

Original Research Paper

Penerapan Algoritma Knuth-Morris-Pratt pada Fungsi Pencarian Dokumen untuk Sistem Informasi Administrasi Sekolah Berbasis Website

Imam Maulana¹, Normalisa¹

¹ Program Studi Teknik Informatika, Fakultas Teknik, Universitas Pamulang, Tangerang Selatan, Indonesia

Article History

Received:
09.01.2019

Revised:
20.02.2019

Accepted:
03.03.2019

*Corresponding Author:

Normalisa
Email:
dosen00377@unpam.ac.id

This is an open access article,
licensed under: [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/)



Abstrak: Algoritma Knuth-Morris-Pratt merupakan algoritma pencarian yang akan mencocokkan pattern atau susunan kata yang akan dicari dari kiri ke kanan pada awal teks dan kemudian menggeser susunan kata sampai susunan kata tersebut berada di ujung teks. Algoritma KMP memiliki keunggulan pencarian kecocokan pada file yang berukuran besar. Metode pengembangan sistem yang digunakan yaitu metode Rapid Application Development. Metode RAD ini adalah strategi siklus hidup yang ditujukan untuk menyediakan pengembangan yang jauh lebih cepat dan mendapatkan hasil dengan kualitas yang lebih baik dibandingkan dengan hasil yang dicapai melalui siklus tradisional. Aplikasi Sistem Informasi Administrasi Sekolah ini diharapkan dapat mengatasi permasalahan yang timbul dari proses pengendalian dokumen secara konvensional agar lebih mudah dan tepat.

Kata Kunci: Knuth-Morris-Pratt, Pencarian Dokumen, Rapid Application Development.

Applying of Knuth-Morris-Pratt Algorithm to the Document Search Function for Website-Based School Administration Information Systems

Abstract: Knuth-Morris-Pratt algorithm is a search algorithm that will match the pattern or arrangement of words to be searched from left to right at the beginning of the text and then shift the order of words until the word order is at the end of the text. The KMP algorithm has the advantage of matching matches on large files. The system development method used is the Rapid Application Development method. RAD method is a life cycle strategy aimed at providing development that is much faster and gets results with better quality compared to results achieved through traditional cycles. Application of School Administration Information System is expected to be able to overcome problems arising from conventional document control processes to make it easier and more precise.

Keywords: Document Search, Knuth-Morris-Pratt, Rapid Application Development.



1. Pendahuluan

Saat ini proses pengendalian dokumen pada SMK Teknologi Informatika YPML masih dilakukan secara konvensional yaitu dokumen dibuat dengan menggunakan program pengolah kata, kemudian dokumen dicetak untuk selanjutnya dikelola sesuai dengan prosedur pengendalian dokumen yang ada. Dokumen yang dikelola akan terus bertambah seiring berjalannya waktu sehingga menyimpan dokumen secara konvensional di dalam map arsip tentu akan menyulitkan ketika mencari dokumen yang sedang dibutuhkan tentunya membuat pekerjaan tenaga administrasi sekolah tidak efektif dan efisien. Selain itu, penyimpanan dokumen secara konvensional tanpa adanya salinan secara elektronik memiliki resiko seperti kehilangan dan kerusakan dokumen.

Solusi untuk mengatasi masalah tersebut yaitu diperlukan sistem pengelolaan dokumen yang terkomputerisasi sehingga dapat membantu proses pengendalian dokumen pada SMK Teknologi Informatika YPML. Fungsi pencarian dokumen pada sistem informasi administrasi sekolah menggunakan algoritma Knuth-Morris-Pratt. Algoritma Knuth-Morris-Pratt (KMP) ini merupakan algoritma pencarian String untuk mencari teks berdasarkan urutan dari kiri ke kanan. Algoritma KMP akan mencocokkan pattern atau susunan kata yang akan dicari dari kiri ke kanan pada awal teks dan kemudian menggeser susunan kata sampai susunan kata tersebut berada di ujung teks [1]. Algoritma KMP memiliki keunggulan pencarian kecocokan pada file yang berukuran besar [2].

Metode pengembangan sistem yang digunakan yaitu metode Rapid Application Development (RAD) yang memiliki strategi siklus hidup yang ditujukan untuk menyediakan pengembangan yang jauh lebih cepat dan mendapatkan hasil dengan kualitas yang lebih baik dibandingkan dengan hasil yang dicapai melalui siklus tradisional [3]. Pemodelan sistem menggunakan Unified Modeling Language (UML) dikarenakan UML menyediakan bahasa pemodelan visual yang memungkinkan bagi pengembang sistem untuk membuat cetak biru atas visi mereka dalam bentuk yang baku, mudah dimengerti, serta dilengkapi dengan mekanisme yang efektif untuk berbagi dan mengkomunikasikan rancangan mereka dengan yang lain [4]. Selanjutnya, rancangan sistem tersebut akan diimplementasikan ke dalam aplikasi berbasis website dengan menggunakan bahasa pemrograman PHP dan database menggunakan MySQL.

2. Tinjauan Pustaka

2.1. Algoritma String Matching

Algoritma *String Matching* adalah sebuah algoritma yang digunakan dalam pencocokan suatu pola kata tertentu terhadap suatu kalimat atau teks panjang. Algoritma *string matching* sendiri dapat dilakukan dengan beberapa cara tertentu, antara lain cara *Brute Force* dan cara *Knuth-Morris-Pratt* (KMP) [1].

2.2. Algoritma Knuth-Morris-Pratt

Algoritma *Knuth-Morris-Pratt* dikembangkan secara terpisah oleh Donald E. Knuth pada tahun 1967 dan James H. Morris bersama Vaughan R. Pratt pada tahun 1966, namun keduanya mempublikasikannya secara bersamaan pada tahun 1977 [5]. Algoritma *Knuth-Morris-Pratt* dilakukan dengan cara menghitung fungsi pinggiran dari pola terlebih dulu dan kemudian akan dilakukan perbandingan antara pola dan elemen pertama dari kalimat, jika tidak sesuai, maka perbandingan tidak dilakukan pada elemen kedua, namun tergantung dari nilai yang akan dikeluarkan oleh fungsi pinggiran tersebut [1].

2.3. Dokumen

Dokumen adalah surat yang tertulis atau tercetak yang dapat dipakai sebagai bukti keterangan, barang cetakan atau naskah karangan yang dikirim melalui pos, atau rekaman suara, gambar dalam film, dan sebagainya yang dapat dijadikan bukti keterangan. Dokumen adalah informasi yang dikumpulkan dan bisa diakses serta digunakan [6].

2.4. Konsep Dasar Sistem

Sistem merupakan bagian yang saling berkaitan erat dan membentuk suatu kesatuan yang saling berinteraksi antara bagian satu dengan bagian lainnya untuk mencapai suatu tujuan, artinya apabila salah satu bagian dari sistem tidak ada maka sistem tersebut tidak berfungsi sebagaimana mestinya. Suatu sistem dapat terdiri dari bagian-bagian sistem (subsistem) yang saling berinteraksi, sebagai akibat dari adanya input yang diproses menjadi output / informasi, misalnya sebuah komputer terdiri

dari beberapa komponen [7].

2.5. Sistem Informasi

Sistem informasi dapat merupakan kombinasi teratur dari orang-orang, hardware, software, jaringan komputer dan sumber daya data yang mengumpulkan, mengubah, dan menyebarkan informasi dalam sebuah organisasi [8]. Sistem informasi adalah sebuah sistem yang menyajikan informasi guna mendukung fungsi operasi, manajemen, dan pengambilan keputusan dalam sebuah organisasi [9]. Tujuan dari sistem informasi itu adalah menyajikan informasi untuk mengambil keputusan pada perencanaan, pengorganisasian, pengendalian kegiatan operasi subsistem pada perusahaan dan menyajikan sinergi organisasi [9].

Sistem informasi adalah suatu sistem di dalam organisasi yang mempertemukan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategic suatu organisasi, serta menyediakan pihak luar tertentu dengan laporan-laporan yang dibutuhkan [10]. Sistem informasi adalah suatu sistem di dalam organisasi yang merupakan kombinasi dari orang-orang, teknologi, fasilitas, prosedur dan pengendalian yang ditujukan untuk mengolah data menjadi informasi yang penting sehingga dapat digunakan sebagai pengambilan keputusan [9].

3. Metode Penelitian

3.1. Analisa Sistem

Analisa sistem merupakan kegiatan penguraian suatu sistem informasi yang utuh dan nyata kedalam bagian-bagian atau kelompok komponen-komponen yang bertujuan untuk mengidentifikasi serta mengevaluasi masalah-masalah yang muncul, hambatan-hambatan yang sering terjadi, serta kebutuhan yang diharapkan, sehingga dapat memberikan solusi untuk perbaikan maupun pengembangan kearah yang lebih baik dan sesuai dengan kebutuhan pengguna dan perkembangan teknologi.

3.2. Analisa Sistem Saat Ini

Analisis sistem saat ini diketahui melalui kegiatan wawancara dengan perwakilan SMK Teknologi Informatika YPML serta observasi langsung di lapangan, prosedur pengelolaan dan pengendalian dokumen yang meliputi kegiatan pembuatan, penerbitan, pendistribusian, pemeliharaan, pengubahan, dan pemusnahan dokumen. *Activity diagram* proses pembuatan, penerbitan dan pendistribusian dokumen pada sistem saat ini dapat dilihat pada Gambar 1.

3.3. Analisa Proses Pemeliharaan Dokumen

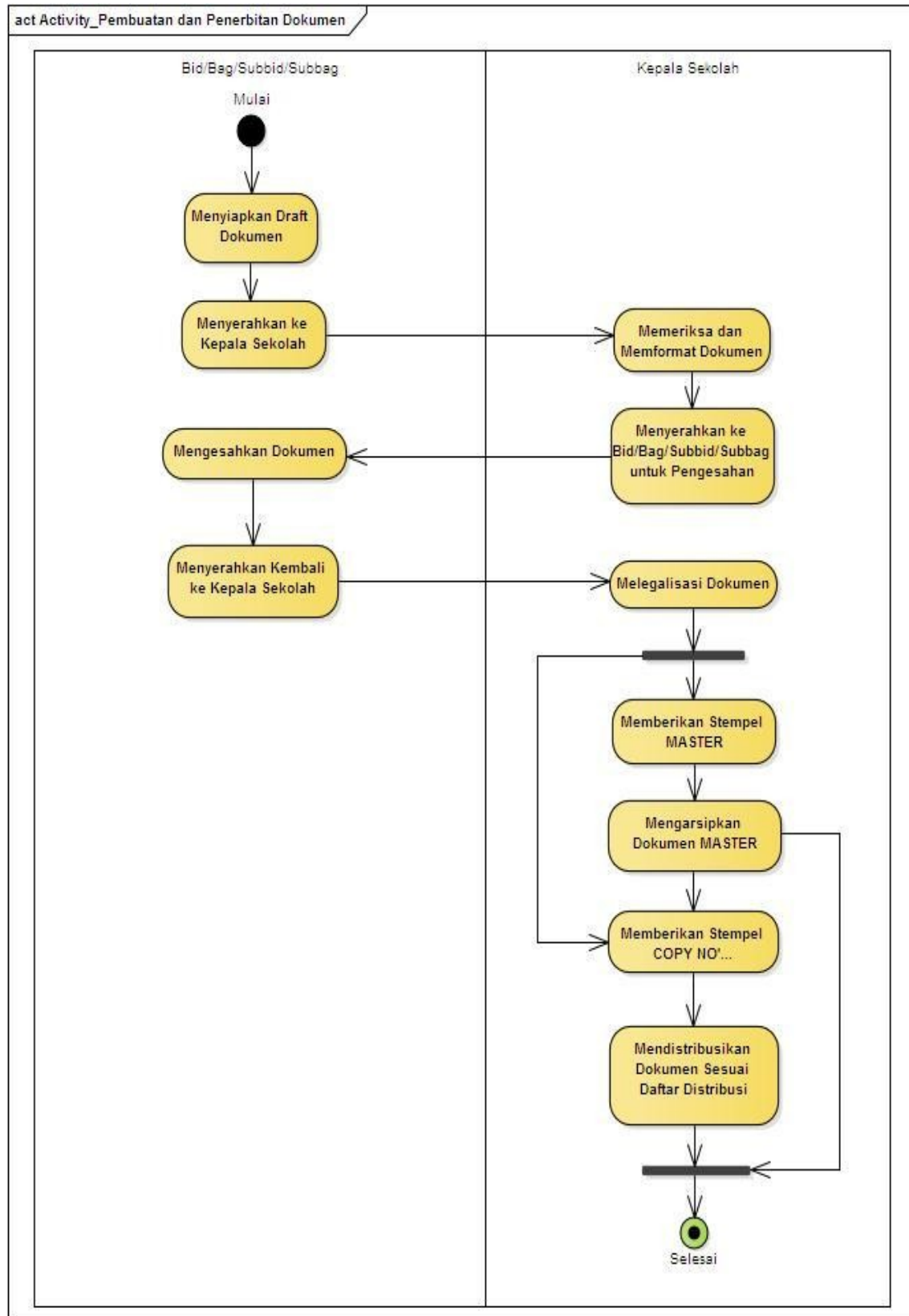
Kepala sekolah secara berkala mengidentifikasi kesesuaian semua dokumen yang digunakan dalam daftar induk termasuk dokumen *eksternal*. Apabila ditemukan dokumen yang tidak sesuai dan atau sudah tidak berlaku, maka kepala sekolah menarik dokumen tersebut, memasukkan ke dalam folder tidak berlaku atau kadaluwarsa dan diganti dengan dokumen yang sesuai. Kepala sekolah menyerahkan dokumen yang sudah tidak berlaku atau kadaluwarsa ke Sub-bagian yang berhubungan dengan penyimpanan atau pemusnahan sesuai dengan jadwal arsip sekolah. *Activity diagram* pemeliharaan dokumen yang dilakukan pada sistem saat ini dapat dilihat pada Gambar 2.

3.4. Analisa Proses Pengubahan Dokumen

Proses pengubahan yang bersifat amandemen, dibuat dalam lembar amandemen sesuai Formulir Amandemen. Kepala sekolah menggandakan hasil amandemen, diberi cap copy dan no, dan mendistribusikannya sesuai daftar distribusi dokumen untuk disisipkan pada dokumen yang diamandemen.

3.5. Analisa Proses Pemusnahan Dokumen

Pemusnahan dokumen dapat dilakukan dengan cara dibakar, dicacah atau dijadikan bubur kertas, atau cara lain sehingga fisik dan informasinya tidak dapat dikenali lagi.



Gambar 1. Activity Diagram Pembuatan, Penerbitan dan Pendistribusian Dokumen

3.6. Evaluasi Sistem Saat Ini

Setelah dilakukan observasi, ditemukan beberapa hal yang terjadi diantaranya:

1. Belum adanya sistem terkomputerisasi yang digunakan sebagai rekomendasi kepada kepala sekolah dalam hal melakukan pengendalian dokumen.
2. Belum adanya database terkomputerisasi sehingga beresiko terjadi kerusakan atau kehilangan ketika menyimpan dokumen di dalam map arsip.
3. Belum adanya metode pencarian dokumen untuk membantu dalam mencari dokumen yang dibutuhkan.

3.7. Analisa Sistem yang Diusulkan

Aplikasi Sistem Informasi Administrasi Sekolah yang dapat membantu SMK Teknologi Informatika YPML untuk melakukan pengelolaan dan pengendalian dokumen adalah sebagai berikut:

1. Dibuatnya sistem informasi untuk memudahkan pengendalian dokumen SMK Teknologi Informatika YPML.
2. Dibangunnya struktur *database* yang mampu menyimpan data dokumen secara digital sebagai backup apabila dokumen fisik hilang atau rusak.
3. Proses pencarian dokumen menggunakan algoritma *Knuth-Morris-Pratt* (KMP) agar proses menjadi lebih tepat.

3.8. Perancangan Sistem

Pada bagian ini dijelaskan tentang perancangan sistem meliputi perancangan basis data (*database*), perancangan alur sistem keseluruhan dan perancangan tampilan user interface.

3.8.1. Perancangan Basis Data

Perancangan basis data meliputi perancangan Entity Relationship Diagram (ERD), Transformasi ERD ke LRS, dan Logical Record Structure (LRS). *Entity Relationship Diagram* disajikan pada Gambar 5.

Transformasi ERD ke LRS disajikan pada Gambar 6. Setelah ditransformasikan ERD ke LRS, maka bentuk *Logical Record Structure (LRS)* yang sudah terbentuk seperti dapat dilihat pada Gambar 7.

3.8.2. Spesifikasi Basis Data

Spesifikasi basis data merupakan penguraian field-field dari setiap tabel yang meliputi, nama field, tipe data, panjang data, dan keterangan. Spesifikasi basis data meliputi data user, Organisasi, Pangkat Golongan, Dokumen, Autentikasi Dokumen, Jenis Dokumen, Daftar Induk Dokumen, File Distribusi, dan Usulan Dokumen.

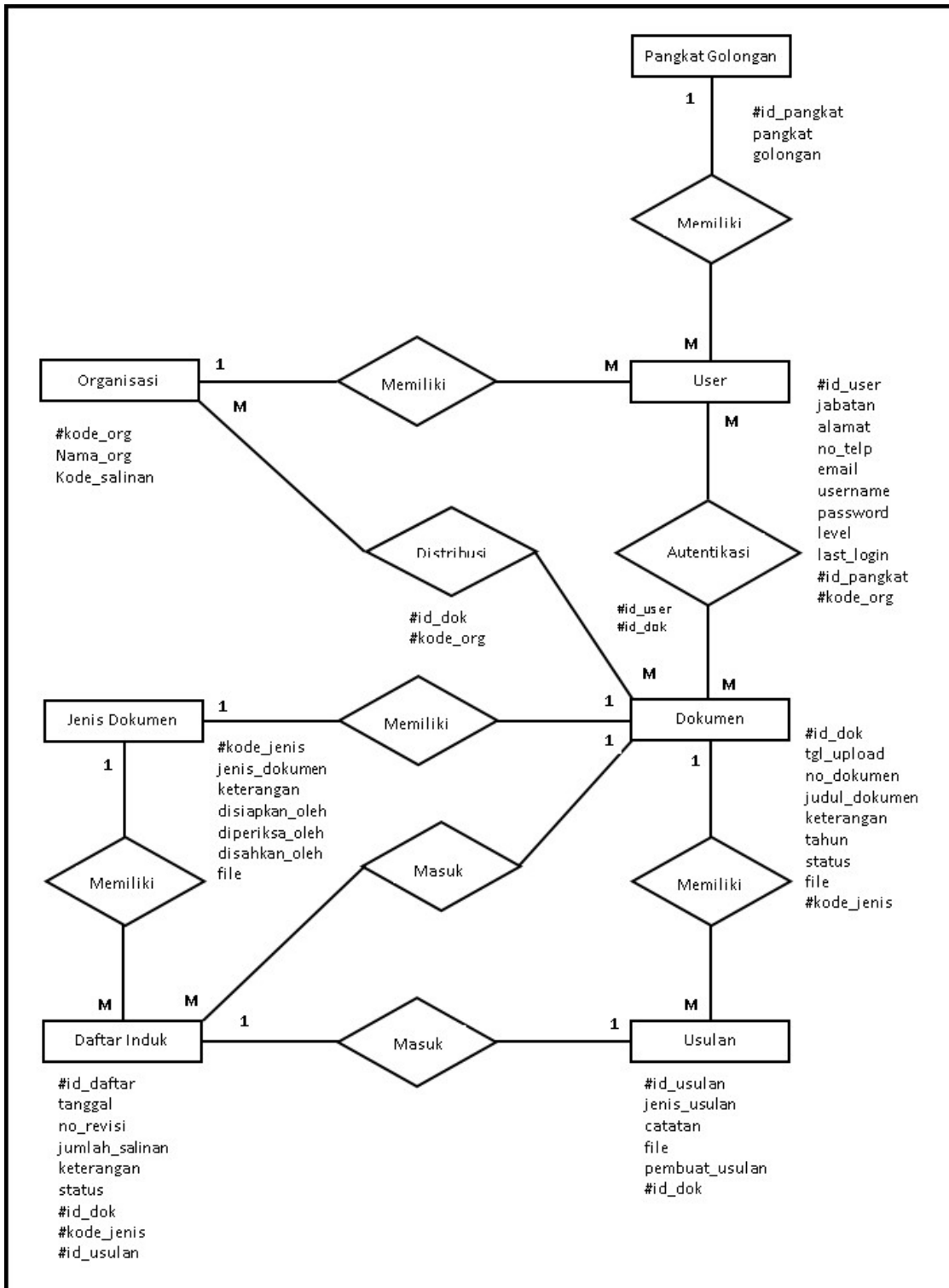
Contoh spesifikasi data User dapat dilihat pada Tabel 1.

3.9. Perancangan Aplikasi

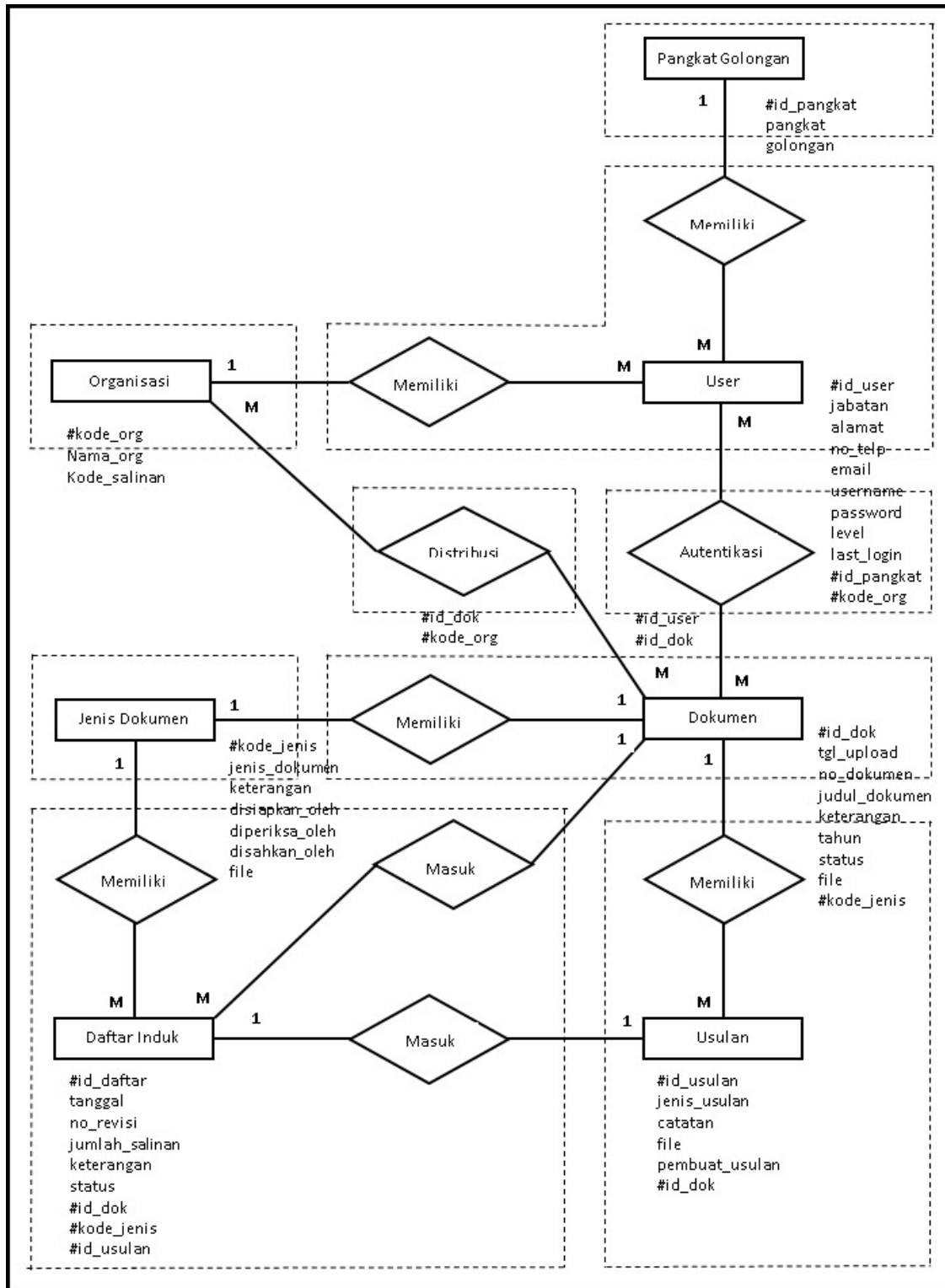
3.9.1. Use Case Diagram

Use case diagram dari sistem yang dirancang dan dibangun disajikan pada Gambar 8. Deskripsi Use Case sebagai berikut:

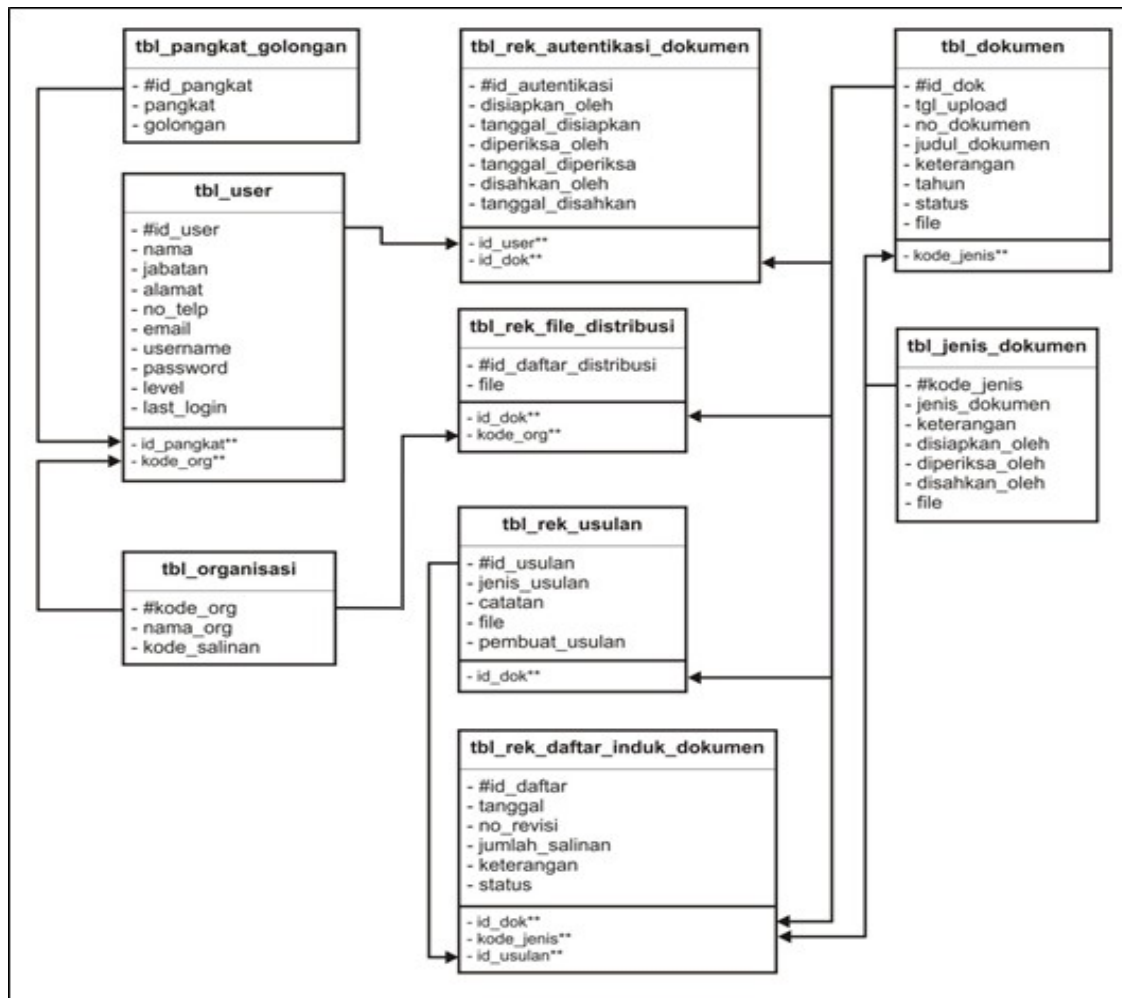
1. *Users* (Admin, User) melakukan *login* pada Sistem Informasi Administrasi Sekolah.
2. *Users* dapat memilih menu Tambah Dokumen untuk menambah dokumen baru.
3. User melakukan Autentikasi Dokumen berupa penyiapan dokumen, pemeriksaan dokumen, dan pengesahan dokumen sesuai wewenang User.
4. Admin melakukan Kendali Dokumen yang telah disiapkan, diperiksa, dan disahkan oleh user yang berwenang.
5. *Users* dapat membuat Usulan Dokumen berupa amandemen, maupun pemusnahan dokumen yang tidak berlaku.
6. *Users* dapat Melihat Dokumen Terkendali yang telah dipublikasi oleh Admin.
7. Admin dapat Mengelola Data *Users* yang menggunakan Sistem Informasi Administrasi Sekolah.
8. Admin dapat Mengelola Data Template Dokumen yang akan diunduh oleh *Users*.
9. *Users* dapat melakukan pencarian data dengan algoritma *Knuth-Morris-Pratt*.
10. *Users* dapat melakukan Logout untuk keluar dari Sistem Informasi Administrasi Sekolah.



Gambar 2. Entity Relationship Diagram



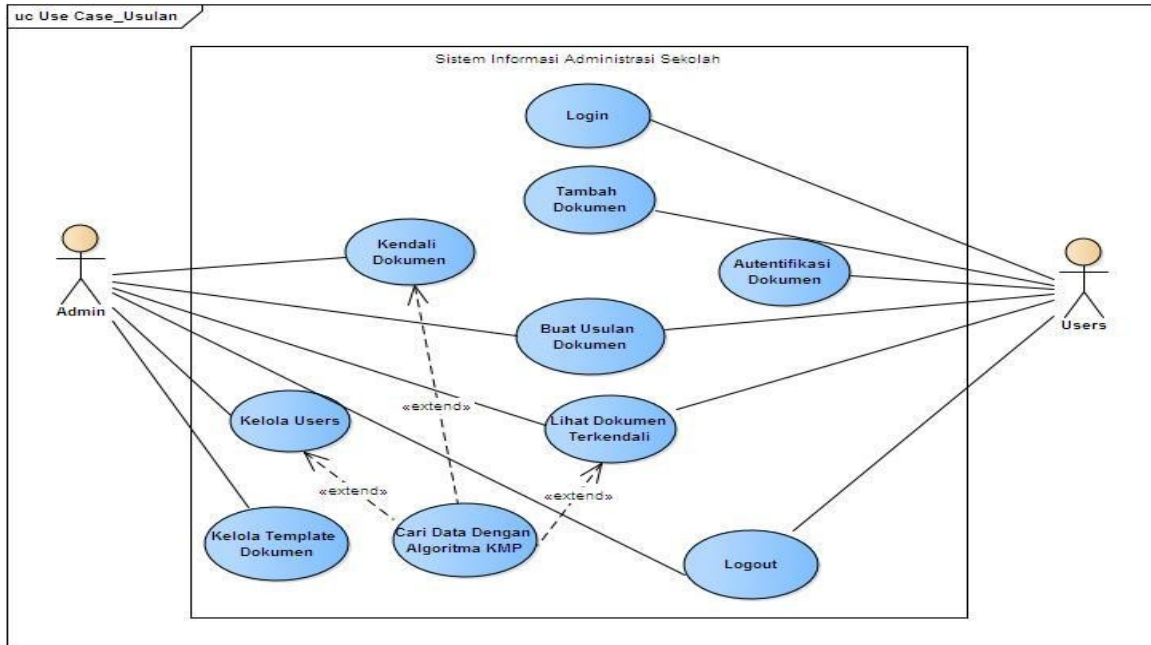
Gambar 3. Transformasi ERD ke LRS



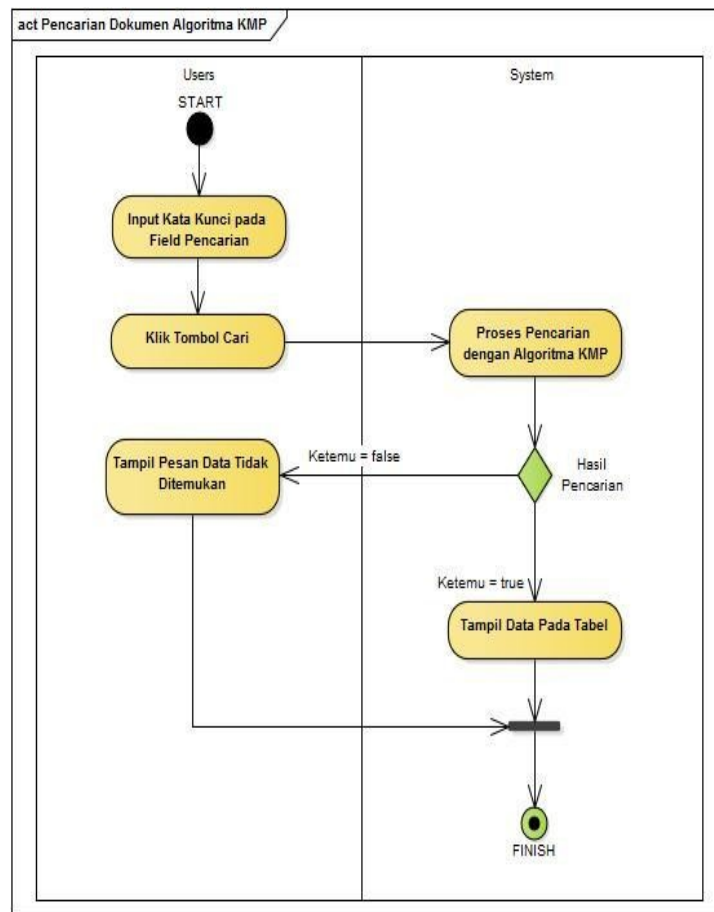
Gambar 4. Logical Record Structure

Tabel 1. Tabel Database User

Field	Type	Length	Indeks
id_user	Int	11	Primary Key
Nama	Varchar	50	-
Jabatan	Enum	Kepala, Staff	-
Alamat	text	-	-
no_telp	Varchar	13	-
Email	Varchar	30	-
Username	Varchar	50	Unique
Password	Varchar	255	-
Level	Enum	Admin, User	-
last_login	Varchar	30	-
kode_org	Varchar	30	Foreign Key
id_pangkat	Int	11	Foreign Key



Gambar 5. Use Case Diagram



Gambar 6. Activity Diagram Pencarian Data Algoritma KMP

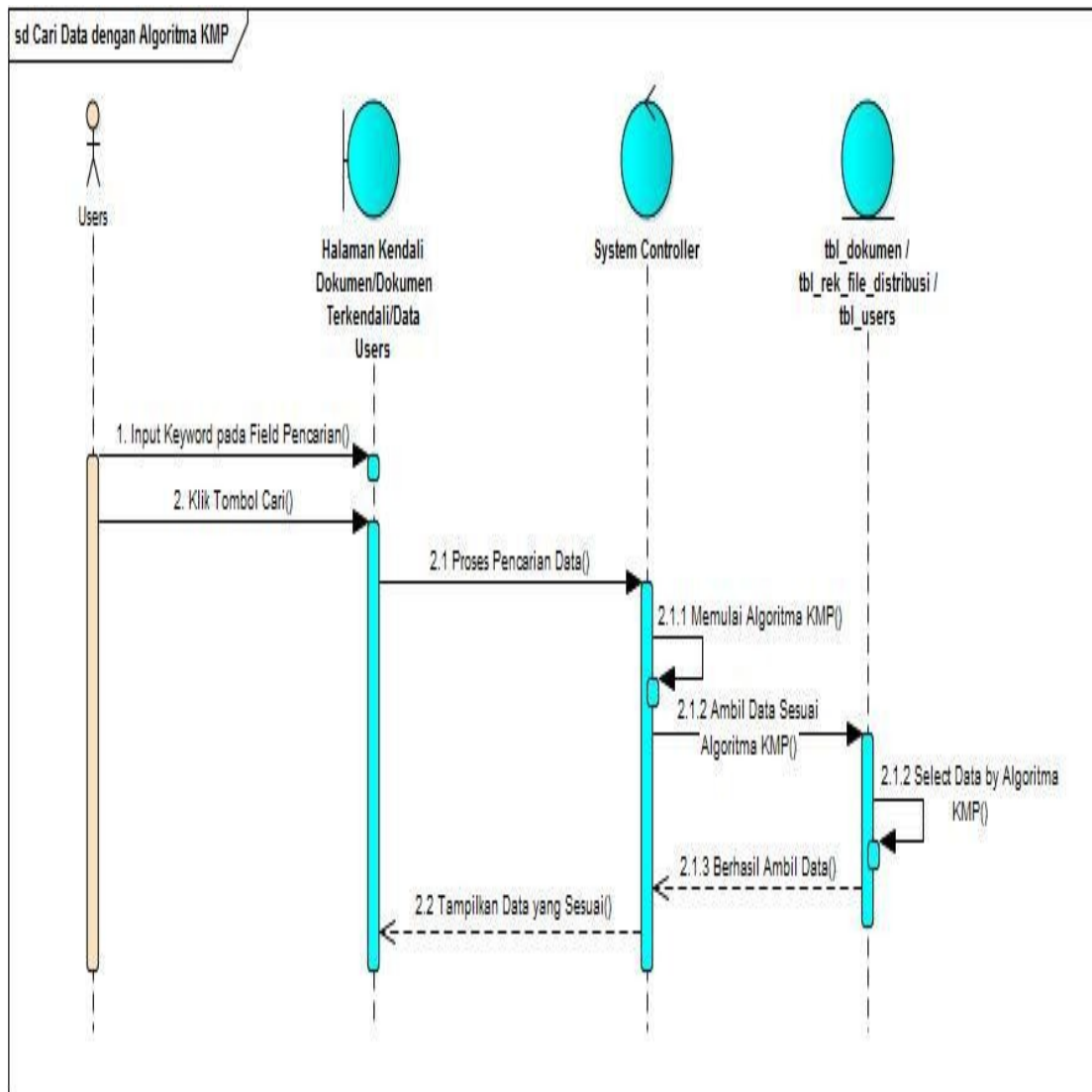
3.9.2. Activity Diagram

Beberapa *activity diagram* pada penelitian ini adalah *activity diagram* Login dan Tambah Dokumen, Autentikasi Dokumen dan Kendali Dokumen, Lihat Dokumen Terkendali dan Pencarian Data Algoritma KMP, Usulan Dokumen dan Kelola User, Kelola Template Dokumen dan Logout.

Activity diagram pencarian data algoritma KMP disajikan pada Gambar 6.

3.9.3. Sequence Diagram

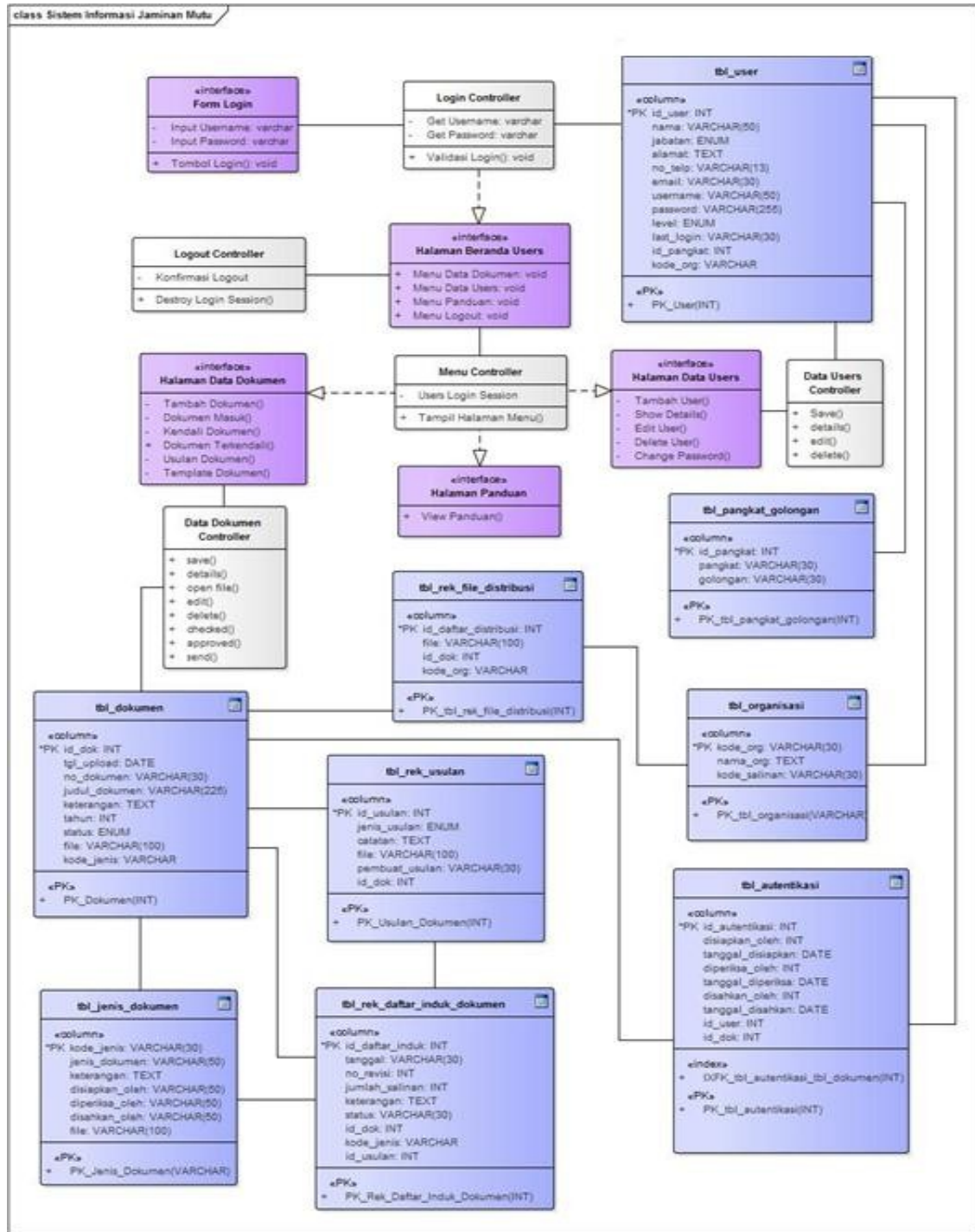
Diagram sequence merupakan penjelasan secara detail dari masing-masing *activity diagram* yang telah dijelaskan sebelumnya. Contoh *sequence diagram* Pencarian Data Algoritma KMP dapat dilihat pada Gambar 7.



Gambar 7. Sequence Diagram Cari Data dengan Algoritma

3.9.4. Class Diagram

Class Diagram struktur rancangan sistem yang akan diterapkan dalam Sistem Informasi Administrasi Sekolah disajikan pada Gambar 8.



Gambar 8. Class Diagram

3.10. Analisa Algoritma Knuth-Morris-Pratt

Algoritma yang digunakan untuk pencocokan *string* pada fungsi pencarian Sistem Informasi Administrasi Sekolah yaitu menggunakan algoritma *Knuth-Morris-Pratt* (KMP). Algoritma KMP merupakan algoritma yang dikembangkan secara terpisah oleh D. E. Knuth pada tahun 1967, dan J.H. Morris bersama V.R. Pratt pada tahun 1966, namun keduanya mempublikasikan secara bersamaan

pada tahun 1977 [4].

Pseudocode algoritma KMP.

Fase Pra-Pencarian	Fase Pencarian
<pre> Procedure preKMP (input P : array[0...n-1] of char, input n : integer, input/output kmpNext : array[0...n] of integer) Deklarasi : i,j : integer Algoritma i := 0; j := kmpNext(0) := -1; while (i < n) { while (j > -1 and not (P[i] = P[j])) j := kmpNext[j]; i := i+1; j := j+1; if (p[i] = P[j]) kmpNext[i] := kmpNext [j]; else kmpNext[i] := j; endif endwhile </pre>	<pre> Procedure KMPSearch(input m, n : integer, input P : array[0...n-1] of char, input T : array[0...m-1] of char, output ketemu : array[0...m- 1 of boolean) Deklarasi: i,j,next : integer kmpNext : array[0...n] of integer Algoritma: preKMP(n,P,kmpNext) i:=0 while (i<=m-n) do j:=0 while (j<n and T[i+j] = P[j]) do j:=j+1 endwhile if(j>=n) then ketemu[j]:=true; endif next:=j-kmpNext[j] i:=i+next endwhile </pre>

Berdasarkan *pseudocode* tersebut, dapat diketahui langkah-langkah yang dilakukan algoritma *Knuth-Morris-Pratt* pada saat mencocokkan *string* adalah sebagai berikut:

Algoritma *Knuth-Morris-Pratt* mulai mencocokkan *pattern* pada awal teks.

1. Dari kiri ke kanan, algoritma ini akan mencocokkan karakter per karakter *pattern* dengan karakter di teks yang bersesuaian, sampai salah satu kondisi berikut dipenuhi adalah Karakter di *pattern* dan di teks yang dibandingkan tidak cocok (*mismatch*) serta Semua karakter di *pattern* cocok. Kemudian algoritma akan memberitahukan penemuan di posisi ini.
2. Algoritma kemudian menggeser *pattern* berdasarkan tabel *next*, lalu mengulangi langkah b sampai *pattern* berada di ujung teks. Penerapan algoritma *Knuth-Morris-Pratt* pada fungsi pencarian dokumen untuk Sistem Informasi Administrasi Sekolah dijelaskan dalam beberapa langkah berikut:

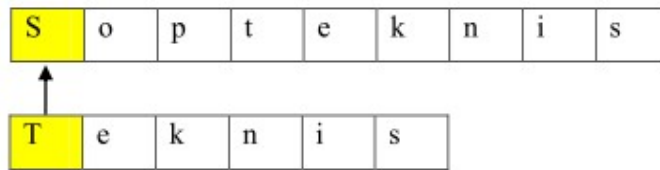
String S:

S	o	p	t	e	k	n	i	s
---	---	---	---	---	---	---	---	---

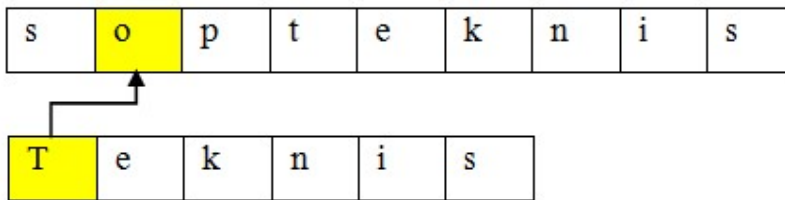
Pattern P:

T e k n i s

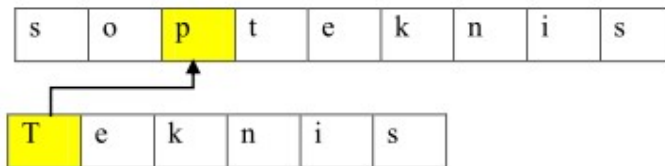
Langkah 1, bandingkan P(1) dengan S(1)



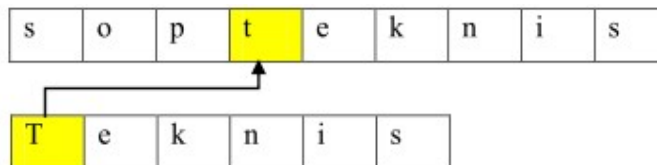
P(1) tidak cocok dengan S(1), maka pattern akan bergeser satu posisi ke kanan. Langkah 2, bandingkan P(1) dengan S(2)



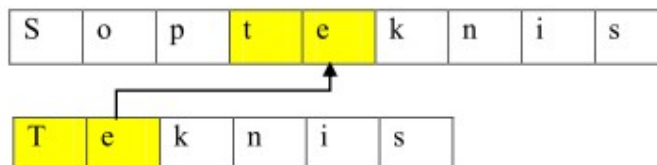
P(1) tidak cocok dengan S(2), maka pattern akan bergeser satu posisi ke kanan. Langkah 3, bandingkan P(1) dengan S(3)



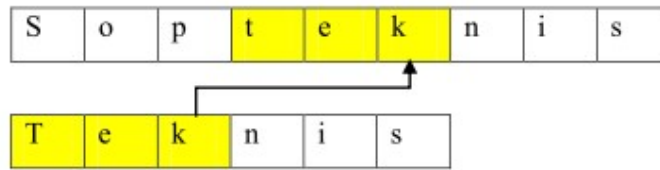
P(1) tidak cocok dengan S(3), maka pattern akan bergeser satu posisi ke kanan. Langkah 4, bandingkan P(1) dengan S(4)



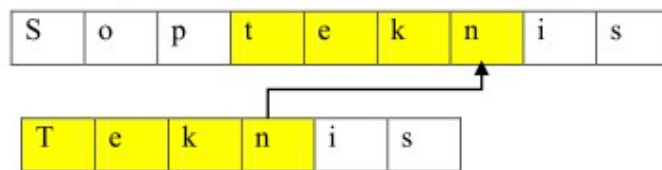
P(1) cocok dengan S(4), maka algoritma Knuth-Morris-Pratt akan menyimpan informasi ini, dan pattern tidak akan melakukan pergeseran dan melanjutkan pencocokan P(2) dengan S(5). Langkah 5, bandingkan P(2) dengan S(5)



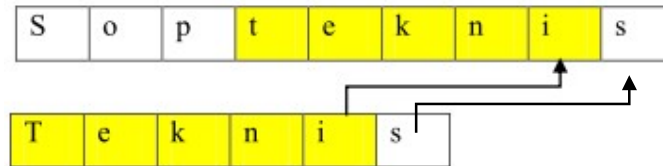
$P(2)$ cocok dengan $S(5)$, maka algoritma *Knuth-Morris-Pratt* akan menyimpan informasi ini, dan *pattern* tidak akan melakukan pergeseran dan melanjutkan pencocokan $P(3)$ dengan $S(6)$. Langkah 6, bandingkan $P(3)$ dengan $S(6)$



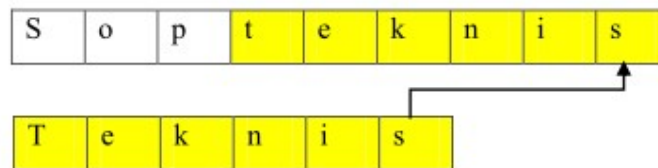
$P(3)$ cocok dengan $S(6)$ maka algoritma *Knuth-Morris-Pratt* akan menyimpan informasi ini, dan *pattern* tidak akan melakukan pergeseran dan melanjutkan pencocokan $P(4)$ dengan $S(7)$. Langkah 7, bandingkan $P(4)$ dengan $S(7)$



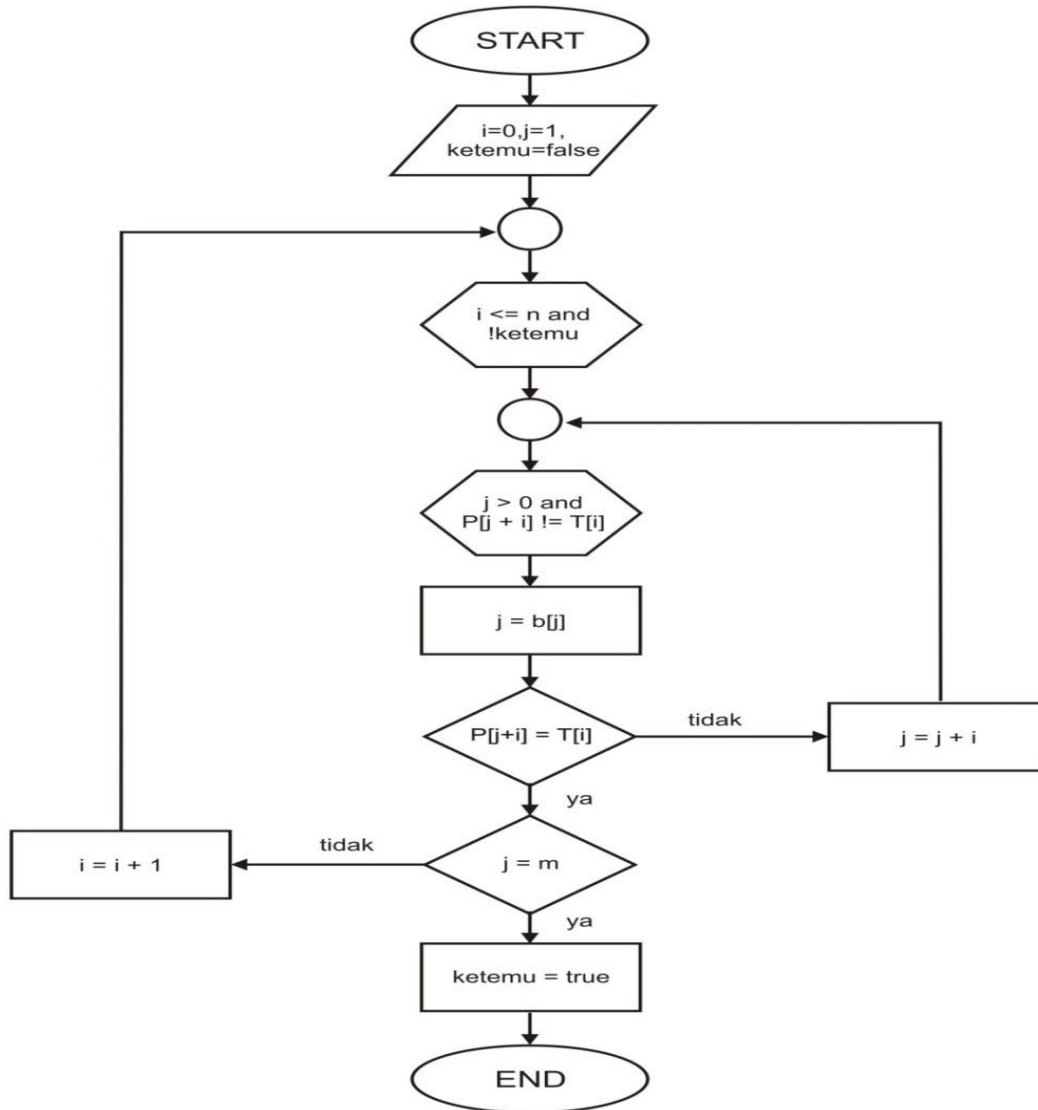
$P(4)$ cocok dengan $S(7)$ maka algoritma *Knuth-Morris-Pratt* akan menyimpan informasi ini, dan *pattern* tidak akan melakukan pergeseran dan melanjutkan pencocokan $P(5)$ dengan $S(8)$. Langkah 8, bandingkan $P(5)$ dengan $S(8)$.



$P(5)$ cocok dengan $S(8)$ maka algoritma *Knuth-Morris-Pratt* akan menyimpan informasi ini, dan *pattern* tidak akan melakukan pergeseran dan melanjutkan pencocokan $P(6)$ dengan $S(9)$



$P(6)$ cocok dengan $S(9)$, maka algoritma *Knuth-Morris-Pratt* akan menyimpan informasi ini, dan *pattern* tidak akan melakukan pergeseran dan melanjutkan pencocokan $P(7)$ dengan $S(10)$. Namun karena jumlah *pattern* hanya 6 huruf, maka pencarian akan dihentikan dan diperoleh hasil bahwa *pattern* P terdapat kecocokan dengan *string* S sebesar 100 persen. Jika semua huruf pada *pattern* sudah dicocokkan dengan *string*, maka akan ditemukan sebuah pola kosakata di dalam *string*. Dalam menemukan sebuah pola *pattern* di dalam *string* akan dilakukan pergeseran beberapa kali untuk mencocokkan setiap huruf pada *pattern* yang dimulai dari sebelah kiri untuk mencocokkan setiap huruf pada *string*. *Flowchart* Algoritma *Knuth-Morris-Pratt* disajikan pada Gambar 24.



Gambar 9. Flowchart Algoritma Knuth-Morris-Pratt

4. Implementasi dan Pengujian

4.1. Implementasi Perangkat Keras

Website yang telah dirancang maka dibutuhkan perangkat keras sebagai tempat untuk menerapkannya. Adapun Perangkat keras yang dibutuhkan disajikan pada Tabel 2.

Tabel 2. Perangkat Keras

No.	Nama	Spesifikasi Minimum
1.	Processor	AMD C-70 1.00 GHz (Atau setara)
2.	RAM	2 GB
3.	Monitor	11.3" Radeon™ HD Graphics

4.2. Implementasi Perangkat Lunak

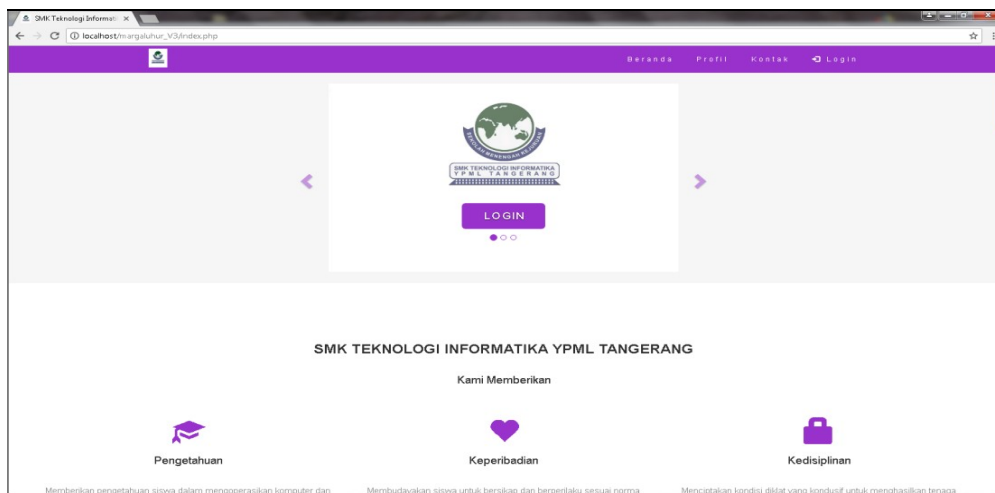
Perangkat lunak yang digunakan untuk mengimplementasikan sistem disajikan pada Tabel 3.

Tabel 3. Perangkat Lunak

No.	Tools	Software Pendukung
1.	Web Browser	Mozilla Firefox 52.0.1 (32 bit)
2.	XAMPP	XAMPP for Windows 5.6.14
3.	PHP	PHP Version 5.6.14
4.	Database	MariaDB (MySQLi Extentions)

4.3. Implementasi Antarmuka

Implementasi antarmuka halaman beranda disajikan pada Gambar 10.



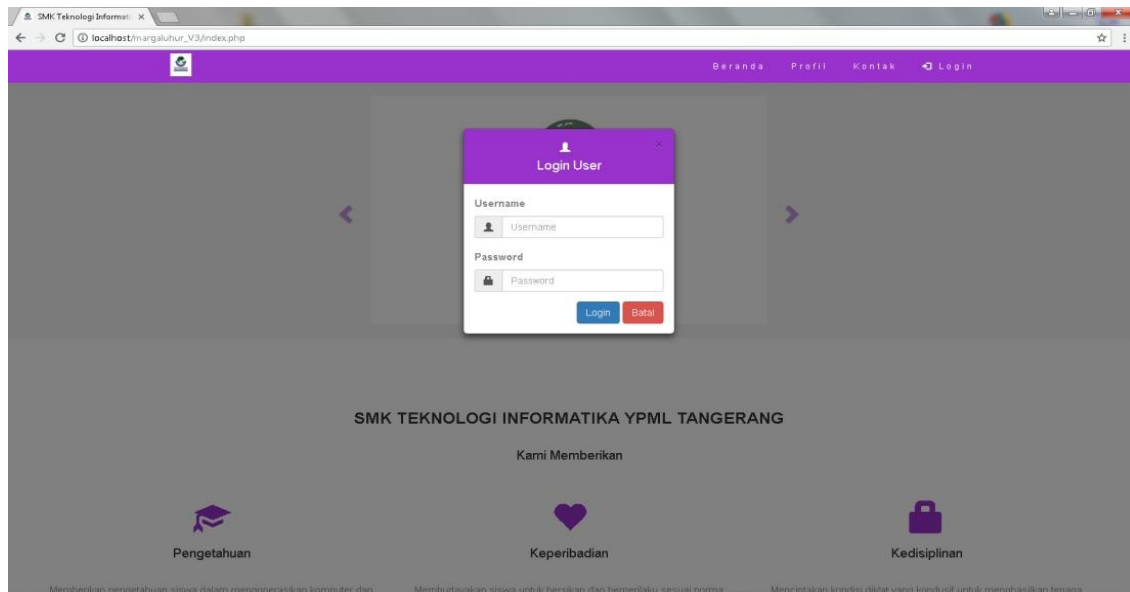
Gambar 10. Implementasi Halaman Beranda

Struktur menu yang terdapat pada halaman beranda yang dapat diakses oleh pengguna Sistem Informasi Administrasi Sekolah dapat dilihat pada Tabel 4.

Tabel 4. Struktur Menu Halaman Beranda

Menu	Deskripsi
Beranda	Menu ini menampilkan halaman beranda Sistem Informasi Administrasi Sekolah.
Profil	Menu ini menampilkan profil dari instansi terkait.
Kontak	Menu ini menampilkan kontak staf yang berwenang pada SMK TI YPML Tangerang.
Login	Menu ini menampilkan form Login <i>users</i> untuk masuk ke sistem.

Implementasi antarmuka form *login* dapat dilihat pada Gambar 11.



Gambar 11. Implementasi Form *Login*

Struktur komponen yang terdapat pada form *Login* yang dapat diakses oleh pengguna Sistem Informasi Administrasi Sekolah disajikan pada Tabel 5.

Tabel 5. Struktur Komponen Form *Login*

Komponen	Deskripsi
<i>Login Header</i>	Komponen ini menampilkan judul form dan tombol <i>close</i> untuk menutup form <i>Login</i> .
<i>Field Username</i>	Komponen ini digunakan untuk menampung <i>input</i> dari users berupa <i>username</i> .
<i>Field Password</i>	Komponen ini digunakan untuk menampung <i>input</i> dari users berupa <i>password</i> .
Tombol <i>Login</i>	Komponen ini digunakan untuk memproses <i>input</i> dari users berupa validasi <i>Login</i> .
Tombol Batal	Komponen ini digunakan untuk menutup form <i>Login</i> .

4.4. Pengujian *Blackbox* dan *Whitebox*

4.4.1. *Blackbox*

Pengujian *Blackbox* merupakan pengujian *software* berfokus pada persyaratan fungsionalnya. Rencana pengujian *blackbox* disajikan pada Tabel 6.

Tabel 6. Rencana Pengujian

ITEM UJI	JENIS PENGUJIAN
<i>Login</i>	<i>Blackbox</i>
Tambah Dokumen	<i>Blackbox</i>
Kendali Dokumen	<i>Blackbox</i>
Buat Usulan Dokumen	<i>Blackbox</i>
Kelola <i>Template</i> Dokumen	<i>Blackbox</i>
Kelola Data <i>Users</i>	<i>Blackbox</i>
Pencarian Data dengan Algoritma KMP	<i>Blackbox</i>

Contoh Pengujian ujian *login* disajikan pada Tabel 7.

Tabel 7. Pengujian *Login*

KASUS DAN HASIL UJI (DATA BENAR)			
DATA YANG DIMASUKAN	YANG DIHARAPKAN	PENGAMATAN	KESIMPULAN
<i>Input username dan password sesuai data user.</i>	Dapat <i>login</i> dan masuk ke halaman beranda <i>user</i> .	Menampilkan pesan <i>Berhasil Login</i> dan halaman beranda <i>user</i> , seperti yang diharapkan.	(√) Sesuai () Tidak Sesuai
KASUS DAN HASIL UJI (DATA SALAH)			
<i>Input username dan password tidak sesuai data user.</i>	Tidak dapat <i>login</i> dan Masuk ke halaman beranda <i>user</i> .	Menampilkan pesan <i>Gagal Login</i> dan kembali ke halaman beranda <i>website</i> .	(√) Sesuai () Tidak Sesuai
<i>Username dan password di- biarkan kosong lalu tekan tombol Login.</i>	Tidak dapat memproses <i>Login</i> .	Menampilkan pesan <i>username dan password harus diisi pada field username dan password.</i>	(√) Sesuai () Tidak Sesuai

Pengujian Pencarian Data dengan Algoritma KMP disajikan pada Tabel 8.

Tabel 8. Pengujian Pencarian Data dengan Algoritma KMP

KASUS DAN HASIL UJI (DATA BENAR)			
DATA YANG DIMASUKAN	YANG DIHARAPKAN	PENGAMATAN	KESIMPULAN
<i>Input keyword pada field pencarian lalu klik tombol Cari.</i>	Dapat memproses pencarian data dengan algoritma KMP.	Menampilkan data yang ditemukan pada tabel.	(√) Sesuai () Tidak Sesuai
KASUS DAN HASIL UJI (DATA SALAH)			
<i>Form Input pencarian data dibiarkan kosong lalu tekan tombol Cari.</i>	Tidak dapat memproses pencarian dengan algoritma KMP.	Menampilkan pesan <i>form input harus diisi.</i>	(√) Sesuai () Tidak Sesuai

4.4.2. Whitebox

Uji *Whitebox* Sistem Informasi Administrasi Sekolah pada bagian validasi data, login user dan pencarian dokumen dengan KMP, disajikan pada Tabel 9.

Tabel 9. Pengujian *Whitebox* Sistem Informasi Administrasi Sekolah

PENGUJIAN	TEST CASE BENAR	TEST CASE SALAH
<pre>if (isset(\$_POST['username'])) { \$username = \$_POST['username']; \$password = md5(\$_POST['password']);</pre>	Untuk validasi data	Tidak dapat melakukan
<pre>\$sql = "SELECT * FROM tbl_user WHERE username = 'username' AND password = '\$password'"; \$query = mysqli_query(\$connect,\$sql); if (\$query- >num_rows > 0) { \$row = \$query->fetch_assoc(); \$id = \$row['id']; if(\$_SESSION['level'] == 'admin') { echo "<script>alert('Berhasil Login');</script>"; echo "<meta http-equiv='refresh' content='0; url=on- admin'>"; } elseif(\$_SESSION['level'] == 'user') { echo "<script>alert('Berhasil Login');</script>"; echo "<meta http-equiv='refresh' content='0; url=on- user'>"; } else { echo "<script>window.location.href='in dex.php'</script>"; } } else { echo "<script>alert('Username & Password Salah'); window.location.href='index.php'< /script>"; } \$connect->close(); exit();</pre>	<i>login user.</i>	validasi data <i>login user.</i>
<pre>\$keyword = \$_GET['keyword']; \$search = explode(' ', \$keyword); \$sql = "SELECT * FROM tbl_dokumen WHERE status='Telah Didistribusi' AND ("; \$parts = array(); \$kata = array(); foreach(\$search as \$search_word) { \$parts[] = 'tgl_upload LIKE "%'.\$search_word.'%"; \$parts[] = 'kode_jenis LIKE "%'.\$search_word.'%"; \$parts[] = 'no_dokumen LIKE "%'.\$search_word.'%"; \$parts[] = 'judul_dokumen LIKE "%'.\$search_word.'%"; \$parts[] = 'keterangan_output LIKE "%'.\$search_word.'%"; \$parts[] = 'tahun_prioritas LIKE "%'.\$search_word.'%"; \$kata[] = \$search_word; } \$dapat .= implode(' ', \$kata); \$sql .= implode(' OR ',</pre>	<i>Sistem akan mencari data dengan algoritma KMP.</i>	Tidak ada reaksi.

5. Kesimpulan

Berdasarkan hasil perancangan, implementasi dan pengujian Sistem Informasi Administrasi Sekolah dengan menerapkan algoritma *Knuth-Morris-Pratt* untuk fungsi pencarian dokumen dan data lain yang berkaitan dengan sistem, maka dapat diambil kesimpulan bahwa perancangan dan pembangunan Sistem Informasi Administrasi Sekolah terbukti dapat membantu SMK TI YPML Tangerang dalam mengendalikan dokumen sehingga lebih efektif dan efisien. Penyimpanan data dokumen pada *database* terbukti dapat menyediakan *backup* data secara digital sehingga aman ketika terjadi kerusakan atau kehilangan terhadap dokumen fisik. Penerapan algoritma *Knuth-Morris-Pratt* pada fungsi pencarian dokumen di *database* terbukti menampilkan hasil yang lebih tepat.

Daftar Pustaka

- [1] M. Rossaria, B. Susilo, and Ernawati, "Implementasi Algoritma Pencocokan String Knuth-Morris-Pratt dalam Aplikasi Pencarian Dokumen Digital Berbasis Android", *Jurnal Rekursif*, vol. 3, no. 2, pp. 183-195, 2015.
- [2] H. T. Sa'diah, "Implementasi Algoritma *Knuth-Morris-Pratt* pada Fungsi Pencarian Judul Tugas Akhir Repository", *Jurnal Komputasi*, vol. 14, no. 1, pp. 1-9, 2017.
- [3] R. J. McLeod, *System Development: A Project Management Approach*, New York: Leigh Publishing LLC, 2002.
- [4] Munawar, *Pemodelan Visual dengan UML*, Yogyakarta: Graha Ilmu, 2005.
- [5] Ramdhani, A. Sukmaji, and T. Sutanto, "Rancang Bangun Aplikasi Pencarian Ayat Al-S.Qur'an Menggunakan Metode Algoritma String Matching Knuth-Morris-Pratt Berbasis Android," *JSIKA*, vol. 4, no. 1, pp. 1-7, 2015.
- [6] M. S. Badri, *Manajemen Administrasi Perkantoran Modern*. Jakarta: Penerbit Erlangga, 2007.
- [7] Jogiyanto, *Analisis dan Desain Sistem Informasi*, Jakarta: Andi, 2005
- [8] M. C. Utami, and Y. T. Hutomo, "Penerapan Waterfall dalam Analisis dan Perancangan Sistem Informasi Manajemen Dokumen Surat Menyurat pada Bank BJB Kantor Cabang BSD Tangerang", *Jurnal Sains, Teknologi dan Industri*, vol. 12, no. 1, pp. 129-135, 2014.
- [9] D. A. Prpto, and M. Purwaningsih, "Pengembangan Aplikasi Dokumen Manajemen untuk Departemen Pemerintahan," *Seminar Nasional Informatika 2008 (semnasIF 2008)*, pp. 82-87, 2008.
- [10] Tukino, "Perancangan Sistem Informasi Manajemen Proyek Pengaksesan Dokumen Perakitan PCBA," di *PT Surya Teknologi Batam Berbasis Web, TEKNOSI (Jurnal Teknologi dan Sistem Informasi)*, vol. 2, no. 3, pp. 67-84, 2016.