

Penerapan Algoritma Pencarian *Knuth-Morris-Pratt* dalam Sistem Informasi Perpustakaan pada SMAN 1 Babelan

Nur Azizah Diti ¹, Fata Nidaul Khasanah ^{1,*}, Aida Fitriyani ¹

* Korespondensi: e-mail: fatanidaul@gmail.com

¹ Informatika: Universitas Bhayangkara Jakarta Raya;
Jl.Raya Perjuangan Bekasi Utara, Kota Bekasi,
88955882 / 88955871: e-mail:
nur.azizah.diti18@mhs.ubharajaya.ac.id
fatanidaul@gmail.com,
aida.fitriyani@dsn.ubharajaya.ac.id

Submitted : **16 Agustus 2022**
Revised : **20 September 2022**
Accepted : **24 Oktober 2022**
Published : **30 November 2022**

DOI:
<https://doi.org/10.31599/jsracs.v3i2.1507>

Abstract

The library is a place where information can be obtained easily. Because in the library there are lots of books, journals, and research results that can be used. As is the case in the library of SMAN 1 Babelan. But in fact the library of SMAN 1 Babelan still has not implemented a digital technology information system. So that in the recording process, they still use a manual system or recording in a ledger. Therefore, a system is needed to overcome this problem by implementing the Knuth-Morris-Pratt algorithm which is a string search algorithm, and the Waterfall search development method.

Keywords: *Knuth-Morris-Pratt, Library, Waterfall*

Abstrak

Perpustakaan merupakan salah satu tempat dimana sebuah informasi dapat didapatkan dengan mudah. Karena dalam perpustakaan banyak sekali buku-buku, jurnal, serta hasil penelitian yang dapat dimanfaatkan. Seperti halnya pada perpustakaan SMAN 1 Babelan. Namun nyatanya perpustakaan SMAN 1 Babelan masih belum menerapkan sistem informasi teknologi digital. Sehingga dalam proses pencatatan, masih menggunakan sistem manual atau pencatatan pada buku besar. Maka dari itu dibutuhkan suatu sistem untuk mengatasi masalah tersebut dengan cara mengimplementasikan algoritma *Knuth-Morris-Pratt* yaitu algoritma pencarian string, dan dengan metode pengembangan *Waterfall*.

Kata kunci: *Knuth-Morris-Pratt, Perpustakaan, Waterfall*

1. Pendahuluan

Memasuki zaman era digital seperti ini, penggunaan teknologi sudah banyak berkembang dan banyak digunakan pada bidang apapun. Yang bertujuan untuk mempermudah suatu pekerjaan manusia dan untuk membantu menyampaikan sebuah informasi secara mudah dan cepat. Perpustakaan merupakan salah satu sarana pendidikan yang bertujuan untuk membantu proses belajar mengajar (Sinaga, 2022). Perpustakaan umumnya berisikan buku-buku, jurnal, hasil penelitian, serta lain-lainnya yang terus berkembang seiring dengan perkembangan zamannya. Demikian pula untuk pelayanan penunjang bagi setiap anggota perpustakaan, seringkali dilengkapi dengan sistem pencarian buku (Afriyanti & Retnoningsih, 2022). Seperti halnya pada perpustakaan yang berada pada SMAN 1 Babelan yang menyediakan berbagai macam buku pelajaran guna untuk menunjang proses pembelajaran, seperti untuk mencari berbagai macam referensi buku, peminjaman buku, serta pengembalian buku. Namun perpustakaan SMAN 1 Babelan masih belum menerapkan sistem informasi. Sehingga pada saat proses pencarian, peminjaman, serta pengembalian buku masih menggunakan sistem manual yang menyebabkan kurangnya efektifitas pada pelayanan perpustakaan, serta memungkinkan terjadinya kesalahan pada proses penelitian.

Algoritma *Knuth-Morris-Pratt (KMP)* merupakan algoritma pencarian *string*, yang dikembangkan pada tahun 1977 oleh Donald E. Knuth, James H. Morris, dan Vaughan R. Pratt. Algoritma *Knuth-Morris-Pratt* merupakan pengembangan dari algoritma *Brute Force*. Algoritma ini melakukan pencocokan terhadap *string* dari karakter per-karakter yang melakukan pencarian pada *pattern* dan teks (Khairan & Ahmadian, 2019; Sadiah, 2017). Dengan algoritma Knuth-Morris-Pratt dalam perpustakaan membuat penelusuran sistem lebih cepat pelayanan perpustakaan dapat berjalan lancar (Nursobah & Pahrudin, 2019)

Cara kerja Algoritma *Knuth-Morris-Pratt (KMP)* menurut pada saat mencocokkan *string* menurut (Islamiyati & Fikri, 2022) adalah sebagai berikut: (a) Masukkan *Query* atau kata yang ingin dicari, dengan pemisalan $P = Pattern$ yang merupakan susunan kata yang akan diperiksa, $T = Teks$ yang merupakan kalimat yang akan dicari kecocokannya; (b) Kemudian algoritma akan mencocokkan susunan *pattern* yang dijadikan pada awal kalimat; (c) Pencocokan *pattern* ini dilakukan dari ke kanan, dengan karakter per-karakter. Hingga salah satu kondisi berikut terpenuhi yaitu karakter pada *pattern* dan *teks* yang dibandingkan tidak cocok (*mismatch*) atau semua karakter pada *pattern* cocok, Lalu algoritma akan menampilkan hasil pencarian; (d) Algoritma kemudian akan menggeser *pattern* atau susunan kata berdasarkan tabel, lalu mengulangi langkah no.2 hingga *pattern* berada pada ujung teks.

2. Metode Penelitian

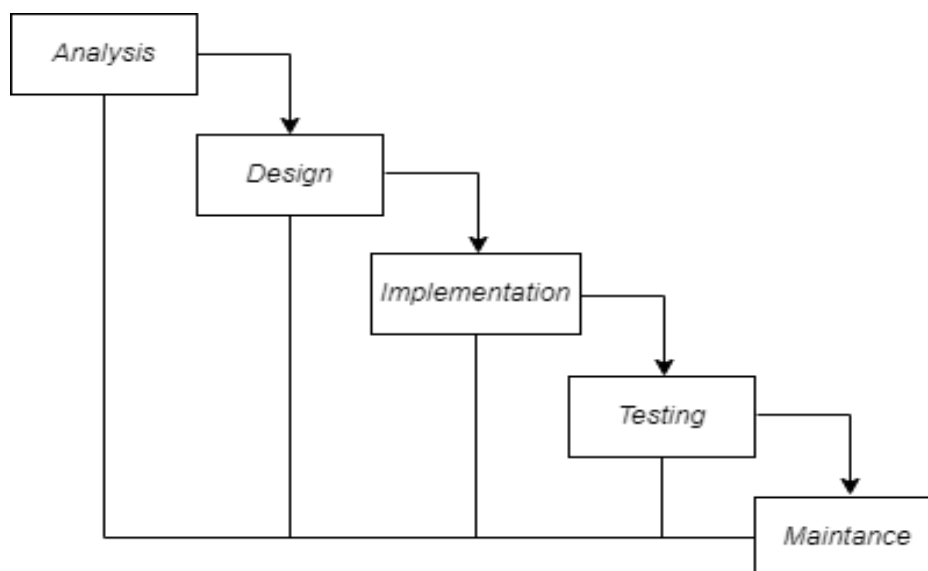
Dalam melakukan penelitian diperlukan adanya sebuah metode agar suatu data dapat bersifat faktual serta dapat memecahkan suatu permasalahan yang ada. Berikut metode penelitian yang digunakan.

2.1. Pengumpulan Data

Terdapat 3 (tiga) metode pengumpulan data yang digunakan, yaitu: (a) Observasi yang dilakukan dengan cara melakukan pengamatan secara langsung terkait permasalahan objek yang terjadi (Khasanah & Herlawati, 2021); (b) Wawancara yang dilakukan dengan cara tanya jawab yang ditujukan untuk narasumber, terkait tentang permasalahan yang terjadi (Khasanah et al., 2022); (c) Studi pustaka yang dilakukan dengan cara mencari informasi yang terkait dengan permasalahan yang juga dapat menjadi sebuah acuan penyelesaian masalah.

2.2. Pengembangan Sistem

Metode *Waterfall* (metode air terjun) adalah suatu gambaran pendekatan yang sistematis pada pengembangan perangkat lunak, yang diawali dari tahapan menganalisa kebutuhan pengguna, lalu tahapan perencanaan (*planning*), pemodelan (*modelling*), konstruksi (*construction*), dan yang terakhir dukungan perangkat lunak yang dihasilkan (Swastika & Khasanah, 2017).



Sumber: Hasil Penelitian (2022)

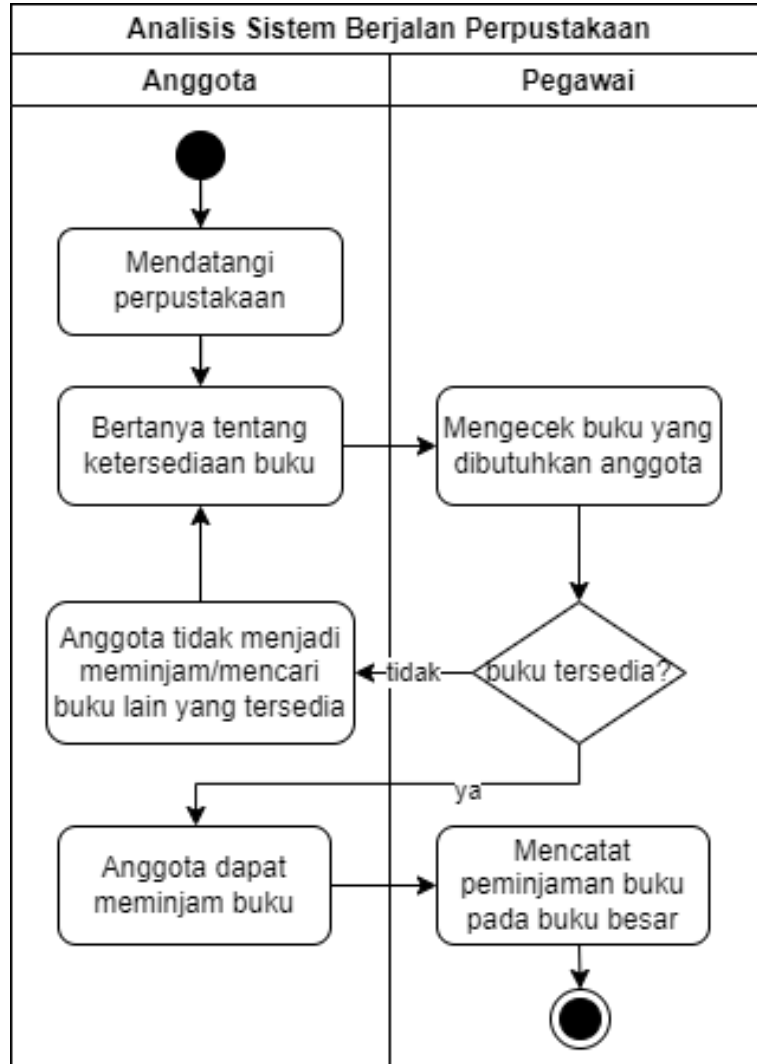
Gambar 1. Pengembangan Metode *Waterfall*

Dalam pengembangannya, metode *waterfall* mempunyai tahapan-tahapan didalamnya. Berikut penjelasan dari tahapan-tahapannya: (a) *Analysis* merupakan tahapan awal yang digunakan untuk mendapatkan suatu informasi dengan melakukan wawancara, observasi, dan studi pustaka; (b) *Design* merupakan tahapan perancangan desain yang digunakan untuk membantu pada tahap pengembangan; (c) *Implementation* merupakan tahapan pengembangann aplikasi, dengan cara mengimplementasikan bahasa pemrograman yang digunakan; (d) *Testing* merupakan tahapan pengujian terhadap aplikasi yang sudah dikembangkan sebelumnya; (e) *Maintenance* merupakan tahapan yang meminta beberapa calon pengguna untuk menggunakan aplikasi, lalu memberikan *feedback* agar aplikasi tetap dapat dikembangkan kembali menjadi yang lebih baik.

3. Hasil dan Pembahasan

3.1. Analisis Sistem Berjalan

Dalam mengembangkan sistem informasi perpustakaan, tentunya diperlukan rancangan sistem dari proses sebelumnya agar rancangan sistem selanjutnya dapat menyelesaikan permasalahan yang ada. Pada Gambar 2 merupakan rancangan dari sistem berjalan yang ada.



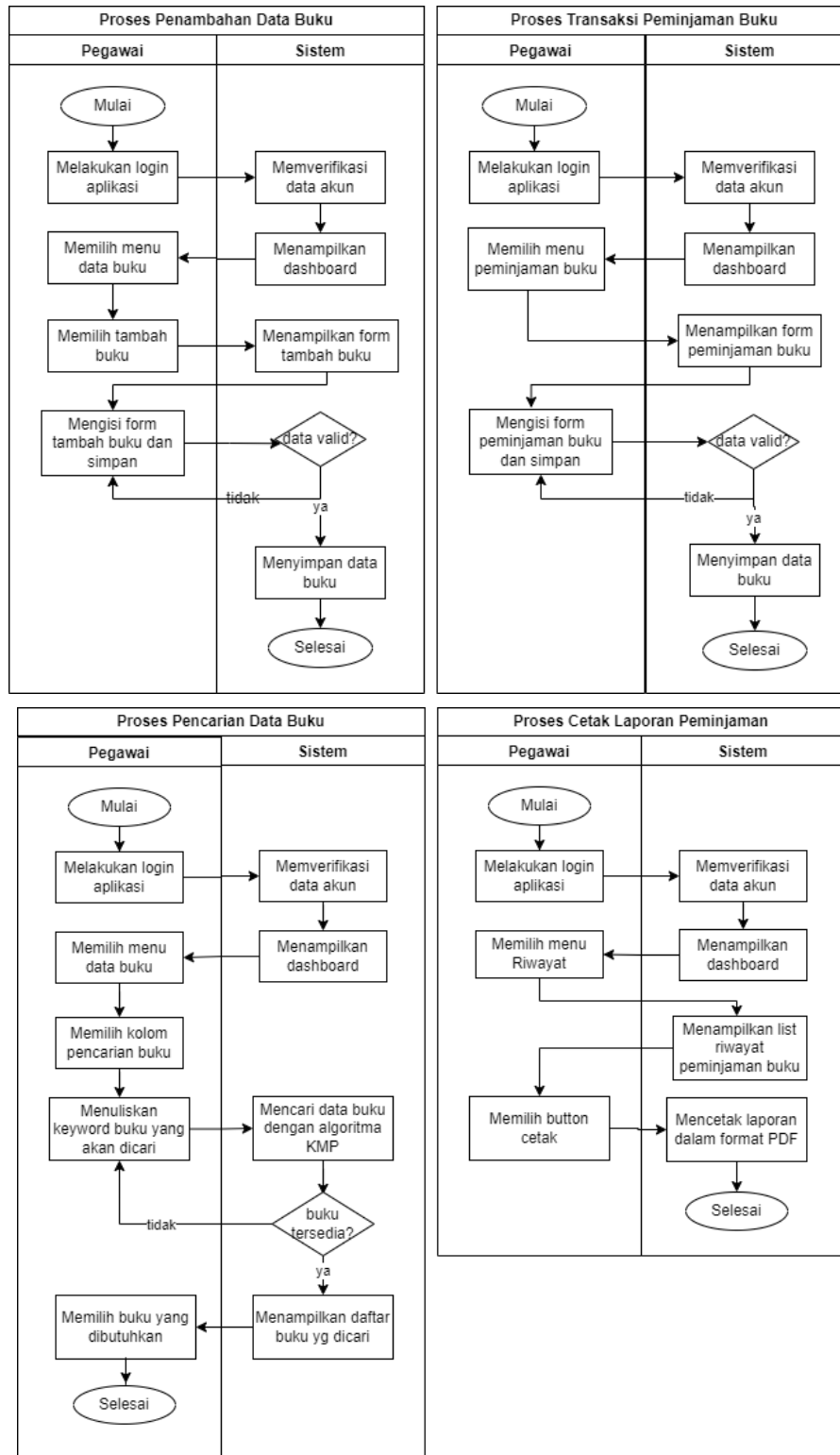
Sumber: Hasil Penelitian (2022)

Gambar 2. Flowmap Analisis Sistem Berjalan

Pada Gambar 2 menunjukkan bahwa sistem berjalan perpustakaan diawali dengan anggota yang mengunjungi perpustakaan, kemudian bertanya tentang ketersediaan buku yang dibutuhkannya, lalu pegawai akan mengecek buku yang dibutuhkan oleh anggota. Jika buku tidak tersedia dalam perpustakaan, maka anggota tidak dapat meminjam buku atau memilih buku lain yang tersedia pada perpustakaan. Namun jika buku tersedia pada perpustakaan, maka anggota dapat meminjam buku. Pegawai akan mencatat transaksi peminjaman pada buku besar.

3.2 Sistem Usulan

Setelah mengetahui permasalahan yang terjadi pada perpustakaan, maka peneliti mengusulkan sistem sebagaimana pada Gambar 3.

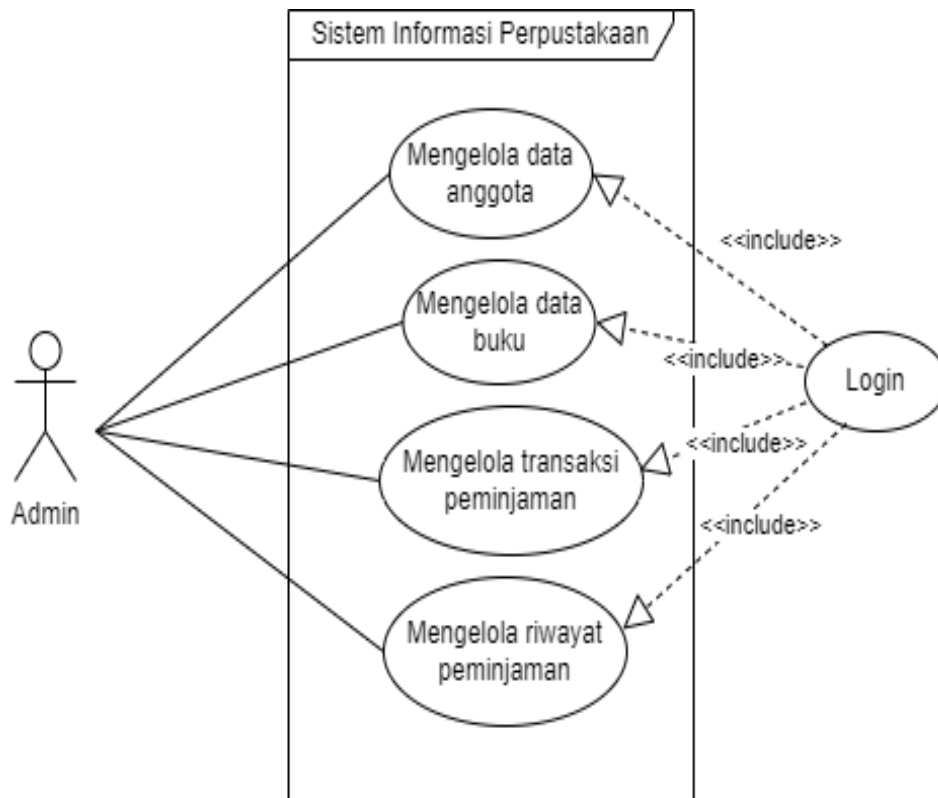


Sumber: Hasil Penelitian (2022)

Gambar 3. Flowmap Sistem Usulan

Use Case Diagram

Use case diagram diatas menjelaskan bagaimana cara berinteraksi antara aktor dengan sistem. Use case diagram pada sistem usulan kali ini, terdapat aktor yang bekerja didalamnya, yaitu admin (pegawai perpustakaan). Admin dapat mengelola data anggota, mengelola data buku, mengelola transaksi peminjaman buku, serta mengelola riwayat peminjaman buku. Semua pengelolaan tersebut tentunya harus dilakukan dengan melakukan login aplikasi terlebih dahulu.



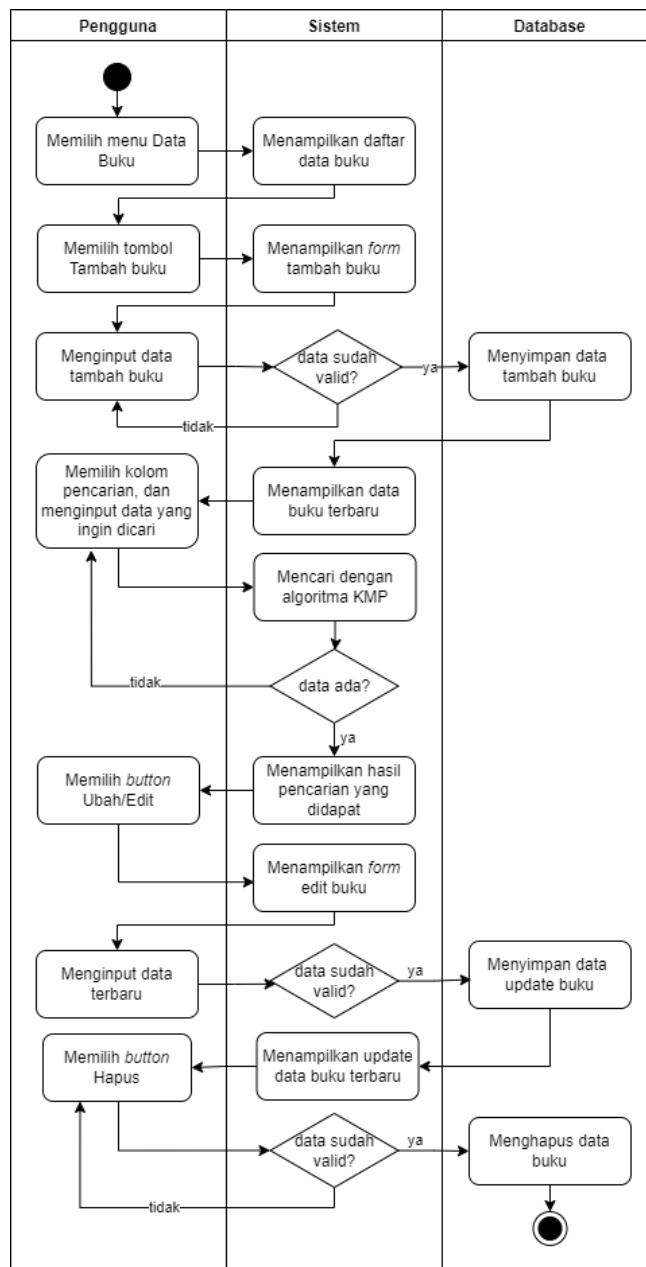
Sumber: Hasil Penelitian (2022)

Gambar 4. Use Case Sistem Informasi Perpustakaan

Activity Diagram

a. Activity Diagram Mengelola Data Buku

Berikut ini merupakan desain UML dari Activity Diagram Mengelola Data Buku pada sistem perpustakaan. Yang diawali dengan user memilih menu Data Buku, yang kemudian sistem akan menampilkan halaman tabel data buku. Lalu user dapat memilih tombol Tambah Data, dan sistem akan menampilkan form tambah data buku. Kemudian user diminta untuk menginput form penambahan buku. Jika sudah di input, maka sistem akan menampilkan pop-up konfirmasi apakah data yang di input sudah benar. Lalu jika sudah maka sistem akan menyimpan data buku pada database dan menampilkannya pada tabel data buku. Dan user dapat mengelola data buku tersebut dengan cara melihat detail, mengubah, menghapus, mencari, serta pinjam buku.

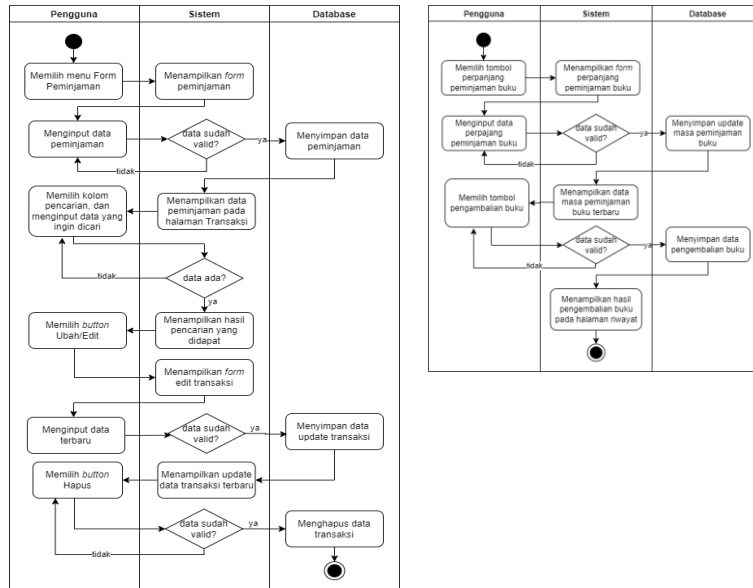


Sumber: Hasil Penelitian (2022)

Gambar 5. Activity Diagram Mengelola Data Buku

b. Activity Diagram Mengelola Data Transaksi

Gambar 6 merupakan desain UML dari Activity Diagram Mengelola Transaksi Peminjaman Buku pada sistem perpustakaan. Diawali dengan user memilih menu Form Peminjaman, sistem akan menampilkan form peminjaman yang harus di input oleh user. Jika sudah di input, maka sistem akan menampilkan pop-up konfirmasi apakah data yang di input sudah benar. Jika sudah benar maka sistem akan menyimpan data peminjaman pada database dan menampilkannya pada tabel transaksi. User dapat mengelola data transaksi tersebut dengan cara mengubah, menghapus, mencari, memperpanjang masa peminjaman, dan mengembalikan peminjaman buku.



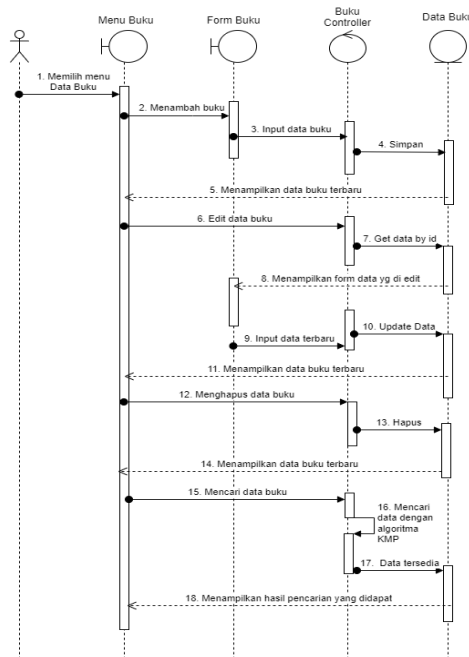
Sumber: Hasil Penelitian (2022)

Gambar 6. Activity Diagram Mengelola Data Transaksi

Sequence Diagram

a. Sequence Diagram Mengelola Data Buku

Berikut Gambar 7 merupakan desain UML dari Sequence Diagram Mengelola Data Buku, yaitu dengan cara user memilih menu Data Buku dan sistem akan menampilkan halaman tabel data buku. Lalu user dapat memilih tombol Tambah Data, dan sistem akan menampilkan form tambah data buku yang harus di input oleh user. Jika sudah, maka sistem akan menyimpan data buku pada database dan menampilkannya pada tabel data buku. Dan admin dapat mengelola data tersebut.

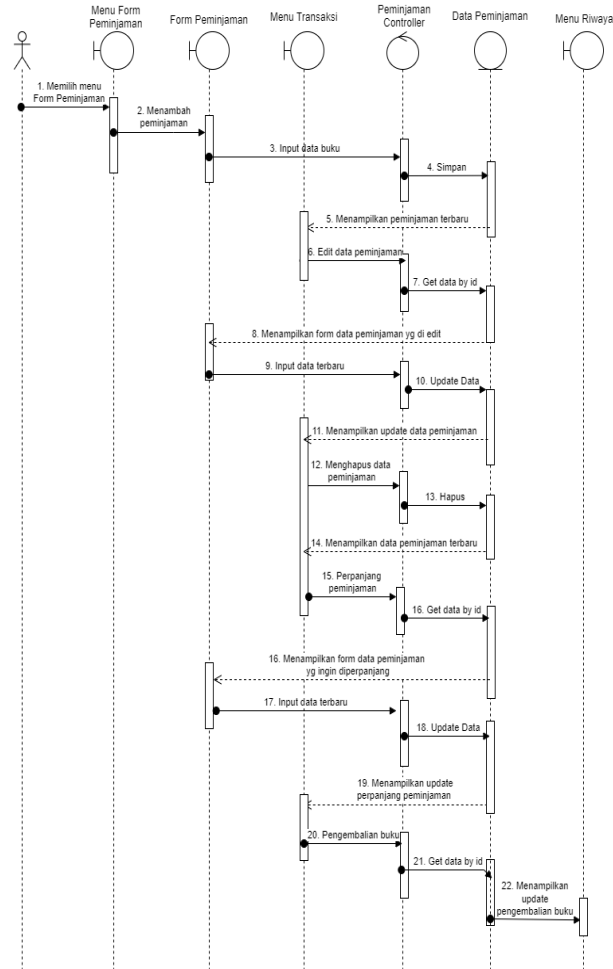


Sumber: Hasil Penelitian (2022)

Gambar 7. Sequence Diagram Mengelola Data Buku

b. *Sequence Diagram* Mengelola Transaksi Peminjaman

Berikut ini Gambar 8 merupakan desain UML dari *Sequence Diagram* Mengelola Transaksi Peminjaman, yaitu dengan cara user memilih menu *Form* Peminjaman dan sistem akan menampilkan halaman *form* peminjaman yang harus di *input* oleh user. Jika sudah di *input* dan data sudah benar. Maka sistem akan menyimpan data peminjaman pada *database* dan menampilkannya pada tabel transaksi. Dan user dapat mengelola data tersebut.

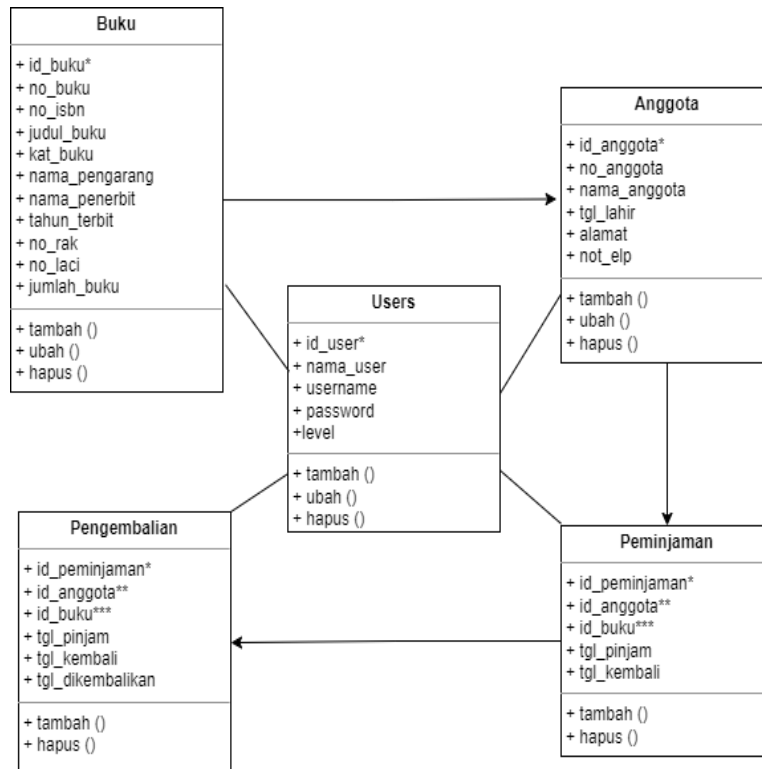


Sumber: Hasil Penelitian (2022)

Gambar 8. *Sequence Diagram* Mengelola Transaksi Peminjaman

Class Diagram

Class diagram merupakan penggambaran dari struktur sistem yang mendefinisikan kelas-kelas yang akan dibuat untuk sistem. Kelas-kelas tersebut terdiri dari *atribut* dan *method*. *Class diagram* pada sistem informasi perpustakaan terdiri dari 5 *class* didalamnya yaitu *user*, *buku*, *anggota*, *peminjaman*, dan *pengembalian* seperti pada Gambar 5. *Class user* menjadi peran utama dalam proses jalannya *class-class* yang lainnya. Karena tanpa adanya akses *class user*, pengguna tidak dapat mengakses menu lainnya pada sistem.



Sumber: Hasil Penelitian (2022)

Gambar 9. *Class Diagram* Sistem Informasi Perpustakaan

3.4. Implementasi

Implementasi merupakan suatu tahapan penerapan sistem informasi perpustakaan yang nantinya dapat digunakan pada tahap pengujian (*testing*).

a. Tampilan *dashboard*

Setelah *user* berhasil untuk memasuki sistem, maka selanjutnya sistem akan menampilkan halaman *dashboard* yang berisikan menu-menu disampingnya (*navbar*) serta menu *user* dan *log out* pada bagian atas menu.

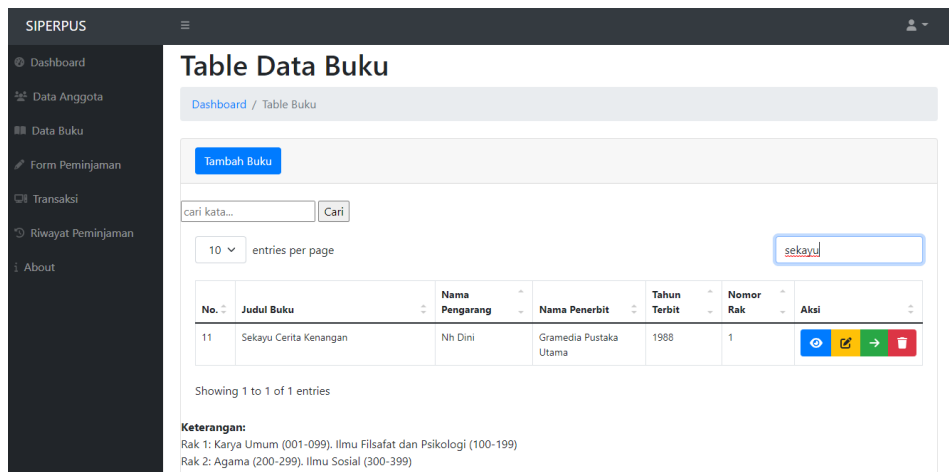


Sumber: Hasil Penelitian (2022)

Gambar 10. Tampilan Sistem Perpustakaan *Login*

b. Tampilan Pencarian Data Buku

Saat user memilih menu data buku, user juga dapat mencari data buku pada kolom pencarian. Lalu sistem akan mencari data buku dengan menggunakan algoritma *Knuth-Morris-Pratt*, dan akan menampilkan hasil pencarian pada tabel.

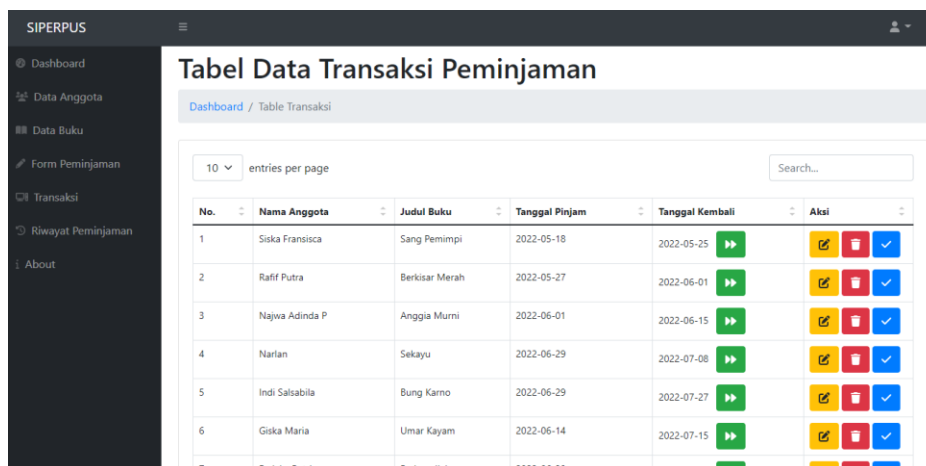


Sumber: Hasil Penelitian (2022)

Gambar 11. Tampilan Sistem Perpustakaan Pencarian Data Buku

c. Tampilan Data Transaksi Peminjaman Buku

Saat user memilih menu transaksi, maka sistem akan menampilkan data-data transaksi yang sudah di *input* dalam *database*. Serta menampilkan aksi perpanjangan, *edit*, *delete*, dan kembalikan buku.



Sumber: Hasil Penelitian (2022)

Gambar 12. Tampilan Sistem Perpustakaan Transaksi Peminjaman

d. Implementasi Algoritma *Knuth-Morris-Pratt* (KMP)

Algoritma *Knuth-Morris-Pratt* digunakan untuk melakukan pencarian pada *string* data dengan cara membandingkan karakter pada *teks* dari kiri hingga ke kanan. Berikut Gambar 13 proses atau alur perhitungan algoritma *Knuth-Morris-Pratt*.

```

53 function KMPSearch(pat,txt)
54 {
55     var M = pat.length;
56     var N = txt.length;
57
58     // create lps[] that will hold the longest
59     // prefix suffix values for pattern
60     var lps = [];
61     var j = 0; // index for pat[]
62
63     // Preprocess the pattern (calculate lps[])
64     // array)
65     computeLPSArray(pat, M, lps);
66
67     var i = 0; // index for txt[]
68     while (i < N) {
69         if (pat.charAt(j) == txt.charAt(i)) {
70             j++;
71             i++;
72         }
73         if (j == M) {
74             console.log("Found pattern " + "at index " + (i - j) + "\n");
75             j = lps[j - 1];
76         }
77         // mismatch after j matches
78         else if (i < N && pat.charAt(j) != txt.charAt(i)) {
79             // Do not match lps[0..lps[j-1]] characters,
80             // they will match anyway
81             if (j != 0)
82                 j = lps[j - 1];
83             else
84                 i = i + 1;
85         }
86     }
87 }
88

```

Sumber: Hasil Penelitian (2022)

Gambar 13. Implementasi Algoritma KMP

Kemudian setelah menginput *source code* algoritma KMP, maka akan menampilkan *output* sebagai berikut pada Gambar 14.

Found pattern at index 10

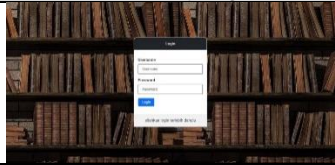

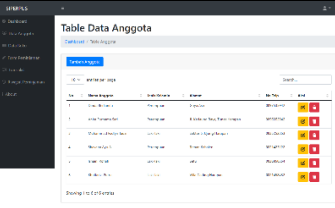
Sumber: Hasil Penelitian (2022)

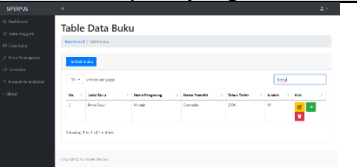
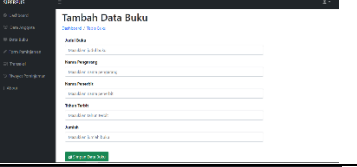
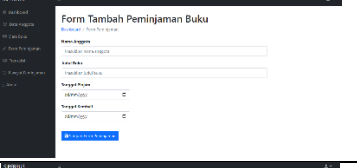

Gambar 14. Output Algoritma KMP

3.5. Pengujian Black Box Testing

Pada pengujian sistem informasi perpustakaan ini peneliti menggunakan dua metode yaitu *Black-box Testing* dan kuesioner, yang digunakan untuk menguji apakah kinerja dan fungsional sistem sudah sesuai dengan harapan (Khasanah, 2018).

Table 1. Pengujian Black Box Testing

No.	Menu/Fungsi	Hasil yang diharapkan	Kesimpulan	Gambar Tampilan yang dihasilkan
1.	Login Aplikasi dengan <i>username</i> dan <i>password</i> yang benar.	Sistem akan menampilkan halaman <i>dashboard</i> .	Valid.	
2.	Login Aplikasi dengan <i>username</i> dan <i>password</i> yang salah.	Sistem akan menampilkan <i>notifikasi password/username</i> salah, dan tidak akan membuka halaman <i>dashboard</i> .	Valid.	
3.	Memilih menu data anggota.	Sistem akan menampilkan tabel data anggota.	Valid.	

No.	Menu/Fungsi	Hasil yang diharapkan	Kesimpulan	Gambar Tampilan yang dihasilkan
4.	Mencari buku pada tabel data buku	Sistem akan menampilkan hasil pencarian buku yang diharapkan.	Valid.	
5.	Menambah buku pada tabel data buku.	Sistem akan menampilkan form tambah data buku.	Valid.	
6.	Menambah form peminjaman buku.	Sistem akan menampilkan form peminjaman buku, dan menyimpan data peminjaman pada transaksi.	Valid.	
7.	Memilih pengembalian buku.	Sistem akan menampilkan data pada riwayat peminjaman buku.	Valid.	

Hasil data yang didapat dalam pengujian *black box testing* ini adalah Valid. Semua proses sistem yang diperintahkan oleh *user* berjalan sesuai dengan kebutuhan. Sehingga aplikasi sudah siap untuk digunakan.

4. Kesimpulan

Berdasarkan penjelasan-penjelasan yang sudah dibahas serta telah dilakukannya sebuah pengujian aplikasi, maka kesimpulan yang dapat diambil oleh peneliti adalah: (a) Dapat mengembangkan sistem informasi perpustakaan berbasis website; (b) Dapat menerapkan algoritma *Knuth-Morris-Pratt* dalam proses pencarian data; (c) Dapat mengembangkan sistem dengan metode pengembangan *Waterfall*; (d) Hasil pengujian sistem dengan metode *Black-Box Testing* menunjukkan bahwa aplikasi telah berjalan sesuai dengan hasil yang diharapkan.

Daftar Pustaka

- Islamiyati, D. S., & Fikri, A. (2022). Penerapan Algoritma Knuth-Morris-Pratt dalam Mendeteksi Tingkat Kemiripan Judul Skripsi Berbasis Web. *Journal of Information System Research (JOSH)*, 3(2), 58–63. <https://doi.org/10.47065/josh.v3i2.1168>
- Khairan, K. A., & Ahmadian, H. (2019). Penerapan Algoritma Knuth-Morris-Pratt Pada Fitur Pencarian Definisi Istilah Standar Operasional Prosedur (SOP) Pada Lembaga Penjaminan Mutu UIN Ar-Raniry. *Cyberspace: Jurnal Pendidikan Teknologi Informasi*, 3(1), 26. <https://doi.org/10.22373/cj.v3i1.4723>
- Khasanah, F. N. (2018). Pengujian Fungsional dan Non Fungsional Aplikasi Informasi Telepon Darurat Berbasis Android. *Information System For Educators And Professionals*, 3(1), 79–90. <https://ejournal-binainsani.ac.id/index.php/ISBI/article/view/1040>

- Khasanah, F. N., & Herlawati, H. (2021). Culinary Places Recommendation System in Bekasi City Using the Simple Additive Weighting Method. *PIKSEL : Penelitian Ilmu Komputer Sistem Embedded and Logic*, 9(1), 63–74. <https://doi.org/10.33558/piksel.v9i1.2621>
- Khasanah, F. N., Murdowo, S., Untari, D. T., Nurmanto, D., & Arifin, W. (2022). Optimization of Simple Additive Weighting Method in Assessment of Research Reviewer Selection. *JUITA : Jurnal Informatika*, 10(2), 283. <https://doi.org/10.30595/juita.v10i2.15030>
- Murdowo, S., & Khasanah, F. N. (2022). *Buku Ajar Pemrograman Berbasis Objek 1 Microsoft Visual Basic 6*. CV Pena Persada.
- Nursobah, N., & Pahrudin, P. (2019). Penerapan Algoritma Pencarian Knuth-Morris-Pratt (KMP) Dalam Sistem Informasi Perpustakaan SMK TI Pratama. *Sebatik*, 23(1), 112–115. <https://doi.org/10.46984/sebatik.v23i1.451>
- Sadiyah, H. T. (2017). Implementasi Algoritma Knuth-Morris-Pratt Pada Fungsi Pencarian Judul Tugas Akhir Repository. *Jurnal Komputasi*, 14(1), 115–124. <https://doi.org/10.33751/komputasi.v14i1.262>
- Sinaga, D. (2022). *Mengelola Perpustakaan Sekolah*. Pustaka Putra Khatulistiwa & Kiblat Buku Utama.
- Swastika, R. H., & Khasanah, F. N. (2017). Sistem Informasi Reservasi Lapangan Futsal Pada Futsal Corner Menggunakan Metode Waterfall. *Jurnal Mahasiswa Bina Insani*, 1 No.2(2), 251–266.