

# PEMBANGUNAN APLIKASI KLASIFIKASI KODE SURAT BERBASIS ANDROID MENGGUNAKAN ALGORITMA BOYER-MOORE DI KANTOR KECAMATAN CIPARAY

Yosep Bahtiar, Rustiyana, Yusuf Muharam

**ABSTRAK:** (1) Arsiparis merupakan sebutan untuk orang yang memiliki kompetensi dalam bidang kearsipan, biasanya profesi arsiparis dapat ditemui di lingkungan kantor pemerintahan. Salah satu tugas arsiparis adalah melakukan pengelolaan arsip masuk atau keluar. Dalam kegiatan pengelolaan tersebut arsiparis akan melakukan pencocokan kode pada arsip/surat menggunakan daftar klasifikasi kode arsip yang memiliki lebih dari 2000 indeks secara manual, dengan indeks sebanyak itu, tentunya pencocokan kode yang dilakukan oleh arsiparis tidak efisien. Penggunaan aplikasi berbasis Android dapat membuat pencarian dan pencocokan kode arsip lebih cepat dan efisien. Agar kinerja pencarian aplikasi lebih cepat dibutuhkan algoritma khusus yang mencocokkan string. Algoritma pencocokan string yang masih populer saat ini adalah algoritma Boyer-Moore, algoritma ini dianggap memiliki hasil paling baik, karena dalam praktiknya, algoritma tersebut bergerak mencocokkan string dari arah kanan ke kiri. Penelitian ini dilakukan dengan membangun aplikasi bernama Klasifikasi Kode Surat berbasis Android yang dapat mencari kode arsip dengan mengimplementasikan algoritma Boyer-Moore sebagai algoritma yang melakukan pencocokan string. (2) Metode penelitian yang digunakan adalah metode Waterfall, metode ini memiliki pendekatan sistematis dan berurutan pada proses pengembangan perangkat lunaknya. (3) Setelah pembangunan aplikasi selesai, aplikasi diuji dengan menggunakan metode black-box testing dan pengujian penerimaan oleh pengguna. Black-box testing dilakukan agar aplikasi tidak memiliki kesalahan pada antarmuka dan kinerja, sedangkan pengujian penerimaan dilakukan agar tingkat kelayakan aplikasi dapat dinilai secara langsung oleh pengguna. (4) Diketahui bahwa berdasarkan hasil black-box testing dan pengujian penilaian yang dilakukan pengguna, aplikasi dapat berjalan secara lancar dan dinilai efektif serta layak untuk digunakan dalam mencari klasifikasi kode surat.

**Kata Kunci:** algoritma Boyer-Moore, algoritma pencarian string, Android, Eclipse, Klasifikasi Kode Surat, SQLite.

**ABSTRACT:** (1) Archivist is a term for people who have competence in the field of archives, usually the archivist profession can be found in government offices. One of the archivist's duties is to manage incoming or outgoing records. In these management activities, archivists will perform code matching on archives/letters using a list of archive code classifications that have more than 2000 indexes manually, with that many indexes, of course the code matching performed by archivists is not efficient. The use of Android-based applications can make searching and matching archive code faster and more efficient. For faster application search performance, a special algorithm that matches strings is needed. The string matching algorithm that is still popular today is the Boyer-Moore algorithm, this algorithm is considered to have the best results, because in practice, the algorithm moves to match strings from right to left. This research was conducted by building an application called Android-based Letter Code Classification that can search for archive codes by implementing the Boyer-Moore algorithm as an algorithm that performs string matching. (2) The research method used is the Waterfall method, this method has a systematic and sequential approach to the software development process. (3) After the application development is complete, the application is tested using the black-box testing method and user acceptance testing. Black-box testing is carried out so that the application does not have errors in the interface and performance, while acceptance testing is carried out so that the feasibility level of the application can be assessed directly by the user. (4) It is known that based on the results of black-box testing and assessment tests carried out by users, the application can run smoothly and is considered effective and feasible to use in searching for letter code classifications.

**Keywords:** Boyer-Moore algorithm, string search algorithm, Android, Eclipse, Letter Code Classification, SQLite.

## PENDAHULUAN

Arsip (atau sering disebut dengan surat), merupakan naskah-naskah yang dibuat dan diterima oleh lembaga negara dan badan pemerintahan dalam bentuk corak apapun, baik dalam keadaan tunggal maupun berkelompok dalam rangka pelaksanaan kegiatan pemerintahan. Pada sebuah instansi pemerintahan, pengarsipan seringkali diterjemahkan sebagai suatu kegiatan manajemen naskah-naskah yang dianggap penting atau bersejarah sesuai dengan prosedur yang ada. Salah satunya pada instansi pemerintah tingkat kecamatan yaitu Kantor Kecamatan Ciparay. Kantor kecamatan ciparay seperti halnya kantor kecamatan lain di Indonesia, mempunyai tugas melaksanakan sebagian kewenangan pemerintah kabupaten di wilayah kerjanya, yang mencakup bidang pemerintahan, ekonomi, pembangunan, kesejahteraan rakyat dan pembinaan kehidupan masyarakat serta urusan pelayanan umum lainnya yang dilimpahkan oleh pemerintah kabupaten.

Dalam proses pelimpahan wewenangnnya pemerintah kabupaten berkomunikasi dengan kantor kecamatan biasanya menggunakan arsip yang ditujukan ke camat atau bidang tertentu yang mewakili tugas khusus atau disebut juga SKPD (satuan kerja perangkat daerah) di kantor kecamatan berdasarkan SOP yang berlaku. Arsip yang dikirimkan terlebih dahulu akan diterima dan dikelola oleh Arsiparis, yang merupakan sebutan untuk orang yang memiliki kompetensi dalam bidang kearsipan. Tugas pokok seorang arsiparis adalah pengelolaan arsip dinamis dan arsip statis, pembinaan kearsipan, serta pengelolaan dan penyajian arsip menjadi informasi, tetapi jika tidak ada yang memegang jabatan arsiparis, sekretariat lain di kantor kecamatan akan merangkap sebagai arsiparis.

Ketika melakukan pengelolaan arsip masuk atau keluar, arsiparis akan mencocokkan kode pada arsip dengan yang ada pada daftar klasifikasi kode arsip agar arsip masuk atau keluar dapat didata, klasifikasi kode arsip itu sendiri memiliki jumlah lebih dari 2000 indeks, akibatnya kegiatan pencocokan atau pencarian yang dilakukan arsiparis akan tidak efisien. Dengan dibangunnya sebuah aplikasi yang dapat mencari klasifikasi kode arsip, maka pekerjaan arsiparis bisa terbantu karena pengoperasiannya hanya tinggal menginputkan klasifikasi kode yang ingin dicari tanpa terlebih dahulu melihat kepada buku klasifikasi kode arsip. Namun dikarenakan indeks data

klasifikasi kode arsip yang berjumlah sangat banyak, proses pencarian aplikasi akan berjalan dengan lambat, belum lagi aplikasi tersebut haruslah dibangun pada sistem operasi yang mendukung mobilitas agar dapat dibawa dan dibuka dimana saja.

Untuk menjawab permasalahan terkait mobilitas aplikasi, sistem operasi yang mendukung mobilitas dan yang banyak digunakan oleh masyarakat saat ini adalah Android OS pada *smartphone*. Sedangkan untuk mempercepat dan mempermudah suatu proses pencarian, dibutuhkan suatu algoritma yang dapat memaksimalkan proses pencarian tersebut. Algoritma pencarian yang dianggap memiliki hasil paling baik dalam praktiknya, yaitu algoritma yang bergerak mencocokkan *string* dari arah kanan ke kiri. Algoritma Boyer-Moore merupakan salah satu contoh algoritma yang menggunakan pencocokan *string* dari arah kanan ke kiri. Berdasarkan pemaparan sebelumnya, aplikasi klasifikasi kode arsip harus mengimplementasikan algoritma Boyer-Moore dan berbasis Android, maka judul yang diambil oleh peneliti adalah “PEMBANGUNAN APLIKASI KLASIFIKASI KODE SURAT BERBASIS ANDROID MENGGUNAKAN ALGORITMA BOYER-MOORE DI KANTOR KECAMATAN CIPARAY”.

Berdasarkan latar belakang di atas, diperoleh rumusan masalah diantaranya:

1. Apakah aplikasi Klasifikasi Kode Surat dapat dirancang menggunakan basis Android?
2. Apakah algoritma Boyer-Moore dapat diimplementasikan pada aplikasi Klasifikasi Kode Surat berbasis Android?
3. Apakah penggunaan aplikasi Klasifikasi Kode Surat efektif dan layak digunakan?

Berdasarkan rumusan masalah yang telah dikemukakan di atas, tujuan dari penelitian ini yaitu:

1. Merancang aplikasi yang dapat mempermudah pencarian klasifikasi kode surat dinas resmi berbasis Android.
2. Mengimplementasikan algoritma Boyer-Moore pada aplikasi Klasifikasi Kode Surat berbasis Android.
3. Mengetahui efektifitas penggunaan dan tingkat kelayakan aplikasi Klasifikasi Kode Surat.

Penelitian Terdahulu yang digunakan sebagai landasan teori oleh peneliti adalah sebagai berikut:

1. (Yogyawan, 2016), dalam papernya yang berjudul "Implementasi Boyer-Moore Pada Aplikasi Pencarian Rumus Matematika dan Fisika", diketahui bahwa metode *string matching* dengan algoritma Boyer-Moore dapat diterapkan pada aplikasi pencarian rumus matematika dan fisika tingkat SMA, 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 yang cocok dengan soal yang dikerjakan oleh pengguna.
2. (Rifqo & Andilala, 2020), dalam papernya yang berjudul "Implementasi Algoritme Boyer-Moore pada Aplikasi Kamus Istilah Komputer Berbasis Android", diketahui bahwa algoritma Boyer-Moore dapat diterapkan dalam perancangan aplikasi kamus istilah komputer sehingga dapat memudahkan pengguna untuk mencari kata atau istilah komputer yang ingin diterjemahkan.
3. (Lestari et al., 2016), dalam papernya yang berjudul "Perancangan Aplikasi Kamus Istilah Medis Berbasis Android dengan Algoritma Boyer-Moore", diketahui bahwa algoritma Boyer-Moore dapat diterapkan dalam perancangan aplikasi kamus istilah medis sehingga dapat memudahkan pengguna untuk mencari kata yang ingin diterjemahkan.
4. (Sari, 2019), dalam papernya yang berjudul "Perancangan Aplikasi Kamus Bahasa Indonesia ke Bahasa Arab dengan Algoritma Boyer Moore berbasis Android", diketahui bahwa penerapan algoritma pencocokan string Boyer-Moore dapat digunakan untuk mencari kata-kata bahasa Arab dengan melakukan pencocokan terhadap kalimat yang di-input-kan menggunakan bahasa Indonesia.
5. (Daeli & Hondro, 2017), dalam papernya yang berjudul "Perancangan Aplikasi Pencarian Kata dengan Kombinasi Algoritma Knuth Morris Pratt dan Algoritma Boyer Moore", diketahui bahwa aplikasi pencarian kata yang dirancang dengan algoritma Knuth-Morris-Pratt dan Boyer-Moore mampu menampilkan hasil berdasarkan kata

kunci pada file berekstensi .txt di komputer.

Kajian teori yang digunakan dalam penelitian ini diantaranya adalah:

1. Klasifikasi Kearsipan, Menurut (Tasyhar, 2013) Kearsipan berarti penyimpanan secara tetap dan teratur terhadap warkat-warkat penting mengenai kemajuan organisasi. Penyimpanan arsip dilakukan dengan menggunakan cara tertentu secara sistematis dengan maksud untuk membantu dan mempermudah dalam penyimpanan dan penemuan kembali arsip tersebut. Metode penyimpanan yang sistematis tersebut sering disebut dengan sistem penyimpanan arsip (*filing system*).
2. Algoritma Boyer-Moore, Menurut (Sari, 2019) Algoritma Boyer-Moore adalah salah satu algoritma untuk mencari suatu string di dalam teks yang dibuat oleh R.M Boyer dan J.S Moore. Algoritma Boyer-Moore melakukan perbandingan dimulai dari kanan ke kiri, tetapi pergeseran window tetap dari kiri ke kanan. Jika terjadi kecocokkan maka dilakukan perbandingan karakter teks dan karakter pola yang sebelumnya, yaitu dengan sama-sama mengurangi indeks teks dan pola masing-masing sebanyak satu. Dengan menggunakan algoritma ini, secara rata-rata proses pencarian akan menjadi lebih cepat jika dibandingkan dengan algoritma lainnya.
3. Menurut (Safaat H., 2015), Android merupakan subset perangkat lunak untuk ponsel yang meliputi sistem operasi, middleware dan aplikasi kunci yang dirilis oleh Google. Saat ini disediakan Android SDK sebagai alat bantu untuk mulai mengembangkan aplikasi pada platform Android menggunakan bahasa pemrograman Java.
4. SQLite, Menurut (Setiawan, 2007), SQLite adalah RDBMS (Sistem Manajemen Basis Data Relasional) alternatif yang bersifat portable (tidak memerlukan proses instalasi), cepat, gratis, dan didukung oleh banyak bahasa pemrograman.
5. Eclipse ADT Bundle, Menurut (Juhara, 2016) Eclipse ADT adalah plugin Eclipse IDE (Integrated Development Environment) yang menambahkan fungsionalitas pengembangan aplikasi Android di Eclipse IDE.

## METODA

Metode penelitian yang dilakukan dalam penelitian ini adalah menggunakan metode pengembangan perangkat lunak Waterfall. Menurut (A.S. & Shalahuddin, 2019), metode waterfall sering juga disebut dengan model sekuensial linier atau alur hidup yang terurut dari tahap analisis, desain, pengkodean, dan pengujian.

Metode pengumpulan data yang dilakukan pada penelitian ini meliputi:

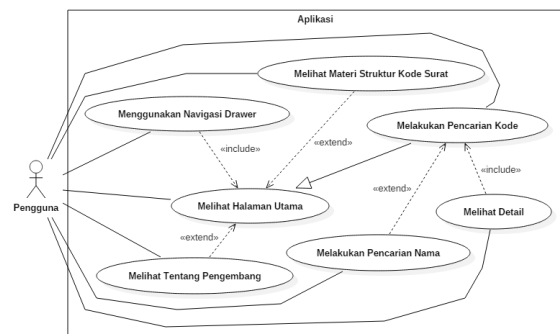
1. Observasi, peneliti melakukan pengamatan dan analisis secara langsung ke Kantor Kecamatan Ciparay untuk memperoleh informasi dan gambaran yang jelas terhadap objek penelitian.
2. Studi pustaka, peneliti mencari dan mempelajari pustaka berbentuk jurnal, buku, serta artikel di internet yang relevan terkait dengan objek penelitian.
3. Kuesioner, peneliti melakukan pengumpulan data menggunakan lembar pertanyaan isian yang diisi oleh subjek penelitian terkait dengan kebutuhan aplikasi dan penilaian terhadap aplikasi.

- d. Algoritma pencocokan string yang perlu dipakai adalah algoritma Boyer-Moore, karena dalam praktiknya algoritma ini dianggap memiliki hasil paling baik.

## 2. Perancangan Aplikasi

Perancangan proses aplikasi ini digunakan Unified Modelling Language pada perangkat lunak StarUML yang terdiri dari use case diagram, class diagram, activity diagram, dan sequence diagram.

- 1) Use case diagram dibuat untuk mengetahui gambaran fungsi apa saja yang ada di dalam sistem aplikasi. Berikut adalah use case diagram pada aplikasi Klasifikasi Kode Surat.



Gambar 1 Use Case Aplikasi

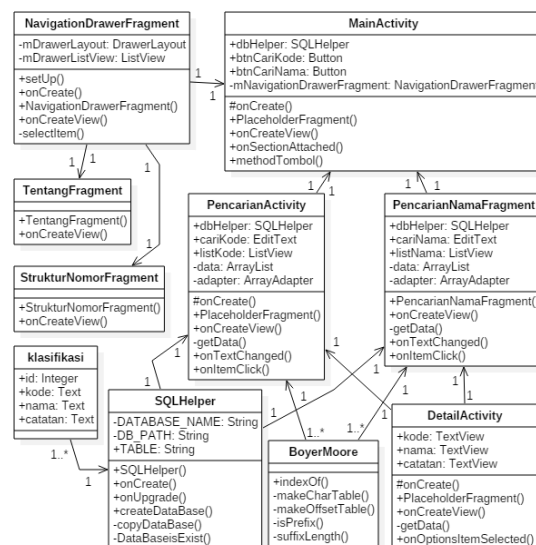
## HASIL DAN PEMBAHASAN

### 1. Analisis Masalah

Analisis yang dilakukan pada tahap ini adalah analisis terkait pengelolaan surat masuk dan keluar yang dilakukan oleh petugas kearsipan. Berikut adalah hasil dari tahap analisis:

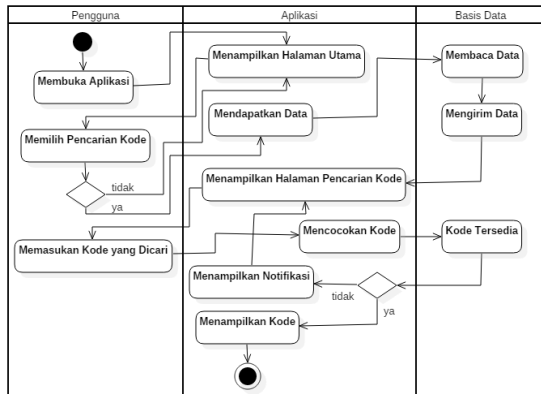
- a. Petugas kearsipan membutuhkan aplikasi yang dapat mencari klasifikasi kode arsip, karena pencarian kode yang dilakukan oleh arsiparis masih manual sehingga proses pencarian memakan waktu yang tidak sebentar.
- b. Aplikasi yang dibutuhkan adalah aplikasi berbasis Android, karena aplikasi pencarian klasifikasi kode arsip ini harus mendukung konsep mobilitas, yaitu mudah diakses dan dapat dibawa kemana-mana.
- c. Aplikasi membutuhkan algoritma pencocokan string, mengingat bahwa jumlah indeks dalam klasifikasi kode arsip tidak sedikit.

- 2) Class diagram dibuat untuk menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun aplikasi.



Gambar 2 Class Diagram Aplikasi

- 3) Activity diagram adalah model aktifitas yang ada dalam sistem. Pada Activity Diagram Proses Pencarian, activity meliputi gambaran aktifitas pencarian yang dilakukan oleh aplikasi.



Gambar 3 Activity Diagram Pencarian

### 3. Implementasi Algoritma

Pembahasan Algoritma Boyer Moore: Contoh pencocokan string yang terjadi jika diketahui sebuah teks memiliki string SAYASUKAMAKAN, dengan pattern yang dicari adalah MAKAN, maka pencarian yang dilakukan algoritma Boyer-Moore adalah sebagai berikut:

- Membuat tabel delta dari pattern yang dicari (occurrence heuristic). Karakter pada pattern yang diketahui adalah M-A-K-N, maka dengan menggunakan teknik looking glass (melihat dari kanan) pada Boyer-Moore, karakter N akan memiliki indeks = 0, A = 1 dan 3 (1 merupakan nilai yang muncul paling awal dari pada 3), K = 2, dan M = 4 seperti pada tabel berikut:

Tabel 1 Occurrence Heuristic (OH)

Indeks	M	A	K	N
0				
1				
2				
3				
4				

Karakter Pattern	M	A	K	N
Kemunculan Paling Awal (OH)	4	1	2	0

- Setelah tabel delta diketahui, proses pencocokan string dengan menggunakan algoritma Boyer-Moore adalah sebagai berikut:

Tabel 2 Proses Pencocokan String

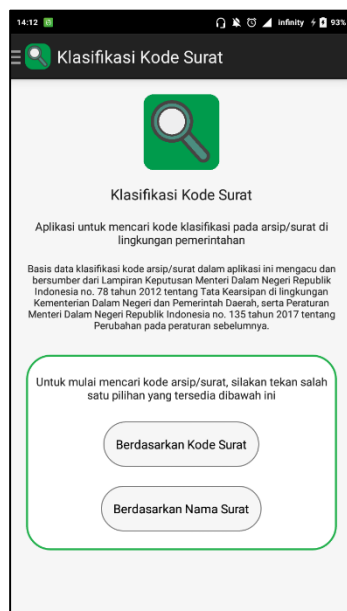
Indeks	0	1	2	3	4	5	6	7	8	9	10	11	12
Teks	S	A	Y	A	S	U	K	A	M	A	K	A	N
Pattern	M	A	K	A	N								

- Proses pencocokan string yang pertama berdasarkan pada tabel diatas adalah proses mencocokkan karakter dari pattern paling kanan yang memiliki indeks 4 dengan karakter dari teks yang memiliki indeks sama seperti pattern, yaitu karakter N dengan karakter S. Karena karakter  $N \neq S$  dan karakter S pada tabel delta tidak ada, maka pattern digeser sebanyak jumlah karakter yang dimiliki oleh pattern, yaitu 5.
- Pencocokan kedua adalah pencocokan karakter N pada pattern dengan karakter A pada teks. Karena karakter  $N \neq A$  tetapi karakter A terdapat pada tabel delta dengan nilai  $OH(A) = 1$ , maka pattern digeser sebanyak jumlah OH pada karakter A, yaitu 1.
- Pencocokan ketiga adalah pencocokan karakter N pada pattern dengan karakter K pada teks. Karena karakter  $N \neq K$  tetapi karakter K terdapat pada tabel delta dengan nilai  $OH(K) = 2$ , maka pattern digeser sebanyak jumlah OH pada karakter A, yaitu 2.
- Pencocokan keempat adalah pencocokan karakter N pada pattern dengan karakter N pada teks. Karena karakter  $N=N$ , maka pencocokan karakter sukses dan dilanjutkan dengan mencocokkan karakter pada indeks teks dan pattern sebelumnya, yaitu karakter A pada pattern dengan karakter A pada teks.
- Setelah dilakukan pencocokan dari kanan ke kiri teks terhadap pattern dan ditemukan kecocokan seluruh pattern pada teks. Maka algoritma akan berhenti dan mengembalikan mengembalikan nilai 8, yaitu nilai dimana indeks pada teks ditemukan kecocokan yang terakhir kalinya dengan pattern.

Setelah mengetahui cara kerja algoritma Boyer-Moore, peneliti lalu mengimplementasikan algoritma tersebut kepada kelas dengan nama file BoyerMoore.java, kelas tersebut akan mengembalikan nilai integer antara  $0 \dots n-1$ , dimana  $n$  adalah banyaknya indeks pada teks yang merupakan tempat pattern yang akan dicari, atau dalam hal ini teks tersebut dianggap sebagai suatu baris pada suatu kolom field kode atau nama surat dalam basis data.

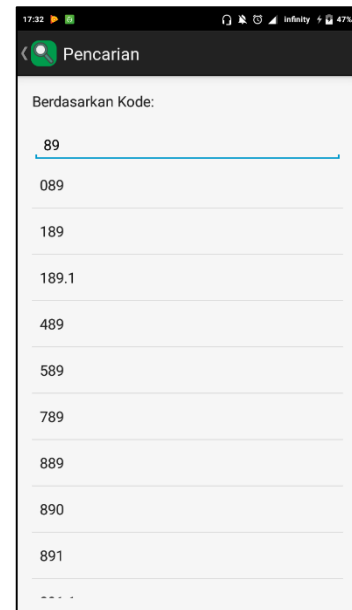
Hasil dari implementasi algoritma, basis data, antarmuka, dan prosedur-prosedur lain tersebut adalah aplikasi Klasifikasi Kode Surat.apk yang dapat dipasang dan dijalankan pada sistem operasi Android. Berikut adalah hasil tangkapan layar dari aplikasi yang dijalankan peneliti pada perangkat smartphone:

- 1) Halaman Utama, merupakan tampilan yang akan tampil pertama kali saat aplikasi dibuka oleh pengguna, terlihat pada gambar dibawah bahwa komponen-komponen user interface Halaman Utama telah dibangun sesuai dengan rancangan pada tahap perancangan sebelumnya.



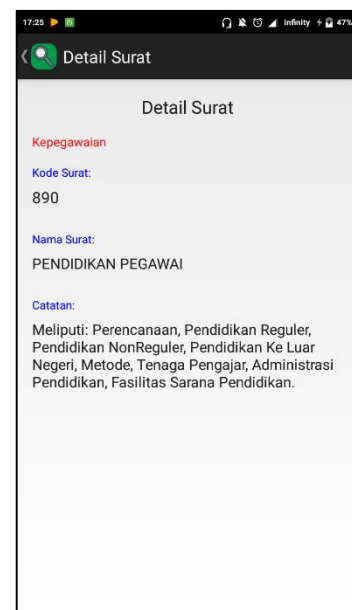
**Gambar 4 Tampilan Halaman Utama**

- 2) Halaman Pencarian Berdasarkan Kode ini merupakan halaman yang digunakan oleh pengguna untuk mencari klasifikasi surat berdasarkan kodenya.



**Gambar 5 Tampilan Pencarian**

- 3) Halaman Detail Surat ini merupakan halaman yang tampil setelah sebuah list view ditekan pada halaman pencarian, baik halaman Pencarian Berdasarkan Kode maupun halaman Pencarian Berdasarkan Nama.



**Gambar 6 Tampilan Detail Surat**

- 4) Halaman Tentang Pengembang ini mempunyai komponen user interface gambar dan teks. Informasi yang ditampilkan adalah informasi tentang peneliti dan tujuan tentang pembuatan aplikasi.



**Gambar 7 Tampilan Tentang**

#### 4. Pengujian Aplikasi

Setelah aplikasi selesai dipasang dan dijalankan pada perangkat keras yang digunakan peneliti, selanjutnya adalah pelaksanaan tahap pengujian. Tahap pengujian ini dilakukan dengan metode black-box testing dan pengujian kelayakan terhadap aplikasi yang telah dibangun.

Pengujian black-box digunakan untuk mengetahui unjuk kerja dari aplikasi dengan cara menguji fungsionalitas dari aplikasi. Berikut hasilnya:

**Tabel 3 Pengujian *Black-Box***

Halaman Utama	
Menekan tombol Menu Navigasi	+Berhasil
Menekan tombol Pencarian Berdasarkan Kode	+Berhasil
Menekan tombol Pencarian Berdasarkan Nama	+Berhasil
Halaman Pencarian Berdasarkan Kode	
Menekan tombol Up/ Back Button	+Berhasil
Mencari klasifikasi berdasarkan kode	+Berhasil
Menampilkan notifikasi data tidak tersedia	+Berhasil

Menekan data yang tampil pada List View	+Berhasil
Halaman Pencarian Berdasarkan Nama	
Menekan tombol Up/ Back Button	+Berhasil
Mencari klasifikasi berdasarkan nama	+Berhasil
Menampilkan notifikasi data tidak tersedia	+Berhasil
Menekan data yang tampil pada List View	+Berhasil
Halaman Detail Surat	
Menekan tombol Up/ Back Button	+Berhasil
Halaman Menu Navigasi	
Menekan tombol Halaman Utama	+Berhasil
Menekan tombol Halaman Struktur Nomor Surat	+Berhasil
Menekan tombol Tentang Pengembang	+Berhasil
Halaman Struktur Nomor Surat	
Menekan tombol Menu Navigasi	+Berhasil
Halaman Tentang Pengembang	
Menekan tombol Menu Navigasi	+Berhasil

Sedangkan berdasarkan kuesioner penilaian pengguna terhadap aplikasi, dapat disimpulkan bahwa aplikasi Klasifikasi Kode Surat yang dibangun peneliti layak digunakan oleh petugas kearsipan untuk mencari kode klasifikasi surat.

#### SIMPULAN DAN SARAN

Berdasarkan hasil analisis, perancangan, pengujian dengan menggunakan metode black-box testing dan pengujian efektifitas dan kelayakan aplikasi menggunakan kuesioner pada penelitian ini, didapatkan hasil bahwa:

1. Aplikasi dapat mengimplementasikan pencocokan string dengan menggunakan algoritma Boyer-Moore.
2. Pencarian pada aplikasi dengan menggunakan algoritma Boyer-Moore dapat menampilkan keseluruhan indeks yang mengandung pattern yang dicari,

dan tentunya hasil pencarian tersebut ditampilkan dengan sangat cepat oleh aplikasi.

3. Aplikasi Klasifikasi Kode Surat layak digunakan untuk mencari klasifikasi kode arsip/surat di lingkungan pemerintahan daerah.

Dalam proses pembangunan aplikasi Klasifikasi Kode Surat berbasis Android menggunakan algoritma Boyer-Moore di Kantor Kecamatan Ciparay ini banyak sekali memiliki kekurangan dan jauh dari sempurna, sehingga peneliti berharap agar peneliti selanjutnya dapat mengembangkan lagi aplikasi ini, seperti:

1. Basis data aplikasi yang bisa diperbaharui secara online agar menyesuaikan dengan ketentuan tata kearsipan di lingkungan pemerintahan.
2. Penampilan data terkait tingkatan klasifikasi kode surat/arsip pada halaman Pencarian Berdasarkan Kode dan Nama, supaya klasifikasi kode pokok masalah utama dan kode sub pokok masalah dapat dibedakan dengan mudah.

## PUSTAKA ACUAN

- A.S., R., & Shalahuddin, M. (2019). *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek* (Revisi). Informatika.
- Daeli, M. M. Y., & Hondro, R. K. (2017). Perancangan Aplikasi Pencarian Kata dengan Kombinasi Algoritma Knuth Morris Pratt dan Algoritma Boyer Moore. *Majalah Ilmiah INTI*, XII(2), 271–275. <https://ejurnal.stmik-budidarma.ac.id/index.php/inti/article/view/380/362>
- Juhara, Z. P. (2016). *Panduan Lengkap Pemrograman ANDROID* (P. S. Wibowo (ed.); 1 ed.). ANDI OFFSET.
- Lestari, C. P., Hasibuan, N. A., & Ginting, G. L. (2016). Perancangan Aplikasi Kamus Istilah Medis Berbasis Android Dengan Algoritma Boyer-Moore. *Jurnal INFOTEK*, II(3), 1–6. <https://ejurnal.stmik-budidarma.ac.id/index.php/komik/article/view/1661/1337>
- Rifqo, M. H., & Andilala. (2020). Implementasi Algoritme Boyer-Moore pada Aplikasi Kamus Istilah Komputer Berbasis Android. *Jurnal Pseudocode*, VII(1), 69–77. <https://ejournal.unib.ac.id/index.php/pseudocode/article/download/10478/5283>
- Safaat H., N. (2015). *ANDROID: Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*. Informatika.
- Sari, L. (2019). Perancangan Aplikasi Kamus Bahasa Indonesia ke Bahasa Arab dengan Algoritma Boyer Moore berbasis Android. *Jurnal Pelita Informatika*, VIII(2), 189–192. <https://ejurnal.stmik-budidarma.ac.id/index.php/pelita/article/view/1820>
- Setiawan, D. (2007). Mengenal SQLite Command Line. In *IlmuKomputer.Com* (Vol. 3). IlmuKomputer.Com. <https://ilmukomputer.org/wp-content/uploads/2013/09/di2k-SQLite-Command-Line.pdf>
- Tasyhar, M. (2013). *Kearsipan 1 Bahan Ajar Kurikulum 2013 Sekolah Menengah Kejuruan Program Keahlian Administrasi Perkantoran* (W. Sucipto & S. Prihatin (ed.)). [https://bsd.pendidikan.id/data/2013/kelas\\_10smk/Kelas\\_10\\_SMK\\_Kearsipan\\_1.pdf](https://bsd.pendidikan.id/data/2013/kelas_10smk/Kelas_10_SMK_Kearsipan_1.pdf)
- Yogyawan, H. A. (2016). Implementasi Boyer-Moore Pada Aplikasi Pencarian Rumus Matematika dan Fisika. *Jurnal Ilmiah Teknologi Informasi Terapan (JITTER)*, III(1), 74–85. <http://journal.widyatama.ac.id/index.php/jitter/article/view/124/114>