yang dipakai tidak akan terbuang sia-sia saja karena menunggu satu proses yang seharusnya bisa dilakukan dengan cepat, seperti halnya pada agama islam, waktu adalah salah satu nikmat yang agung dari Allah SWT kepada manusia. Sudah sepantasnya manusia memanfaatkannya secara baik, efektif dan semaksimal mungkin untuk amal shalih. Islam menganjurkan agar manusia

memanfaatkan waktu dan kesempatan yang dimiliki sehingga ia tidak termasuk golongan orang yang merugi. Hal itu tercantum dalam Qur'an Surat. 'Ashr dan Rasulullah Saw juga menganjurkan agar manusia memanfaatkan kesempatan yang ia miliki. Allah Ta'ala Berfirman:

وَالْعَصْرِ (١) إِنَّ الْإِنْسَانَ لَفِي خُسْرٍ (٢) إِلَّا الَّذِينَ آمَنُوا وَعَمِلُوا الصَّالِحَاتِ وَتَوَّصُوا بِالحَقِّ وَتَوَّصُوا  
بِالصَّبْرِ (٣)

*“Demi masa. Sesungguhnya manusia itu benar-benar dalam kerugian, kecuali orang-orang yang beriman dan mengerjakan amal saleh dan nasehat menasehati supaya mentaati kebenaran dan nasehat menasehati supaya menepati kesabaran.”* (QS. Al' Ashr:1-3).

Pada Tafsir Ibnu Katsir QS. Al' Ashr:1-3 menjelaskan bahwa masa yang didalamnya berbagai aktivitas anak cucu adam berlangsung, baik dalam wujud kebaikan maupun keburukan. Allah Ta'ala telah bersumpah dengan masa tersebut bahwa manusia itu dalam kerugian, yakni benar-benar merugi dan binasa. *“Kecuali orang-orang yang beriman dan mengerjakan amal shalih”*. Allah memberikan pengecualian dari kerugian itu bagi orang-orang yang beriman dengan hati mereka dan mengerjakan amal shalih melalui anggota tubuhnya. *“Dan nasihat-menasihati supaya mentaati kebenaran”*. Yaitu, mewujudkan semua bentuk ketaatan dan meninggalkan semua yang diharamkan. *“Dan nasihat-menasihati supaya menepati kesabaran”*. Yakni bersabar atas segala macam cobaan, takdir, serta gangguan yang dilancarkan kepada orang-orang yang menegakkan amar ma'ruf nahi munkar.

Berdasarkan latar belakang diatas, pada penelitian mengusulkan bagaimana penerapan algoritma *boyer moore* untuk ketepatan pencarian data pada Sistem Informasi Manajemen Laboratorium Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Diharapkan dengan menggunakan algoritma ini dapat menyelesaikan permasalahan dalam pencarian data pada Sistem Informasi Manajemen Laboratorium Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.

## 1.2 Pernyataan Masalah

Berdasarkan latar belakang yang sudah dijelaskan pada sub bagian sebelumnya, dapat diambil pertanyaan penelitian sebagai berikut:

- a. Seberapa berguna sistem informasi laboratorium untuk kebutuhan *user* ?
- b. Bagaimana implementasi algoritma *boyer moore* untuk pencarian data pada Sistem Informasi Manajemen Laboratorium fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang?

## 1.3 Tujuan Penelitian

Dari pertanyaan rumusan masalah yang sudah dipaparkan di subbab sebelumnya maka dapat diambil tujuan penelitian sebagai berikut :

- a. Mengetahui kegunaan Sistem Informasi Manajemen Laboratorium untuk kebutuhan *user*.
- b. Mengetahui implementasi algoritma *boyer moore* untuk pencarian data pada Sistem Informasi ManajemenLaboratorium fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.

#### 1.4 Manfaat Penelitian

Adapun manfaat yang diperoleh dalam penelitian ini adalah sebagai berikut:

- a. Memudahkan dalam pengecekan data yang berhubungan dengan sistem informasi manajemen laboratorium.
- b. Memudahkan proses pencarian data yang ada dalam sistem informasi manajemen laboratorium sesuai dengan pencarian yang diinginkan.

#### 1.5 Batasan Masalah

Berdasarkan masalah-masalah yang sudah dijelaskan pada sub bagian sebelumnya, pengembangan sistem dapat dibatasi dengan ruang lingkup sebagai berikut:

- a. Penelitian digunakan untuk pencarian data pada Sistem Informasi Manajemen Laboratorium Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
- b. Sistem Informasi Manajemen Laboratorium dibangun dalam bentuk Website dengan menggunakan *framework CodeIgniter* dan *database MySQL*.

#### 1.6 Sistematika Penulisan

Dalam penulisan skripsi ini, secara keseluruhan terdiri dari lima bab yang masing-masing bab disusun dengan sistematika sebagai berikut :

### **BAB I PENDAHULUAN**

Bab ini merupakan bagian awal, dalam bab ini berisi latar belakang, pernyataan masalah, tujuan penelitian, batasan masalah, manfaat penelitian dan sistematika penulisan laporan.

## **BAB II STUDI PUSTAKA**

Bab ini berisi tentang teori-teori yang berhubungan dengan permasalahan yang diangkat dari penelitian ini, antara lain pencarian, sistem informasi manajemen laboratorium, algoritma *Boyer-Moore* dan penelitian-penelitian terkait.

## **BAB III METODOLOGI PENELITIAN**

Bab ini menjelaskan tentang analisis sistem dan perancangan sistem yang akan dibuat sekaligus batasan-batasan sistem serta di dalamnya juga terdapat beberapa diagram-diagram sistem.

## **BAB IV UJI COBA DAN PEMBAHASAN**

Bab ini berisi mengenai pengujian dan analisis dari hasil pengujian dari sistem yang telah dibangun berdasarkan hasil perancangan pada bab 3 sebelumnya.

## **BAB V KESIMPULAN DAN SARAN**

Bab ini berisi kesimpulan dan saran seluruh penelitian yang telah dilakukan.

## BAB II

### STUDI PUSTAKA

#### 2.1 Pencarian

Pencarian merupakan proses yang fundamental dalam pengolahan data. Proses pencarian adalah menemukan nilai tertentu di dalam sekumpulan data yang bertipe sama (baik bertipe dasar atau bertipe bentukan). Sebagai contoh untuk mengubah data tertentu, langkah pertama yang harus dilakukan adalah mencari keberadaan data tersebut didalam kumpulannya. Jika data yang dicari ditemukan, maka data tersebut dapat diubah nilainya dengan data yang baru. (Rinaldi Munir, 2011 : 395).

Metode pencarian data dapat dilakukan dengan dua cara yaitu pencarian *internal* dan pencarian *eksternal*. Pencarian internal adalah pencarian terhadap sekumpulan data yang disimpan di dalam memori utama, sedangkan pencarian *eksternal* adalah pencarian terhadap sekumpulan data yang disimpan di dalam memori sekunder seperti tape atau disk.

Selain itu metode pencarian data dapat dikelompokkan menjadi pencarian statis dan pencarian dinamis. Pada pencarian statis, banyaknya rekaman yang diketahui dianggap tetap, pada pencarian dinamis, banyaknya rekaman yang diketahui bisa berubah-ubah yang disebabkan oleh penambahan atau penghapusan suatu rekaman.

Menurut ( Rinaldi Munir, 2011 : 397 - 418 ) Ada dua macam teknik pencarian data yaitu pencarian beruntun (*sequential search*) dan pencarian bagi

dua (*binary search*). Perbedaan dari dua teknik ini terletak pada keadaan data. Pencarian sekuensial digunakan apabila data dalam keadaan acak atau tidak teratur. Sebaliknya pencarian biner digunakan pada data yang sudah dalam keadaan teratur.

### 2.1.1 Algoritma Pencarian String

#### 2.1.1.1 Algoritma *Brute Force*

Algoritma brute force merupakan Algoritma pencocokan string yang ditulis tanpa memikirkan peningkatan performa. Algoritma ini sangat jarang dipakai dalam praktik, namun berguna dalam studi perbandingan dan studi-studi lainnya. Secara sistematis, langkah-langkah yang dilakukan Algoritma bruteforce pada saat mencocokkan string adalah:

1. Algoritma brute force mulai mencocokkan pattern pada awal teks.
2. Dari kiri ke kanan, Algoritma ini akan mencocokkan karakter per karakter pattern dengan karakter di teks yang bersesuaian, sampai salah satu kondisi berikut dipenuhi:
3. Karakter di pattern dan di teks yang dibandingkan tidak cocok (*mismatch*).
4. Semua karakter di pattern cocok. Kemudian Algoritma akan memberitahukan penemuan di posisi ini.
5. Algoritma kemudian terus menggeser pattern sebesar satu ke kanan.
6. mengulangi langkah ke-2 sampai pattern berada di ujung teks.

Algoritma *brute force string match* adalah Algoritma yang paling sederhana untuk memecahkan masalah string match. Cara kerja Algoritma ini

adalah dengan mencoba setiap posisi pattern (kata yang akan dicocokkan) terhadap teks, kemudian dilakukan proses pencocokan setiap karakter dan teks pada posisi tersebut.

### 2.1.1.2 Algoritma Knuth Morris Pratt

Algoritma Knuth Morris Pratt merupakan salah satu algoritma yang sering digunakan untuk menyelesaikan masalah pencocokan string. Algoritma ini adalah penyempurnaan dari algoritma pencocokan string dengan menggunakan algoritma *brute force*. Pada algoritma *brute force*, setiap kali ditemukan ketidakcocokan pattern dengan teks, maka pattern akan digeser satu ke kanan. Sedangkan pada algoritma Knuth-Morris-Pratt, kita memelihara informasi yang digunakan untuk melakukan jumlah pergeseran. Algoritma menggunakan informasi tersebut untuk membuat pergeseran angka lebih jauh, tidak hanya satu karakter seperti pada algoritma *brute force*. Dengan algoritma Knuth Morris Pratt ini, waktu pencarian dapat dikurangi secara signifikan. Algoritma Knuth Morris Pratt ini dikembangkan oleh D. E. Knuth, bersama-sama dengan J. H. Morris dan V. R. Pratt. Algoritma Knuth-Morris-Pratt memelihara informasi yang digunakan saat melakukan pergeseran. Informasi ini digunakan untuk melakukan pergeseran yang lebih jauh, tidak seperti *brute force* yang melakukan pergeseran per karakter. Pergeseran dilakukan berdasarkan *suffix* kesamaan *suffix* dan *prefix* dalam pattern dan yang ditemukan di dalam teks. Dalam algoritma Knuth Morris Pratt ini kita akan menemui beberapa definisi yang nantinya akan digunakan dalam algoritma ini.

### 2.1.2 Algoritma *Boyer Moore*

Algoritma *Boyer Moore* diciptakan oleh R.M Boyer dan J.S Moore. Algoritma ini terkenal karena banyak diterapkan pada algoritma pencocokan untuk banyak string (*multi pattern*) (Imam Sulisty, 2010). *The Boyer-Moore algorithm is one of the famous exact string matching algorithms that used in single pattern matching and it considers very fast in its performance.* (Awsan Abdulrahman Hasan dan Nur'Aini Abdul Rashid, 2012).

Algoritma *Boyer Moore* telah dibuktikan sebagai salah satu algoritma yang paling efisien dalam aplikasi pencarian string dengan menggunakan *natural language* (bukan *binary language*). Algoritma ini telah sering diimplementasikan untuk fungsi “*Search*” dan “*Subtitute*” pada text editor. Padasarnya cara kerja algoritma ini mirip dengan algoritma *Knuth-MorrisPrat* (KMP) dimana kedua algoritma ini akan melakukan lompatan pengecekan dalam proses pencarian string. Namun berbeda dengan algoritma KMP, algoritma *Boyer Moore* ini melakukan perbandingan *pattern* mulai dari kanan ke kiri. (Evlyn Dwi Tambun, 2010) *Idea of the Boyer-Moore Algorithm The algorithm uses knowledge of the search text to improve search speed significantly. It uses a step of preprocessing to create an occurance-function and shifting function which are used to perform bad character heuristics and good suffix heuristics respectively.* (Prabhakar Gupta, et al. 2010).

Ide dari algoritma *Boyer-Moore* menggunakan pengetahuan tentang pencarian teks untuk meningkatkan kecepatan pencarian secara signifikan. Algoritma *Boyer-Moore* menggunakan suatu langkah sebelum proses untuk

membuat *occurance-function* dan *shifting-function* yang digunakan untuk melakukan bad character heuristics dan good suffix heuristics masing-masing. *Boyer Moore* menggunakan dua heuristik untuk memutuskan seberapa jauh melompat: *badcharacter heuristic*, juga sering disebut *occurrence heuristic*, dan heuristik yang *good-character*, juga disebut *match heuristic*. Informasi untuk heuristik setiap dipertahankan dalam sebuah array yang dibangun pada awal operasi yang cocok. (Jon Orwant, et al, 1999)

Algoritma yang digunakan untuk pencarian string semakin berkembang dari hari ke hari. Tujuan utamanya tentu saja untuk mencari string seakurat dan secepat mungkin. Hingga saat ini algoritma pencarian string terbagi atas 3 kategori berdasarkan arah pencocokan string yaitu dari kiri ke kanan, kanan ke kiri, dan dari arah yang ditentukan secara spesifik. Metode pencocokan dari kiri ke kanan merupakan metode yang paling natural karena sesuai dengan arah membaca, pencocokan string dari kanan ke kiri merupakan metode yang dianggap paling efisien dalam praktiknya, dan pencocokan string dari arah yang ditentukan secara spesifik merupakan algoritma yang memiliki hasil yang paling baik secara teoritis (Kumara, 2009). Algoritma yang dianggap memiliki hasil yang paling baik dalam praktiknya merupakan algoritma yang bergerak mencocokkan string dari arah kanan ke kiri. Algoritma *Boyer-Moore* adalah salah satu contoh algoritma yang menggunakan arah dari kanan ke kiri. Algoritma ini telah banyak dikenal oleh masyarakat dan dianggap paling efisien untuk pencarian string. Ide dibalik algoritma ini adalah bahwa dengan memulai pencocokkan karakter dari kanan, dan bukan dari kiri, maka akan lebih banyak informasi yang didapat (Fauzy, 2011). Algoritma ini kemudian dikembangkan sehingga tercipta algoritma *Turbo*

*Boyer-Moore* dan *Tuned Boyer-Moore*. Berdasarkan dokumentasi MySQL yang dinyatakan pada kalimat *If you use ... LIKE '%string%' and string is longer than three characters, MySQL uses the Turbo Boyer-Moore algorithm to initialize the pattern for the string and then uses this pattern to perform the search more quickly*. Menyatakan bahwa algoritma mysql *pattern matching* menggunakan algoritma *Turbo Boyer Moore* yang merupakan pengembangan dari algoritma *boyer moore*. Algoritma *Turbo Boyer-Moore* adalah sebuah variasi dari algoritma *Boyer-Moore*. Bila dibandingkan dengan algoritma *Boyer-Moore*, algoritma ini membutuhkan ruang lebih tapi tidak memerlukan pemrosesan ekstra. Ruang ekstra yang diperlukan berguna untuk mengingat faktor dari teks yang cocok dengan akhiran dari string yang dicari selama attempt terakhir dan hanya jika *good-suffix shift* dilakukan (Charras & Lecroq, 2009). Pada Penelitian yang dilakukan oleh Vina Sagita dan Maria Irminda Prasetiyowati dimana mereka melakukan perbandingan Implementasi Algoritma *Boyer-Moore*, *Turbo Boyer-Moore*, dan *Tuned Boyer-Moore* dalam Pencarian String untuk mengetahui manakah algoritma yang mampu mencari string paling cepat. Dari hasil penelitian yang didapat, bisa disimpulkan bahwa algoritma *Boyer-Moore* memiliki waktu pencarian tercepat dari tiga varian *Boyer-Moore* yang telah ditentukan. Algoritma *Turbo Boyer-Moore* merupakan algoritma tercepat kedua dan yang paling lambat adalah *Tuned Boyer-Moore*.

Dari pernyataan diatas penulis akan menggunakan algoritma *boyer moore* pada proses pencarian dari sistem informasi manajemen laboratorium fakultas saintek, dimana algoritma ini telah banyak dinyatakan sebagai algoritma yang

efisien dalam melakukan pencarian, karena pergeseran pattern berdasarkan perbandingan nilai OH dan MH.

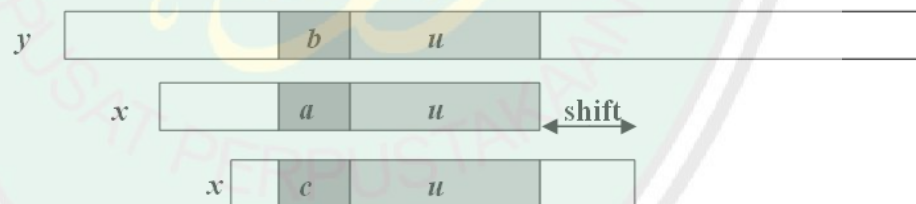
### 2.1.2.1 Deskripsi Kerja Algoritma *Boyer Moore*

Untuk menjelaskan konsep dari *good-suffix shift* dan *bad-character shift* diperlukan contoh kasus, seperti kasus ketidakcocokan ditengah pencocokan karakter pada teks dan *pattern*. Karakter *pattern*  $x[i]=a$  tidak cocok dengan karakter teks  $y[i+j]=b$  saat pencocokan pada posisi  $j$ . Maka  $x[i+1 .. m-1]=y[i+j+1 .. j+m-1]=u$  dan  $x[i] \neq y[i+j]$ .

### 2.1.2.2 *Good-suffix shift*

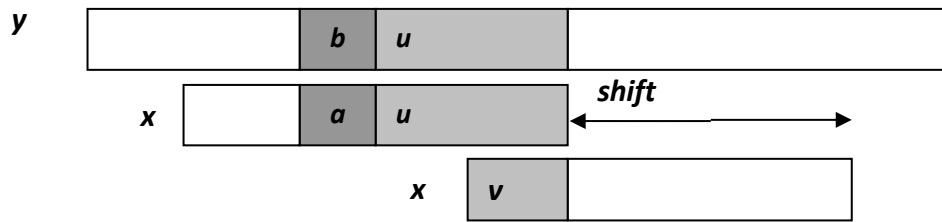
Konsep dari fungsi *good-suffix shift* adalah sebagai berikut:

1. *Good-suffix shift* adalah pergeseran yang dibutuhkan dari  $x[i]=a$  ke karakter lain yang letaknya lebih kiri dari  $x[i]$  dan terletak di sebelah kiri segmen  $u$ . Kasus ini ditunjukkan pada Gambar 2.1



Gambar 2.1 *Good-suffix shift*,  $u$  terjadi lagi didahului karakter  $c$  berbeda dari  $a$

2. Jika tidak ada segmen yang sama dengan  $u$ , maka dicari  $u$  yang merupakan *suffix* terpanjang  $u$ . Kasus ini ditunjukkan pada Gambar 2.2



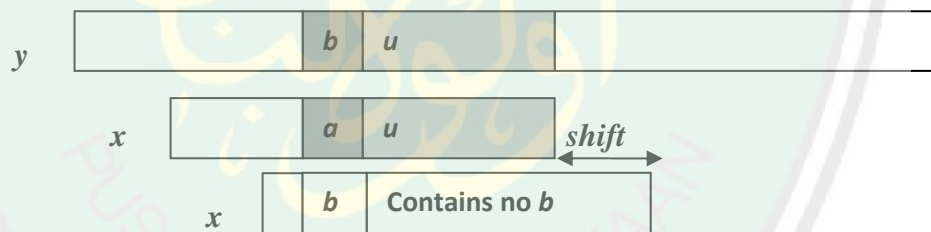
Gambar 2.2 Good-suffix shift, hanya suffix dari u yang terjadi lagi di pattern x

2.1.2.3 Bad-character shift

Berdasarkan contoh kasus di atas, bad-character adalah karakter pada teks yaitu  $y[i+j]$  yang tidak cocok dengan karakter pada pattern.

Konsep dari fungsi bad-character shift adalah sebagai berikut:

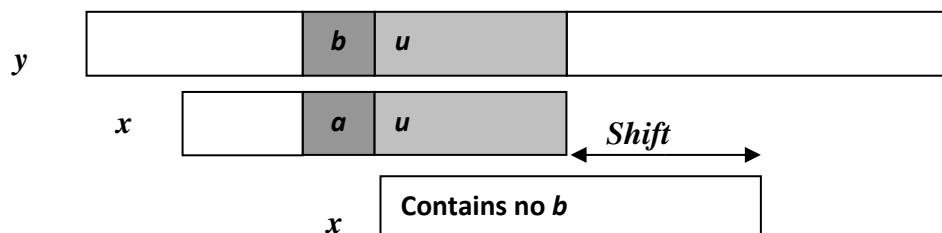
1. Jika bad-character  $y[i+j]$ terdapat pada pattern di posisi terkanan  $k$  yang lebih kiri dari  $x[i]$ maka pattern digeser ke kanan sejauh  $i-k$ . Kasus ini ditunjukkan pada Gambar 2.3



Gambar 2.3 Bad-character shift, b terdapat di pattern x

2. Jika bad-character  $y[i+j]$ tidak ada pada pattern sama sekali, maka pattern digeser ke kanan sejauh  $i$ . Kasus ini ditunjukkan pada Gambar

2.4



Gambar 2.4 Bad-character shift, b tidak ada di pattern x

3. Jika *bad-character*  $y[i+j]$  terdapat pada *pattern* di posisi terkanan  $k$  yang lebih kanan dari  $x[i]$  maka *pattern* seharusnya digeser sejauh  $i-k$  yang hasilnya negatif (*pattern* digeser kembali ke kiri). Maka bila kasus ini terjadi, akan diabaikan.

Pada kasus ketidakcocokan di atas, algoritma akan membandingkan langkah yang diambil oleh fungsi *good-suffix shift* dan *bad-character shift* di mana langkah yang paling besar yang akan digunakan

#### 2.1.2.4 Cara kerja algoritma *Boyer Moore*.

Cara kerja dari algoritma *Boyer Moore* adalah sebagai berikut:

1. Menjalankan prosedur *preBmBc* dan *preBmGs* untuk mendapatkan inisialisasi.
  - a. Menjalankan prosedur *preBmBc*. Fungsi dari prosedur ini adalah untuk menentukan berapa besar pergeseran yang dibutuhkan untuk mencapai karakter tertentu pada *pattern* dari karakter *pattern* terakhir/terkanan. Hasil dari prosedur *preBmBc* disimpan pada tabel *BmBc* yang berisi nilai *Occurance Heuristic* (OH)
  - b. Menjalankan prosedur *preBmGs*. Sebelum menjalankan isi prosedur ini, prosedur *suffix* dijalankan terlebih dulu pada *pattern*. Fungsi dari prosedur *suffix* adalah memeriksa kecocokan sejumlah karakter yang dimulai dari karakter terakhir/terkanan dengan sejumlah karakter yang dimulai dari setiap karakter yang lebih kiri dari karakter terkanan tadi. Hasil dari prosedur *suffix* disimpan pada tabel *suff*. Jadi  $suff[i]$  mencatat panjang dari *suffix* yang cocok dengan segmen dari

*pattern* yang diakhiri karakter ke-*i*. Dengan prosedur preBmGs, dapat diketahui berapa banyak langkah pada *pattern* dari sebuah segmen ke segmen lain yang sama yang letaknya lebih kiri dengan karakter di sebelah kiri segmen yang berbeda. Prosedur preBmGs menggunakan tabel *suff* untuk mengetahui semua pasangan segmen yang sama. Hasil dari prosedur preBmGs disimpan pada tabel BmGs yang berisi nilai *Match Heuristic* (MH).

2. Dilakukan proses pencarian *string* dengan menggunakan hasil dari prosedur preBmBc dan preBmGs yaitu tabel BmBc dan BmGs.

Berikut ini diberikan contoh untuk menjelaskan proses inialisasi dari algoritma *Boyer Moore* dengan *pattern* **gcagagag** yang akan dicari pada *string* **gcacgcagagagtatacagtagc**.

Dengan prosedur preBmBc, didapatkan jumlah pergeseran pada *pattern* yang dibutuhkan untuk mencapai karakter a,c,g,t dari posisi terkanan. Berdasarkan contoh diketahui untuk mencapai masing-masing karakter tadi dibutuhkan pergeseran sebanyak 1, 6, 2 dan 8.

Dengan prosedur preBmGs, dijalankan prosedur *suffix* terlebih dulu. Dengan prosedur *suffix* akan diketahui:

- $\text{suff}[0] = 1$ , 1 karakter g posisi 7 cocok dengan 1 karakter g posisi 0.
- $\text{suff}[1] = 0$ , karakter g posisi 7 tidak cocok dengan karakter c posisi 1.
- $\text{suff}[2] = 0$ , karakter g posisi 7 tidak cocok dengan karakter a posisi 2.
- $\text{suff}[3] = 2$ , 2 karakter dimulai dari karakter g posisi 7 cocok dengan 2 karakter dimulai dari karakter g posisi 3, yang artinya karakter a,g posisi 6,7 cocok dengan karakter a,g posisi 2,3.

- $\text{suff}[4] = 0$ , karakter g posisi 7 tidak cocok dengan karakter a posisi 4.
- $\text{suff}[5] = 4$ , 4 karakter dimulai dari karakter g posisi 7 cocok dengan 4 karakter dimulai dari karakter 5, artinya karakter a,g,a,g posisi 4,5,6,7 cocok dengan karakter a,g,a,g posisi 2,3,4,5.
- $\text{suff}[6] = 0$ , karakter g posisi 7 tidak cocok dengan karakter a posisi 6.
- $\text{suff}[7] = 8$ , 8 karakter g,c,a,g,a,g,a,g posisi 0,1,2,3,4,5,6,7 cocok dengan 8 karakter g,c,a,g,a,g,a,g posisi 0,1,2,3,4,5,6,7.

Dengan prosedur BmGs akan didapatkan:

0 1 2 3 4 5 6 7

g c a g a g a g

- $\text{bmGs}[0] = 7$ , karakter ke-0 g adalah karakter sebelah kiri segmen cagagag. Tidak ada segmen cagagag lain dengan karakter sebelah kiri bukan g maka digeser 7 langkah.
- $\text{bmGs}[1] = 7$ , karakter ke-1 c adalah karakter sebelah kiri segmen agagag. Tidak ada segmen agagag lain dengan karakter sebelah kiri bukan c maka digeser 7 langkah.
- $\text{bmGs}[2] = 7$ , karakter ke-2 a adalah karakter sebelah kiri segmen gagag. Tidak ada segmen gagag lain dengan karakter sebelah kiri bukan a maka digeser 7 langkah.
- $\text{bmGs}[3] = 2$ . karakter ke-3 g adalah karakter sebelah kiri segmen agag. Karena ada segmen agag posisi 2,3,4,5 dengan karakter sebelah kiri bukan g yaitu c posisi 1 maka digeser 2 langkah.

- $bmGs[4]= 7$ , karakter ke-4 a adalah karakter sebelah kiri segmen gag. Karena tidak ada seamen gag lain dengan karakter sebelah kiri bukan a maka digeser 7 langkah.
- $bmGs[5]= 4$ . karakter ke-5 g adalah karakter sebelah kiri seamen ag. Karena ada segmen ag posisi 2,3 dengan karakter sebelah kiri bukan g yaitu c posisi 1 maka digeser 4 langkah.
- $bmGs[6]= 7$ , karakter ke-6 a adalah karakter sebelah kiri segmen yaitu a posisi 7. Karena tidak ada segmen g dengan karakter sebelah kirinya bukan a maka digeser 7 langkah.
- $bmGs[7]= 1$ , karakter ke-7 g adalah karakter sebelah kiri segmen dan karena segmen tidak ada maka digeser 1 langkah

## 2.2 Sistem Informasi Manajemen Laboratorium

Definisi sistem informasi manajemen menurut Jogiyanto Hartono (2000:700) Sistem Informasi Manajemen adalah kumpulan dari interaksi sistem-sistem informasi yang bertanggung jawab mengumpulkan dan mengolah data untuk menyediakan informasi yang berguna untuk semua tingkat manajemen didalam kegiatan perencanaan dan pengendalian. Pada dasarnya sistem informasi mempunyai tiga kegiatan utama yaitu: menerima data sebagai masukan, kemudian memprosesnya dengan melakukan perhitungan, penggabungan unsur-unsur data dan akhirnya dapat diperoleh informasi yang diperlukan sebagai keluaran. Prinsip tersebut berlaku baik bagi sistem informasi manual maupun sistem informasi modern dengan penggunaan perangkat komputer.

Sistem informasi manajemen yang efektif menurut Raymond Coleman dalam Moekijat (1991:40) adalah bahwa sistem tersebut dapat memberikan data

yang cermat, tepat waktu, dan yang penting artinya bagi perencanaan, analisis, dan pengendalian manajemen untuk mengoptimalkan pertumbuhan organisasi. George M. Scott yang diterjemahkan oleh Budiman (2001:100), mengemukakan Sistem Informasi Manajemen adalah serangkaian sub-sistem informasi yang menyeluruh dan terkoordinasi dan secara rasional terpadu yang mampu mentransformasi data sehingga menjadi informasi lewat serangkaian cara guna meningkatkan produktivitas yang sesuai dengan gaya dan sifat manajer atas dasar kriteria mutu yang telah ditetapkan.

Dari pengertian diatas dapat disimpulkan bahwa sistem informasi manajemen laboratorium merupakan sistem berbasis komputer yang menyediakan informasi bagi beberapa pemakai dengan kebutuhan yang serupa diantaranya dekan, ketua program studi, kepala laboratorium, laboran, asisten dosen dan mahasiswa. Dimana para pemakai tersebut merupakan bagian dari organisasi formal program studi sistem informasi. Informasi menjelaskan laboratorium mengenai apa yang terjadi dimasa lalu, apa yang sedang terjadi sekarang dan mungkin apa yang terjadi dimasa depan. Informasi tersebut tersedia dalam bentuk laporan periodik, laporan khusus, dan *output* dari simulasi matematika. Output informasi digunakan oleh pengguna dalam program studi sistem informasi saat mereka membuat keputusan untuk memecahkan masalah mengenai laboratorium.

### **2.3 Receiver Operating Characteristic**

Kurva ROC pertama kali digunakan pada perang dunia II untuk menganalisis sinyal radar sebelum dikembangkan dalam signal detection theory. Berdasarkan serangan di Pearl Harbon tahun 1941, tentara Amerika melakukan riset untuk meningkatkan ketepatan prediksi dalam mendeteksi sinyal radar

pesawat Jepang. Akhir-akhir ini penggunaan kurva ROC semakin populer dalam berbagai aplikasi terutama dalam bidang medis, radiologi, dan *processing image*. *Receiver Operating Characteristic* (ROC) adalah hasil pengukuran klasifikasi dalam bentuk 2-dimensi. Berikut ada empat peluang yang dapat diformulasikan dalam tabel kontingensi 2 x 2 untuk menganalisis ROC :

Tabel 2.1 Tabel Kontingensi ROC

		Kelas Sebenarnya	
		True	False
Kelas Prediksi	Positive	True Positive	False Positive
	Negative	True Negative	False Negative

Adapun kriteria ROC adalah sebagai berikut:

- True Positive Rate disebut juga Sensitivity (TPR)= $TP/(TP+FN)$  (2.1)
- True Negative Rate disebut juga Specifity (TNR)=  $TN/(TN+FP)$  (2.2)
- Accuracy =  $(TP+TN)/(TP+FP+TN+FN)$ . (2.3)

Dimana:

TP = True Positive yaitu klasifikasi yang dari kelas yang positif

FN = False Negative yaitu kesalahan Type II

FP = False Positive atau kesalahan Type I

## 2.4 Penelitian Terkait

Sagita & Prasetyowati (2013) pada penelitiannya yang berjudul Studi Perbandingan Implementasi Algoritma *Boyer-Moore*, *Turbo Boyer-Moore*, dan *Tuned Boyer-Moore* dalam Pencarian String telah melakukan penelitian untuk mengetahui manakah algoritma yang mampu mencari string paling cepat dalam pencarian string. Dari hasil penelitian yang didapat, bisa disimpulkan bahwa algoritma *Boyer-Moore* memiliki waktu pencarian tercepat dari tiga varian *Boyer-Moore* yang telah ditentukan. Algoritma *Turbo Boyer-Moore* merupakan algoritma tercepat kedua dan yang paling lambat adalah *Tuned Boyer-Moore*.

Rahmanita (2014) melakukan penelitian yang membahas tentang pencarian kata/string dalam sebuah dokumen dengan menggunakan algoritma *boyer moore*, hasil yang didapatkan dalam penelitian ini menyatakan bahwa Algoritma *Boyer Moore* mempunyai keunggulan dalam waktu menemukan pattern yang akan dicari dalam ukuran file yang lebih besar. Pada file berekstensi .txt dengan file size 4.625 byte dengan varian keyword berbeda. Varian keyword yang sedikit dapat ditempuh dengan waktu lebih cepat pada pattern 'a' yaitu 0,228 detik dengan banyak pattern ditemukan 685. Pada file berekstensi .doc dengan file size 39.936 byte, pada pattern 'yang' dapat diproses dengan waktu 0,542 detik dengan ditemukannya pattern sebanyak 18. Sedangkan pada file berestensi .pdf optimalisasi dari pattern matching yang diproses adalah 0,103 detik dengan file size 15.804 byte dan banyak pattern = 1 untuk pencarian pattern 'suatu'.

Ginting (2014) pada penelitiannya yang berjudul Penerapan Algoritma *Boyer Moore* pada Aplikasi Pengajuan Judul Skripsi Berbasis Web telah melakukan penelitian untuk menyelesaikan permasalahan pengajuan judul

skripsi yang selama ini dikerjakan secara manual dalam memutuskan diterima atau ditolak judul yang diajukan oleh mahasiswa. Adapun cara yang dilakukan adalah dengan melakukan pencocokan judul skripsi yang diajukan dengan judul skripsi yang telah diterima. Jika judul skripsi yang diajukan dengan judul skripsi yang telah diterima menghasilkan karakter (string) yang sama dalam rentang tertentu maka judul skripsi dinyatakan tidak diterima dan jika judul skripsi yang diajukan dengan judul yang telah diterima menghasilkan karakter (string) yang tidak sama dalam rentang tertentu maka judul dinyatakan diterima. Algoritma yang digunakan dalam penelitian ini adalah algoritma *boyer moore* dengan hasil Algoritma *Boyer Moore* cocok digunakan untuk melakukan pencarian string tetapi tidak cocok digunakan untuk mengambil informasi dari kecocokan string berdasarkan satu kata atau lebih, dan untuk menentukan judul diterima ditentukan dari hasil persentase kesamaan string, hal tersebut tidak terlalu efektif dikarenakan judul dapat memiliki kesamaan string yang besar tetapi tidak memiliki arti yang sama atau sebaliknya.

Halim Agung, Y. (2016) telah melakukan penelitian tentang bagaimana cara melakukan pendeteksian kata kunci yang terdapat pada soal matematika dan fisika sekaligus dicocokkan dengan menggunakan metode pencocokan kata. Algoritma yang digunakan dalam penelitian ini adalah algoritma *boyer moore*. Hasil dari penelitian ini menyatakan bahwa Metode string matching dengan algoritma *boyer moore* dapat diterapkan pada aplikasi pencarian rumus matematika dan fisika tingkat SMA yang disertai dengan pengujian algoritma sebanyak 30 kali dimana aplikasi dapat mencocokkan pola kata kunci pada soal serta menampilkan list atau

kumpulan rumus matematika dan fisika tingkat SMA yang cocok dan sesuai dengan soal yang dikerjakan.

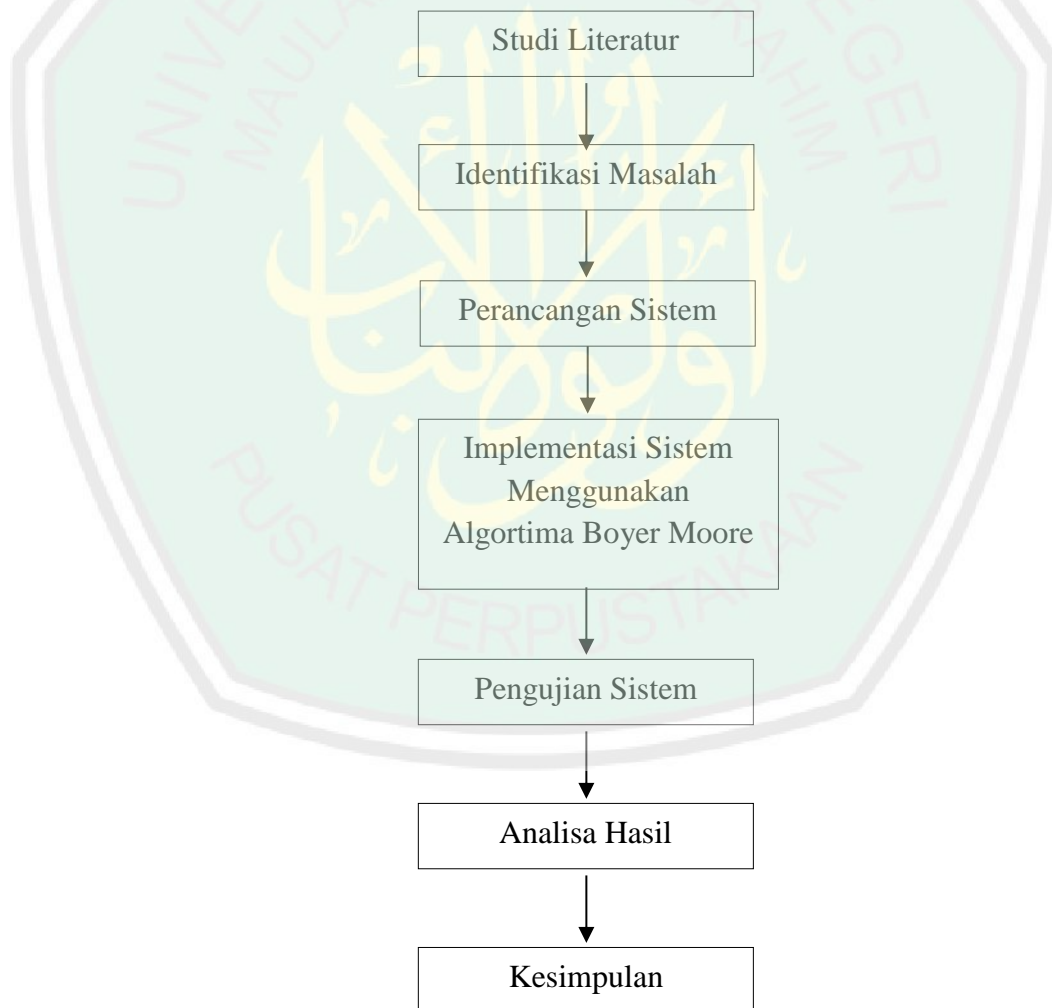
Alwin Fau, M. G. (2017). Melakukan analisa perbandingan *boyer moore* dan *knuth morris pratt* dalam pencarian judul buku menerapkan metode perbandingan eksponensial pada kasus pencarian judul buku pada perpustakaan Perpustakaan STMIK Budidarma Medan dimana proses pencarian yang dilakukan masih membutuhkan waktu yang sangat lama. penelitian ini memberikan hasil dari Nilai dari Metode Perbandingan Eksponensial (MPE) yang dilakukan pada penelitian terbukti bahwa Algoritma *Boyer Moore* merupakan Algoritma yang tercepat dalam melakukan pencarian.



## BAB III METODOLOGI PENELITIAN

### 3.1 Prosedur Penelitian

Prosedur penelitian merupakan rangkaian tahapan penelitian yang tersusun secara sistematis. Tujuan dari metodologi penelitian adalah agar pelaksanaan penelitian mendapatkan hasil yang sesuai dengan tujuan penelitian. Adapun rangkaian tahapan yang akan dilakukan dalam penelitian ini adalah sebagai berikut:



**Gambar 3.1** Prosedur Penelitian

Berdasarkan Gambar 3.1 terdapat tujuh proses yang harus dilakukan untuk menyelesaikan penelitian ini. Penelitian dimulai dengan studi literatur yang dilakukan untuk menemukan dan mengumpulkan data atau informasi kasus dari referensi-referensi terkait. Referensi–referensi ini dapat berupa buku-buku tentang metode pencarian, jurnal-jurnal atau tulisan penelitian tentang pencarian data menggunakan algoritma *boyer moore*, atau artikel-artikel yang membahas kasus yang sama dengan kasus dalam penelitian ini. Tahap selanjutnya yaitu mengidentifikasi masalah dengan menentukan pertanyaan-pertanyaan penelitian dalam penelitian. Setelah menentukan pertanyaan-pertanyaan penelitian, dilakukan perancangan sistem untuk menentukan alur proses pada sistem yang selanjutnya diimplementasikan pada sistem. Tahap berikutnya adalah melakukan pengujian sistem sehingga dapat dilakukan analisa dari hasil pengujian tersebut.

### **3.2 Perancangan Sistem**

Perancangan sistem pada sistem informasi manajemen laboratorium meliputi analisa sistem dengan menggunakan pemodelan UML, perancangan database, dan penjelasan penggunaan algoritma *boyer moore* pada sistem.

#### **3.2.1. Pemodelan UML**

Unified Modeling Language (UML) adalah suatu himpunan secara struktur dan teknik untuk pemodelan desain program berorientasi objek (OOP) serta aplikasinya. Pada Pemodelan UML yang digunakan meliputi beberapa diagram pemodelan seperti *usecase diagram*, *activity diagram*, *sequence diagram* dan *class diagram*.

Pemodelan UML ini akan memudahkan dalam membaca alur dari sistem yang akan dibuat, dan aktifitas- aktifitas apa saja yang ada didalam sistem informasi manajemen laboratorium.

### 3.2.1.1 Use Case Diagram



Gambar 3.2 Use Case Diagram

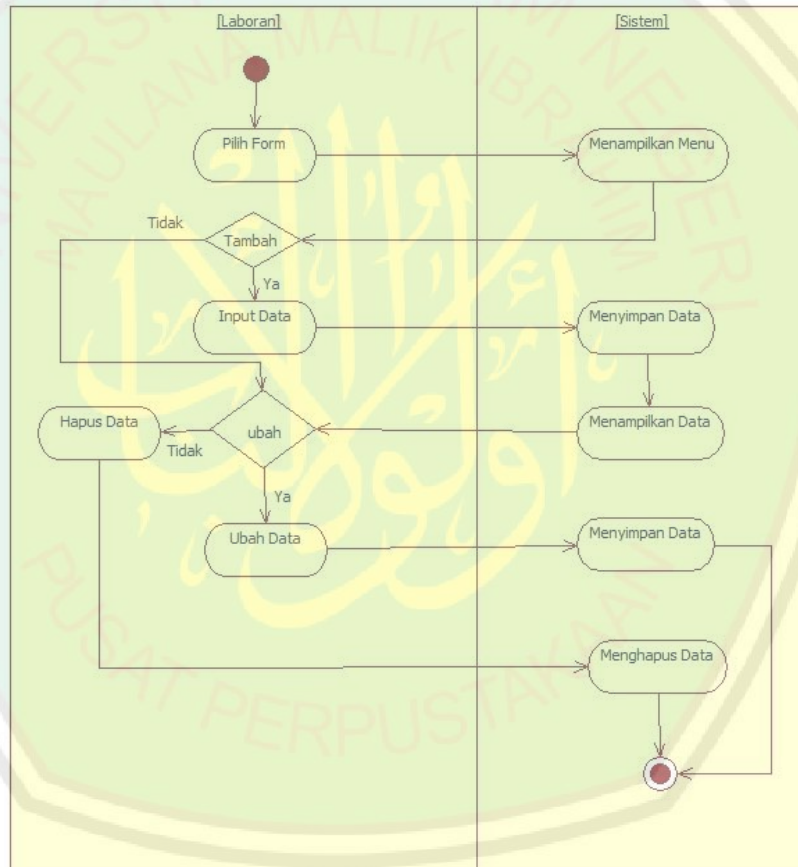
Usecase diagram menunjukkan aktifitas yang dilakukan oleh pengguna terhadap sistem. Pada usecase diagram diambil user yaitu laboran dan mahasiswa yang banyak melakukan pencarian data atau berkaitan dengan pencarian data. Pada Gambar 3.2 laboran dapat melakukan pencarian data pada alat dan bahan, dimana alat dan bahan sebelumnya telah diinput oleh laboran untuk mengecek data alat dan bahan. Mahasiswa melakukan peminjaman alat dan bahan praktikum yang akan masuk kedalam sistem, dan Laboran dapat mengecek data mahasiswa yang melakukan peminjaman untuk konfirmasi peminjaman dan laporan peminjaman penggunaan alat dan bahan praktikum.

### 3.2.1.2 Activity Diagram

*Activity diagram* menggambarkan logika prosedural, proses bisnis, jalur kerja, dan urutan aktivitas dalam suatu proses.

#### a. *Activity diagram* kelola data

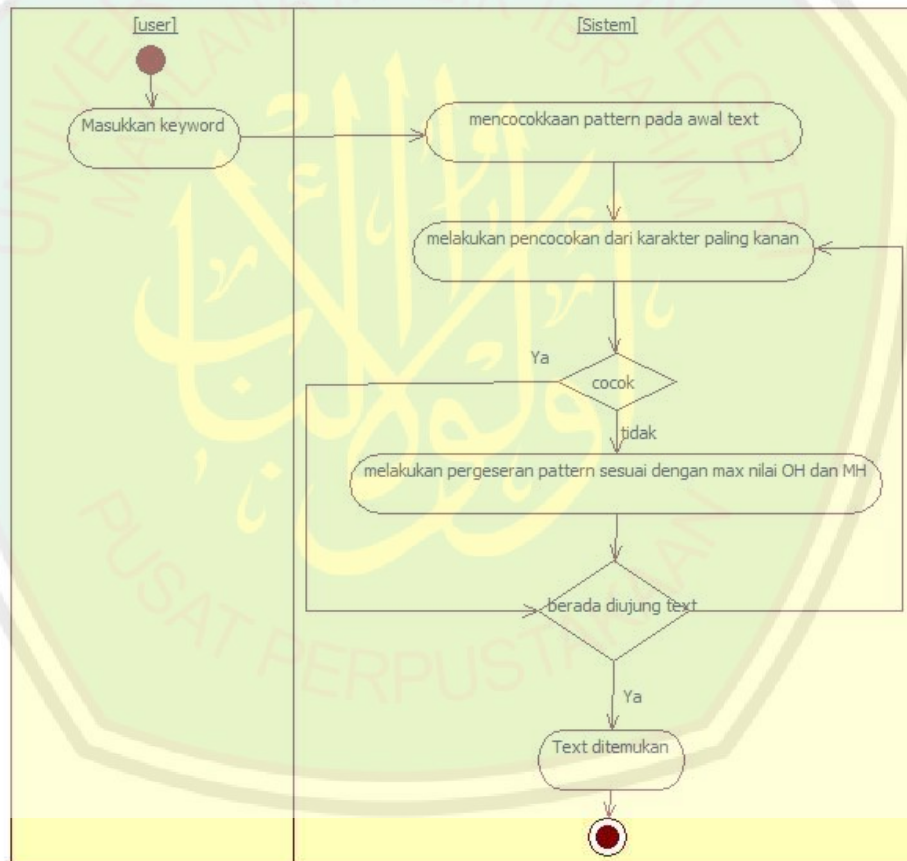
Pada *activity diagram* pada gambar 3.3 Laboran dapat menambahkan data seperti data alat dan bahan, mengubah data dan menghapus data.



**Gambar 3.3** Activity Diagram Kelola Data

b. *Activity diagram* pencarian data

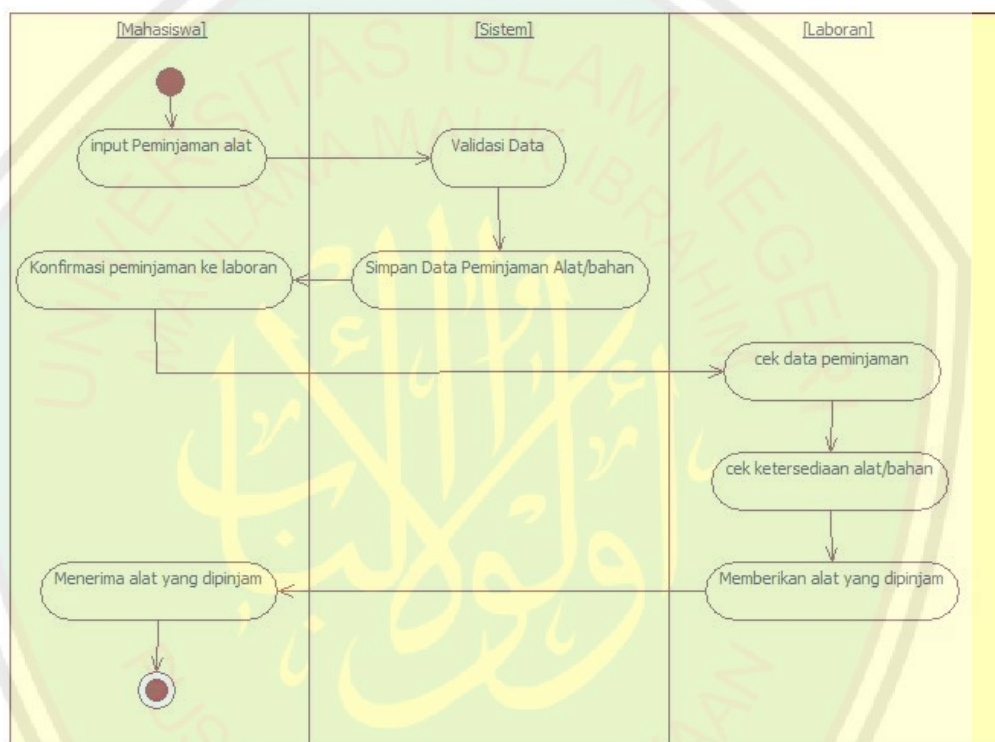
Pada gambar 3.4 pencarian data pada sistem informasi manajemen laboratorium dapat dilakukan oleh *user* yang terkait didalamnya, *user* menginputkan kata kunci dari data yang akan dicari, maka sistem akan melakukan pencarian data dengan menggunakan algoritma *boyer moore* yang mana algoritma tersebut akan melakukan pencocokan *pattern* dari karakter paling kanan ke kiri. Jika terdapat karakter yang tidak cocok maka *pattern* akan digeser berdasarkan nilai maksimal dari OH dan MH.



**Gambar 3.4** *Activity Diagram* Pencarian Data

c. *Activity diagram* peminjaman alat/bahan

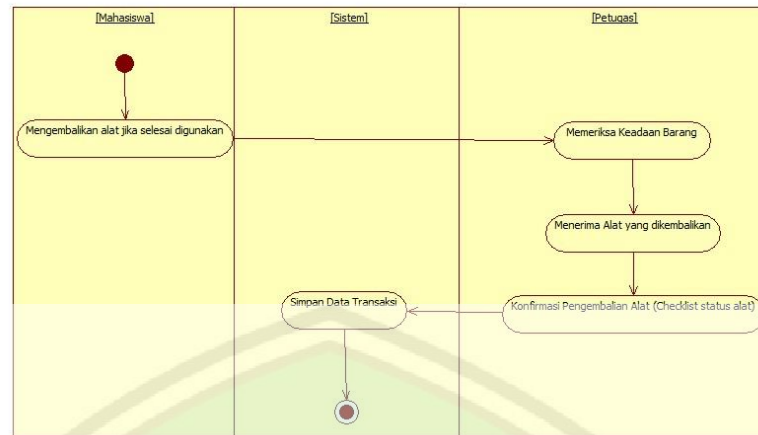
Mahasiswa memilih menu peminjaman kemudian menginputkan data pada *form* peminjaman, kemudian sistem menyimpan data peminjaman. Setelah itu mahasiswa melakukan konfirmasi peminjaman kepada laboran, dan laboran mengkonfirmasi peminjaman alat atau bahan praktikum. *activity diagram* peminjaman alat/bahan dapat dilihat pada Gambar 3.5



Gambar 3.5 *Activity Diagram* Peminjaman Alat/Bahan

d. *Activity diagram* pengembalian alat

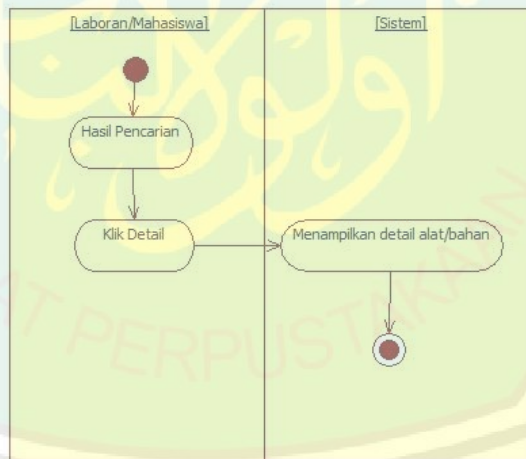
Setelah selesai melakukan peminjaman alat praktikum, mahasiswa melakukan pengembalian alat kepada laboran, laboran memeriksa kondisi dari alat yang telah dipakai dan menerima alat yang telah dipinjam. Kemudian mengkonfirmasi pengembalian alat praktikum.



**Gambar 3.6 Activity Diagram Pengembalian Alat**

e. *Activity diagram detail alat/bahan praktikum*

Mahasiswa ataupun laboran dan user lainnya dapat melihat detail alat dan bahan yang ada untuk mengetahui lebih lanjut tentang alat atau bahan yang dicari. Sistem akan menampilkan detail dari alat dan bahan yang dimaksud kepada *user*.



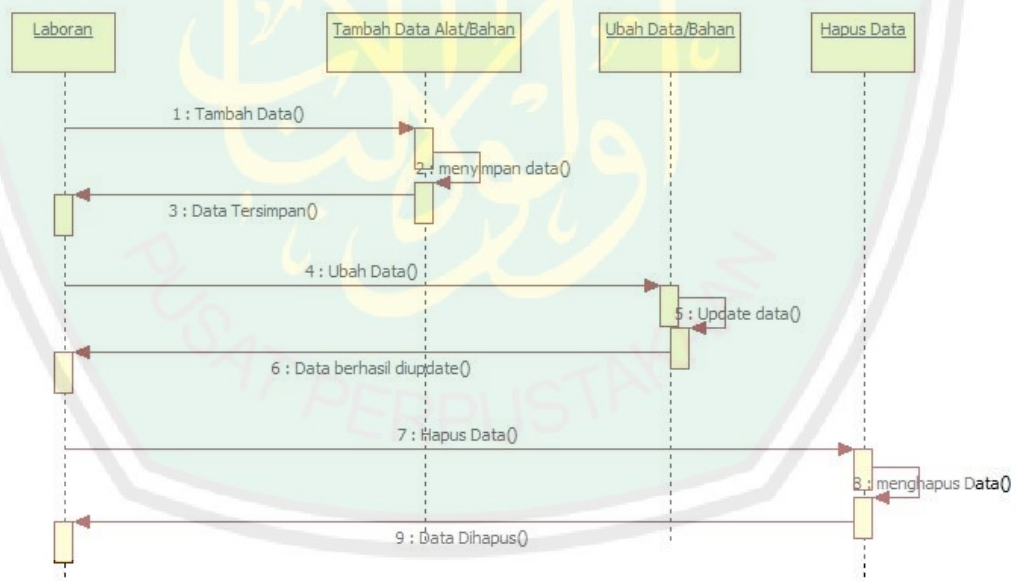
**Gambar 3.7 Activity Diagram Detail Alat/Bahan Praktikum**

### 3.2.1.3 Sequence Diagram

Didalam sequence diagram, akan digambarkan mengenai keterkaitan antar komponen. Dimana masing-masing komponen yang berelasi akan menghasilkan informasi sesuai aktifitas didalam sistem.

#### a. Sequence diagram kelola data

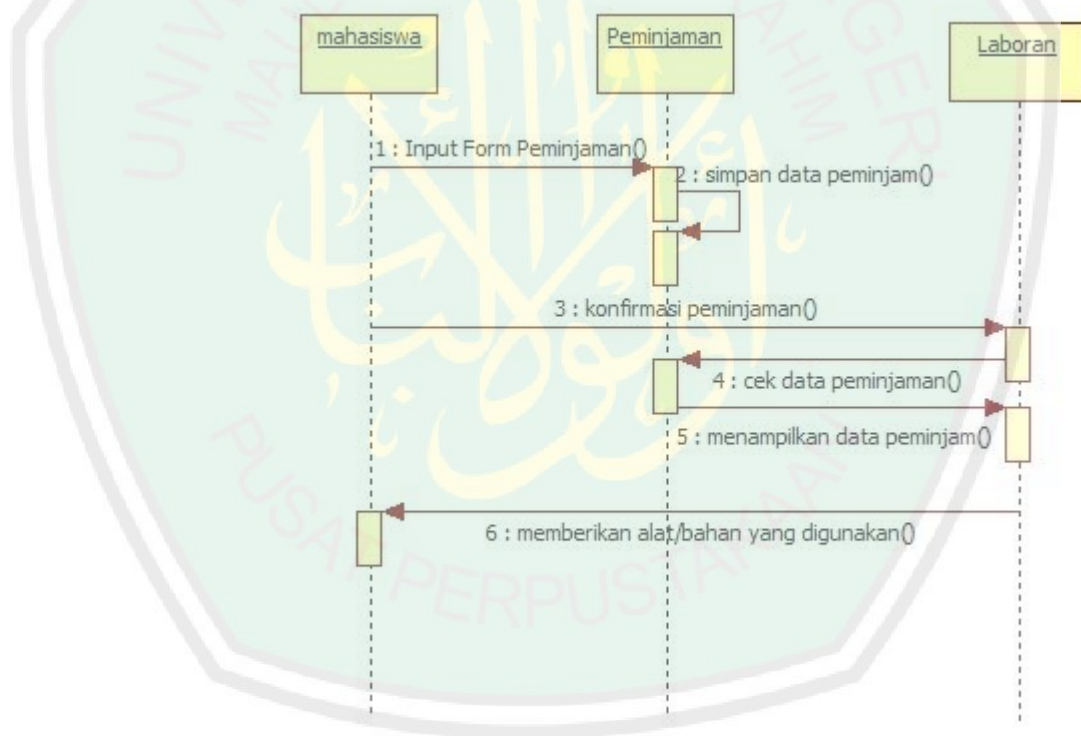
Saat laboran menginputkan data-data baru seperti alat dan bahan praktikum yang baru datang, maka sistem akan menyimpan data ke dalam *database*, data yang sudah diinputkan juga dapat di *update* jika terdapat kesalahan atau pembaharuan data. Data yang tidak perlu dapat dihapus dalam sistem. *Sequence diagram* kelola data dapat dilihat pada Gambar 3.8



Gambar 3.8 Sequence Diagram Kelola Data

b. *Sequence diagram* peminjaman alat/bahan

Mahasiswa yang akan meminjam alat atau menggunakan bahan praktikum dapat menginputkan data peminjaman pada *form* peminjaman sistem informasi manajemen laboratorium, setelah menginputkan data maka sistem akan menyimpan data peminjaman. Kemudian mahasiswa melakukan konfirmasi peminjaman ke laboran, dan laboran melakukan validasi data peminjam pada sistem, setelah divalidasi maka laboran memberikan alat atau bahan praktikum yang akan digunakan oleh mahasiswa. *Sequence diagram* peminjaman alat/bahan dapat dilihat pada Gambar 3.9

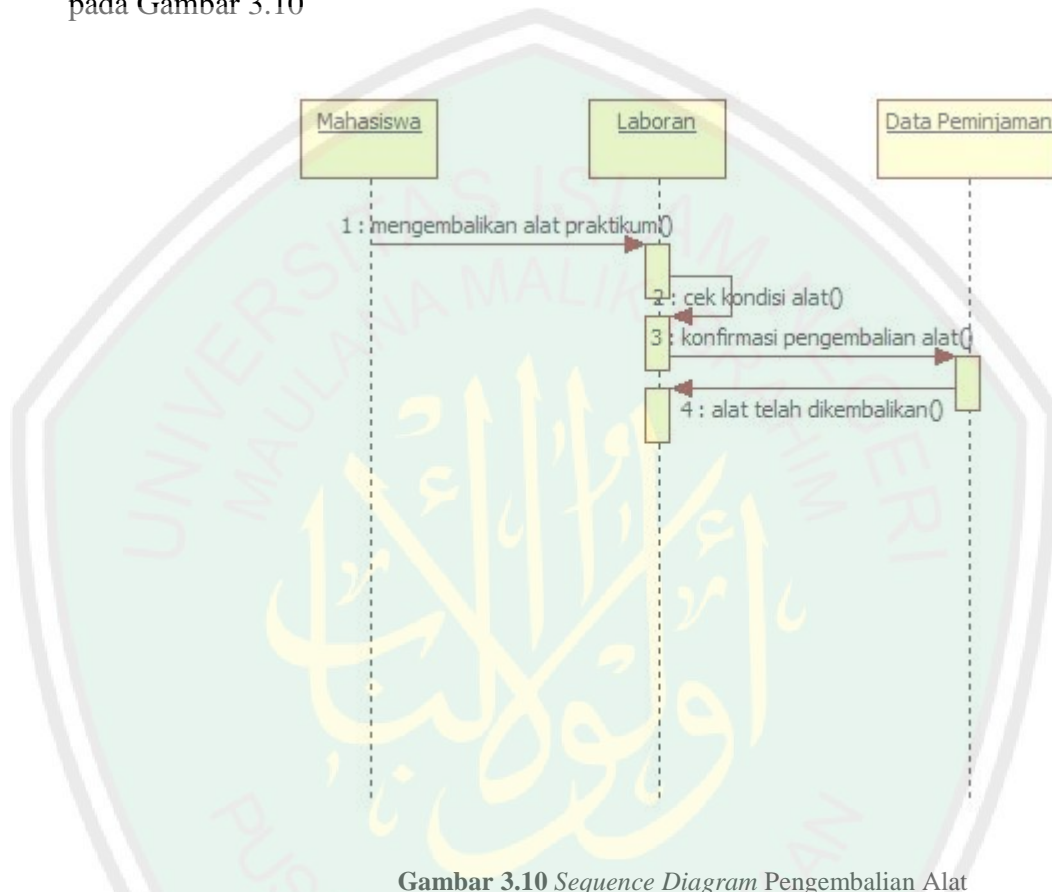


**Gambar 3.9** *Sequence Diagram* Peminjaman Alat/Bahan

c. *Sequence diagram* pengembalian alat

Mahasiswa yang telah selesai menggunakan alat diharuskan untuk mengembalikan alat ke ruang alat tersebut agar bisa dipakai oleh mahasiswa lain.

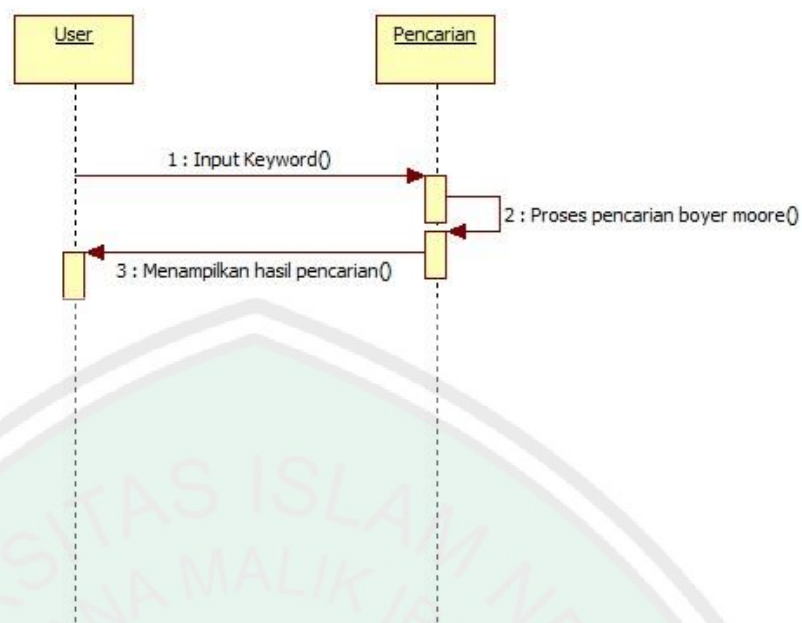
Proses pengembalian alat praktikum di kembalikan ke laboran, kemudian laboran mengecek kondisi alat praktikum, dan mengkonfirmasi pengembalian alat ke sistem, sehingga sistem mencatat bahwa alat tersebut sudah dikembalikan oleh mahasiswa yang bersangkutan. *Sequence diagram* pengembalian alat dapat dilihat pada Gambar 3.10



Gambar 3.10 Sequence Diagram Pengembalian Alat

d. *Sequence diagram* pencarian data

*User* yang berhubungan dengan sistem dapat melakukan pencarian data, seperti data alat dan bahan praktikum atau data peminjaman agar proses pencarian lebih efektif. Sistem pencarian menggunakan algoritma *boyer moore*. *Sequence diagram* pencarian data dapat dilihat pada Gambar 3.11.

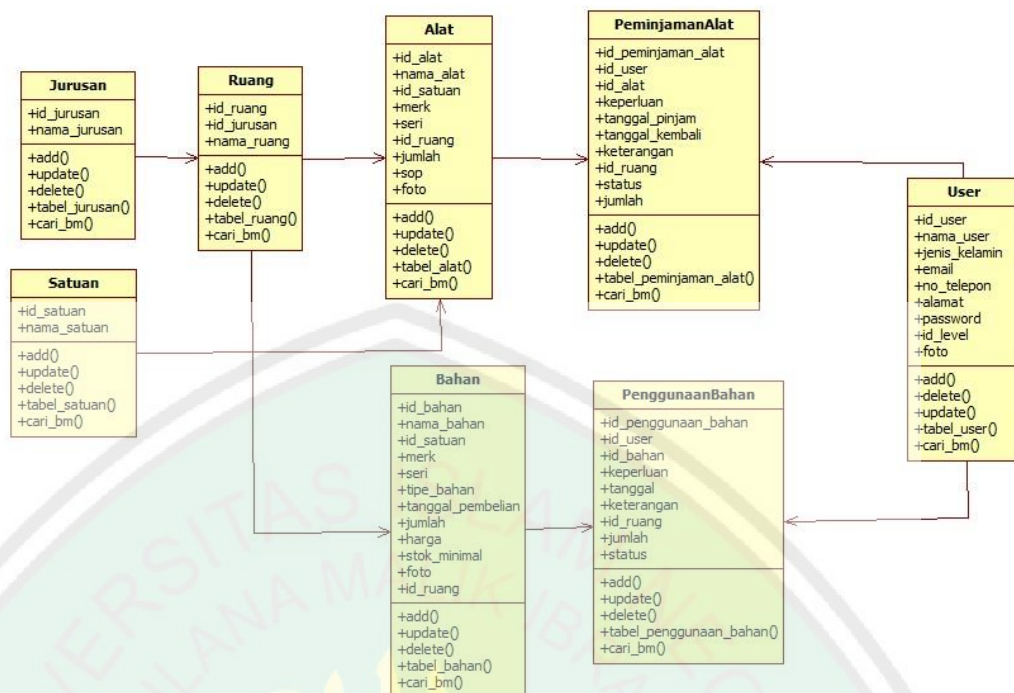


Gambar 3.11 Sequence Diagram Pencarian Data

#### 3.2.1.4 Class Diagram

*Class diagram* adalah diagram yang menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. kelas memiliki 3 bagian utama yaitu *attribute*, *operation*, dan *name*. kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem.

Pada Gambar 3.12 merupakan dari *class diagram* dari sistem yang berhubungan dengan pencarian data, pada baris pertama merupakan nama dari *class* yang akan digunakan, baris kedua adalah *attribute* yang digunakan pada *class* tersebut, dan baris ketiga adalah operator atau *function* yang terdapat dalam *class* yang bersangkutan. Tiap *class* satu dengan *class* yang lain saling berhubungan seperti yang ditampilkan pada Gambar 3.12.



Gambar 3.12 Class Diagram Sistem

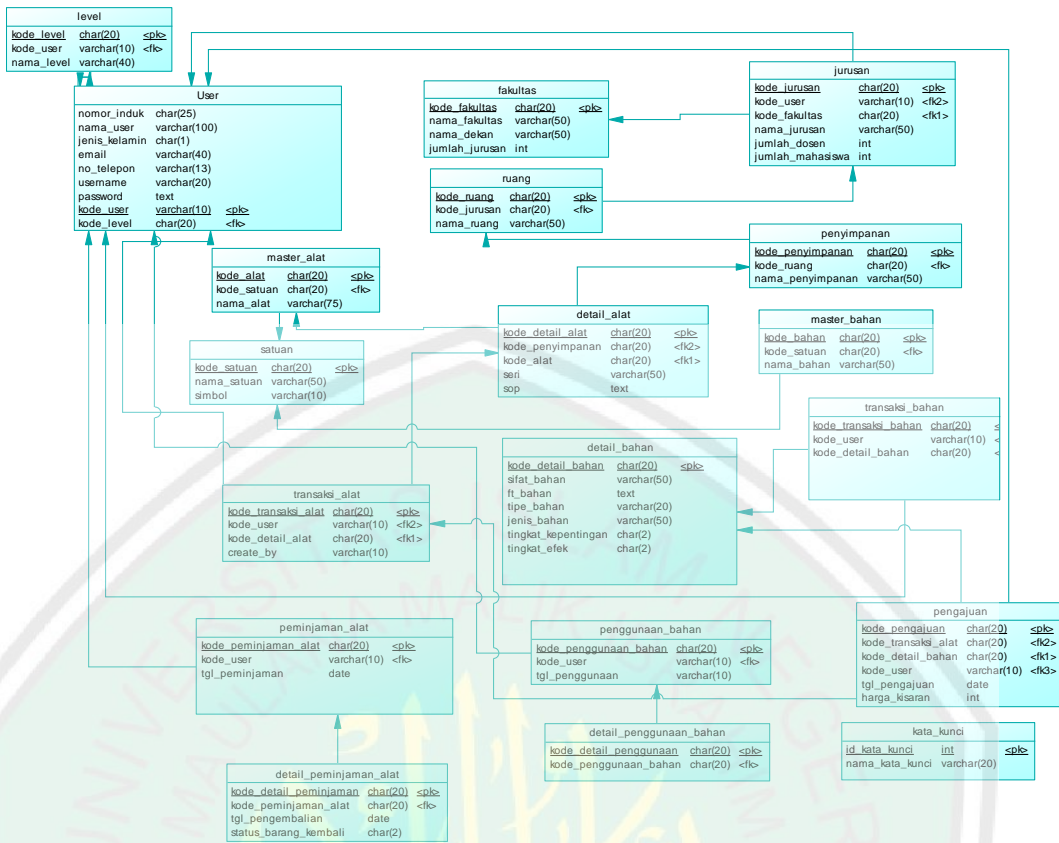
### 3.2.2. Perancangan Database

Perancangan *database* digunakan untuk merancang masukan data yang bertujuan untuk membentuk sistem basis data yang saling terhubung dan berelasi antara tabel yang satu dengan yang lain. Selain itu desain *database* digunakan juga untuk menentukan panjang karakter masing-masing *fields* dan tipe datanya. Berikut rancangan dari database yang akan digunakan.

#### 3.2.2.1 Conceptual Data Model

CDM adalah model yang dibuat berdasarkan anggapan bahwa dunia nyata terdiri dari koleksi obyek-obyek dasar yang dinamakan entitas (*entity*) serta hubungan (*relationship*) antara entitas-entitas itu. Pada CDM yang dibuat terdiri dari 19 entitas. CDM dapat dilihat pada Gambar 3.13

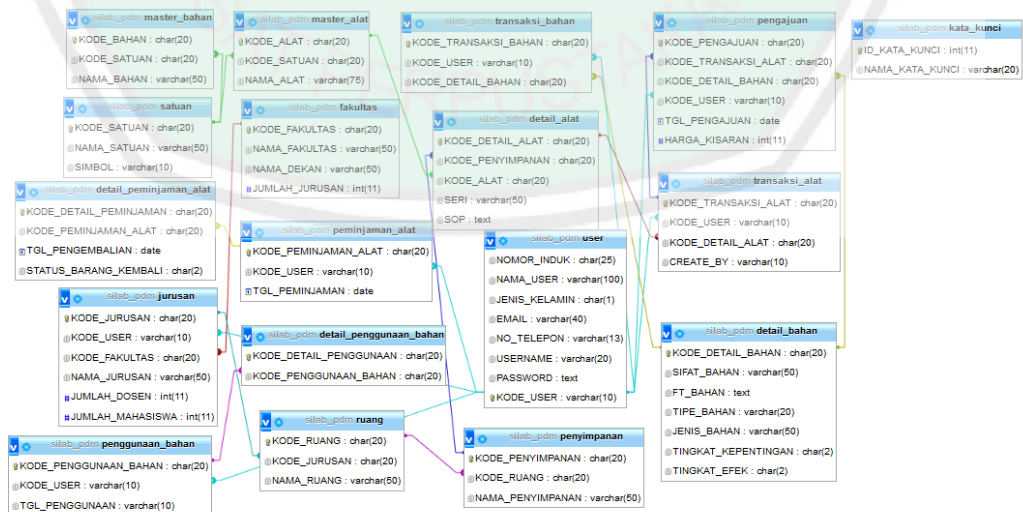




Gambar 3.14 Physical Data Model

### 3.2.2.3 Database

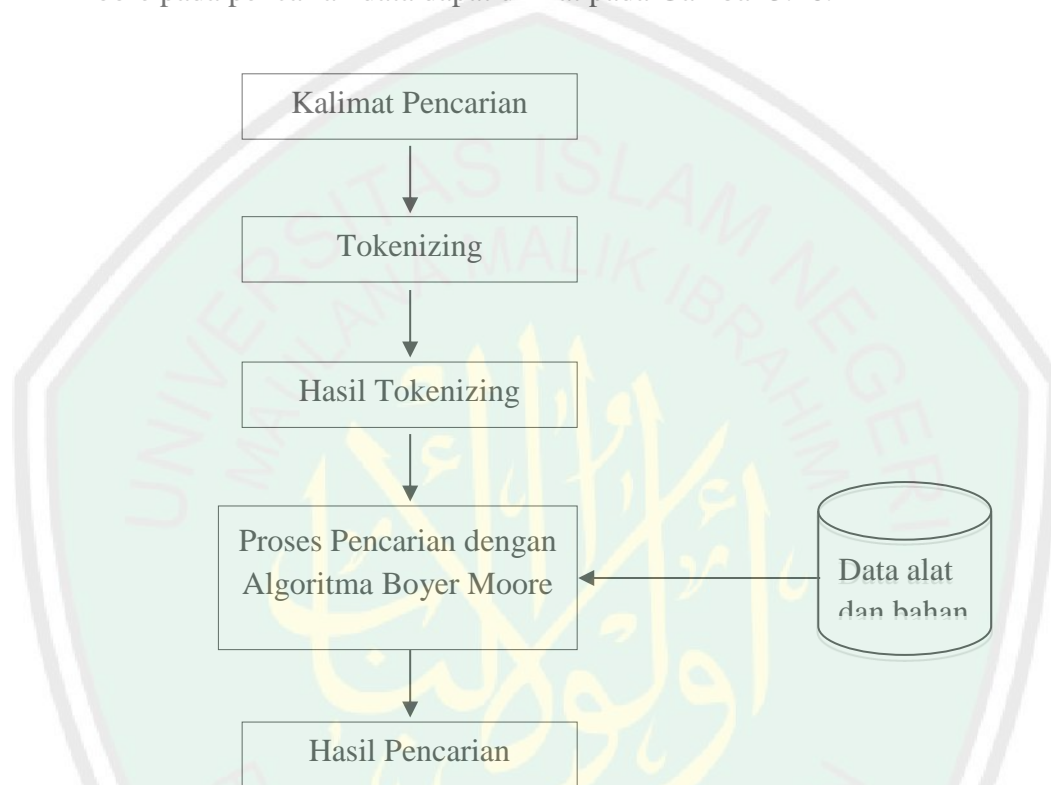
Hasil dari Physical Data Model akan di generate menjadi database yang menjadi tempat penyimpanan data.



Gambar 3.15 Database

### 3.2.3. Perancangan Sistem Dengan Algoritma *Boyer Moore*

Sebelum dilakukan implementasi algoritma di dalam sistem, maka dilakukan pemodelan atau desain sistem dalam implementasi algoritma yang digunakan. Secara umum desain sistem untuk implementasi algoritma boyer moore pada pencarian data dapat dilihat pada Gambar 3.16.



Gambar 3.16 Desain Sistem

Proses pencarian pada sistem menggunakan algoritma *boyer moore*, yang mana tahapan sebelum dilakukan proses pencarian adalah pemecahan kalimat menjadi kata (*tokenizing*), sehingga akan terbentuk beberapa kata dari kalimat tersebut. Hasil dari *tokenizing* akan di cocokkan dengan data kata kunci menggunakan algoritma boyer moore, sehingga dari hasil pencocokan dapat diseleksi lagi dan hasil pencarian akan ditampilkan berdasarkan query sql yang sesuai.

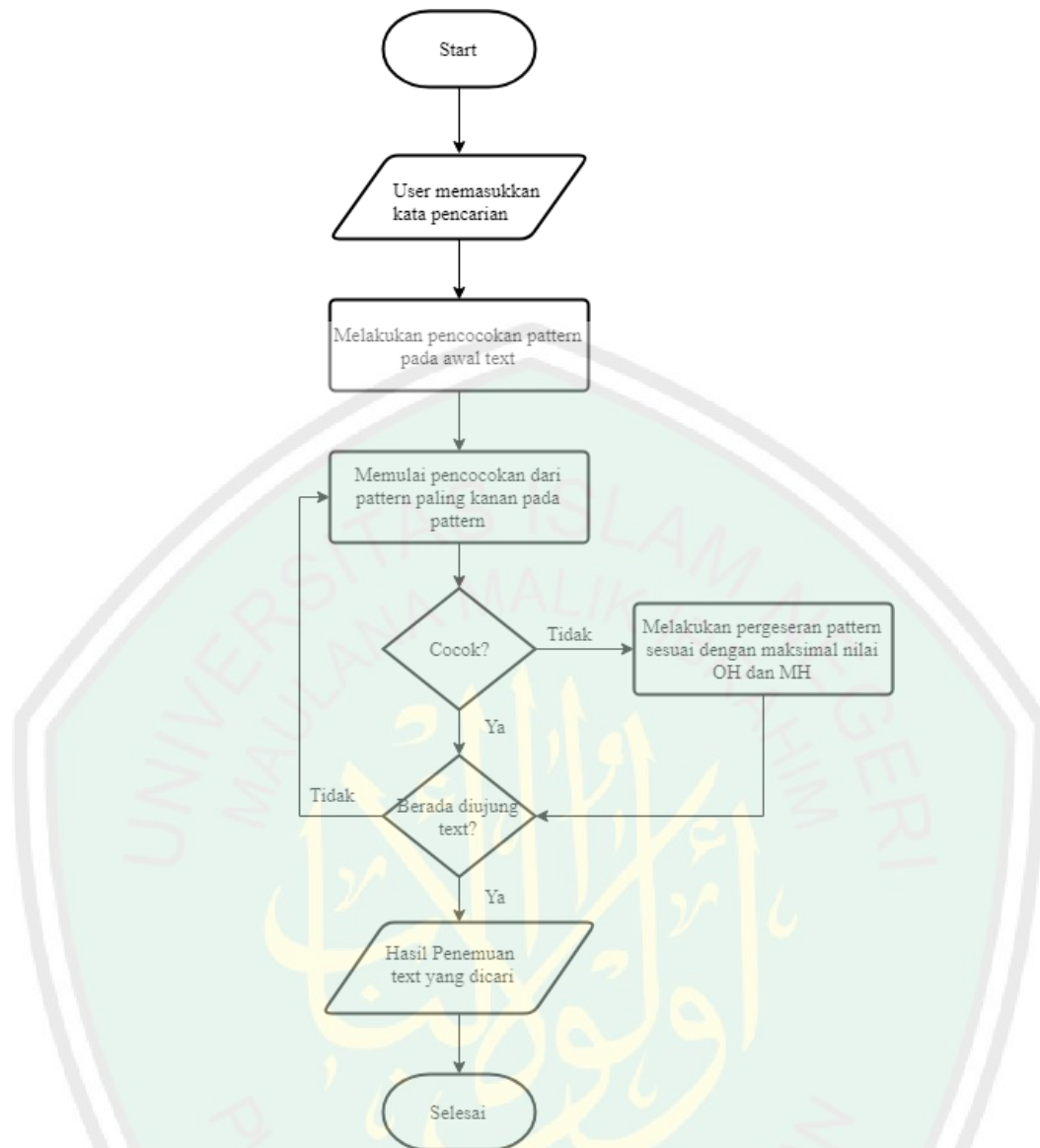
### 3.2.3.1 Contoh Implementasi Algoritma Boyer Moore

Data pencarian yang akan digunakan berupa tabel. Pada contoh diberikan contoh tabel data alat.

**Tabel 3.1 Contoh Data Alat**

No	Nama Alat	Seri	Merk	Satuan	Stok
1	Abcdmikroskop	A	Merk A	Buah	10
2	Gelas Ukur	B	Merk B	Buah	100
3	Pipet	C	Merk C	Buah	120

Pada Tabel 3.1 merupakan contoh data dari alat yang tersedia, kemudian user akan memasukkan kalimat pencarian pada *field* pencarian. Kalimat pada pencarian akan di pecah menjadi beberapa kata. Kata tersebut kemudian akan dicocokkan dengan data yang ada pada database menggunakan algoritma *boyer moore*. Berikut proses dari pencocokan kata dengan algoritma *boyer moore*.



Gambar 3.17 Flowchart Pencarian dengan *Boyer Moore*

Secara Umum, langkah-langkah dari teknik penerapan *Boyer-Moore* pada pencocokan *pattern* pada teks terdiri dari 3 tahap seperti tertera pada studi pustaka.

a. Prosedur preBmB

Prosedur preBmBc memiliki tiga nilai penting, diantaranya :

1. *Pattern*, sebagai subjek pencocokan terhadap teks.
2. Karakter, sebagai karakter-karakter yang terdapat pada *pattern*.
3. *Occurrence Heuristic* (OH), sebagai nilai pergeseran yang diperoleh ketika menemukan ketidakcocokan karakter.
4. Pergeseran, sebagai nilai yang dicapai ketika melakukan pergeseran dari kanan ke kiri *pattern*.

Langkah-langkah pelaksanaan prosedur :

1. *Create* atau berikan nilai kosong ke stack BmBc.
2. Lakukan perhitungan terhadap panjang *pattern*. Jika panjang tidak lebih dari satu, maka hentikan proses dengan menambahkan langsung nilai OH dan karakter ke dalam stack BmBc. Jika tidak, maka lanjutkan ke proses selanjutnya.
3. Cacahan karakter *pattern* mulai dari karakter ke-2 paling kanan.
4. Bandingkan setiap karakter yang dicacah terhadap stack BmBc, jika karakter yang dicacah tidak ditemukan di dalam stack, maka tambahkan karakter tersebut ke dalam stack dimana OH sama dengan jumlah pergeseran karakter yang telah dilakukan.
5. Lakukan langkah-4 kembali, dengan melakukan perpindahan 1 karakter ke kiri hingga mencapai karakter paling kiri secara terus menerus.
6. Jika telah mencapai karakter paling kiri, cacah karakter paling kanan lalu kembali ke langkah-4.

**Contoh kasus:**

*Pattern* : MIKROSKOP

Penyelesaian :

<b>Pattern</b>	M	I	K	R	O	S	K	O	P
<b>Pergeseran</b>								1	

**StackBmBc**  
Karakter  
Nilai OH

<b>Stack BmBc</b>	
Karakter	O
Nilai OH	1

<b>Pattern</b>	M	I	K	R	O	S	K	O	P
<b>Pergeseran</b>							2		

**Stack BmBc**  
Karakter  
Nilai OH

<b>Stack BmBc</b>		
Karakter	O	K
Nilai OH	1	2

<b>Pattern</b>	M	I	K	R	O	S	K	O	P
<b>Pergeseran</b>						3			

**Stack BmBc**  
Karakter  
Nilai OH

<b>Stack BmBc</b>			
Karakter	O	K	S
Nilai OH	1	2	3

<b>Pattern</b>	M	I	K	R	O	S	K	O	P
<b>Pergeseran</b>					4				

**Stack BmBc**  
Karakter  
Nilai OH

<b>Stack BmBc</b>				
Karakter	O	K	S	R
Nilai OH	1	2	3	5

<b>Pattern</b>	M	I	K	R	O	S	K	O	P
<b>Pergeseran</b>					5				

**Stack BmBc**  
Karakter  
Nilai OH

<b>Stack BmBc</b>					
Karakter	O	K	S	R	I
Nilai OH	1	2	3	5	7

<b>Pattern</b>	M	I	K	R	O	S	K	O	P
<b>Pergeseran</b>									6

**Stack BmBc**  
Karakter  
Nilai OH

<b>Stack BmBc</b>						
Karakter	O	K	S	R	I	M
Nilai OH	1	2	3	5	7	8

<b>Pattern</b>	M	I	K	R	O	S	K	O	P
<b>Pergeseran</b>									7

**Stack BmBc**  
Karakter  
Nilai OH

<b>Stack BmBc</b>							
Karakter	O	K	S	R	I	M	P
Nilai OH	1	2	3	5	7	8	0

<b>Pattern</b>	M	I	K	R	O	S	K	O	P
<b>Pergeseran</b>									8

**Stack BmBc**  
Karakter  
Nilai OH

<b>Stack BmBc</b>								
Karakter	O	K	S	R	I	M	P	
Nilai OH	1	2	3	5	7	8	0	

<b>Pattern</b>	M	I	K	R	O	S	K	O	P
<b>Nilai OH</b>	8	7	2	5	1	3	2	1	0

**Gambar 3.18** Penyelesaian contoh kasus Prosedur preBmBc

Pada gambar di atas, terlihat bahwa karakter yang ditunjuk akan ditambahkan ke dalam stack BmBc jika tidak ditemukan pada stack BmBc dengan nilai OH yang berkesesuaian dengan nilai pergeseran.

b. Prosedur preBmGs

Prosedur preBmGs memiliki enam nilai penting, diantaranya:

1. *Pattern*, sebagai subjek pencocokan terhadap teks.
2. *Match Heuristic* (MH), sebagai nilai pergeseran yang diperoleh ketika menemukan kecocokan suffix.
3. *Compare*, sebagai akhiran atau sejumlah karakter sebelah kanan dari sebuah karakter *pattern* yang diperoleh dari pergeseran kanan ke kiri.
4. *Prefix*, sebagai awalan atau karakter *pattern* yang diperoleh dari pergeseran dari kiri ke kanan.
5. *Suffix*, sebagai akhiran sebelah kanan *prefix*.
6. Pergeseran, sebagai nilai yang dicapai ketika melakukan pergeseran dari *compare*.

Langkah-langkah pelaksanaan prosedur :

1. *Create* atau berikan nilai kosong ke stack iBmGs dan Compare.
2. Lakukan aksi yang sama dengan preBmBc jika panjang *pattern* tidak lebih dari satu.
3. Cacah *compare* terhadap *pattern*. Lalu simpan hasil pencacah dengan menambahkan ke dalam stack compare.

4. Cacah *prefix* dan *suffix* terhadap *pattern*. Lalu simpan hasil pencacahan *prefix* ke dalam stack BmGs, dan bandingkan *suffix* terhadap compare dengan melihat kecocokan antara masing-masing *prefix*.
5. Jika panjang kedua *suffix* tidak sama besar, cacah salah satu *suffix* yang memiliki panjang terbesar ke arah kanan sepanjang *suffix* dengan panjang terkecil. Jika ditemukan kecocokan antar *suffix*, bandingkan *prefix* kedua *suffix*. Jika *prefix* yang diperbandingkan adalah sama, maka kecocokan tidak dapat diterima.
6. Untuk *compare*, *prefix* didapat ketika terjadi pencacahan pada *compare*, *prefix* akan bernilai sama dengan karakter paling kanan dari hasil pencacahan terhadap *compare* sebelum terjadinya pencacahan. Jika tidak terjadi pemotongan, maka *prefix-compare* bernilai null.
7. *Indeks prefix* paling kanan, nilai MH harus selalu bernilai 1. Sedangkan nilai MH pada *prefix* lainnya, diberikan ketika ditemukan ketidakcocokan antara *prefix-suffix* terhadap *prefix-compare* dan ditemukannya kecocokan di antara kedua *suffix* sebesar nilai pergeseran terbentuknya *compare*
8. Ulangi langkah 5 hingga *compare* yang terakhir.
9. Ulangi langkah 5 hingga *suffix* yang terakhir.

#### Contoh Kasus

*Pattern* : MIKROSKOP

Penyelesaian :

**Tabel 3.2** Daftar pencacahan *prefix* dan *suffix* dari *pattern*

Suffix (kanan-kiri)	Prefix	Suffix(kiri-kanan)
Null	M	IKROSKOP
M	I	KROSKOP
MI	K	ROSKOP
MIK	R	OSKOP
MIKR	O	SKOP
MIKRO	S	KOP
MIKROS	K	OP
MIKROSK	O	P
MIKROSKO	P	Null

Dari tabel di atas, diasumsikan bahwa telah terjadi pemotongan pada kedua sisi *suffix* terhadap *prefix* pada *pattern*. Terlihat bahwa masing-masing *suffix* dipersatukan oleh *prefix* hingga menjadi satu kesatuan *pattern* utuh. Sedangkan *null* diartikan kosong, yakni semua kondisi pencocokan akan bernilai benar jika dilakukannya perbandingan terhadap karakter tersebut.

## Stack Compare

Compare	MIKROSKO	MIKROSK	MIKROS	MIKRO	MIKR	MIK	MI	M	Null
Pergeseran	1	2	3	4	5	6	7	8	9

Pattern	M	I	K	R	O	S	K	O	P

## Stack BmGs

Karakter	M	I	K	R	O	S	K	O	P
Nilai MH	-	-	-	-	-	-	-	-	1

MH selalu bernilai 1

Pattern	M	I	K	R	O	S	K	O	P
---------	---	---	---	---	---	---	---	---	---

Prefix	O
Suffix	P

Pattern	M	I	K	R	O	S	K	O	P
---------	---	---	---	---	---	---	---	---	---

Prefix	K
Suffix	OP

Pattern	M	I	K	R	O	S	K	O	P
---------	---	---	---	---	---	---	---	---	---

Prefix	S
Suffix	KOP

Pattern	M	I	K	R	O	S	K	O	P
---------	---	---	---	---	---	---	---	---	---

Prefix	O
Suffix	SKOP

Pattern	M	I	K	R	O	S	K	O	P
---------	---	---	---	---	---	---	---	---	---

Prefix	R
Suffix	OSKOP

Pattern	M	I	K	R	O	S	K	O	P
---------	---	---	---	---	---	---	---	---	---

Prefix	K
Suffix	ROSKOP

Pattern	M	I	K	R	O	S	K	O	P
---------	---	---	---	---	---	---	---	---	---

Prefix	I
Suffix	KROSKOP

**Stack BmGs**

Karakter	M	I	K	R	O	S	K	O	P
Nilai MH	-	-	-	-	-	-	-	9	1

Suffix compare Null  
 Pergeseran Suffix Comparator ke 9

**Stack BmGs**

Karakter	M	I	K	R	O	S	K	O	P
Nilai MH	-	-	-	-	-	-	9	9	1

Suffix compare Null  
 Pergeseran Suffix Comparator ke 9

**Stack BmGs**

Karakter	M	I	K	R	O	S	K	O	P
Nilai MH	-	-	-	-	-	9	9	9	1

Suffix compare Null  
 Pergeseran Suffix Comparator ke 9

**Stack BmGs**

Karakter	M	I	K	R	O	S	K	O	P
Nilai MH	-	-	-	-	9	9	9	9	1

Suffix compare Null  
 Pergeseran Suffix Comparator ke 9

**Stack BmGs**

Karakter	M	I	K	R	O	S	K	O	P
Nilai MH	-	-	-	9	9	9	9	9	1

Suffix compare Null  
 Pergeseran Suffix Comparator ke 9

**Stack BmGs**

Karakter	M	I	K	R	O	S	K	O	P
Nilai MH	-	-	9	9	9	9	9	9	1

Suffix compare Null  
 Pergeseran Suffix Comparator ke 9

**Stack BmGs**

Karakter	M	I	K	R	O	S	K	O	P
Nilai MH	-	9	9	9	9	9	9	9	1

Suffix compare Null  
 Pergeseran Suffix Comparator ke 9

Pattern	M	I	K	R	O	S	K	O	P
---------	---	---	---	---	---	---	---	---	---

Prefix	M
Suffix	IKROSKOP

<b>Stack BmGs</b>									
Karakter	M	I	K	R	O	S	K	O	P
Nilai MH	9	9	9	9	9	9	9	9	1

Suffix compare	Null
Pergeseran Suffix Comparator	ke 9

Gambar 3.19 Penyelesaian contoh kasus Prosedur preBmGs

Pada gambar di atas terlihat bahwa pergeseran pada compare memiliki peran utama dalam penentuan nilai MH. Berbeda dengan nilai OH, nilai MH selalu diberikan pada karakter-karakter yang ada pada *pattern*. Penyusunan stack BmGs pun berdasarkan urutan indeks posisi karakter pada *pattern*.

c. Prosedur BM

Prosedur BM memiliki empat nilai penting, diantaranya :

1. *Pattern*, merupakan subjek pencocokan terhadap teks.
2. *Teks*, sebagai objek pencocokan.
3. *Stack BmBc*, sebagai pembanding ketika ditemukan ketidakcocokan.
4. *Stack BmGs*, sebagai pembanding ketika ditemukan kecocokan.

Langkah-langkah pelaksanaan prosedur :

1. Pencocokan dimulai dari indeks teks terkecil atau dari karakter paling kiri.
2. Pencocokan per karakter dimulai dari karakter paling kanan *pattern*.
3. Setiap kali ditemukan ketidakcocokan karakter, maka ambil nilai OH pada stack BmBc dengan karakter yang berkesesuaian terhadap teks, lalu nilai OH dikurangi dengan jumlah kecocokan yang telah terjadi. Ambil nilai MH pada stack BmGs dengan indeks karakter yang ditemukan ketidakcocokan, lalu

bandingkan dengan nilai OH yang telah dioperasikan. Dengan nilai terbesar, geser *pattern* ke arah kiri teks dan ulangi pencocokan.

4. Jika nilai kecocokan sama dengan panjang *pattern* maka *pattern* telah ditemukan pada teks, geser *pattern* sepanjang jumlah karakter *pattern* ke arah kiri teks untuk melanjutkan pencocokan selanjutnya.

### Contoh kasus:

*Pattern* : MIKROSKOP

Teks : ABCD MIKROSKOP

Penyelesaian :

Stack BMBC									
Karakter	M	I	K	R	O	S	K	O	P
Nilai OH	8	7	2	5	1	3	2	1	0

Stack BMGS									
Karakter	M	I	K	R	O	S	K	O	P
Nilai MH	9	9	9	9	9	9	9	9	1

Index	0	1	2	3	4	5	6	7	8	9	10	11	12
Text	A	B	C	D	M	I	K	R	O	S	K	O	P
Pattern	M	I	K	R	O	S	K	O	P				

Max BMBC[O] = 1 banding BMGS[P] = 1

Shift = 1 pergeseran

Text	A	B	C	D	M	I	K	R	O	S	K	O	P
Pattern		M	I	K	R	O	S	K	O	P			

Max BMBC[S] = 3 banding BMGS[P] = 1

Shift = 3 pergeseran

Text	A	B	C	D	M	I	K	R	O	S	K	O	P
Pattern					M	I	K	R	O	S	K	O	P

**Gambar 3.20** Penyelesaian contoh kasus Prosedur BM

Pada proses pencarian diatas memberikan hasil bahwa kata mikroskop ditemukan pada data alat yang ada. Sehingga data alat yang dimaksud akan dimunculkan dalam bentuk tabel.

**Tabel 3.3 Contoh Hasil Pencarian**

No	Nama Alat	Seri	Merk	Satuan	Stok
1	Abcdmikroskop	A	Merk A	Buah	10

Pada Gambar 3.19 terlihat bahwa pergeseran pada *pattern* selalu mengikuti keputusan dari hasil perbandingan OH dan MH. Pada umumnya, pencocokan standar akan mencocokkan karakter per karakter, ketika ditemukan ketidakcocokan maka *pattern* akan digeser satu langkah ke arah selanjutnya. Namun tidak demikian pada algoritma *Boyer-Moore*. Inilah yang menjadi alasan utama penulis menerapkan algoritma ini sebagai algoritma pencarian data pada sistem informasi manajemen laboratorium ketika dibutuhkannya sebuah proses pencocokan string.

## BAB IV

### UJI COBA DAN PEMBAHASAN

#### 4.2 Implementasi Sistem

Tahap implementasi merupakan tahap kelanjutan dari kegiatan perancangan sistem. Wujud dari hasil implementasi ini nantinya adalah sebuah sistem yang siap untuk diuji dan digunakan.

##### 4.1.1. Implementasi Perangkat Keras

Spesifikasi perangkat keras (*hardware*) yang digunakan untuk mengimplementasikan Sistem Informasi Manajemen Laboratorium dapat dilihat pada tabel 4.1

Tabel 4.1 Perangkat Keras Yang Digunakan

Perangkat Keras	Keterangan
<i>Processor</i>	Intel Core i7 3.1 GHz
<i>Memory</i>	DDR4 8GB
<i>Harddisk</i>	1000GB HDD
VGA	Intel HD Graphics 520+NVIDIA GeForce 940MX (2 GB)
<i>Monitor</i>	15,6" HD (1366 x 768)

#### 4.1.2. Implementasi Perangkat Lunak

Spesifikasi perangkat lunak (*software*) yang digunakan untuk mengimplementasikan Sistem Informasi Manajemen Laboratorium dapat dilihat pada tabel 4.2.

Tabel 4.2 Perangkat Lunak Yang Digunakan

Perangkat Lunak	Keterangan
Sistem Operasi	Windows 10
IDE	Netbeans IDE 8.1
Web Server	XAMPP version 3.2.2
Database Server	MySQL
Web Browser	Mozilla Firefox versi 59.0.3

#### 4.2.4 Implementasi Algoritma *Boyer Moore*

Algoritma *Boyer Moore* di implementasikan kedalam proses pencarian pada sistem. Berikut form pencarian pada sistem.

Pencarian

Gambar 4.1 Form Pencarian

Pencarian

Gambar 4.2 Contoh Kalimat Pencarian

No	Nama Bahan	Tipe Bahan	Aksi
1	RJ 45	padat	<a href="#">Edit</a> <a href="#">Hapus</a>
2	Calcium chloride	padat	<a href="#">Edit</a> <a href="#">Hapus</a>
3	Copper (II) acetate	padat	<a href="#">Edit</a> <a href="#">Hapus</a>
4	Sputit 20 ml	padat	<a href="#">Edit</a> <a href="#">Hapus</a>
5	Membran filter	padat	<a href="#">Edit</a> <a href="#">Hapus</a>
6	NAA	padat	<a href="#">Edit</a> <a href="#">Hapus</a>
7	Thidiazuron	padat	<a href="#">Edit</a> <a href="#">Hapus</a>
8	Scalpel Blade	padat	<a href="#">Edit</a> <a href="#">Hapus</a>
9	Masker	padat	<a href="#">Edit</a> <a href="#">Hapus</a>
10	Handgloves	padat	<a href="#">Edit</a> <a href="#">Hapus</a>
11	tube PCR	padat	<a href="#">Edit</a> <a href="#">Hapus</a>
12	Negrosin	padat	<a href="#">Edit</a> <a href="#">Hapus</a>
13	Bahan A	padat	<a href="#">Edit</a> <a href="#">Hapus</a>
14	Serbuk tembaga murni	padat	<a href="#">Edit</a> <a href="#">Hapus</a>
15	Serbuk NaCl teknis	padat	<a href="#">Edit</a> <a href="#">Hapus</a>
16	CuSO4 teknik	padat	<a href="#">Edit</a> <a href="#">Hapus</a>
17	Serbuk Al2SO4 (Alumina)	padat	<a href="#">Edit</a> <a href="#">Hapus</a>
No	Nama Bahan	Tipe Bahan	Aksi

Gambar 4.3 Hasil Pencarian

Selanjutnya akan dilakukan proses *tokenizing* pada kalimat yang dicari sehingga kalimat pencarian akan dipecah-pecah menjadi beberapa kata. Berikut source code untuk pemecahan kata.

```
$string = $this->input->post('pencarian');
$PecahStr = explode(" ", $string);
$data = $this->bahan->getallbahan();
```

Gambar 4.4 Source Code Tokenizing

#### 4.2.4.1 Proses Pencocokan String dalam Algoritma *Boyer Moore*

Dalam proses pencocokan dengan menggunakan Algoritma *Boyer Moore* ini, sistem terlebih dahulu memberikan nilai OH dan MH pada pattern pencocokan sebelum memasuki tahap pencocokan string dalam Algoritma Boyer Moore itu sendiri.

a. Proses preBmBc

Proses preBmBc dilakukan untuk mendapatkan nilai OH. Pemberian nilai OH dimulai dari karakter paling kanan pada *pattern*. Dalam prosesnya dilakukan pemecahan *pattern* yang kemudian masing-masing karakter diberi nilai sesuai dengan *index* dari kanan ke kiri, apabila menemukan karakter yang sama dengan karakter sebelumnya yang telah diberi nilai, maka nilai karakter tersebut sama dengan nilai OH karakter sebelumnya. Proses ini dapat dilihat pada Gambar 4.5

```

    $m = strlen($pattern);
    for ($i = 0; $i < $m - 1; ++$i) {
        $badChars[$pattern{$i}] = $m - $i - 1;
    }
    return $badChars;
}

```

Gambar 4.5 Potongan *SourceCode* Pemberian Nilai OH

b. Proses preBmGs

Proses preBmBc dilakukan untuk mendapatkan nilai MH. Pemberian nilai MH pada *pattern* dimulai dengan membentuk *suffix* kanan ke kiri dan *suffix* pembandingan dari kiri ke kanan. Kemudian memberikan nilai pergeseran untuk *suffix* kanan ke kiri (Gambar 4.6). Selanjutnya pemberian nilai MH dimulai dari karakter paling kanan pada *pattern* yang selalu diberi nilai MH=1. Kemudian proses berjalan ke kiri karakter dan dilakukan pencocokan terhadap *suffix* pembandingan, apabila terdapat kesamaan *suffix*, maka nilai MH karakter sama dengan nilai pergeseran pada *suffix*, jika tidak sama maka nilai MH sebanyak jumlah *pattern*. Proses ini dapat dilihat pada Gambar 4.7

```

$m = strlen($pattern);
$suffixes[$m - 1] = $m;
$g = $m - 1;
for ($i = $m - 2; $i >= 0; --$i) {
if ($i > $g && $suffixes[$i + $m - 1 - $f] < $i - $g) {
$suffixes[$i] = $suffixes[$i + $m - 1 - $f];
} else {
if ($i < $g) {
$g = $i;
}
$f = $i;
while ($g >= 0 && $pattern[$g] == $pattern[$g + $m - 1 - $f]) {
$g--;
}
$suffixes[$i] = $f - $g;
}
}

```

**Gambar 4.6** Potongan Source Code Pembentukan Suffix dan Nilai Pergeseran

```

$m = strlen($pattern);
$suff = array();
$this->suffixes($pattern, $suff);
for ($i = 0; $i < $m; $i++) {
    $goodSuffixes[$i] = $m;
}
for ($i = $m - 1; $i >= 0; $i--) {
if ($suff[$i] == $i + 1) {
for ($j = 0; $j < $m - $i - 1; $j++) {
if ($goodSuffixes[$j] == $m) {
            $goodSuffixes[$j] = $m - $i - 1;
        }
    }
}
}
for ($i = 0; $i < $m - 2; $i++) {
    $goodSuffixes[$m - 1 - $suff[$i]] = $m - $i - 1 - 1;
}
}

```

**Gambar 4.7** Potongan SourceCode Pemberian Nilai MH pada Karakter

c. Proses BM

Setelah mendapatkan nilai OH dan MH, tahap selanjutnya adalah memasuki proses pencocokan string. Dalam prosesnya, Algoritma *BoyerMoore* melakukan pengecekan dari awal teks dimulai dari karakter paling kanan pada pattern.Selanjutnya melakukan pergeseran pattern sesuai dengan

maksimal nilai OH karakter teks dan nilai MH karakter pattern. Potongan *sourcecode* proses ini dapat dilihat pada Gambar 4.8

```

$n = strlen($text);
$m = strlen($pattern);
$j = 0;
while ($j < $n - $m) {
for ($i = $m - 1; $i >= 0 && $pattern[$i] == $text[$i + $j];
$i--);
if ($i < 0) {
echo $j;
$j += $goodSuffixes[0];
} else {
$j += max($goodSuffixes[$i], $badCharacters[$text[$i + $j]] -
$m + $i + 1);
}
}

```

Gambar 4.8 Potongan *Source Code* untuk Pencocokan String

### 4.3 Pengujian Sistem

Pengujian merupakan hal terpenting yang bertujuan untuk menemukan kesalahan-kesalahan atau kekurangan-kekurangan pada perangkat lunak dan bermaksud untuk mengetahui perangkat lunak yang dibuat sudah memenuhi kriteria yang sesuai dengan tujuan perancangan perangkat yang akan iuji

#### 4.3.2 Pengujian *Black Box*

Pengujian fungsional yang digunakan untuk menguji sistem yang baru adalah metode pengujian alpha. Metode yang digunakan dalam pengujian ini adalah pengujian *black box* yang berfokus pada persyaratan fungsional dari sistem yang dibangun.

a. *Form Login*

Menu Login berfungsi untuk melakukan verifikasi dan membatasi hak – hak penggunaan aplikasi yang dimiliki oleh user.

**Tabel 4.3** Uji Coba Form Login

No	Tujuan	Input	Output Diharapkan	Output Sistem
1	Deskripsi username dan password valid	- username : admin - password : admin123	Menu login tertutup, dan menu admin aktif	1. Sukses 2. Login Berhasil 3. Tampil Dashboard admin
2	Deskripsi username dan password tidak valid	- username : admin - password : admintest	Tampil alert username dan password salah	1. Sukses 2. Login tidak berhasil 3. Muncul pesan seperti yang diharapkan

The screenshot shows a web application interface for registration and login. At the top, there is a teal navigation bar with the text 'REGISTER / LOGIN' and a 'Beranda' link. Below the navigation bar, there are two tabs: 'Form Pendaftaran' and 'Login Form'. The 'Login Form' tab is selected, displaying a form with the following fields: 'Nomor Induk' (text input), 'Nama' (text input), 'Jenis Kelamin' (dropdown menu), 'Jurusan' (dropdown menu), 'Nomor HP' (text input), 'Alamat' (text input), 'Email' (text input), 'Username' (text input), and 'Password' (text input). A 'Login' button is located to the right of the password field. The background of the page features a large, faint watermark of the Maulana Malik Ibrahim State Islamic University of Malang logo.

**Gambar 4.9** Form Login

Gambar 4.10 Login Gagal

b. *Form* tambah bahan

Pada *form* ini berfungsi untuk menambahkan data bahan praktikum yang belum ada pada daftar bahan. *Form* ini terdiri dari beberapa *textfield* yang berhubungan dengan variabel data yang dibutuhkan pada bahan praktikum.

Tabel 4.4 Uji Coba *Form* Tambah Bahan

No	Tujuan	Input	Output Diharapkan	Output Sistem
1	Menambahkan data baru.	Memasukkan data kemudian menekan tombol simpan.	Data berhasil disimpan	1. Sukses 2. Muncul alert Data berhasil disimpan 3. Data tersimpan di database
2	Menghindari data input kosong	Tidak memasukkan data kemudian tekan tombol simpan	Tampil alert data tidak boleh kosong	1. Sukses 2. Muncul Alert data tidak boleh kosong.

The screenshot shows a web application interface for adding material details. The page title is 'Detail Bahan' and the sub-header is 'Tambah Detail Bahan'. A search bar is present in the top right. The form contains the following fields:

- Nama Jurusan\***: Dropdown menu with 'Pilih Jurusan' selected.
- Nama Bahan\***: Text input field with 'Nama Bahan' placeholder.
- Merk\***: Text input field with 'Merk Bahan' placeholder.
- Seri/Ukuran/Jenis**: Text input field with 'Seri/Ukuran/Jenis' placeholder.
- Harga Satuan\***: Text input field with 'Harga Satuan' placeholder.
- Nama Satuan\***: Dropdown menu with 'Pilih Satuan' selected.
- Tanggal Expired**: Date input field with 'mm / dd / yyyy' placeholder.
- Foto\***: File upload field with 'Browse' button and 'No file selected' text.
- Tipe Bahan**: Radio buttons for 'Padat' and 'Cair'.
- Jenis Bahan\***: Radio buttons for 'Habis Pakai' and 'Tidak Habis Pakai'.
- Sifat Bahan**: Dropdown menu with 'Pilih Keterangan Bahan' selected.

Buttons at the bottom include 'Simpan' (Save) and 'Cancel'.

Gambar 4.11 Form Tambah Bahan

The screenshot shows the same 'Tambah Detail Bahan' form, but with a success message overlay. The message is a white box with a green checkmark icon and the text 'Data Berhasil Disimpan' and an 'OK' button. The form fields are filled with the following data:

- Nama Jurusan\***: Biologi
- Nama Bahan\***: Asam Asetat Distilat
- Merk\***: merk
- Seri/Ukuran/Jenis**: Seri/Ukuran/Jenis
- Harga Satuan\***: 227600
- Nama Satuan\***: L (Liter)
- Tanggal Expired**: mm / dd / yyyy
- Foto\***: Browse button with 'asa' text
- Tipe Bahan**: Radio button for 'Cair' is selected.
- Jenis Bahan\***: Radio button for 'Habis Pakai' is selected.
- Sifat Bahan**: Dropdown menu with 'Pilih Keterangan Bahan' selected.

Buttons at the bottom include 'Simpan' (Save) and 'Cancel'.

Gambar 4.12 Data Bahan Berhasil di Tambahkan

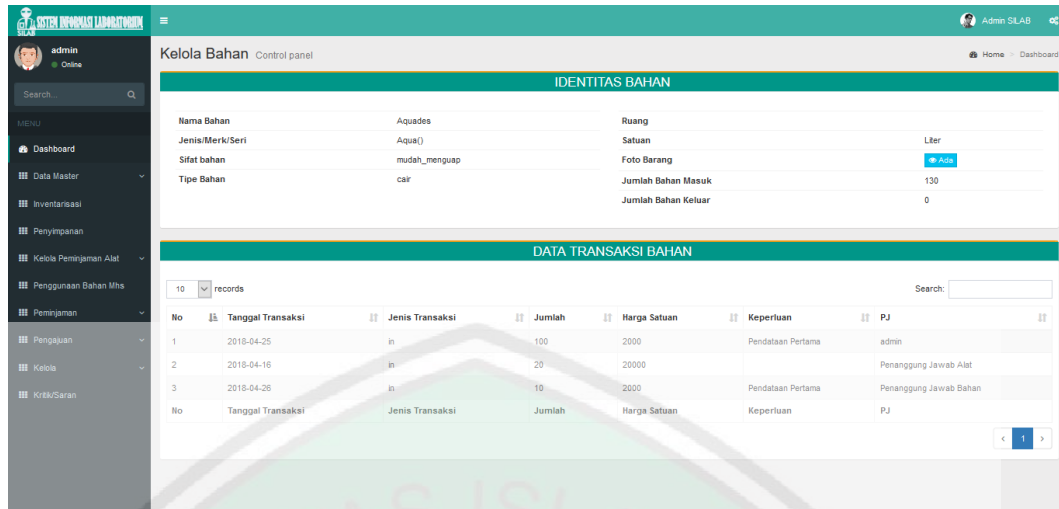
**Gambar 4.13** Alert Data Tidak Boleh Kosong

c. *Form detail bahan*

Menu ini berfungsi untuk menampilkan detail informasi dari bahan praktikum dan detail transaksi dari bahan

**Tabel 4.5** Uji Coba Menu Detail Bahan

No	Tujuan	Input	Output Diharapkan	Output Sistem
1	Menampilkan data bahan praktikum	Klik tombol detail bahan	Detail bahan ditampilkan	1. Sukses 2. Informasi detail bahan ditampilkan



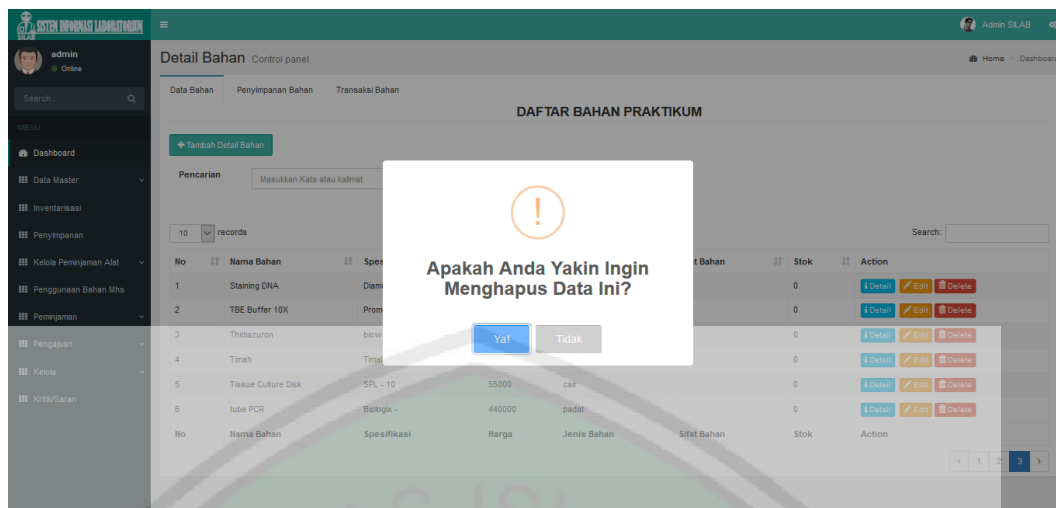
Gambar 4.14 Tampilan Detail Bahan

d. Menu hapus bahan

Menu ini berfungsi untuk menghapus bahan yang tidak dibutuhkan pada daftar bahan praktikum.

Tabel 4.6 Uji Coba Menu Hapus Bahan

No	Tujuan	Input	Output Diharapkan	Output Sistem
1	Menampilkan peringatan hapus bahan	Klik tombol Hapus pada bahan yang ingin di hapus	Peringatan hapus bahan	1. Sukses 2. Peringatan hapus bahan ditampilkan
2	Menghapus data bahan praktikum	Klik option “Ya” pada peringatan penghapusan bahan .	Data bahan terhapus	1. Sukses 2. Data bahan terhapus dari database



Gambar 4.15 Peringatan Penghapusan Bahan

e. *Form* tambah alat praktikum

Pada *form* ini berfungsi untuk menambahkan data alat praktikum yang belum ada pada daftar alat. *Form* ini terdiri dari beberapa *textfield* yang berhubungan dengan variabel data yang dibutuhkan pada alat praktikum.

Tabel 4.7 Uji Coba *Form* Tambah Alat

No	Tujuan	Input	Output Diharapkan	Output Sistem
1	Menambahkan data alat baru.	Memasukkan data pada form kemudian menekan tombol simpan.	Data berhasil disimpan	1. Sukses 2. Muncul alert Data berhasil disimpan 3. Data tersimpan di database
2	Menghindari data input kosong	Tidak memasukkan data kemudian tekan tombol simpan	Tampil alert data tidak boleh kosong	1. Sukses 2. Muncul Alert data tidak boleh kosong.

Gambar 4.16 Form Tambah Alat

Gambar 4.17 Data Alat Berhasil di Tambahkan

f. *Form edit alat*

*Form* ini berfungsi untuk melakukan pembaharuan data pada alat praktikum. Berisi tentang data alat praktikum yang akan diubah isinya.

Tabel 4.8 Uji Coba *Form* Edit Alat

No	Tujuan	Input	Output Diharapkan	Output Sistem
1	Memperbaharui data alat praktikum	Mengganti data alat praktikum yang dirubah kemudian klik simpan	Data berhasil diperbaharui	1. Sukses 2. Data berhasil diperbaharui

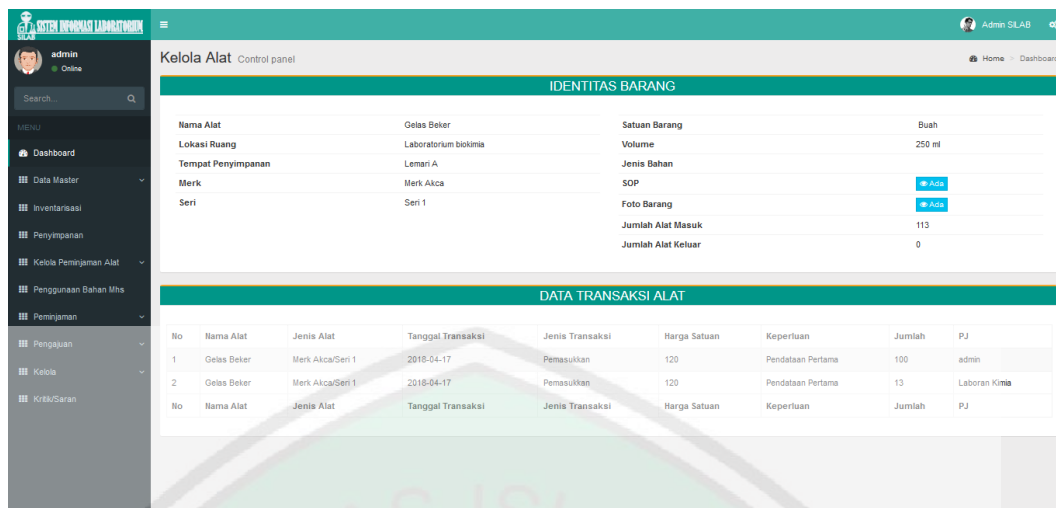
Gambar 4.18 *Form* Edit Alat

g. Menu detail alat

Menu ini berfungsi untuk menampilkan detail informasi dari alat praktikum dan detail transaksi dari alat.

Tabel 4.9 Uji Coba Menu Detail Alat

No	Tujuan	Input	Output Diharapkan	Output Sistem
1	Menampilkan detail alat praktikum	Klik tombol detail	Detail alat ditampilkan	1. Sukses 2. Informasi detail alat ditampilkan



Gambar 4.19 Detail Alat Praktikum

h. Menu hapus alat

Menu ini berfungsi untuk menghapus bahan yang tidak dibutuhkan pada daftar bahan praktikum.

Tabel 4.10 Uji Coba Menu Hapus Alat

No	Tujuan	Input	Output Diharapkan	Output Sistem
1	Menampilkan peringatan hapus alat	Klik tombol Hapus pada alat yang ingin di hapus	Peringatan hapus alat	1. Sukses 2. Peringatan hapus alat ditampilkan
2	Menghapus data alat praktikum	Klik option "Ya" pada peringatan penghapusan alat	Data alat terhapus	1. Sukses 2. Data alat terhapus dari database

i. *Form* peminjaman bahan

Pada form ini berfungsi untuk mahasiswa dalam melakukan peminjaman bahan praktikum, data peminjaman akan masuk ke daftar peminjaman, dan akan diverifikasi oleh laboran jika peminjaman berhasil di lakukan.

Tabel 4.11 Uji Coba Form Peminjaman Bahan

No	Tujuan	Input	Output Diharapkan	Output Sistem
1	Menyimpan data peminjaman bahan	Mengisi data seluruh data pada form peminjaman kemudian Klik button “Simpan & Selesai”	Peminjaman bahan berhasil di lakukan	<ol style="list-style-type: none"> <li>1. Sukses</li> <li>2. Alert Peminjaman berhasil dilakukan</li> <li>3. Data peminjaman berhasil di simpan di database</li> </ol>
2	Menghindari data input kosong	Tidak memasukkan data kemudian tekan tombol simpan	Proses peminjaman tidak dilanjutkan, dan muncul alert data kosong	<ol style="list-style-type: none"> <li>1. Sukses</li> <li>2. Proses peminjaman tidak dapat dilanjutkan</li> </ol>

Data Peminjaman Bahan

Data Peminjaman Bahan

Step 1/2

Identitas Peminjaman

Kategori Peminjaman: Penelitian

Tanggal Peminjaman: 2018-05-15

Nama Ruang/Lab: Laboratorium biokimia

Keterangan: untuk penelitian bahan penelitian

Next

Gambar 4.20 Form Peminjaman Bahan Step1

Data Peminjaman Bahan

Data Peminjaman Bahan

Step 2/2

Bahan Yang Digunakan

Bahan: Aquadest/Aqua aqua123

Jumlah: 1

Penjelasan

Nama Bahan: Isi Bahan yang akan dipinjam

Jumlah: Jumlah yang dimasukkan tidak boleh melebihi stok yang ada

Info Bahan

Nama Bahan: Aquadest (Aqua aqua123)

Foto Bahan

Stok Bahan: 130

Simpan & Selesai

Gambar 4.21 Form Peminjaman Bahan Step2

Data Peminjaman Bahan

Data Peminjaman Bahan

Step 2/2

Bahan Yang Digunakan

Bahan: Aquadest/Aqua aqua123

Jumlah: 1

Penjelasan

Nama Bahan: Isi Bahan yang akan dipinjam

Jumlah: Jumlah yang dimasukkan tidak boleh melebihi stok yang ada

Info Bahan

Nama Bahan: Aquadest (Aqua aqua123)

Foto Bahan

Stok Bahan: 130

Simpan & Selesai

Data Berhasil Disimpan

OK

Gambar 4.22 Proses Peminjaman Berhasil

**Gambar 4.23** Peringatan Input Data Peminjaman Bahan Kosong

j. *Form Peminjaman Alat*

Pada *form* ini berfungsi untuk mahasiswa dalam melakukan peminjaman alat praktikum, data peminjaman akan masuk ke daftar peminjaman alat, dan akan diverifikasi oleh laboran jika peminjaman berhasil di lakukan.

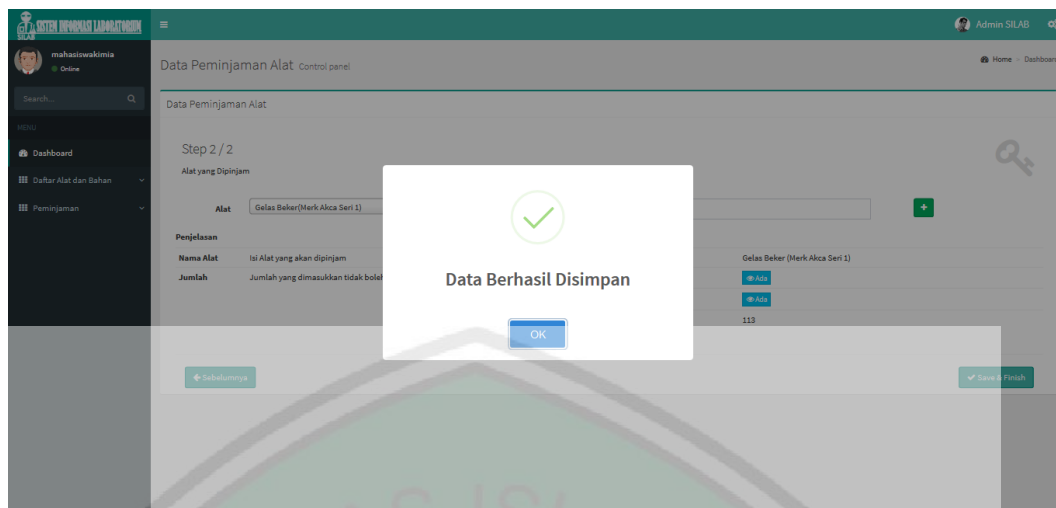
**Tabel 4.12** Uji Coba Form Peminjaman Alat

No	Tujuan	Input	Output Diharapkan	Output Sistem
1	Menyimpan data peminjaman alat	Mengisi data seluruh data pada form peminjaman kemudian Klik button “Simpan & Selesai”	Peminjaman bahan berhasil di lakukan	<ol style="list-style-type: none"> <li>1. Sukses</li> <li>2. Alert Peminjaman berhasil dilakukan</li> <li>3. Data peminjaman berhasil di simpan di database</li> </ol>
2	Menghindari data input kosong	Tidak memasukkan data kemudian	Proses peminjaman tidak	<ol style="list-style-type: none"> <li>1. Sukses</li> <li>2. Proses peminjaman</li> </ol>

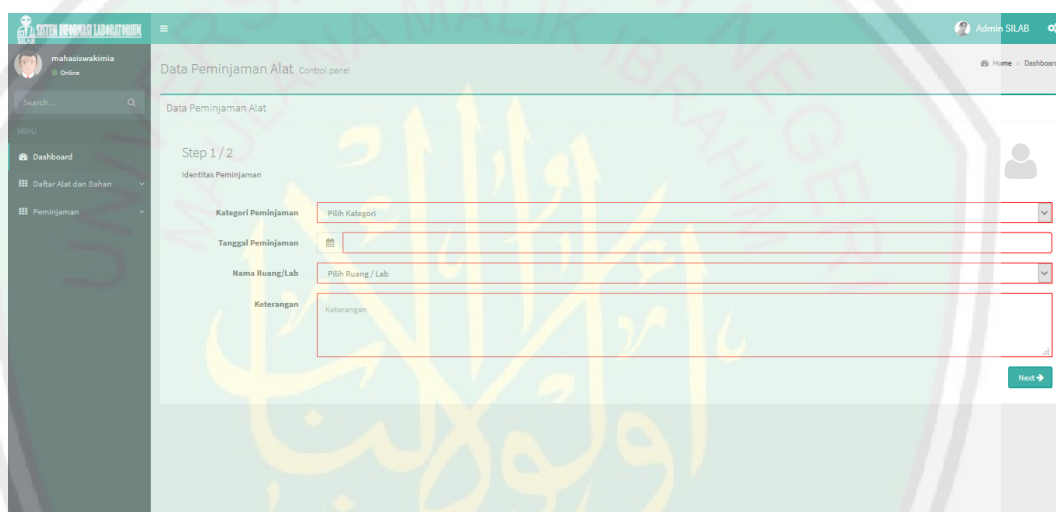
		tekan tombol simpan	dilanjutkan, dan muncul alert data kosong	tidak dapat dilanjutkan
--	--	------------------------	---	----------------------------

Gambar 4.24 Form Peminjaman Alat Step 1

Gambar 4.25 Form Peminjaman Alat Step 2



Gambar 4.26 Data Peminjaman Alat Berhasil di Simpan



Gambar 4.27 Peringatan Input Data Peminjaman Alat Kosong

### 4.3.3 Pengujian User

Pengujian dilakukan dengan tujuan untuk mengetahui sejauh mana kualitas dari perangkat lunak yang dibangun, apakah sudah sesuai dengan harapan atau belum. Untuk itu dalam pengujian beta dilakukan penelitian dengan cara memberikan kuesioner pada calon pengguna perangkat lunak yang dibangun. Contoh kuesioner bisa dilihat pada lampiran 1. Adapun metode penelitian yang digunakan adalah metode kuantitatif.

Kuesioner diberikan kepada 10 sample calon pengguna perangkat lunak yang dibangun. Kuesioner terdiri dari 2 jenis dengan 20 pertanyaan dan saran untuk setiap kuisisioner. Kuisisioner dibuat menggunakan skala likert dengan skala 1 sampai 5. Tabel bobot nilai bisa dilihat pada Tabel 4.13

**Tabel 4.13** Bobot Nilai

A	1
B	2
C	3
D	4
E	5

**Tabel 4.14** Keterangan Nilai Kuesioner

Jawaban	Keterangan
0% - 19.99%	Sangat (Tidak Setuju, Buruk atau Kurang Sekali)
20% - 39.99%	Tidak Setuju atau Kurang Baik
40% - 59.99%	Cukup atau Netral
60% - 79.99%	Setuju, Baik atau Suka
80% - 100%	Sangat (Setuju, Baik, Suka)

Untuk mendapatkan hasil interpretasi, harus diketahui dulu skor tertinggi (X) dan angka terendah (Y) untuk item penilaian dengan rumus sebagai berikut :

$Y = \text{Skor tertinggi likert} \times \text{jumlah responden}$

$X = \text{Skor terendah likert} \times \text{jumlah responden}$

**Rumus Index % = Total Skor / Y x 100** (4.1)

Jumlah skor tertinggi untuk item sangat setuju ialah  $5 \times 10 = 50$ , sedangkan item sangat tidak setuju ialah  $1 \times 10 = 10$

Berikut ini adalah hasil prosentase masing-masing jawaban yang sudah dihitung nilainya dengan menggunakan rumus diatas.

1. Apakah informasi yang disediakan oleh aplikasi mudah dimengerti?

**Tabel 4.15** Hasil Pengujian Kuesioner Pertanyaan Nomor 1

Bobot Soal	Keterangan	Responden	Skor	Hasil
5	Sangat Setuju	1	5	76%
4	Setuju	6	24	
3	Netral	3	9	
2	Tidak Setuju			
1	Sangat Tidak Setuju			
Total		10	38	

Dari hasil di atas maka dapat disimpulkan responden setuju bahwa aplikasi mudah untuk dipahami.

2. Apakah penggunaan menu atau fitur aplikasi menu mudah digunakan?

**Tabel 4.16** Hasil Pengujian Kuesioner Pertanyaan Nomor 2

Bobot Soal	Keterangan	Responden	Skor	Hasil
5	Sangat Setuju	1	5	76%
4	Setuju	6	24	
3	Netral	3	9	
2	Tidak Setuju			
1	Sangat Tidak Setuju			
Total		10	38	

Dari hasil di atas maka dapat disimpulkan responden setuju bahwa penggunaan menu atau fitur aplikasi menu mudah digunakan.

3. Apakah aplikasi nyaman digunakan?

**Tabel 4.17** Hasil Pengujian Kuesioner Pertanyaan Nomor 3

Bobot Soal	Keterangan	Responden	Skor	Hasil
5	Sangat Setuju	1	5	78%
4	Setuju	7	28	
3	Netral	2	6	
2	Tidak Setuju			
1	Sangat Tidak Setuju			
Total		10	39	

Dari hasil di atas maka dapat disimpulkan responden setuju bahwa aplikasi nyaman digunakan.

4. Secara keseluruhan apakah penggunaan aplikasi ini memuaskan?

**Tabel 4.18** Hasil Pengujian Kuesioner Pertanyaan Nomor 4

Bobot Soal	Keterangan	Responden	Skor	Hasil
5	Sangat Setuju	1	5	78%
4	Setuju	7	28	
3	Netral	2	6	
2	Tidak Setuju			
1	Sangat Tidak Setuju			
Total		10	39	

Dari hasil di atas maka dapat disimpulkan responden setuju bahwa secara keseluruhan **aplikasi memuaskan**.

## 5. Apakah aplikasi ini sesuai dengan kebutuhan?

**Tabel 4.19** Hasil Pengujian Kuesioner Pertanyaan Nomor 5

Bobot Soal	Keterangan	Responden	Skor	Hasil
5	Sangat Setuju	3	15	86%
4	Setuju	7	28	
3	Netral			
2	Tidak Setuju			
1	Sangat Tidak Setuju			
Total		10	43	

Dari hasil di atas maka dapat disimpulkan responden sangat setuju bahwa **aplikasi sesuai dengan kebutuhan.**

## 6. Apakah aplikasi dapat dengan mudah dipelajari?

**Tabel 4.20** Hasil Pengujian Kuesioner Pertanyaan Nomor 6

Bobot Soal	Keterangan	Responden	Skor	Hasil
5	Sangat Setuju	1	5	78%
4	Setuju	7	28	
3	Netral	2	6	
2	Tidak Setuju			
1	Sangat Tidak Setuju			
Total		10	39	

Dari hasil di atas maka dapat disimpulkan responden setuju bahwa aplikasi mudah untuk dipelajari.

## 7. Apakah aplikasi mudah dioperasikan?

**Tabel 4.21** Hasil Pengujian Kuesioner Pertanyaan Nomor 7

Bobot Soal	Keterangan	Responden	Skor	Hasil
5	Sangat Setuju	1	5	72%
4	Setuju	4	16	
3	Netral	5	15	
2	Tidak Setuju			
1	Sangat Tidak Setuju			
Total		10	36	

Dari hasil di atas maka dapat disimpulkan responden setuju bahwa aplikasi mudah untuk dioperasikan.

## 8. Apakah dapat dengan mudah menghindari kesalahan dalam menggunakan aplikasi?

**Tabel 4.22** Hasil Pengujian Kuesioner Pertanyaan Nomor 8

Bobot Soal	Keterangan	Responden	Skor	Hasil
5	Sangat Setuju	1	5	70%
4	Setuju	3	12	
3	Netral	6	18	
2	Tidak Setuju			
1	Sangat Tidak Setuju			
Total		10	35	

Dari hasil di atas maka dapat disimpulkan responden setuju bahwa dengan menggunakan aplikasi dapat dengan mudah menghindari kesalahan.

## 9. Apakah aplikasi bermanfaat bagi pengguna?

**Tabel 4.23** Hasil Pengujian Kuesioner Pertanyaan Nomor 9

Bobot Soal	Keterangan	Responden	Skor	Hasil
5	Sangat Setuju	2	10	78%
4	Setuju	5	20	
3	Netral	3	9	
2	Tidak Setuju			
1	Sangat Tidak Setuju			
Total		10	39	

Dari hasil di atas maka dapat disimpulkan responden setuju bahwa aplikasi bermanfaat bagi pengguna.

## 10. Apakah tampilan menu dalam aplikasi mudah untuk dikenali?

**Tabel 4.24** Hasil Pengujian Kuesioner Pertanyaan Nomor 10

Bobot Soal	Keterangan	Responden	Skor	Hasil
5	Sangat Setuju	1	5	78%
4	Setuju	7	28	
3	Netral	2	6	
2	Tidak Setuju			
1	Sangat Tidak Setuju			
Total		10	39	

Dari hasil di atas maka dapat disimpulkan responden setuju bahwa tampilan menu dalam aplikasi mudah untuk dikenali.

11. Apakah aplikasi mempunyai kemampuan dan fungsi sesuai yang diharapkan?

**Tabel 4.25** Hasil Pengujian Kuesioner Pertanyaan Nomor 11

Bobot Soal	Keterangan	Responden	Skor	Hasil
5	Sangat Setuju			74%
4	Setuju	7	28	
3	Netral	3	9	
2	Tidak Setuju			
1	Sangat Tidak Setuju			
Total		10	37	

Dari hasil di atas maka dapat disimpulkan responden setuju bahwa aplikasi mempunyai kemampuan dan fungsi sesuai yang diharapkan.

12. Apakah dengan adanya aplikasi ini membantu pengelolaan data menjadi lebih mudah?

**Tabel 4.26** Hasil Pengujian Kuesioner Pertanyaan Nomor 12

Bobot Soal	Keterangan	Responden	Skor	Hasil
5	Sangat Setuju	3	15	80%
4	Setuju	4	16	
3	Netral	3	9	
2	Tidak Setuju			
1	Sangat Tidak Setuju			
Total		7	40	

Dari hasil di atas maka dapat disimpulkan bahwa responden sangat setuju bahwa dengan adanya aplikasi ini membantu pengelolaan data menjadi lebih mudah.

13. Apakah dengan adanya aplikasi ini memberikan informasi data secara akurat?

**Tabel 4.27** Hasil Pengujian Kuesioner Pertanyaan Nomor 13

Bobot Soal	Keterangan	Responden	Skor	Hasil
5	Sangat Setuju			72%
4	Setuju	6	24	
3	Netral	4	12	
2	Tidak Setuju			
1	Sangat Tidak Setuju			
Total		10	36	

Dari hasil di atas maka dapat disimpulkan responden setuju bahwa dengan adanya aplikasi ini memberikan informasi data secara akurat.

14. Apakah tampilan website cukup menarik dan mudah dipahami (user friendly)?

**Tabel 4.28** Hasil Pengujian Kuesioner Pertanyaan Nomor 14

Bobot Soal	Keterangan	Responden	Skor	Hasil
5	Sangat Setuju	1	5	78%
4	Setuju	7	28	
3	Netral	2	6	
2	Tidak Setuju			
1	Sangat Tidak Setuju			
Total		10	39	

Dari hasil di atas maka dapat disimpulkan bahwa responden setuju bahwa tampilan website cukup menarik dan mudah dipahami.

15. Apakah dengan adanya aplikasi ini dapat menghasilkan laporan yang tepat seperti yang dibutuhkan?

**Tabel 4.29** Hasil Pengujian Kuesioner Pertanyaan Nomor 15

Bobot Soal	Keterangan	Responden	Skor	Hasil
5	Sangat Setuju			72%
4	Setuju	6	24	
3	Netral	4	12	
2	Tidak Setuju			
1	Sangat Tidak Setuju			
Total		10	36	

Dari hasil di atas maka dapat disimpulkan responden setuju bahwa dengan adanya aplikasi ini dapat menghasilkan laporan yang tepat seperti yang dibutuhkan.

16. Apakah dengan adanya aplikasi ini dapat memperoleh informasi yang dibutuhkan dengan tepat waktu?

**Tabel 4.30** Hasil Pengujian Kuesioner Pertanyaan Nomor 16

Bobot Soal	Keterangan	Responden	Skor	Hasil
5	Sangat Setuju			74%
4	Setuju	7	28	
3	Netral	3	9	
2	Tidak Setuju			
1	Sangat Tidak Setuju			
Total		10	37	

Dari hasil di atas maka dapat disimpulkan responden setuju bahwa dengan adanya aplikasi ini dapat memperoleh informasi yang dibutuhkan dengan tepat waktu.

17. Apakah aplikasi ini mampu memberikan informasi sesuai dengan format yang dibutuhkan?

**Tabel 4.31** Hasil Pengujian Kuesioner Pertanyaan Nomor 17

Bobot Soal	Keterangan	Responden	Skor	Hasil
5	Sangat Setuju			76%
4	Setuju	8	32	
3	Netral	2	6	
2	Tidak Setuju			
1	Sangat Tidak Setuju			
Total		10	38	

Dari hasil di atas maka dapat disimpulkan responden setuju bahwa aplikasi mampu memberikan informasi sesuai dengan format yang dibutuhkan

18. Apakah aplikasi ini mampu menghasilkan informasi yang dapat dipahami secara jelas?

**Tabel 4.32** Hasil Pengujian Kuesioner Pertanyaan Nomor 18

Bobot Soal	Keterangan	Responden	Skor	Hasil
5	Sangat Setuju			76%
4	Setuju	8	32	
3	Netral	2	6	
2	Tidak Setuju			
1	Sangat Tidak Setuju			
Total		10	38	

Dari hasil di atas maka dapat disimpulkan responden setuju bahwa aplikasi mampu menghasilkan informasi yang dapat dipahami secara jelas.

19. Apakah aplikasi mudah dipelajari oleh orang yang baru pertama kali digunakan?

**Tabel 4.33** Hasil Pengujian Kuesioner Pertanyaan Nomor 19

Bobot Soal	Keterangan	Responden	Skor	Hasil
5	Sangat Setuju	1	5	76%
4	Setuju	6	24	
3	Netral	3	9	
2	Tidak Setuju			
1	Sangat Tidak Setuju			
Total		10	38	

Dari hasil di atas maka dapat disimpulkan responden setuju bahwa aplikasi mudah dipelajari oleh orang yang baru pertama kali digunakan.

20. Apakah aplikasi mampu menghasilkan informasi yang bersifat *mutakhir/up to date*?

**Tabel 4.34** Hasil Pengujian Kuesioner Pertanyaan Nomor 20

Bobot Soal	Keterangan	Responden	Skor	Hasil
5	Sangat Setuju			72%
4	Setuju	6	24	
3	Netral	4	12	
2	Tidak Setuju			
1	Sangat Tidak Setuju			
Total		10	36	

Dari hasil di atas maka dapat disimpulkan responden setuju bahwa aplikasi mampu menghasilkan informasi yang bersifat mutakhir/ *up to date*.

#### 4.3.4 Pengujian Algoritma *Boyer Moore*

Algoritma *Boyer Moore* pada aplikasi Sistem Informasi Manajemen Laboratorium, di implementasikan pada pencarian data pada sistem. Pencarian Pengujian dalam pencarian data menggunakan algoritma *boyer moore* ini dilakukan dengan melihat keakuratan sistem dalam mendapatkan data hasil pencarian. Jumlah data yang digunakan adalah sebagai berikut :

Tabel 4.35 Jumlah Data Yang Digunakan

No	Nama Data	Jumlah Data
1	Data Bahan	40
2	Penyimpanan Bahan	8
3	Transaksi Bahan	16
4	Data Ruang Laboratorium	25
5	Tempat Penyimpanan	16
6	Data Alat	20
7	Penyimpanan Alat	7
8	Transaksi Alat	22
9	Peminjaman Alat	15
10	Penggunaan Bahan	6
Total		175

Total data keseluruhan sebanyak 175, uji coba akan dilakukan sebanyak 50 kali percobaan dengan menggunakan kalimat-kalimat pencarian yang mungkin akan sering digunakan. Perhitungan akurasi menggunakan rumus akurasi pada *receiver operating characteristic* yang telah dijelaskan pada bab Studi Pustaka

$$\text{Akurasi} = \frac{\Sigma \text{ True Positive (TP)} + \Sigma \text{ True Negative (TN)}}{\Sigma \text{ Total Population}} \quad (2)$$

Sehingga diperoleh hasil pengujian seperti pada Tabel 4.36



Tabel 4.36 Hasil Uji Coba Pencarian

No	Kalimat Pencarian	Jumlah Data	Hasil Pencarian yang benar (TP)	Data Seharusnya yang benar	Hasil Pencarian Salah	Data yang tidak terseleksi dengan benar (TN)	Waktu	Ket	Akurasi
1	Tampilkan bahan berjenis cair	175	21	21	0	154	0.000015	Tepat	100%
2	Tampilkan bahan berjenis padat	175	19	19	0	156	0.000016	Tepat	100%
3	Tampilkan data bahan dan stok nya	175	0	40	16	135	0.000014	Tidak Tepat	77%
4	Tampilkan bahan yang habis	175	32	32	0	143	0.000015	Tepat	100%
5	Tampilkan bahan limit	175	2	2	0	173	0.000017	Tepat	100%

Tabel 4.37 Hasil Uji Coba Pencarian (Lanjutan)

No	Kalimat Pencarian	Jumlah Data	Hasil Pencarian yang benar (TP)	Data Seharusnya yang benar	Hasil Pencarian Salah	Data yang tidak terseleksi dengan benar (TN)	Waktu	Ket	Akurasi
6	Tampilkan data alat dan stok nya	175	20	20	0	155	0.000014	Tepat	100%
7	Tampilkan alat yang stoknya habis	175	12	12	0	163	0.000012	Tepat	100%
8	Tampilkan alat yang stoknya limit	175	1	1	0	174	0.0000238	Tepat	100%
9	Nama mahasiswa yang belum mengembalikan alat	175	17	17	0	158	0.000015	Tepat	100%
10	Nama mahasiswa yang sudah mengembalikan alat	175	1	1	0	174	0.000022	Tepat	100%

Tabel 4.38 Hasil Uji Coba Pencarian (Lanjutan)

No	Kalimat Pencarian	Jumlah Data	Hasil Pencarian yang benar (TP)	Data Seharusnya yang benar	Hasil Pencarian Salah	Data yang tidak terseleksi dengan benar (TN)	Waktu	Ket	Akurasi
11	Tampilkan nama bahan dan merk	175	40	40	0	135	0.0000138	Tepat	100%
12	tampilkan nama bahan dan harganya	175	40	40	0	135	0.0000141	Tepat	100%
13	tampilkan nama bahan dan jurusan	175	40	40	0	135	0.0000219	Tepat	100%
14	tampilkan nama bahan di fisika	175	13	13	0	162	0.000016	Tepat	100%
15	tampilkan bahan kimia	175	2	2	0	173	0.000042	Tepat	100%

Tabel 4.39 Hasil Uji Coba Pencarian (Lanjutan)

No	Kalimat Pencarian	Jumlah Data	Hasil Pencarian yang benar (TP)	Data Seharusnya yang benar	Hasil Pencarian Salah	Data yang tidak terseleksi dengan benar (TN)	Waktu	Ket	Akurasi
16	tampilkan bahan di jurusan biologi	175	23	23	0	152	0.000021	Tepat	100%
17	Nama bahan yang berada di jurusan teknik informatika	175	2	2	0	173	0.0000141	Tepat	100%
18	Bahan yang harganya kurang dari 10000	175	8	8	0	167	0.000015	Tepat	100%
19	Tampilkan bahan yang harganya lebih dari 50000	175	23	23	0	152	0.0000138	Tepat	100%
20	Tampilkan tempat alat di fisika	175	20	20	0	155	0.000021	Tepat	100%

Tabel 4.40 Hasil Uji Coba Pencarian (Lanjutan)

No	Kalimat Pencarian	Jumlah Data	Hasil Pencarian yang benar (TP)	Data Seharusnya yang benar	Hasil Pencarian Salah	Data yang tidak terseleksi dengan benar (TN)	Waktu	Ket	Akurasi
21	Tampilkan nama alat beserta merk nya	175	20	20	0	155	0.0000288	Tepat	100%
22	Nama alat dengan letak jurusan	175	19	19	0	156	0.000018	Tepat	100%
23	Nama alat yang berada di fisika	175	0	2	10	173	0.000037	Tidak Tepat	97%
24	Tampilkan nama alat kimia	175	5	5	0	170	0.0000281	Tepat	100%
25	Nama alat biologi	175	0	0	0	175	0.000021	Tepat	100%

Tabel 4.41 Hasil Uji Coba Pencarian (Lanjutan)

No	Kalimat Pencarian	Jumlah Data	Hasil Pencarian yang benar (TP)	Data Seharusnya yang benar	Hasil Pencarian Salah	Data yang tidak terseleksi dengan benar (TN)	Waktu	Ket	Akurasi
26	Tampilkan alat yang berada di teknik informatika	175	1	1	0	174	0.0000172	Tepat	100%
27	Tampilkan data semua alat	175	20	20	0	155	0.0000229	Tepat	100%
28	Tampilkan transaksi bahan	175	16	16	0	159	0.0000498	Tepat	100%
29	Tampilkan transaksi pemasukan bahan	175	12	12	0	163	0.0000198	Tepat	100%
30	Tampilkan transaksi pengeluaran bahan	175	4	4	0	174	0.0000241	Tepat	100%

Tabel 4.42 Hasil Uji Coba Pencarian (Lanjutan)

No	Kalimat Pencarian	Jumlah Data	Hasil Pencarian yang benar (TP)	Data Seharusnya yang benar	Hasil Pencarian Salah	Data yang tidak terseleksi dengan benar (TN)	Waktu	Ket	Akurasi
31	Tampilkan transaksi alat	175	22	22	0	153	0.0000169	Tepat	100%
32	Data transaksi pengeluaran alat	175	7	7	0	168	0.0000181	Tepat	100%
33	Data transaksi pemasukan alat	175	15	15	0	160	0.0000129	Tepat	100%
34	Tampilkan data peminjaman alat	175	18	18	0	157	0.0000141	Tepat	100%
35	Tampilkan data penggunaan bahan	175	7	7	0	168	0.000016	Tepat	100%

Tabel 4.43 Hasil Uji Coba Pencarian (Lanjutan)

No	Kalimat Pencarian	Jumlah Data	Hasil Pencarian yang benar (TP)	Data Seharusnya yang benar	Hasil Pencarian Salah	Data yang tidak terseleksi dengan benar (TN)	Waktu	Ket	Akurasi
36	Tampilkan semua data ruang diseluruh jurusan	175	25	25	0	150	0.00002	Tepat	100%
37	Tampilkan ruangan di jurusan teknik informatika	175	7	7	0	168	0.000025	Tepat	100%
38	Data ruang di jurusan biologi	175	6	6	0	169	0.000016	Tepat	100%
39	Tampilkan ruangan di jurusan fisika	175	6	6	0	169	0.0000138	Tepat	100%
40	Tampilkan ruangan di jurusan kimia	175	6	6	0	169	0.0000119	Tepat	100%

Tabel 4.44 Hasil Uji Coba Pencarian (Lanjutan)

No	Kalimat Pencarian	Jumlah Data	Hasil Pencarian yang benar (TP)	Data Seharusnya yang benar	Hasil Pencarian Salah	Data yang tidak terseleksi dengan benar (TN)	Waktu	Ket	Akurasi
41	Tampilkan seluruh tempat penyimpanan di lab	175	16	16	0	159	0.0000219	Tepat	100%
42	dimana letak bahan disimpan	175	8	8	0	167	0.000012	Tepat	100
43	Tampilkan tempat bahan di biologi	175	3	3	0	172	0.000014	Tepat	100%
44	Tampilkan status alat	175	18	18	0	157	0.0000141	Tepat	100%
45	Tampilkan semua data bahan	175	40	40	0	135	0.000019	Tepat	100%

Tabel 4.45 Hasil Uji Coba Pencarian (Lanjutan)

No	Kalimat Pencarian	Jumlah Data	Hasil Pencarian yang benar (TP)	Data Seharusnya yang benar	Hasil Pencarian Salah	Data yang tidak terseleksi dengan benar (TN)	Waktu	Ket	Akurasi
46	Tampilkan tempat bahan di kimia	175	3	3	0	172	0.000018	Tepat	100%
47	Tampilkan tempat bahan di fisika	175	1	1	0	174	0.000011	Tepat	100%
48	Tampilkan tempat bahan di teknik informatika	175	1	1	0	174	0.0000145	Tepat	100%
49	dimana saja letak alat disimpan	175	7	7	0	168	0.0000191	Tepat	100%
50	Tampilkan tempat alat di kimia	175	0	4	3	171	0.000017	Tidak Tepat	97%
<b>Rata-Rata Hasil</b>							<b>0.00016</b>		<b>98%</b>

Berdasarkan hasil ujicoba didapatkan hasil akurasi sebesar 98% pada 50 kali percobaan pencarian. Rata-rata hasil pencarian memberikan *output* yang sesuai. Karena algoritma ini bersifat *Match Case* jadi algoritma *Boyer Moore* dapat memudahkan pencocokan kata pada kata kunci jadi tidak hanya mencari data yang sama persis dengan kata kunci, namun juga mengikutkan data yang mengandung kata kunci, meskipun hasilnya tidak *MatchCase*. Waktu pencarian yang dibutuhkan yang sangat cepat dengan rata-rata waktu sebesar 0.00016 detik.



## BAB V

### KESIMPULAN DAN SARAN

#### 5.2 Kesimpulan

Dari hasil pembahasan yang telah dilakukan maka penelitian dapat diambil kesimpulan sebagai berikut:

- a. Sistem Informasi Manajemen Laboratorium Fakultas Sains dan Teknologi sudah berjalan dengan baik dan sesuai kebutuhan pengguna dengan rata-rata hasil kuesioner menggunakan skala likert sebesar 76% yang menunjukkan responden setuju bahwa sistem ini memudahkan pengguna untuk pengolahan data alat dan bahan praktikum, serta peminjaman alat dan bahan.
- b. Algoritma *boyer moore* pada sistem informasi manajemen laboratorium pada implementasinya di pencarian data memberikan hasil akurasi pencarian sebesar 98 % dari 50 kali percobaan dengan rata-rata waktu pencarian sebesar 0.00016 detik.

#### 5.3 Saran

Setelah berbagai kegiatan yang dilakukan pada penelitian ini, berikut ini adalah beberapa saran sebagai bahan pertimbangan untuk penelitian selanjutnya:

- a. Menambahkan beberapa fitur lain pada sistem informasi manajemen laboratorium sesuai kebutuhan di masa depan.
- b. Untuk meningkatkan akurasi pada proses pencarian dapat membandingkan dengan algoritma yang lain atau mengembangkan algoritma *boyer moore*. Sehingga di hasilkan proses pencarian yang lebih cepat dan tepat.

## DAFTAR PUSTAKA

- A, Hornby. S. (2010). *Oxford Advanced Learner's Dictionary*. Oxford University Press.
- Al-Sheikh, A. b. (1994). *Tafsir Ibnu Katsir Jilid 1*. Bogor: Pustaka Imam asy-Syafi'i.
- Alwin Fau, M. G. (2017). Analisa Perbandingan Boyer Moore Dan Knuth Morris Pratt Dalam Pencarian Judul Buku Menerapkan Metode Perbandingan Eksponensial. *Jurnal Technology Informatics & Computer System (Jurnal TIMES)*, 12-22.
- Anti Damayanti, I. K. (2008). *Manajemen & Teknik Laboratorium*. Yogyakarta: Prodi Biologi, Fakultas Saintek UIN SUKA.
- Assidiq, A. K. (2008). *Kamus Biologi*. Yogyakarta: Panji Pustaka.
- Awsan Abdulrahman Hasan, N. A. (2012). Hash - Boyer-Moore - Horspool String Matching Algorithm for Intrusion Detection System. *IACSIT Press*, 20-25.
- Budiman, A. N. (2001). *Prinsip Prinsip Sistem Informasi Manajemen*. Jakarta: Raja Grafindo Persada.
- Diana Efendy, T. H. Penerapan String Matching Menggunakan Algoritma Boyer Moore Pada Translator Bahasa Pascal ke C. *Majalah Ilmiah UNIKOM, XI*, 262-275.
- Elfraim Turban, L. E. (1992). *Expert System and Applied Artificial Intelligence*. Macmillan Pub. Co.
- Ginting, G. L. (2014). Penerapan Algoritma Boyer Moore pada Aplikasi Pengajuan Judul Skripsi Berbasis Web. *Informasi dan Teknologi Ilmiah (INTI)*, 123-132.
- H Emha Saleh, d. (2002). *Pedoman Penggunaan Laboratorium Sekolah*. Bandung: PT Remaja Rosdakarya.
- Halim Agung, Y. (2016). Implementasi Boyer Moore pada aplikasi pencarian Rumus Matematika dan Fisika. *Jurnal Ilmiah Teknologi Informasi Terapan*, 74-85.
- Imam Sulisty, A. P. (2008). Algoritma Boyer Moore Dalam Pencarian String.
- Jon Orwant, J. H. (1999). *mastering Algorithms*. O'Reilly United States.
- Kadir, A. (2014). *Pengenalan Sistem Informasi Edisi Revisi*. Yogyakarta: Andi.
- Kertiasa, N. (2006). *Laboratorium Sekolah dan Pengelolaannya*. Bandung: Pundak Scientific.

- Koesmaji W, d. (2004). *Teknik Laboratorium*. Bandung: Jurusan Pendidikan Biologi FMIPA.
- Moekijat. (1994). *Pengantar Sistem Informasi Manajemen*. Bandung: Remaja Rosdakarya.
- Munir, R. (2011). *Algoritma dan Pemrograman dalam Bahasa Pascal dan C (Edisi Revisi)*. Bandung: Informatika Bandung.
- Nuryani, R. (2005). *Strategi Belajar Mengajar Biologi*. Malang: Penerbit Universitas Negeri Malang.
- Prabhakar Gupta, V. A. (2012). *Design and Analysis of Algorithms*. New Delhi: PHI Learning Private Limited.
- Press, C. U. (2008). *Cambridge Advanced Learner's Dictionary*. Singapore: Green Gian.
- Prima Fithri, M., & Berlian, Y. (2015). Pengelolaan Persediaan Barang Suku Cadang (Spare Parts) Pada Gudang PT Semen Padang. *Jurnal Ilmiah Teknik Industri*, 1-42.
- Rahmanita, E. (2014). Pencarian String menggunakan Algoritma Boyer Moore pada Dokumen. *Jurnal Imiah NERO*, 15-26.
- Retno Sari, T. R. (2017). *Aplikasi Sistem Informasi dan Manajemen Laboratorium*. Jakarta: Kementerian Kesehatan Republik Indonesia.
- T.Sutojo, E. M. (2011). *Kecerdasan Buatan*. Yogyakarta: ANDI.
- Tambun, E. D. (2010). *Perbandingan Penggunaan Algoritma BM dan Algoritma Horspool pada Pencarian String dalam Bahasa Medis*.
- Vina Sagita, M. I. (2013). Studi Perbandingan Implementasi Algoritma Boyer-Moore, Turbo Boyer-Moore, dan Tuned Boyer-Moore dalam Pencarian String. *ULTIMATICS*, 31-37.
- Wikipedia. (2018, June 27). *Receiver Operating Characteristic*. Retrieved June 5, 2018, from Wikipedia the free encyclopedia:  
[https://en.wikipedia.org/wiki/Receiver\\_operating\\_characteristic](https://en.wikipedia.org/wiki/Receiver_operating_characteristic)

## LAMPIRAN 1

### KUESIONER PENELITIAN SISTEM INFORMASI MANAJEMEN LABORATORIUM FAKULTAS SAINS DAN TEKNOLOGI UIN MAULANA MALIK IBRAHIM MALANG

Nama :

No Identitas :

Jabatan :

Berikan tanda silang (x) pada nilai yang di anggap sesuai

1 : Sangat Tidak Setuju

2 : Tidak Setuju

3 : Netral

4 : Setuju

5 : Sangat Setuju

No	Pertanyaan	Nilai				
		1	2	3	4	5
1	Apakah informasi yang disediakan oleh aplikasi mudah dimengerti?					
2	Apakah penggunaan menu atau fitur aplikasi menu mudah digunakan?					
3	Apakah aplikasi nyaman digunakan?					
4	Secara keseluruhan apakah penggunaan aplikasi ini memuaskan?					
5	Apakah aplikasi ini sesuai dengan kebutuhan?					
6	Apakah aplikasi dapat dengan mudah dipelajari?					
7	Apakah aplikasi mudah dioperasikan?					
8	Apakah dapat dengan mudah menghindari kesalahan dalam menggunakan aplikasi?					
9	Apakah aplikasi bermanfaat bagi pengguna?					
10	Apakah tampilan menu dalam aplikasi mudah untuk dikenali?					

11	Apakah aplikasi mempunyai kemampuan dan fungsi sesuai yang diharapkan?					
12	Apakah dengan adanya aplikasi ini membantu pengelolaan data menjadi lebih mudah?					
13	Apakah dengan adanya aplikasi ini memberikan informasi data secara akurat?					
14	Apakah tampilan website cukup menarik dan mudah dipahami (user friendly)?					
15	Apakah dengan adanya aplikasi ini dapat menghasilkan laporan yang tepat seperti yang dibutuhkan?					
16	Apakah dengan adanya aplikasi ini dapat memperoleh informasi yang dibutuhkan dengan tepat waktu?					
17	Apakah aplikasi ini mampu memberikan informasi sesuai dengan format yang dibutuhkan?					
18	Apakah aplikasi ini mampu menghasilkan informasi yang dapat dipahami secara jelas?					
19	Apakah aplikasi mudah dipelajari oleh orang yang baru pertama kali digunakan?					
20	Apakah aplikasi mampu menghasilkan informasi yang bersifat mutakhir/ <i>up to date</i> ?					