

IMPLEMENTASI ALGORITMA *SEQUENTIAL SEARCHING* PADA APLIKASI *E-OFFICE*

Ali Akbar Rismayadi¹, Liah Jamaliah²

Teknik Informatika

Universitas BSI Bandung, Jl Sekolah Internasional No.1-6 Antapani Bandung

rismayadi.ali@gmail.com¹, liahjamaliah29@gmail.com²

Abstrak

Pengolahan dan pengarsipan data serta nomor surat masih menggunakan cara manual yaitu dengan mencatat (tulis tangan) pada buku dan melihat ulang urutan surat sebelumnya untuk memberi nomor pada surat baru. Cara tersebut tentu tidak efisien karena memerlukan buku yang tebal dan membutuhkan waktu yang cukup lama dalam proses perekapan. Masalah lainnya yaitu penyimpanan berkas yang terlalu banyak sehingga menjadi kendala saat melakukan pencarian data. Maksud penelitian ini adalah merancang dan membangun aplikasi *e-Office* serta mengimplementasikan algoritma *sequential searching* pada aplikasi tersebut. *E-Office* adalah suatu sistem yang berhubungan dengan administrasi secara maya memusatkan komponen-komponen sebuah organisasi dimana data, informasi, dan komunikasi dibuat melalui media telekomunikasi. Algoritma *Sequential Searching* adalah proses membandingkan setiap elemen *array* satu persatu secara beruntun dimulai dari elemen pertama hingga elemen yang di cari ditemukan atau hingga elemen terakhir *array*. Implementasi algoritma *sequential searching* pada aplikasi *e-office* ini mampu meningkatkan efektifitas dalam pengolahan, pendistribusian, serta pencarian dokumen persuratan.

Kata Kunci: Surat, Pencarian, *E-Office*, *Sequential Searching*

Abstract

Processing and archiving of data and letter numbers still use manual method by taking notes (handwritten) on the book and reviewing the order of previous letter to give number on new letter. The way is certainly not efficient because it requires a thick book and takes a long time in the process of recording. Another problem is the storage of files that are too many to be a constraint when performing data searches. The purpose of this research is to design and build e-Office applications and implement sequential searching algorithm in the application. E-Office is a system that deals with the administration of a virtual concentrate component of an organization where data, information, and communication is made through telecommunication media. Sequential Searching algorithm is the process of comparing each array elements one by one in a row from the first element to the found element found or up to the last element of the array. Implementation of sequential searching algorithm in e-office application is able to improve the effectiveness in processing, distribution, as well as search documents.

Key Word: Mail, Search, E-Office, Sequential Searching

I. PENDAHULUAN

Ketertiban dan kelancaran merupakan faktor penting dalam kepengurusan administrasi tata usaha perkantoran. Tata usaha adalah segenap aktifitas menghimpun, mencatat, mengolah, mengirim dan menyimpan keterangan-keterangan yang di perlukan dalam setiap organisasi. Salah satu tugas tata usaha dari pengertian di atas berkaitan dengan keterangan, surat menyurat dan warkat-warkat, oleh karena itu tata usaha sering disebut pekerjaan tulis menulis maka tercipta warkat yang kemudian terkumpul menjadi arsip [1].

Masalah lainnya dari sistem manual ini adalah tempat penyimpanan dan pencarian dokumen. Penyimpanan dokumen membutuhkan banyak ruangan dikantor yang menjadikan ruangan kantor menjadi semakin sempit dan berantakan. Dalam hal pencarian dokumen pun, akan semakin sulit apabila dokumen tersebut tidak disimpan dengan teratur. Membutuhkan waktu yang lama hanya untuk mencari dokumen [2].

Pengolahan dan pengarsipan nomor surat di Kantor POS Udara Soekarno-Hatta Jakarta 19000 masih menggunakan cara manual yaitu dengan mencatat (tulis

tangan) pada buku dan melihat ulang urutan surat sebelumnya untuk memberi nomor pada surat baru. Cara tersebut tentu tidak efisien karena memerlukan buku yang tebal terlebih lagi jika untuk mencatat dalam waktu yang lama dan banyaknya jenis surat yang akan di arsipkan. Apalagi permasalahan yang rentan terjadi adalah penyimpanan berkas yang terlalu banyak sehingga menjadi kendala saat melakukan pencarian data. Dengan masalah tersebut, penelitian ini akan membuat sistem *e-office* yang mampu mengolah data surat dan mampu menampilkan nomor surat sehingga pengarsipan dan pemberian nomor surat menjadi lebih efisien. Sistem *e-office* yang akan dibangun tentunya memiliki fitur pencarian (*searching*) data surat.

Pencarian merupakan proses yang fundamental dalam pengolahan data. *Sequential Searching* merupakan algoritma pencarian beruntun yang prosesnya membandingkan setiap elemen satu persatu secara beruntun, mulai dari elemen pertama sampai elemen yang dicari ditemukan atau seluruh elemen sudah diperiksa. Dalam sistem *e-office* yang akan dibuat dalam penelitian ini, fitur pencarian menggunakan algoritma *Sequential Searching* untuk mencari nomor surat yang telah di arsipkan. Data arsip yang telah tersimpan didalam *database*

di kumpulkan pada sebuah *Array* lalu dari *Array* tersebut akan di lakukan proses pencarian dengan algoritma *sequential searching*. Jika data yang dicari ada di dalam *array* maka data tersebut akan di tampilkan. Jika tidak ada maka akan menampilkan pesan bahwa data tidak ada [3].

Implementasi algoritma *sequential searching* pada aplikasi *e-office* ini hanya untuk Kantor Pos Udara Soekarno Hatta Jakarta 19000, Implementasi algoritma *sequential searching* pada aplikasi *e-office* ini dibuat berbasis *web* menggunakan bahasa pemrograman PHP dan *user interface* menggunakan *framework laravel* serta *MySQL* sebagai *server database*.

II. TINJAUAN PUSTAKA

A. Algoritma *Sequential Searching*

Sequential Search adalah proses membandingkan setiap elemen *array* satu persatu secara beruntun dimulai dari elemen pertama hingga elemen yang dicari ditemukan atau hingga elemen terakhir *array* [4].

Metode *Sequential Search* atau disebut pencarian beruntun dapat digunakan untuk melakukan pencarian data baik pada *array* yang sudah terurut maupun yang belum terurut. Proses yang terjadi pada metode pencarian ini adalah sebagai berikut:

1. Membaca *array* data.
2. Menentukan data yang dicari.
3. Mulai dari data pertama sampai dengan data terakhir, data yang dicari dibandingkan dengan masing-masing data di dalam *array*. Jika data yang dicari tidak ditemukan maka semua data atau elemen *array* dibandingkan sampai selesai. Jika data yang dicari ditemukan maka perbandingan akan dihentikan.

B. Aplikasi *E-Office*

Electronic Office (E-Office) adalah suatu sistem yang berhubungan dengan administrasi, secara maya memusatkan komponen-komponen sebuah organisasi dimana data, informasi, dan komunikasi dibuat melalui media telekomunikasi. *E-office* merupakan *software* yang digunakan untuk mengatur pola pekerjaan yang sudah atau akan dilakukan oleh pegawai, menyimpan, dan mengontrol konten-konten yang ada didalam suatu organisasi secara elektronik [5].

C. Pemrograman *WEB*

Pemrograman *web* diambil dari dua suku kata yaitu pemrograman dan *web*. Pemrograman di artikan proses, cara, perbuatan program. Definisi *web* adalah jaringan komputer yang terdiri kumpulan situs internet yang menawarkan teks, grafik, suara dan sumber daya

animasi melalui protocol transfer *hypertext*. Orang banyak mengenal *web* dengan istilah *WWW (World Wide Web)*, *world wide web* adalah layanan internet yang paling populer saat ini internet mulai dikenal dan digunakan secara luas setelah adanya layanan *WWW*. *WWW* adalah halaman-halaman *website* yang dapat saling terkoneksi satu dengan lainnya (*hyperlink*) yang membentuk samudra belantara informasi. *WWW* berjalan dengan *protocol Hypertext Transfer Protokol (HTTP)* [1].

D. *XAMPP*

XAMPP termasuk paket *server* yang paling mudah untuk digunakan sebagai paket untuk pengembangan aplikasi *web*. *XAMPP* termasuk yang paling bagus updatenya, sehingga paling baik dipilih untuk digunakan untuk development ataupun untuk produksi [6].

E. *PHP (Personal Home Page)*

PHP (Personal Home Page) secara umum dikenal sebagai bahasa pemrograman *script-script* yang membuat dokumen *HTML* secara *on the fly* yang dieksekusi di *server web*, dokumen *HTML* yang dihasilkan dari suatu aplikasi bukan dokumen *HTML* yang dibuat dengan menggunakan editor text atau editor *HTML* [6].

F. *Framework Laravel*

Framework PHP adalah suatu kerangka kerja untuk pembuatan program dengan menggunakan *PHP*. *Framework laravel* di desain berbasis *model, view, dan controller*. Dengan *MVC*, pengembangan aplikasi *web* akan dapat dibangun lebih cepat, lebih mudah di *manage*. Jadi beberapa alasan membangun aplikasi berbasis *MVC* adalah:

1. Solusi mudah untuk hampir semua aplikasi *web*.
2. Pembuatan aplikasi *web* akan lebih fleksibel dan lebih mudah dikembangkan.
3. Mendukung pembuatan aplikasi besar dan kompleks.
4. Pengelolaan logika aplikasi lebih mudah [7].

G. *MySQL*

MySQL merupakan *software database* yang termasuk paling populer dilingkungan *Linux*, kepopulerannya ini karena ditunjang performansi *query* dari *database* nya yang saat itu bisa dikatakan paling cepat, dan jarang bermasalah. *MySQL* telah tersedia juga di lingkungan *Windows* [6].

III. METODE PENELITIAN

A. Algoritma *Sequential Searching*

Sequential Search adalah proses membandingkan setiap elemen *array* satu persatu secara beruntun dimulai

dari elemen pertama hingga elemen yang dicari ditemukan atau hingga elemen terakhir *array* [4].

Metode *Sequential Search* atau disebut pencarian beruntun dapat digunakan untuk melakukan pencarian data baik pada *array* yang sudah terurut maupun yang belum terurut. Proses yang terjadi pada metode pencarian ini adalah sebagai berikut:

4. Membaca *array* data.
5. Menentukan data yang dicari.
6. Mulai dari data pertama sampai dengan data terakhir, data yang dicari dibandingkan dengan masing-masing data di dalam *array*. Jika data yang dicari tidak ditemukan maka semua data atau elemen *array* dibandingkan sampai selesai. Jika data yang dicari ditemukan maka perbandingan akan dihentikan.

Proses Algoritma *Sequential Searching* adalah sebagai berikut:

- a. Pertama data melakukan perbandingan satu persatu secara beruntun dalam kumpulan data dengan data yang di cari sampai data tersebut ditemukan atau tidak ditemukan.
- b. Pada dasarnya, pencarian ini hanya melakukan pengulangan data dari 1 sampai dengan jumlah data (n).
- c. Setiap pengulangan, dibandingkan data ke-i dengan data yang sedang dicari.
- d. Apabila data sama dengan yang di cari, berarti data telah berhasil ditemukan. Sebaliknya apabila sampai akhir melakukan pengulangan tidak ada data yang sama dengan yang dicari, berarti data tidak ada yang ditemukan.

Urutan Algoritma *Sequential Searching*:

1. $i < 0$
2. Ketemu \leftarrow false
3. Selama (tidak ketemu) dan ($i < N$) kerjakan baris 4.
4. Jika (Data [i]=key) maka ketemu \leftarrow true

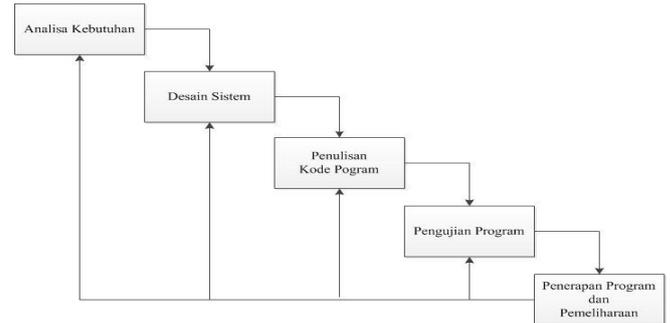
Jika tidak $i \leftarrow i + 1$

Jika (ketemu) maka i adalah indeks dari data yang di cari [3].

B. Metode Pengembangan Sistem

Dalam tahap pengembangan sistem, penelitian ini menggunakan metode *waterfall*. Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut mulai dari analisis, desain pengkodean, pengujian, dan tahap pendukung (*support*). Tahapan-tahapan dalam metode pengembangan sistem *waterfall* adalah sebagai

berikut:



Gambar 1. Siklus *Waterfall*

1. Analisa Kebutuhan Sistem

Analisa kebutuhan sistem dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami serta disesuaikan dengan kebutuhan *user*. Analisa kebutuhan ini di definisikan secara objektif mengenai kebutuhan *input* dan *output*. Langkah yang dilakukan dalam mengidentifikasi input adalah dengan melakukan studi awal mengenai bentuk proses pengadministrasian, pengarsipan, dan pencarian data serta dokumen persuratan pada Kantor Tukar Pos Udara Soekarno Hatta Jakarta yang saat ini sedang dijalankan. Sedangkan *output* yang di dapatkan adalah informasi surat cuti, surat P6, surat P6a dan surat perintah kerja lembur pada setiap bidang di Kantor Tukar Pos Udara Soekarno Hatta Jakarta 19000 serta menampilkan hasil pencarian surat berdasarkan nomor surat.

2. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean. Setelah data terkumpul dan kebutuhan sistem diketahui, kemudian data tersebut diolah menjadi sebuah basis data serta menentukan struktur datanya. Kemudian merancang arsitektur aplikasi *e-office* yang akan dibangun serta mengimplementasikan algoritma *sequential searching* sehingga dapat di aplikasikan sesuai kebutuhan user dengan menggunakan UML (*Unified Modelling Language*).

3. Code Generation

Setelah melewati tahapan desain dan menghasilkan rancangan serta arsitektur keseluruhan implementasi algoritma *sequential searching* pada aplikasi *e-office* yang akan di bangun menggunakan UML, langkah selanjutnya adalah membangun aplikasi *e-office* dengan mengimplementasikan algoritma *sequential searching* dengan menggunakan pemrograman PHP dan *user interface* menggunakan *framework laravel*.

4. Testing

Pengujian fokus pada perangkat lunak dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir

kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan apa yang diinginkan. Pengujian juga dilakukan terhadap aplikasi *e-office*. Pengujian tersebut dilakukan dengan metode *blackbox testing*.

5. Support

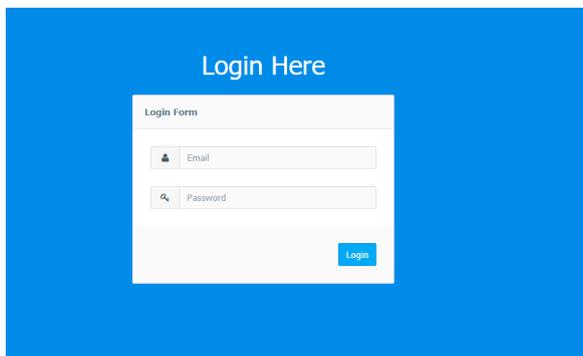
Dalam tahap *support* terhadap aplikasi *e-office* pada Kantor Tukar POS Udara Soekarno-Hatta Jakarta ini dilakukan *backup database* secara berkala. Selain itu, dilakukan perbaikan struktur maupun *coding* apabila terjadi *debugging* ketika aplikasi digunakan atau di implementasikan [8].

IV. ANALISIS DAN PERANCANGAN

1. Hasil Penelitian

Hasil dari penelitian dengan judul Implementasi Algoritma Sequential Searching Pada Aplikasi E-Office Di Kantor Tukar Pos Udara Soekarno Hatta Jakarta 19000 adalah sebagai berikut:

A. Halaman Login



Gambar 2. Tampilan Login

Pada saat user ingin masuk ke sistem, maka harus *login* terlebih dahulu. Hanya *user* yang sudah ditambahkan oleh Super Admin yang dapat masuk ke sistem tentunya dengan menginputkan *username* dan *password*. Pada sistem ini terdapat 6 macam leve pengguna diantaranya:

1. Super Admin

Pengguna dengan level super admin hanya bisa menambah pengguna (*user*) dan memanager pengguna.

2. Cek

Pengguna dengan level cek bertugas sebagai pengecekan surat cuti, menyetujui dan tidak menyetujui surat cuti

3. Kepala Kantor

Pengguna dengan level kepala kantor dapat mengakses keseluruhan menu diantaranya menu surat cuti (tambah surat cuti, surat cuti saya, disetujui, belum disetujui), menu surat P6 (tambah surat P6, manage surat P6), menu surat P6a (tambah surat P6a, manage surat P6a), menu surat SPKL (tambah SPKL, manage SPKL).

4. Manager

Pengguna dengan level manager dapat mengakses keseluruhan menu diantaranya menu surat cuti (tambah surat cuti, surat cuti saya, disetujui, belum disetujui), menu surat P6 (tambah surat P6, manage surat P6), menu surat P6a (tambah surat P6a, manage surat P6a), menu surat SPKL (tambah SPKL, manage SPKL).

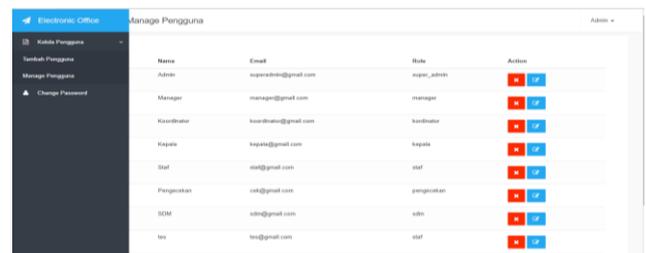
5. Koordinator

Pengguna dengan level koordinator dapat mengakses keseluruhan menu diantaranya menu surat cuti (tambah surat cuti, surat cuti saya, disetujui, belum disetujui), menu surat P6 (tambah surat P6, manage surat P6), menu surat P6a (tambah surat P6a, manage surat P6a), menu surat SPKL (tambah SPKL, manage SPKL).

6. Staff

Pengguna dengan level staff dapat mengakses menu diantaranya menu surat cuti (tambah surat cuti, manage surat cuti), menu surat SPKL (tambah SPKL, manage SPKL).

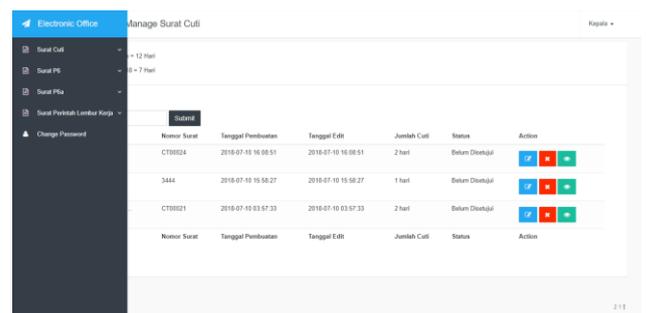
B. Halaman Super Admin



gambar 3. Tampilan Super Admin

Super Admin ketika berada dalam sistem dapat melakukan tambah pengguna dan manage pengguna.

C. Halaman Dashboard E-Office

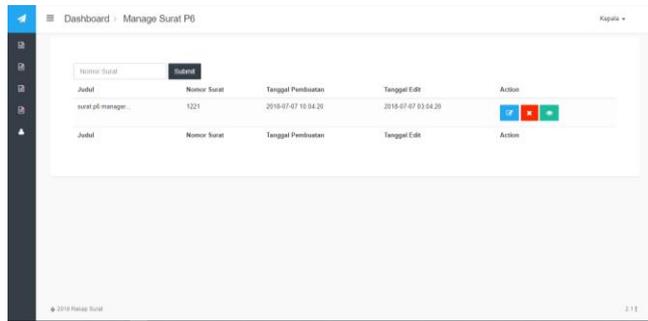


Gambar 4. Tampilan Dashboard E-Office

Ketika *user* masuk ke sistem dengan level kepala, manager dan koordinator dapat mengakses keseluruhan menu diantaranya menu surat cuti (tambah surat cuti, surat cuti saya, disetujui, belum disetujui), menu surat P6 (tambah surat P6, manage surat P6), menu surat

P6a (tambah surat P6a, manage surat P6a), menu surat SPKL (tambah SPKL, manage SPKL).

D. Halaman Manage Surat



Gambar 5. Tampilan Manage Surat

Ketika *user* dengan level kepala, manager, koordinator, dan staff masuk ke halaman manage surat, maka dapat melakukan edit data surat, delete data surat, cetak surat serta pencarian surat berdasarkan nomor surat.

2. Pembahasan

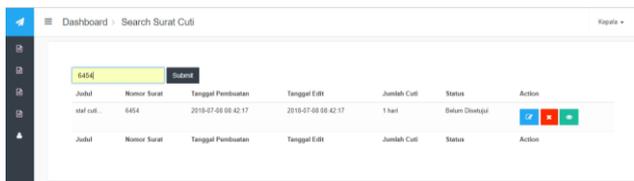
A. Implementasi Algoritma *Sequential Searching*

Sequential searching adalah teknik pencarian data dimana data dicari secara urut dari depan kebelakang atau dari awal sampe akhir. Pada setiap halaman manage surat terdapat fitur pencarian seperti gambar di bawah ini:



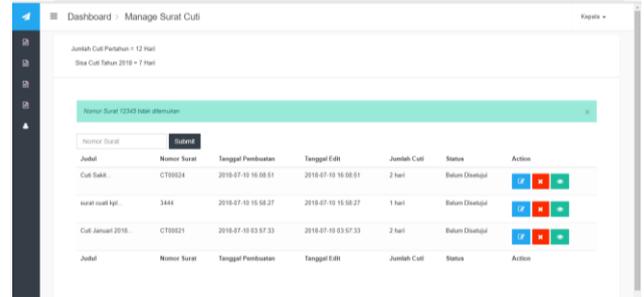
Gambar 6. Kolom pencarian

Pada fitur pencarian inilah *Sequential Searching* bekerja ketika user menginputkan nomor surat pada kolom pencarian maka sistem akan bekerja dan mencari data kemudian menampilkan surat yang sesuai dengan nomor yang di inputkan. Jika data yang di cari ditemukan, maka sistem akan menampilkan seperti gambar di bawah ini:



Gambar 7. Hasil Pencarian Surat Dengan Nomor 6454

Namun jika nomor yang di cari dan inputkan tidak ada maka sistem akan menampilkan seperti gambar dibawah ini:



Gambar 8. Hasil Pencarian Surat Dengan Nomor 12345

Berikut adalah kode algoritma *sequential searching* pada sistem yang dibangun:

```
public function searchAlgoritmaSequential(Request $req){
    $title = 'Search Surat Cuti';

    $SuratCuti = new SuratCuti();
    $data = SuratCuti::all();//select* from surat cuti

    // searched value
    $ketemu = false;// awalnya kosong belum ketemu
    foreach ($data as $item) { //apakah data ini sama
        dengan data yang di cari
            if ($item->nomor_surat == $req->
                input('nomor_surat')) {
                $SuratCuti = $item;
                $ketemu = true;
                break;
            }
    }

    if ($ketemu == false) {
        Session::flash('success','Nomor Surat '.$req->
            input('nomor_surat').' tidak ditemukan');
        return redirect('/Auth::user()->login_role./suratcuti/manage');
    }else{
        return view('admin.suratcuti.surat', ['title'=>$title,
            'data'=>[$SuratCuti], 'sis_cuti'=>'null', 'tahun'=>'null']);
    }
}
```

B. Proses Kerja Algoritma *Sequential Searching*

Semua data yang tersimpan pada *database* dipanggil kemudian disimpan pada *array*. Pada array data yang tersimpan hanya nomor surat saja karena pencarian akan dilakukan berdasarkan nomor surat saja.

Berikut proses kerja algoritma *sequential searching*:

1. Menentukan jumlah data yang tersimpan pada *array*
`$data = SuratCuti::all`
2. Melakukan perulangan sebanyak jumlah data dan melakukan pencarian data beruntun atau sekuensial terhadap data-data yang tersimpan data *array*

```

foreach ($data as $item)
3. Menentukan kecocokan data yang tersimpan di array
dengan kata kunci yang dicari, dengan perintah IF
if ($item->nomor_surat == $req-
>input('nomor_surat')) {
4. Jika data yang dicari ditemukan atau cocok maka nilai
akan dikembalikan dengan nilai pencarian dihentikan
$SuratCuti = $item;
$ketemu = true;
break;
5. Jika data tidak ditemukan, maka nilai akan
dikembalikan
if ($ketemu == false) {
    Session::flash('success','Nomor Surat
'. $req->input('nomor_surat'). ' tidak ditemukan');
    return redirect('/.Auth::user()-
>login_role./suratcuti/manage');
} else {
    return view('admin.suratcuti.surat',
['title'=>$title, 'data'=>[$SuratCuti],
'sisa_cuti'=>'null', 'tahun'=>'null']);

```

V. KESIMPULAN

Kesimpulan dari penelitian ini bahwa Kantor Tukar Pos Udara Soekarno Hatta Jakarta 19000 dalam pengolahan dan pengarsipan data serta nomor surat masih menggunakan cara manual yaitu dengan mencatat (tuliskan) pada buku dan melihat ulang urutan surat sebelumnya untuk memberi nomor pada surat baru. Cara tersebut tentu tidak efisien karena memerlukan buku yang tebal dan membutuhkan waktu yang cukup lama dalam proses perekapan. Masalah lainnya yaitu penyimpanan berkas yang terlalu banyak sehingga menjadi kendala saat melakukan pencarian data. Maka dari itu, penulis mengusulkan untuk membangun sistem *E-office* dengan mengimplementasikan Algoritma *Sequential Searching* sehingga masalah pada sistem sebelumnya dapat ditanggulangi. Berikut hasil dari penelitian ini adalah:

1. Dengan adanya aplikasi *e-office* mengubah pengolahan serta pengarsipan data persuratan dari manual menjadi digital.
2. Meningkatnya efektifitas dalam pengolahan, pendistribusian, serta pencarian dokumen persuratan dengan adanya implementasi algoritma *sequential searching* pada aplikasi *e-office*.

REFERENSI

- [1] Lestariningsih, E., Ardianto, E., & Handoko, W. (2015). Rancang Bangun E-Office Administrasi Surat Di Bagian Humas Universitas STIKUBANK Semarang. *Dinamika Informatika-Vol7.7*, 102.
- [2] Wade, R. (2015). Pengembangan Aplikasi E-Office Pencatatan Dokumen Pada Badan Kepegawaian

- Daerah Provinsi Jawa Barat Dengan Menggunakan Metode RAD. *e-Proceeding of Engineering: Vol 2, No.3*, 3.
- [3] Sonita, A., & Sari, M. (2018). Implementasi Algoritma Sequential Searching Untuk Pencarian Nomor Surat Pada Sistem Arsip Elektronik. *Jurnal Pseudocode, Volume V No 1*, 1.
 - [4] Lamhot, S. (2015). *Algoritma dan Pemrograman*. Yogyakarta: Andi.
 - [5] Sari, I., & Megawati. (2018). Evaluasi Budaya Organisasi Dalam Pemanfaatan Aplikasi E-Office Untuk Mendukung Penerapan E-Government Pada Kesbangpol Provinsi Riau. *Pseudocode Volume 5, No.1*, 2.
 - [6] Sidik, B. (2014). *Pemrograman Web Dengan PHP Revisi Kedua*. Bandung: Informatika.
 - [7] Siregar, E. (2015). *Zend Framework 2.X: Solusi Mempercepat Pengembangan Aplikasi Web Dengan PHP 5.x Framework*. Yogyakarta: C.V Andi Offset.
 - [8] A.S., R., & Shalahuddin, M. (2016). *Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek)*. Bandung: Informatika.