

**IMPLEMENTASI *SELECTION SORT* PADA APLIKASI  
SURAT KELUAR DI KELURAHAN BULUKERTO  
BERBASIS WEB**

**SKRIPSI**

Diajukan Sebagai Salah Satu Syarat  
Untuk Memperoleh Gelar Sarjana Jenjang Strata Satu (S1)  
Pada Program Studi Teknik Informatika Fakultas Teknik  
Universitas Muhammadiyah Ponorogo



**ALIEF VERSA HERDIANSYAH**

20533244

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TEKNIK  
UNIVERSITAS MUHAMMADIYAH PONOROGO**

**2025**

## HALAMAN PENGESAHAN

Nama : Alief Versa Herdiansyah  
NIM : 20533244  
Program Studi : Teknik Informatika  
Fakultas : Teknik  
Judul Skripsi : Implementasi *Selection Sort* pada Aplikasi Surat Keluar di  
Kelurahan Bulukerto Berbasis Web

Isi dan formatnya telah disetujui dan dinyatakan memenuhi syarat  
Untuk melengkapi persyaratan guna memperoleh Gelar Sarjana  
pada Program Studi Teknik Informatika Fakultas Teknik Universitas  
Muhammadiyah Ponorogo

Ponorogo, 10 Februari 2025

Menyetujui,

Adi Fajaryanto Cobantoro, S. Kom, M. Kom.

NIK. 19840924 201309 13

Ismail Abdurrozzaq Zulkarnain, S. Kom., M. Kom

NIK. 19880728 201804 13



Dekan Fakultas Teknik,



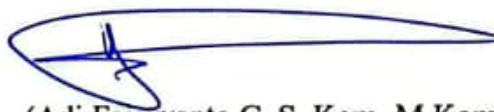
(Edy Kurniawan, ST., MT)

NIK. 19771026 200810 12

Mengetahui,

Ketua Program Studi

Teknik Informatika,



(Adi Fajaryanto C, S. Kom, M. Kom)

NIK. 19840924 201309 13

## PERNYATAAN ORISINALITAS SKRIPSI

Yang bertanda tangan di bawah ini :

Nama : Alief Versa Herdiansyah

NIM : 20533244

Program Studi : Teknik Informatika

Dengan ini menyatakan bahwa Skripsi saya dengan judul : “Implementasi *Selection Sort* pada Aplikasi Surat Keluar di Kelurahan Bulukerto Berbasis Web” bahwa berdasarkan hasil penelusuran berbagai karya ilmiah, gagasan dan masalah ilmiah yang saya rancang/ teliti di dalam Naskah Skripsi ini adalah asli dari pemikiran saya. Tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata di dalam Naskah Skripsi ini dapat dibuktikan terdapat unsur-unsur plagiarisme, saya bersedia Ijazah saya dibatalkan, serta diproses sesuai dengan pengaturan perundangan-undangan yang berlaku

Demikian pernyataan ini dibuat dengan sesungguhnya dan dengan sebenar-benarnya.

Ponorogo, 10 Januari 2025

Mahasiswa,



Alief Versa Herdiansyah

NIM.20533244

## HALAMAN BERITA ACARA UJIAN

Nama : Alief Versa Herdiansyah  
NIM : 20533244  
Program Studi : Teknik Informatika  
Fakultas : Teknik  
Judul Skripsi : Implementasi *Selection Sort* pada Aplikasi Surat Keluar di Kelurahan Bulukerto Berbasis Web

Telah diuji dan dipertahankan dihadapan  
Dosen Penguji tugas akhir jenjang Strata Satu (S1) pada:

Hari : Selasa  
Tanggal : 31 Desember 2024

Dosen Penguji,

Adi Fajaryanto Cobantoro, S. Kom, M. Kom. (Ketua Penguji) : .....  
NIK. 19840924 201309 13  
Khoiru Nurfitri, S. Kom., M. Kom. ( Anggota Penguji I) : .....  
NIK. 19920430 201808 13  
Jamilah Karaman, S. Kom., M. Kom. (Anggota Penguji II) : .....  
NIK. 19900322 201909 13



Dekan Fakultas Teknik,



(Edy Kurniawan, ST., MT)  
NIK. 19771026 200810 12

Mengetahui,

Ketua Program Studi  
Teknik Informatika,



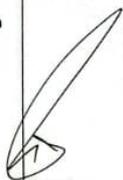
(Adi Fajaryanto C, S. Kom, M. Kom.)  
NIK. 19840924 201309 13

## BERITA ACARA BIMBINGAN SKRIPSI PEMBIMBING I

### BERITA ACARA BIMBINGAN SKRIPSI

Nama : ALIEF VERSA HERDIANSYAH.....  
 NIM : 20533244.....  
 Judul Skripsi : IMPLEMENTASI SELECTION SORT PADA APLIKASI SURAT KELUAR  
 DI KELUAAHAN BILUKERTO BERBASIS WEB.....  
 Dosen Pembimbing I : ADI FAJAR YANTO COBANTORO, S.KOM, M.KOM.....

#### PROSES PEMBIMBINGAN

No	Tanggal	Materi Yang Dikonsultasikan	Saran Pembimbing / Hasil	Tanda Tangan
1	26/24/03	Pengajuan Tema	Tema sudah sesuai, pastikan lingkup penelitian tidak terlalu luas.	
2	28/24/03	Topik Penelitian	Topik penelitian yang fokus dan mudah dikembangkan	
3	2/04	BAB I Latar Belakang Rumusan Masalah	Jelaskan permasalahan lebih rinci	
4	05/24/04	BAB I Rumusan Masalah Tujuan Penelitian	Buat rumusan masalah lebih spesifik Sesuaikan tujuan penelitian agar lebih terukur.	

No.	Tanggal	Materi Yang Dikonsultasikan	Saran Pembimbing / Hasil	Tanda Tangan
5	12/24/09	BAB I & II Penelitian Terdahulu Latar Belakang	Cari referensi jurnal terbaru yang relevan dengan Selection Sort	
6	19/04	BAB I, II & III - Latar Belakang - Rancangan Sistem	Pastikan flowchart dan rancangan sistem mudah dipahami	
7	6/05	BAB III - Perancangan Sistem - Perancangan Tabel	Periksa kesesuaian rancangan tabel dengan kebutuhan sistem.	
8	21/05	Naskah	Acc Simpon	
9	10/24/06	Revisi BAB I - Latar Belakang - Rumusan Masalah	Perjelas latar belakang dengan data pendukung	
10	24/24/06	Revisi BAB II - Tinjauan Pustaka	- Tambah referensi terkait Algoritma.	

No.	Tanggal	Materi Yang Dikonsultasikan	Saran Pembimbing / Hasil	Tanda Tangan
11	8/24 7	Bab III Metode Penelitian	Pastikan Alur metode dijelaskan secara runtut.	
12	22/24 7	Flowchart Selection Sort	Tambahkan penerapan Algoritma Selection Sort	
13	5/24 8	Uji Coba Aplikasi dan Pengumpulan Data	Perbaiki Format Hasil Uji Coba Aplikasi	
14	19/24 8	Revisi BAB 4 Hasil & Pembahasan	Tambahkan Tampilan Interface dari Aplikasi Surat Keluar Kelurahan Bulukerto.	
15	9/24 9	Penyusunan BAB V Kesimpulan dan Saran	Ringkas Kesimpulan sesuai dengan tujuan penelitian	
16	20/24 12	Review kelengkapan skripsi & Persiapan Sidang Akhir	Periksa konsistensi penulisan	

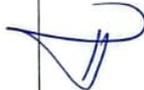
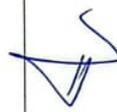
No.	Tanggal	Materi Yang Dikonsultasikan	Saran Pembimbing / Hasil	Tanda Tangan
17	20/12 24	Naskah dan Aplikasi	ACC SIDANG SKRIPSI	

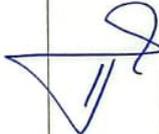
## BERITA ACARA BIMBINGAN SKRIPSI PEMBIMBING II

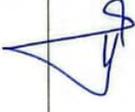
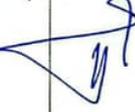
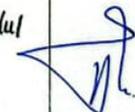
### BERITA ACARA BIMBINGAN SKRIPSI

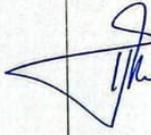
Nama : ALIEF VERSA HERDIANSYAH.....  
NIM : 20533244.....  
Judul Skripsi : IMPLEMENTASI SELECTION SORT PADA APLIKASI SURAT KELUAR  
: DI KELLUKAHAN BULUKERTO BERBASIS WEB.....  
Dosen Pembimbing II : ISMAIL ABDURROZZAQ, Z. S. Kom, M. Kom.....

#### PROSES PEMBIMBINGAN

No	Tanggal	Materi Yang Dikonsultasikan	Saran Pembimbing / Hasil	Tanda Tangan
1	26/3/24	Pengajuan Tema	Tema sudah sesuai	
2	28/3/24	Topik Penelitian	Topik lebih spesifik terkait penerapan Algoritma selection sort	
3	2/4/24	BAB I Latar Belakang Rumusan Masalah	Perkuat latar belakang dengan data sebagai dasar penelitian	
4	5/4/24	BAB I - Rumusan Masalah - Tujuan Penelitian	Rumusan masalah diperjelas. Sesuaikan tujuan penelitian agar menjawab rumusan masalah.	

No.	Tanggal	Materi Yang Dikonsultasikan	Saran Pembimbing / Hasil	Tanda Tangan
5	12/24 1	BAB I & II Penelitian Terdahulu Latar Belakang	Tambahkan Penelitian terkait Algoritma Seleksi sort	
6	19/24 4	BAB I, II & III - Latar Belakang - Rancangan Sistem	Pastikan ada kesinambungan antara latar belakang , tinjauan pustaka, dan rancangan sistem.	
7	6/24 5	BAB III - Perancangan Sistem - Perancangan Tabel	Sesuaikan tabel database dengan kebutuhan data	
8	21/24 E	NASKAH	ACC Lempro	
9	17/24 6	Pemeriksaan Sistematika Penulisan BAB I	Perbaiki Struktur latar belakang agar runtut dan jelas	
10	1/24 7	Penulisan Rumusan Masalah dan Tujuan Penelitian	Gandakan istilah yang sama dengan rumusan dan tujuan penelitian	

No.	Tanggal	Materi Yang Dikonsultasikan	Saran Pembimbing / Hasil	Tanda Tangan
11	15/24 7	Penulisan BAB II Tinjauan Pustaka	Tambahkan Referensi terbaru untuk mendukung teori	
12	29/24 4	BAB III Metode Penelitian	Perjelas langkah algoritma selection sort	
13	12/24 8	BAB IV Hasil & Pembahasan	Partikan tabel diberikan penjelasan yang jelas.	
14	26/24 8	Revisi BAB IV Hasil dan Pembahasan - Penyesuaian format	Periksa keseragaman font, margin, jarak spasi	
15	16/24 9	Penyusunan Daftar Pustaka	Gunakan Format Pedoman Daftar Pustaka sesuai dengan Buku Pedoman.	
16	20/24 12	Review keseluruhan Sistematika Skripsi	Periksa ulang kesesuaian judul, sub judul dan daftar isi.	

No.	Tanggal	Materi Yang Dikonsultasikan	Saran Pembimbing / Hasil	Tanda Tangan
17	20/12 24	Naskah dan Aplikasi	ACC SIDANG SKRIPSI	

# SURAT KETERANGAN HASIL PLAGIASI SKRIPSI



UNIVERSITAS MUHAMMADIYAH PONOROGO  
LEMBAGA LAYANAN PERPUSTAKAAN  
Jalan Budi Utomo No. 10 Ponorogo 63471 Jawa Timur Indonesia  
Telp. (0352) 481124, Fax (0352) 461796, e-mail : lib@umpo.ac.id  
website : [www.library.umpo.ac.id](http://www.library.umpo.ac.id)  
TERAKREDITASI A  
(SK Nomor 000137/ LAP.PT/ III.2020)  
NPP. 3502102D2014337

## SURAT KETERANGAN HASIL *SIMILARITY CHECK* KARYA ILMIAH MAHASISWA UNIVERSITAS MUHAMMADIYAH PONOROGO

Dengan ini kami nyatakan bahwa karya ilmiah ilmiah dengan rincian sebagai berikut :

**Nama** : ALIEF VERSA HERDIANSYAH  
**NIM** : 20533244  
**Judul** : IMPLEMENTASI SELECTION SORT PADA APLIKASI SURAT KELUAR DI  
KELURAHAN BULUKERTO BERBASIS WEB  
**Fakultas / Prodi** : TEKNIK INFORMATIKA

**Dosen pembimbing :**

1. ADI FAJARYANTO COBANTORO, S.Kom., M. Kom
2. ISMAIL ABUDRROZZAQ ZULKARNAIN, S. Kom., M. Kom

Telah dilakukan check plagiasi berupa **SKRIPSI** di Lembaga Layanan Perpustakaan Universitas Muhammadiyah Ponorogo dengan prosentase kesamaan sebesar **20 %**

Demikian surat keterangan dibuat untuk digunakan sebagaimana mestinya.

Ponorogo, 06/02/2025  
Kepala Lembaga Layanan Perpustakaan



Ayu Wulansari, S.Kom, M.A  
NIK. 19760811 201111 21

**NB: Dosen pembimbing dimohon untuk melakukan verifikasi ulang terhadap kelengkapan dan keaslian karya beserta hasil cek Turnitin yang telah dilakukan**

# SURAT KETERANGAN HASIL PLAGIASI ARTIKEL



UNIVERSITAS MUHAMMADIYAH PONOROGO  
LEMBAGA LAYANAN PERPUSTAKAAN  
Jalan Budi Utomo No. 10 Ponorogo 63471 Jawa Timur Indonesia  
Telp. (0352) 481124, Fax (0352) 461796, e-mail : lib@umpo.ac.id  
website : [www.library.umpo.ac.id](http://www.library.umpo.ac.id)  
TERAKREDITASI A  
(SK Nomor 000137/ LAP.PT/ III.2020)  
NPP. 3502102D2014337

## SURAT KETERANGAN HASIL *SIMILARITY CHECK* KARYA ILMIAH MAHASISWA UNIVERSITAS MUHAMMADIYAH PONOROGO

Dengan ini kami nyatakan bahwa karya ilmiah ilmiah dengan rincian sebagai berikut :

**Nama** : ALIEF VERSA HERDIANSYAH  
**NIM** : 20533244  
**Judul** : IMPLEMENTASI SELECTION SORT PADA APLIKASI SURAT KELUAR DI  
KELURAHAN BULUKERTO BERBASIS WEB  
**Fakultas / Prodi** : TEKNIK INFORMATIKA

**Dosen pembimbing :**

1. ADI FAJARYANTO COBANTORO, S. Kom., M. Kom
2. ISMAIL ABDURROZZAQ ZULKARNAIN, S. Kom., M.Kom

Telah dilakukan check plagiasi berupa **ARTIKEL** di Lembaga Layanan Perpustakaan Universitas Muhammadiyah Ponorogo dengan prosentase kesamaan sebesar **14 %**

Demikian surat keterangan dibuat untuk digunakan sebagaimana mestinya.

Ponorogo, 06/02/2025  
Kepala Lembaga Layanan Perpustakaan



Ayu Wulansari, S.Kom, M.A  
NIK. 19760811 201111 21

**NB: Dosen pembimbing dimohon untuk melakukan verifikasi ulang terhadap kelengkapan dan keaslian karya beserta hasil cek Turnitin yang telah dilakukan**

## KATA PERSEMBAHAN

Puji Syukur penulis panjatkan kepada Allah SWT, yang telah memberikan kesehatan, rahmat dan hidayah, sehingga penulis masih diberikan kesempatan untuk menyelesaikan skripsi ini, sebagai salah satu syarat untuk mendapatkan gelar kesarjanaan. Walaupun jauh dari kata sempurna, namun penulis bangga telah mencapai pada titik ini, yang akhirnya skripsi ini bisa selesai diwaktu yang tepat.

Seorang *support system* saya pernah berkata, jika mempunyai sebuah tujuan, maka buatlah batas waktu untuk mencapai tujuan tersebut, sehingga hal inilah yang membuat penulis memacu dirinya sampai batas maksimal sehingga dapat menyelesaikan skripsi ini, diwaktu yang tepat.

Skripsi atau Tugas akhir ini saya persembahkan untuk :

1. Ayah dan Ibu, Bapak Maryoto dan Ibu Hetty Yuliana, terimakasih atas doa, semangat, motivasi, pengorbanan, nasehat serta kasih sayang yang tidak pernah henti sampai saat ini.
2. Adikku Nadya Hellen Anindita, terimakasih telah menjadi penyemangat dalam mengerjakan tugas akhir ini.
3. Keluarga Besar Mbah Soyono, Mbah Sri Wigiyati dan semua keluarga yang tidak bisa disebutkan satu-persatu, terimakasih untuk doa, nasehat, masukan dan semangatnya selama ini.
4. Dosen Pembimbing Bapak Ismail Abdurrozzaq Zulkarnain, S.Kom, M.Kom dan Bapak Adi Fajaryanto Cobantoro, S.Kom., M.Kom yang sudah membimbing serta memberi masukan dan saran selama ini, sehingga saya dapat menyelesaikan skripsi ini.
5. Teman Seangkatan Teknik Informatika 2020
6. Support system yang selalu ada, memberikan motivasi, bantuan, dan kehadiran yang menjadi sumber kekuatan penulis dalam menyelesaikan tugas akhir ini dengan penuh dedikasi. Terima kasih atas setiap doa, semangat, dan kebersamaan yang telah diberikan.

## MOTTO

***“Sepiro gedhening sengsoro yen tinompo amung dadi cobo”***

Seberapapun besarnya kesengsaraan yang kita hadapi, jika diterima dengan ikhlas,  
maka itu hanya akan menjadi cobaan.



# IMPLEMENTASI *SELECTION SORT* PADA SURAT KELUAR DI KELURAHAN BULUKERTO BERBASIS WEB

Alief Versa Herdiansyah<sup>1</sup>, Adi Fajaryanto Cobantoro<sup>2</sup>, Ismail Abdurrozzaq Zulkarnain<sup>3</sup>  
<sup>1,2,3</sup>Program Studi Teknik Informatika, Fakultas Teknik  
<sup>1,2,3</sup>Universitas Muhammadiyah Ponorogo  
Email : [aliefversaherdiansyah@gmail.com](mailto:aliefversaherdiansyah@gmail.com)

## Abstrak

Penelitian ini bertujuan untuk mengimplementasikan algoritma *Selection Sort* pada surat keluar di Kelurahan Bulukerto berbasis Web. Kelurahan Bulukerto menghadapi permasalahan dalam pengelolaan surat keluar, seperti kesulitan dalam pengurutan surat secara manual yang memakan waktu lama dan rentan terhadap kesalahan manusia. Sistem yang diusulkan mengimplementasikan algoritma *Selection Sort* untuk mengurutkan data surat berdasarkan kriteria tertentu, seperti tanggal dan nomor surat. Proses pengurutan dilakukan dengan memilih elemen terkecil dari daftar yang belum terurut dan menempatkannya di posisi awal secara berulang hingga seluruh data tersusun rapi. Sistem ini dirancang menggunakan metode *waterfall*, meliputi tahapan analisis, desain, implementasi, dan pengujian. Pengujian *white box* menunjukkan algoritma *Selection Sort* bekerja sesuai logika dengan memastikan elemen terkecil ditempatkan pada posisi yang tepat. Hasil implementasi membuktikan bahwa sistem ini efektif dalam mengelola data surat keluar, meningkatkan efisiensi kerja, dan meminimalkan kesalahan pengelolaan data. Selain itu, fitur berbasis *web* memungkinkan penggunaan sistem tanpa bergantung pada koneksi internet, mendukung kelancaran operasional di daerah dengan akses internet terbatas. Sistem ini memberikan solusi yang optimal dalam meningkatkan kualitas pelayanan administrasi Kelurahan Bulukerto melalui pengelolaan surat yang lebih terorganisasi dan akurat.

**Kata Kunci : Algoritma *Selection Sort*, Metode *Waterfall*, Pengelolaan Surat Keluar, *Website*.**

## KATA PENGANTAR

Puji syukur ke hadirat Allah SWT karena atas rahmat-Nya yang telah menganugerahkan segala rahmat, taufiq serta hidayah-Nya, sehingga penulis dapat menyelesaikan tugas akhir ini dengan judul “Implementasi *Selection Sort* Pada Aplikasi Surat Keluar Di Kelurahan Bulukerto Berbasis Web”.

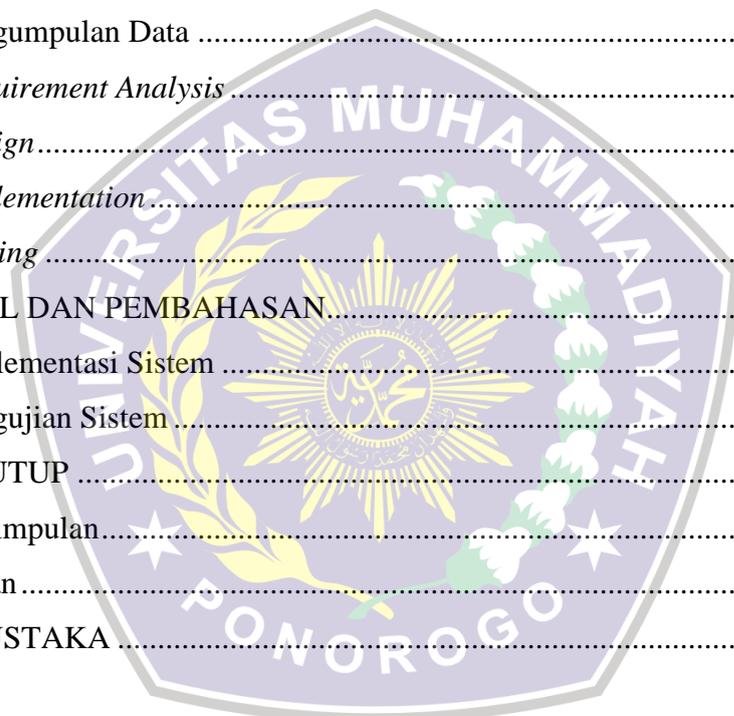
Selama penyusunan tugas akhir ini, penulis banyak memperoleh bantuan, bimbingan serta dorongan dari berbagai pihak. Oleh karena itu, dengan segenap kerendahan hati, pada kesempatan ini penulis sampaikan terima kasih kepada:

1. Kedua orang tua, seluruh keluarga serta orang-orang terdekat yang telah memberikan doa serta dukungan sehingga tugas akhir ini dapat terselesaikan dengan baik.
2. Bapak Dr. Rido Kurnianto, M.Ag.selaku Rektor Universitas Muhammadiyah Ponorogo.
3. Bapak Ir. Aliyadi, MM, M.Kom selaku Dekan Fakultas Teknik Universitas Muhammadiyah Ponorogo.
4. Bapak Adi Fajariyanto, S.Kom, M.Kom selaku Ketua Prodi Teknik Informatika Universitas Muhammadiyah Ponorogo.
5. Bapak Ismail Abdurrozzaq Zulkarnain, S.Kom, M.Kom yang terhormat, selaku dosen pembimbing dengan ketulusan hatinya telah memberikan bimbingan dalam menyelesaikan tugas akhir ini.
6. Bapak dan Ibu Dosen Jurusan Teknik Informatika Universitas Muhammadiyah Ponorogo.
7. Semua pihak yang telah membantu dalam penyusunan laporan skripsi ini yang tidak dapat penulis sebutkan satu per satu.

## DAFTAR ISI

HALAMAN PENGESAHAN.....	ii
PERNYATAAN ORISINALITAS SKRIPSI .....	iii
HALAMAN BERITA ACARA UJIAN .....	iv
BERITA ACARA BIMBINGAN SKRIPSI PEMBIMBING I .....	v
BERITA ACARA BIMBINGAN SKRIPSI PEMBIMBING II .....	viii
SURAT KETERANGAN HASIL PLAGIASI SKRIPSI .....	xi
SURAT KETERANGAN HASIL PLAGIASI ARTIKEL .....	xii
KATA PERSEMBAHAN.....	xiii
MOTTO .....	xiv
Abstrak .....	xv
Penelitian ini bertujuan .....	xv
KATA PENGANTAR .....	xvi
DAFTAR ISI.....	xvii
DAFTAR GAMBAR .....	xix
DAFTAR TABEL.....	xx
BAB 1 PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	3
1.3 Tujuan Penelitian.....	3
1.4 Batasan Masalah.....	3
1.5 Manfaat Penelitian.....	3
BAB 2 TINJAUAN PUSTAKA .....	4
2.1 Penelitian Terdahulu.....	4
2.2 Sistem Pengelolaan Surat Keluar .....	6
2.3 <i>Website</i> .....	7
2.4 HTML.....	7
2.5 PHP.....	7
2.6 MySql .....	8
2.7 <i>Database</i> .....	9

2.8	<i>Selection Sort</i> .....	9
2.9	DFD ( <i>Data Flow Diagram</i> ).....	9
2.10	ERD ( <i>Entity Relationship Diagram</i> ) .....	11
2.11	<i>Flowchart</i> .....	13
2.12	<i>Metode Waterfall</i> .....	14
2.13	<i>White box testing</i> .....	14
BAB 3 METODE PENELITIAN.....		16
3.1	Tahapan Penelitian .....	16
3.1.1	Identifikasi Masalah .....	16
3.1.2	Pengumpulan Data .....	17
3.1.3	<i>Requirement Analysis</i> .....	18
3.1.4	<i>Design</i> .....	19
3.1.5	<i>Implementation</i> .....	30
3.1.6	<i>Testing</i> .....	31
BAB 4 HASIL DAN PEMBAHASAN.....		32
4.1	Implementasi Sistem .....	32
4.2	Pengujian Sistem .....	39
BAB 5 PENUTUP .....		44
5.1	Kesimpulan.....	44
5.2	Saran.....	44
DAFTAR PUSTAKA .....		46



## DAFTAR GAMBAR

Gambar 3.1 Tahapan Penelitian .....	16
Gambar 3.2 <i>Flowchart</i> Sistem Surat Keluar .....	20
Gambar 3.3 <i>Flowchart Selection Sort</i> .....	21
Gambar 3.4 Diagram Konteks.....	22
Gambar 3.5 DFD level 1 .....	23
Gambar 3.6 <i>Usecase Diagram</i> .....	24
Gambar 3.7 <i>Entity Relationship Diagram</i> .....	25
Gambar 3.8 <i>Login</i> .....	27
Gambar 3.9 <i>Dashboard Admin</i> .....	28
Gambar 3.10 Surat Keluar.....	28
Gambar 3.11 Data Penduduk .....	29
Gambar 4.1 Implementasi Algoritma <i>Selection Sort</i> .....	33
Gambar 4.2 Koneksi Database.....	34
Gambar 4.3 <i>Login</i> .....	36
Gambar 4.4 <i>Dashboard Admin</i> .....	36
Gambar 4.5 Data Penduduk .....	37
Gambar 4.6 Tampilan Surat Keterangan Domisili.....	38
Gambar 4.7 Tampilan Buat Surat.....	38
Gambar 4.8 Tampilan Arsip Surat Keluar .....	39
Gambar 4.9 <i>Flow graph Method</i> “\$allSurats = collect()” dari Class “Surat” .....	43

## DAFTAR TABEL

Tabel 2.1 Penelitian Terdahulu .....	4
Tabel 2.2 Komponen <i>Data Flow Diagram</i> .....	10
Tabel 2.3 Komponen ERD .....	11
Tabel 2.4 Komponen <i>Flowchart</i> .....	13
Tabel 3.1 tabel Admin .....	26
Tabel 3.2 tabel data penduduk .....	26
Tabel 3.3 tabel surat keluar .....	27
Tabel 4.1 <i>Pseudocode Method</i> “\$allSurats = collect()” dari Class “Surat” .....	40



# **BAB 1**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Kantor Kelurahan Bulukerto merupakan bagian integral dari struktur administratif desa di Kabupaten Wonogiri, Jawa Tengah, yang berperan dalam menyediakan layanan publik bagi penduduk lokal, khususnya di Desa Bulukerto. Sebagai entitas pemerintahan, Kantor Kelurahan Bulukerto bertanggung jawab dalam pengelolaan administratif desa, termasuk pengurusan surat masuk dan surat keluar yang menjadi saluran komunikasi resmi antara Kelurahan Bulukerto dengan berbagai pihak eksternal, seperti warga, instansi pemerintah, dan lembaga masyarakat. Surat masuk dan keluar ini memastikan terjalannya koordinasi dan informasi yang efektif, baik dalam hal penerimaan dokumen maupun penyampaian kebijakan atau tanggapan resmi dari kelurahan kepada pihak-pihak terkait[1].

Salah satu permasalahan utama yang dihadapi adalah kesulitan dalam mengurutkan dan menyusun surat-surat keluar secara efisien sesuai dengan kriteria tertentu, seperti nomor surat atau tanggal pengiriman. Proses pelayanan surat keluar masih dilakukan secara manual dalam pembuatan dan pengurutan surat, yang memakan waktu lama. Dalam proses manual ini, terdapat risiko kesalahan manusia baik dalam pencatatan maupun pengurutan surat-surat keluar. Pengelolaan surat keluar diprioritaskan karena memiliki pola terstruktur dan volume konsisten rata-rata 7 surat per hari atau 140 surat per bulan sehingga pengelolaan manual menjadi kurang optimal. Sementara itu, surat masuk telah dikelola melalui sistem resmi dari pemerintah yaitu Sistem Informasi Kearsipan Dinamis Terintegrasi (SRIKANDI), sehingga tidak memerlukan pengembangan tambahan. Implementasi sistem yang diusulkan diharapkan dapat mengatasi permasalahan ini dengan mengurutkan surat keluar berdasarkan tanggal dan nomor surat dari yang terkecil. Dengan mengurutkan dari yang terkecil, pencarian surat keluar yang dibutuhkan akan menjadi lebih mudah dan cepat[2].

Pengelolaan surat keluar di tingkat kelurahan merupakan salah satu aspek penting dalam mendukung kelancaran administrasi pemerintahan. Dalam Peraturan Bupati Wonogiri Nomor 25 Tahun 2021 tentang Tata Naskah Dinas, pengelolaan surat keluar diatur melalui proses yang mencakup pembuatan konsep surat, paraf berjenjang, penandatanganan, pencatatan, pengiriman, dan penyimpanan dokumen[3]. Proses ini membutuhkan sistem yang efektif untuk memastikan bahwa setiap tahap dilakukan secara tertib dan efisien. Namun, pengelolaan secara manual sering kali menghadapi kendala seperti kesalahan pencatatan, keterlambatan distribusi, dan sulitnya pelacakan dokumen.

Untuk mengatasi permasalahan tersebut perlu solusi yang menggabungkan teknologi informasi dengan praktik administratif yang ada. Salah satu solusi yang dapat diterapkan adalah penggunaan aplikasi web yang mengimplementasikan algoritma *Selection Sort*. Dengan algoritma *Selection Sort* surat-surat keluar akan diurutkan secara cepat dan efisien berdasarkan kriteria yang dibutuhkan sehingga memudahkan dalam pencarian surat keluar tersebut[4]. Jika surat tidak diurutkan, dampak yang ditimbulkan meliputi sulitnya menemukan surat tertentu saat dibutuhkan, meningkatnya risiko kehilangan data, dan memperpanjang waktu yang diperlukan untuk proses pencarian atau verifikasi. Hal ini dapat menghambat kelancaran pelayanan administrasi, mengurangi keakuratan pencatatan, serta meningkatkan beban kerja petugas.

Dengan teknologi ini proses pengelolaan surat keluar dapat diotomatisasikan yang memungkinkan peningkatan efisiensi dalam penataan, dan pemeliharaan arsip surat. Melalui penggunaan solusi ini, diharapkan Kelurahan Bulukerto dapat meningkatkan kualitas pelayanan kepada masyarakat dan mengoptimalkan efisiensi dalam pengelolaan administrasi surat keluar mereka. Berdasarkan permasalahan yang telah diuraikan dapat disimpulkan bahwa penelitian ini akan disusun dengan judul **“Implementasi *Selection Sort* Pada Aplikasi Surat Keluar di Kelurahan Bulukerto Berbasis Web”**.

## 1.2 Rumusan Masalah

Dengan mempertimbangkan informasi sebelumnya, rumusan masalah pada penelitian ini adalah Bagaimana Implementasi *Selection Sort* Pada Surat Keluar di Kelurahan Bulukerto Berbasis Web ?

## 1.3 Tujuan Penelitian

Penelitian ini bertujuan untuk mengimplementasikan algoritma *Selection Sort* Pada Surat Keluar di Kelurahan Bulukerto Berbasis Web.

## 1.4 Batasan Masalah

Untuk mencapai hasil pengembangan sesuai dengan yang telah dijelaskan, penelitian ini dibatasi oleh beberapa hal seperti berikut:

1. Sistem ini dibangun dengan bahasa pemrograman PHP dan Mysql sebagai basis data.
2. Sistem ini hanya dapat diakses oleh satu *user* yakni administrator.
3. Aplikasi *Selection Sort* Pada Surat Keluar di Kelurahan Bulukerto Berbasis Web.
4. Algoritma *Selection Sort* digunakan khusus untuk pengurutan data surat berdasarkan atribut tertentu, seperti `noSurat`, dan tidak mencakup algoritma pengurutan lain atau skenario pengurutan kompleks.

## 1.5 Manfaat Penelitian

Manfaat dari penelitian ini adalah meningkatkan efisiensi waktu dalam pengelolaan surat keluar dan penghematan sumber daya dalam pelayanan administratif secara keseluruhan di Kelurahan Bulukerto.

## BAB 2

### TINJAUAN PUSTAKA

#### 2.1 Penelitian Terdahulu

**Tabel 2.1 Penelitian Terdahulu**

No.	Judul Jurnal	Penulis	Hasil Penelitian
1.	Rancang Bangun Aplikasi Pengelolaan Arsip Inaktif Dinas Kearsipan Dan Perpustakaan Daerah Kabupaten Grobogan Menggunakan Metode <i>Selection Sort</i> [5].	Susanto (2023)	Hasil penelitian ini menunjukkan bahwa Aplikasi Pengelolaan Arsip Inaktif Dinas Kearsipan dan Perpustakaan Daerah Kabupaten Grobogan dengan Metode <i>Selection Sort</i> telah memberikan kontribusi yang positif dalam meningkatkan efisiensi dan efektivitas pengelolaan arsip inaktif di institusi tersebut.
2.	Aplikasi Pengelolaan Surat Keluar Menggunakan <i>Sequential Search</i> Dan <i>Selection Sort</i> Pada Kpu Kota Blitar[6].	Haming, Nilal dkk., (2021)	Hasil dari penelitian ini menunjukkan bahwa penggunaan metode <i>sequential search</i> dan <i>Selection Sort</i> dalam aplikasi pengelolaan surat keluar KPU Kota Blitar telah berhasil. Dengan hasil pengujian yang sesuai dengan desain dan

algoritma yang direncanakan, aplikasi ini berhasil meningkatkan efisiensi dan efektivitas pengelolaan surat keluar di lingkungan KPU Kota Blitar.

3. Sistem Informasi Apotek Lasriana & Hasil dari penelitian ini Berbasis Web Gunaryati, Aris menunjukkan bahwa Menggunakan Algoritma (2022) *Sequential Search* Dan *Selection Sort*[7].. penerapan sistem informasi apotek dapat memberikan manfaat signifikan bagi pemilik dan karyawan apotek dalam mengelola laporan secara terkomputerisasi. Dengan menggunakan *Selection Sort* membantu mengurutkan data barang secara efisien.
4. Implementasi Algoritma Syahputra (2022) Hasil dari penelitian ini *Selection Sort* Untuk menunjukkan bahwa Pengurutan Nilai Ipk penerapan metode *Selection Sort* dalam Mahasiswa Universitas Potensi Utama[8]. dalam pengurutan IPK mahasiswa kelas TIF B Universitas Potensi Utama efektif dalam menyusun laporan penilaian.

5. Implementasi Algoritma Harahap, Fandhepa Hasil dari penelitian ini *Selection Sort* dalam & Effendy, Irman menunjukkan bahwa Membangun Aplikasi (2023) terwujudnya sebuah Android Pemesanan Jasa aplikasi pemesanan jasa Make-up yang memanfaatkan algoritma *Selection Sort* untuk melakukan *sorting* pada pemesanan paket berdasarkan harga.

---

Berdasarkan tabel 2.1 diatas, dapat disimpulkan bahwa perbedaan utama pada penelitian ini yaitu objek penelitiannya yang mana penelitian ini berfokus mengimplementasikan *Selection Sort* Pada Surat Keluar Di Kelurahan Bulukerto Berbasis Web . Selanjutnya penelitian ini menggunakan pendekatan yang berbeda namun masih memiliki tujuan yang sama yaitu mengurutkan surat keluar di Kelurahan Bulukerto. Metode *Selection Sort* dipilih sebagai pendekatan khusus untuk mengurutkan surat keluar berdasarkan urutan keluar surat tersebut.

## 2.2 Sistem Pengelolaan Surat Keluar

Sistem pengelolaan surat keluar merupakan proses administratif yang mencakup pencatatan, pengarsipan, dan distribusi surat yang dikirim dari suatu organisasi. Proses ini bertujuan untuk memastikan surat keluar terdokumentasi dengan baik, dapat dilacak, dan mudah diakses saat diperlukan. Penggunaan teknologi informasi, seperti aplikasi berbasis web, dalam sistem pengelolaan surat keluar memungkinkan otomatisasi dalam pencatatan, penomoran, dan pengarsipan surat, yang meningkatkan efisiensi, akurasi, dan keamanan data. Selain itu, sistem ini memudahkan pencarian surat berdasarkan kriteria tertentu, seperti tanggal atau nomor surat, sehingga mempercepat proses administrasi dan mendukung pelayanan yang lebih baik[9].

### 2.3 Website

*Website* adalah sebuah halaman atau kumpulan halaman yang dapat diakses melalui internet. Halaman-halaman ini biasanya berisi informasi, konten multimedia, dan berbagai fitur interaktif yang dapat diakses oleh pengguna melalui perangkat seperti komputer, laptop, tablet, atau *smartphone*. *Website* dapat digunakan untuk berbagai tujuan, seperti memberikan informasi tentang suatu perusahaan, organisasi, produk, atau layanan. Selain itu, *website* juga dapat digunakan untuk berbagi konten seperti artikel, blog, foto, video, dan masih banyak lagi. Beberapa jenis *website* yang umum meliputi *website* bisnis, *website e-commerce*, blog pribadi, forum online, dan media sosial. *Website* biasanya terdiri dari berbagai elemen, seperti tata letak, teks, gambar, grafik, audio, video, dan *hyperlink* yang menghubungkan halaman-halaman tersebut. Desain visual dan fungsionalitas *website* dapat bervariasi tergantung pada kebutuhan dan preferensi pemilik *website*[10].

### 2.4 HTML

HTML singkatan dari *HyperText Markup Language*, merupakan sebuah bahasa pemrograman yang digunakan untuk membuat dan mengatur tampilan halaman web di *World Wide Web* (WWW). Semua dokumen HTML dapat dibaca dan dipublikasikan di web, dengan ekstensi file biasanya berakhir dengan *.html* atau *.htm*. Para ahli sepakat bahwa HTML adalah bahasa yang sangat tepat untuk menampilkan informasi pada halaman web karena menggunakan konsep *hypertext* dan menyediakan sejumlah perintah (markup) yang memungkinkan pengaturan tampilan informasi secara fleksibel sesuai kebutuhan. Dengan markup yang dimilikinya, HTML memungkinkan pengguna untuk menambahkan teks, gambar, link, dan elemen-elemen lainnya ke dalam halaman web, sehingga memungkinkan interaksi yang dinamis dan penyajian informasi yang menarik bagi pengguna internet[11].

### 2.5 PHP

PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman yang kuat dan serbaguna yang digunakan untuk mengembangkan situs web dinamis.

Dikembangkan pertama kali pada tahun 1994 oleh Rasmus Lerdorf, PHP awalnya dirancang sebagai alat untuk mengelola situs web pribadinya, namun sejak itu telah berkembang menjadi salah satu bahasa pemrograman paling populer di dunia. Salah satu keunggulan utama PHP adalah kemampuannya untuk disematkan secara langsung ke dalam kode HTML, sehingga memungkinkan pengembang untuk membuat situs web yang dinamis dan interaktif dengan mudah. PHP juga mendukung berbagai basis data populer, seperti MySQL, *PostgreSQL*, dan *SQLite*, serta berbagai protokol jaringan seperti HTTP, FTP, dan IMAP. Dengan dukungan yang luas dari komunitas pengembang yang aktif, PHP terus berkembang dan menjadi pilihan utama dalam pengembangan aplikasi web modern[12].

## 2.6 MySql

Seiring berkembangnya zaman, teknologi semakin berkembang pesat termasuk perangkat lunak. Salah satu contoh perangkat lunak adalah MySQL yang selalu di update oleh produsernya masing-masing. MySQL adalah pengembangan lanjutan dari proyek UNIREG yang dikerjakan oleh Michael Monty Widenius dan TcX (perusahaan perangkat lunak asal Swedia). MySQL adalah DBMS yang open source dengan dua bentuk lisensi, yaitu *Free Software* (perangkat lunak bebas) dan *Shareware* (perangkat lunak berpemilik yang penggunaannya terbatas). Jadi MySQL adalah *database* server yang gratis dengan lisensi GNU *General Public License* (GPL) sehingga dapat dipakai untuk keperluan pribadi atau komersial tanpa harus membayar lisensi yang ada. Seperti yang sudah disebutkan sebelumnya, MySQL masuk ke dalam jenis RDBMS (*Relational database Management System*). Maka dari itu, istilah semacam baris, kolom, tabel, dipakai pada MySQL. Contohnya di dalam MySQL sebuah *database* terdapat satu atau beberapa tabel. MySQL merupakan *database engine* atau server *database* yang mendukung bahasa *database SQL* sebagai bahasa interaktif dalam mengelola data. MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau DBMS yang *multithread, multi-user*[13].

## 2.7 Database

Basis data merupakan kumpulan data terorganisir yang saling terkait sehingga dapat disimpan, dimanipulasi, dan diakses dengan mudah oleh pengguna. Konsep relasi dalam hal ini mengacu pada cara data menggambarkan domain khusus, memfasilitasi penggunaan yang efektif dalam menjawab pertanyaan yang diajukan terhadap basis data tersebut. Sistem basis data, di sisi lain, merupakan koleksi data yang terstruktur sedemikian rupa sehingga memungkinkan penyimpanan dan manipulasi data (termasuk pembaruan, pencarian, pengolahan dengan perhitungan tertentu, dan penghapusan) dengan kemudahan[13].

## 2.8 Selection Sort

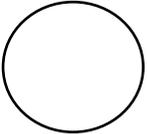
*Selection Sort* merupakan algoritma pengurutan sederhana yang bekerja dengan cara secara berulang memilih elemen terkecil dari sisa *Array* yang belum diurutkan, dan menukarnya dengan elemen pertama dari sisa *Array* tersebut. Proses ini diulangi untuk sisa *Array* yang semakin berkurang setiap iterasi, hingga seluruh *Array* diurutkan. Meskipun sederhana, *Selection Sort* memiliki kompleksitas waktu  $O(n^2)$ , yang membuatnya tidak efisien untuk jumlah data yang besar. Notasi  $O(n^2)$  dalam konteks algoritma *Selection Sort* merujuk pada kompleksitas waktu algoritma tersebut. Dalam notasi Big O,  $O(n^2)$  menunjukkan bahwa waktu eksekusi algoritma *Selection Sort* meningkat secara kuadrat seiring dengan peningkatan ukuran input( $n$ ), di mana  $n$  adalah jumlah elemen dalam *Array* yang diurutkan[2].

## 2.9 DFD (Data Flow Diagram)

*Data Flow Diagram* adalah representasi visual dari sistem yang menunjukkan hubungan antara sistem tersebut dengan entitas eksternal yang berinteraksi dengannya. Diagram ini sering digunakan untuk menggambarkan batasan sistem dan mengidentifikasi masukan (*input*) dan keluaran (*output*) yang terlibat dalam interaksi antara sistem dan entitas eksternal. Diagram konteks menggambarkan aliran informasi antara sistem dan entitas eksternal melalui *input* dan *output*. *Input* adalah masukan yang diberikan kepada sistem,

sedangkan output adalah hasil atau respons yang dihasilkan oleh sistem. Misalnya, *input* dapat berupa data yang dimasukkan oleh pengguna, sementara output dapat berupa tampilan hasil yang diberikan kepada pengguna[14].

**Tabel 2.2 Komponen Data Flow Diagram**

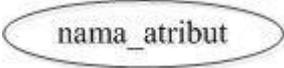
No	Simbol	Nama	Fungsi
1.		Entitas	<p>Simbol ini mepresentasikan dari orang, organisasi, atau sistem yang berada di luar sistem yang sedang dianalisis, tetapi berinteraksi dengan sistem tersebut. Entitas ini bisa berupa pengguna, perangkat keras, atau sistem eksternal lainnya yang memberikan input ke sistem atau menerima output dari sistem.</p>
2.		Arus Data	<p>Simbol ini menggambarkan data tunggal atau kumpulan logis dari data yang mengalir dalam sistem. Arus data selalu bermula atau berakhir di suatu proses dalam DFD. Arus data menunjukkan bagaimana data diproses atau diolah oleh sistem.</p>
3.		Proses	<p>Proses adalah aktivitas atau fungsi dalam sistem yang melakukan operasi tertentu untuk tujuan bisnis yang spesifik. Proses bisa berupa aktivitas manual yang dilakukan oleh pengguna atau proses yang terkomputerisasi. Proses dalam DFD direpresentasikan dengan bentuk lingkaran atau bulatan.</p>

Sumber : (Alhamda et al.,2023)

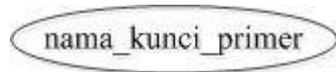
## 2.10 ERD (*Entity Relationship Diagram*)

*Entity Relationship Diagram* adalah sebuah model konseptual yang digunakan untuk merancang basis data relasional. Dikembangkan berdasarkan teori himpunan dalam matematika, ERD memberikan representasi visual tentang entitas-entitas dan hubungan antar entitas dalam sebuah sistem basis data. Setiap entitas direpresentasikan sebagai tabel dengan atribut-atribut yang mencerminkan karakteristiknya. Hubungan antar entitas, seperti satu-ke-satu, satu-ke-banyak, atau banyak-ke-banyak, dijelaskan melalui garis yang menghubungkan tabel-tabel tersebut. ERD memiliki peran penting dalam proses perancangan dan pemahaman struktur basis data, membantu pengembang untuk mengidentifikasi dan memahami interaksi data serta menjaga integritas data melalui penetapan kunci primer dan kunci asing[15].

**Tabel 2.3 Komponen ERD**

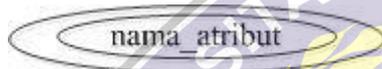
No.	Simbol	Deskripsi
1.	Entitas/ <i>Entity</i> 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel.
2.	Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas.

3. Atribut Kunci Primer



Field atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses record yang diinginkan; biasanya berupa id; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama).

4. Atribut Multinilai/*Multi Value*



*Field* atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.

5. Relasi



Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja.

6.

Assosiasi/*association*



Penghubung antara relasi dan entitas dimana di kedua ujungnya memiliki multiplicity kemungkinan jumlah Pemakaian kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut dengan one to many menghubungkan entitas A dan entitas B.

---

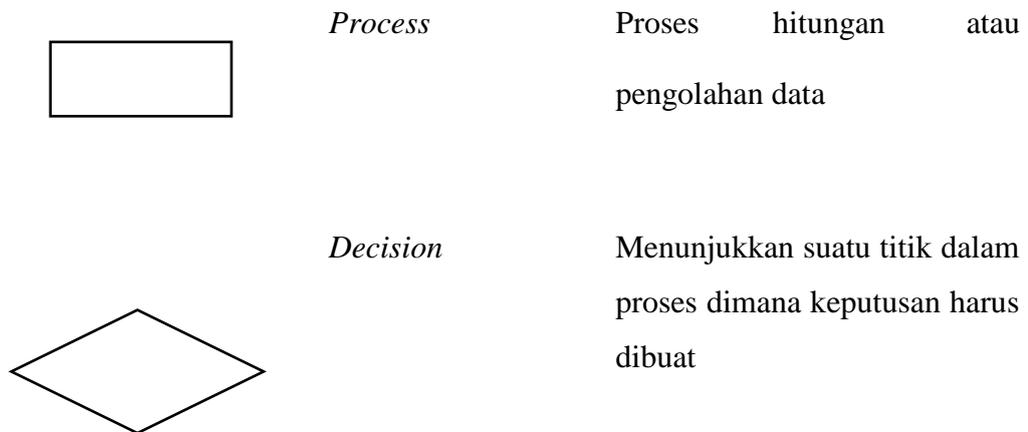
Sumber : (Alhamda et al.,2023)

## 2.11 Flowchart

*Flowchart* atau diagram alir merupakan sebuah representasi visual yang digunakan untuk menunjukkan langkah-langkah instruksi atau algoritma secara berurutan dalam suatu sistem. Biasanya digunakan oleh analis sistem sebagai dokumentasi untuk menjelaskan logika dari sistem yang akan dibangun kepada para programmer. Dengan menggunakan *Flowchart*, solusi untuk berbagai masalah yang mungkin muncul dalam pembangunan sistem dapat diidentifikasi. *Flowchart* terdiri dari simbol-simbol yang mewakili berbagai proses, dan setiap simbol tersebut merepresentasikan suatu tindakan atau langkah. Garis penghubung digunakan untuk menghubungkan satu proses ke proses berikutnya, membentuk urutan logis dari instruksi yang diikuti dalam sistem tersebut[16].

**Tabel 2.4 Komponen Flowchart**

Simbol	Nama	Fungsi
	<i>Terminator</i>	Menandai awal dan akhir dari suatu alur kerja atau proses
	Garis Alir ( <i>Flow Line</i> )	Arah aliran atau urutan dari suatu proses dalam <i>Flowchart</i>
	<i>Input/Output Data</i>	Merepresentasikan masukan atau keluaran dari suatu alur kerja atau proses



Sumber : (Khesya, 2021)

### 2.12 Metode *Waterfall*

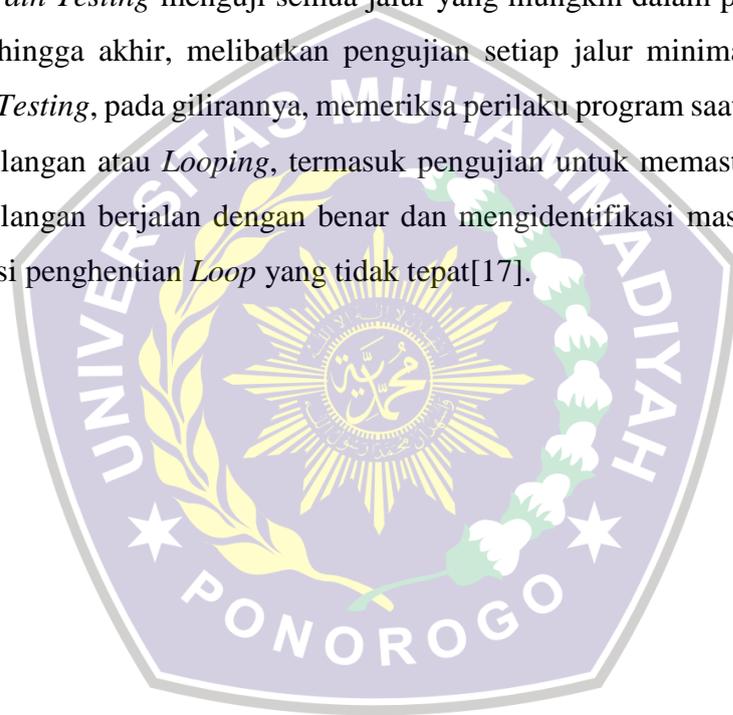
Metode *Waterfall* dalam pengembangan perangkat lunak mengatur proses secara berurutan, dimana setiap langkah dilaksanakan berturut-turut dan tidak bisa dikembalikan ke tahap sebelumnya. Tahapan meliputi analisis kebutuhan perangkat lunak, desain, pembuatan kode program, dan pengujian. Analisis kebutuhan perangkat lunak mencakup pengumpulan informasi untuk menganalisis dan menetapkan kebutuhan perangkat lunak. Tahap desain melibatkan pembuatan desain program perangkat lunak, termasuk struktur data, arsitektur, dan representasi antarmuka, menggunakan notasi seperti *Activity Diagram*, *Use Case Diagram*, DFD, ERD, serta *Logical Record Structure*. Setelah itu, desain diubah menjadi program pada tahap pembuatan kode program, dan diikuti dengan tahap pengujian yang fokus pada pengujian logika dan fungsionalitas perangkat lunak menggunakan metode *blackbox*.

### 2.13 *White box testing*

*White box testing* adalah sebuah metode pengujian perangkat lunak di mana penguji memiliki akses ke dalam struktur internal, kode sumber, dan aliran program dari suatu aplikasi yang diuji. Dalam pengujian ini, penguji memahami struktur dan logika internal aplikasi untuk merancang

dan menjalankan serangkaian tes yang memeriksa apakah semua jalur logika program berjalan dengan benar.

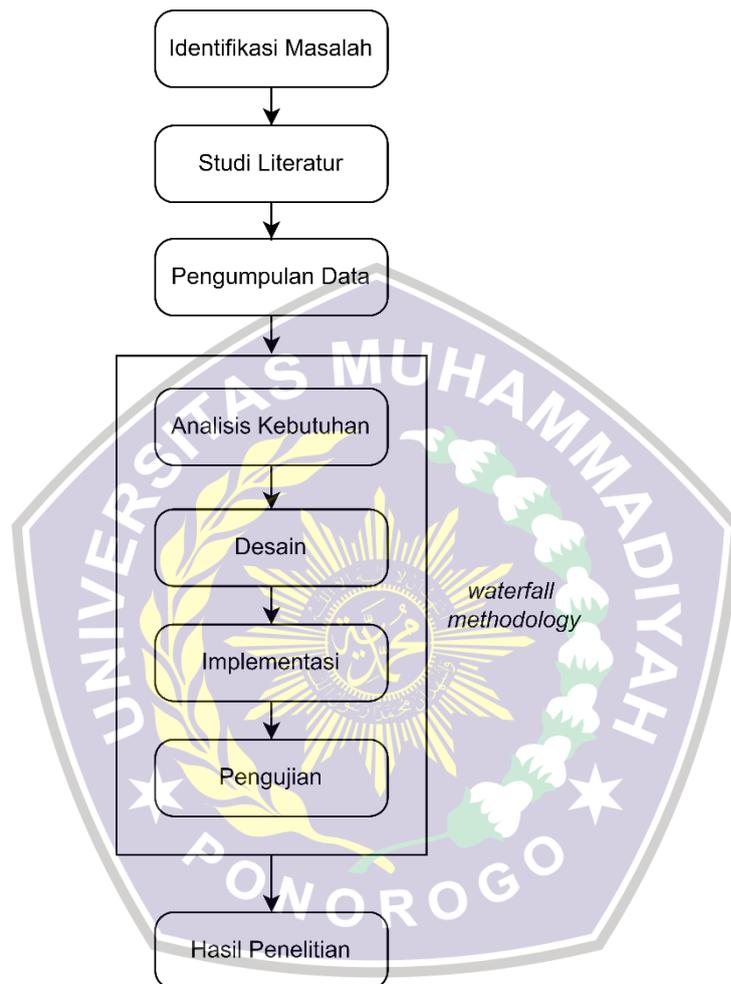
Teknik-teknik yang umum digunakan dalam *white box testing* mencakup *Control Flow Testing*, yang memeriksa jalur-jalur yang diambil oleh program selama eksekusi, termasuk tes untuk semua keputusan logika yang ada dalam kode. *Data Flow Testing*, di sisi lain, memeriksa aliran data dalam program untuk menemukan kebocoran data, penggunaan data yang tidak valid, atau kondisi yang tidak terduga terkait dengan data. *Basic Path* atau *Path Testing* menguji semua jalur yang mungkin dalam program dari awal hingga akhir, melibatkan pengujian setiap jalur minimal satu kali. *Loop Testing*, pada gilirannya, memeriksa perilaku program saat melibatkan pengulangan atau *Looping*, termasuk pengujian untuk memastikan bahwa pengulangan berjalan dengan benar dan mengidentifikasi masalah seperti kondisi penghentian *Loop* yang tidak tepat[17].



## BAB 3

### METODE PENELITIAN

#### 3.1 Tahapan Penelitian



Gambar 3.1 Tahapan Penelitian

##### 3.1.1 Identifikasi Masalah

Pada tahap ini, dilakukan proses identifikasi masalah terhadap pengelolaan surat keluar di Kelurahan Bulukerto. Masalah utama yang teridentifikasi adalah lambatnya pencarian dan pengurutan data surat keluar karena harus mencari berkas-berkas secara manual, yang mengakibatkan efisiensi kerja menjadi terganggu. Selain itu, belum ada sistem pengelolaan berbasis web yang dapat membantu mempercepat proses ini. Ketiadaan

sistem yang terkomputerisasi menyebabkan waktu yang dibutuhkan untuk mengakses dan mengelola data surat menjadi lebih lama, menghambat alur kerja, dan meningkatkan risiko kesalahan dalam administrasi. Untuk mengatasi masalah ini, penerapan sistem pengelolaan surat keluar berbasis web dengan implementasi algoritma *Selection Sort* dapat menjadi solusi yang tepat. Algoritma *Selection Sort* akan membantu mempercepat proses pencarian dan pengurutan surat secara efektif, sehingga diharapkan dapat meningkatkan efisiensi pengelolaan surat keluar dan memperbaiki kinerja administrasi di Kelurahan Bulukerto.

### **3.1.2 Studi Literatur**

Pada tahap studi literatur Pada tahap studi literatur, dilakukan pencarian dan analisis terhadap berbagai sumber informasi yang relevan, seperti jurnal ilmiah, buku teks, serta referensi daring yang membahas pengelolaan surat keluar dan algoritma pengurutan data. Studi ini bertujuan untuk memahami konsep dasar, prinsip kerja, manfaat, serta tantangan dalam penerapan algoritma pengurutan, khususnya *Selection Sort*, dalam sistem pengelolaan surat keluar. Selain itu, studi literatur juga digunakan untuk mengevaluasi praktik terbaik yang telah diterapkan dalam sistem pengelolaan data di berbagai instansi, sehingga dapat menjadi referensi dalam pengembangan sistem berbasis web di Kelurahan Bulukerto. Dengan adanya studi literatur, penelitian ini memiliki dasar teori yang kuat serta dapat mengadopsi solusi yang telah terbukti efektif dalam meningkatkan ketertelusuran dan pencatatan surat keluar secara lebih sistematis.

### **3.1.3 Pengumpulan Data**

Pada tahap pengumpulan data dilakukan dengan dua metode, yaitu observasi dan wawancara, sebagai berikut:

- 1) Observasi dilakukan dengan mengamati secara langsung proses pengelolaan surat keluar di Kelurahan Bulukerto. Dalam tahap ini, peneliti mencermati alur kerja mulai dari pembuatan, pencatatan,

hingga pengarsipan surat keluar. Selain itu, diamati pula hambatan yang sering terjadi, seperti keterlambatan pencatatan atau kesulitan dalam pencarian kembali dokumen. Sistem atau metode yang saat ini digunakan oleh pegawai, baik secara manual maupun digital, turut diperhatikan untuk mengidentifikasi potensi perbaikan. Hasil dari observasi ini menjadi dasar dalam merancang sistem pengelolaan surat keluar berbasis web yang lebih terstruktur dan sesuai dengan kebutuhan pegawai kelurahan.

- 2) Wawancara dilakukan dengan pegawai Kelurahan Bulukerto yang terlibat langsung dalam pengelolaan surat keluar guna mendapatkan pemahaman lebih mendalam mengenai kendala yang dihadapi serta kebutuhan spesifik terhadap sistem yang baru. Dalam wawancara ini, digali informasi mengenai prosedur standar yang diterapkan, tantangan utama dalam pencatatan dan penyimpanan surat, serta harapan pegawai terhadap sistem berbasis web yang akan dikembangkan. Masukan dari pegawai mengenai fitur dan solusi yang dibutuhkan menjadi acuan penting dalam perancangan sistem agar lebih efisien, mudah digunakan, serta mampu mengatasi permasalahan yang ada.

#### **3.1.4 Requirement Analysis**

Pada tahap Requirement Analysis, langkah awal yang dilakukan penelitian adalah melakukan analisis kebutuhan untuk menetapkan prioritas dalam pengelolaan surat keluar di Kelurahan Bulukerto. Mengingat pentingnya efisiensi dalam pengelolaan data surat keluar, peneliti memfokuskan upaya pada implementasi sistem pengelolaan berbasis web yang dapat mempercepat proses pencarian dan pengurutan surat. Selain itu, kebutuhan untuk memastikan keamanan dan integritas data surat juga menjadi prioritas utama.

Berikut beberapa perangkat lunak dan keras yang diperlukan:

- a) Laptop (Lenovo *Ideapad* 320, Processor Intel Core i3 6006u, Nvidia GeForce 920MX, RAM 12GB, 128GB SSD + 1TB Harddisk, Windows 11 Pro)
- b) XAMPP 7.0.0 (untuk *server web* lokal dan *database*)
- c) Bahasa pemrograman PHP (untuk pengembangan aplikasi web)
- d) Algoritma *Selection Sort* untuk pengurutan data surat

Analisis kebutuhan berikutnya melibatkan evaluasi terhadap jenis data yang akan dikelola dalam sistem, termasuk informasi penting seperti tanggal surat, pengirim, dan tujuan. Potensi masalah seperti lambatnya akses data dan risiko kehilangan data akibat pengelolaan manual juga dipertimbangkan dengan seksama.

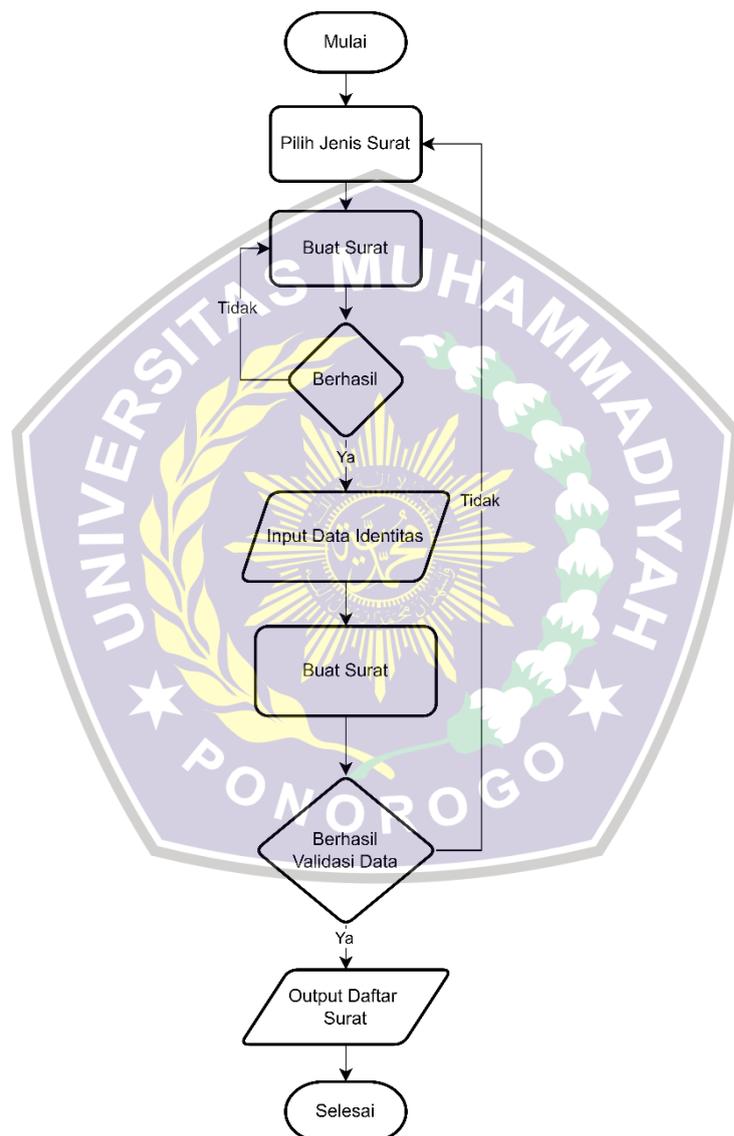
Berdasarkan pemahaman yang mendalam tentang kebutuhan dan potensi masalah, langkah-langkah desain sistem yang efektif dirancang dan diimplementasikan. Ini mencakup pengembangan fitur-fitur utama seperti pencarian cepat, pengurutan otomatis menggunakan algoritma *Selection Sort*, dan backup data secara teratur. Selain itu, sistem ini dirancang dengan antarmuka yang *user-friendly* untuk memastikan pegawai kelurahan dapat menggunakannya dengan mudah dan efisien, sehingga diharapkan dapat meningkatkan efisiensi dan akurasi dalam pengelolaan surat keluar.

### 3.1.5 Design

Tahap desain dalam metode *Waterfall* melibatkan perancangan solusi yang akan diimplementasikan dalam proyek. Dalam konteks sistem surat keluar, desain ini mencakup elemen-elemen seperti *Flowchart* sistem, *Flowchart Selection Sort*, diagram konteks, DFD *level 1*, serta desain *database* dengan ERD dan relasi tabel. *Flowchart* sistem menggambarkan alur keseluruhan proses pembuatan surat keluar dari awal hingga akhir, sementara *Flowchart Selection Sort* menunjukkan metode pengurutan surat berdasarkan kriteria tertentu, seperti tanggal keluar surat. Diagram konteks mengilustrasikan hubungan antara sistem surat keluar dengan entitas

eksternal seperti pengguna, admin, dan *database*. DFD level 1 memperinci alur data utama dalam sistem, termasuk proses login, pembuatan, validasi, dan pengeluaran surat. Desain *database* dengan ERD menggambarkan entitas utama seperti pengguna, surat, dan jenis surat, serta hubungan antar entitas tersebut.

a) *Flowchart* Sistem

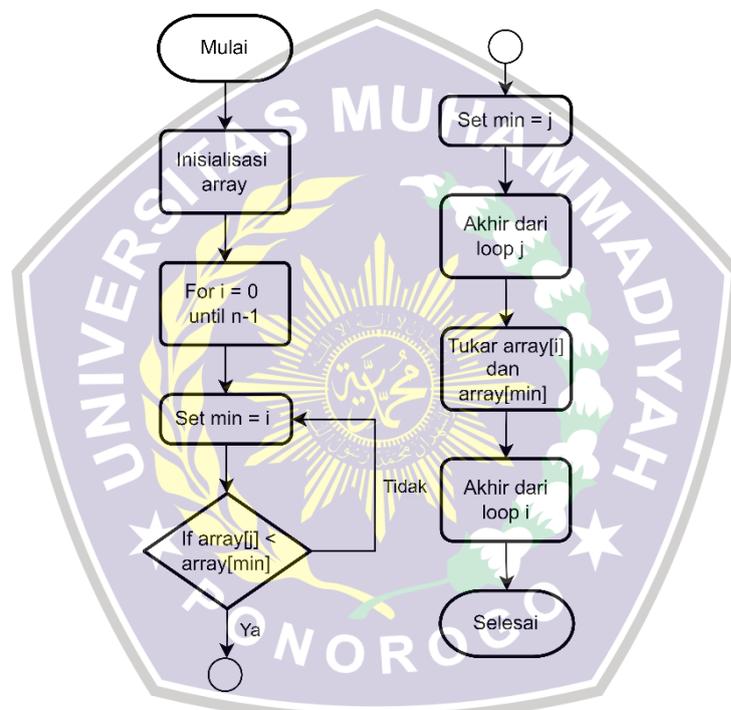


Gambar 3.2 *Flowchart* Sistem Surat Keluar

Flowchart pada gambar 3.2 menggambarkan alur proses pembuatan surat dalam sistem yang dimulai dengan memilih jenis surat, dilanjutkan dengan langkah pembuatan surat. Setelah itu, sistem

memverifikasi apakah pembuatan surat berhasil; jika gagal, proses dihentikan, namun jika berhasil, pengguna memasukkan data berdasarkan KTP. Selanjutnya, data yang diinputkan divalidasi untuk memastikan kebenarannya. Jika validasi gagal, proses kembali ke tahap input data, sedangkan jika validasi berhasil, sistem menghasilkan output data surat. Proses ini ditutup dengan langkah selesai, menunjukkan alur yang terstruktur untuk memastikan akurasi dan keberhasilan pembuatan surat.

b) *Flowchart Selection Sort*



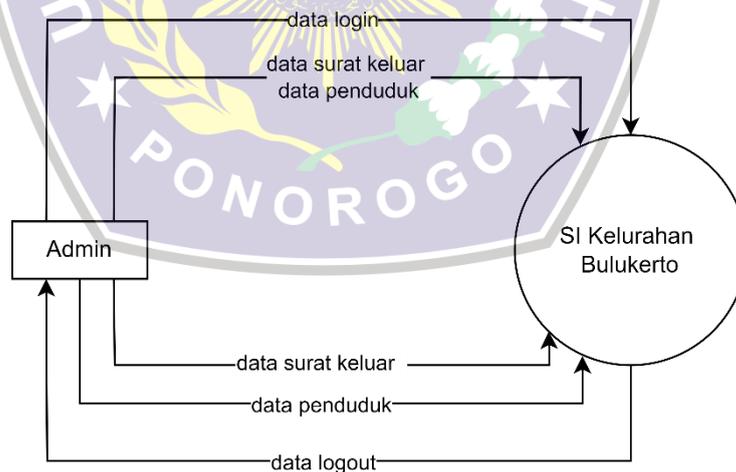
Gambar 3.3 *Flowchart Selection Sort*

Berdasarkan gambar 3.3 di atas, *Flowchart* tersebut menggambarkan algoritma *Selection Sort* untuk mengurutkan elemen-elemen dalam sebuah *Array*. Algoritma dimulai dengan inisialisasi *Array*. *Loop* utama berjalan dari indeks  $i = 0$  hingga  $n-1$ , di mana  $n$  adalah panjang *Array*. Dalam setiap iterasi, *min* diatur ke  $i$ , dan *Loop* kedua berjalan dari indeks  $j = i+1$  hingga  $n$  untuk mencari elemen terkecil di bagian yang belum diurutkan. Jika  $Array[j]$  lebih kecil dari  $Array[min]$ , *min* diatur ke  $j$ . Setelah *Loop* kedua selesai, elemen  $Array[i]$

ditukar dengan elemen  $Array[\min]$ . Proses ini diulangi sampai seluruh  $Array$  terurutkan, dan algoritma berakhir. *Selection Sort* bekerja dengan mencari elemen terkecil dari bagian yang belum diurutkan dan menukarnya dengan elemen pertama dari bagian tersebut, hingga seluruh  $Array$  terurut dalam urutan menaik.

Misalnya, penerapan algoritma *Selection Sort* pada  $Array$  {64, 25, 12, 22, 11} adalah sebagai berikut: pada iterasi pertama, elemen terkecil di bagian yang belum diurutkan (11) ditemukan dan ditukar dengan elemen pertama (64), menghasilkan  $Array$  {11, 25, 12, 22, 64}; iterasi kedua menemukan elemen terkecil berikutnya (12) dan menukarnya dengan elemen di posisi kedua (25), menghasilkan  $Array$  {11, 12, 25, 22, 64}; iterasi ketiga menukar elemen terkecil yang tersisa (22) dengan elemen di posisi ketiga (25), menghasilkan  $Array$  {11, 12, 22, 25, 64}; iterasi keempat dan kelima memastikan bahwa elemen yang tersisa sudah pada posisi yang benar, sehingga  $Array$  akhir terurutkan menjadi {11, 12, 22, 25, 64}.

c) Diagram Konteks

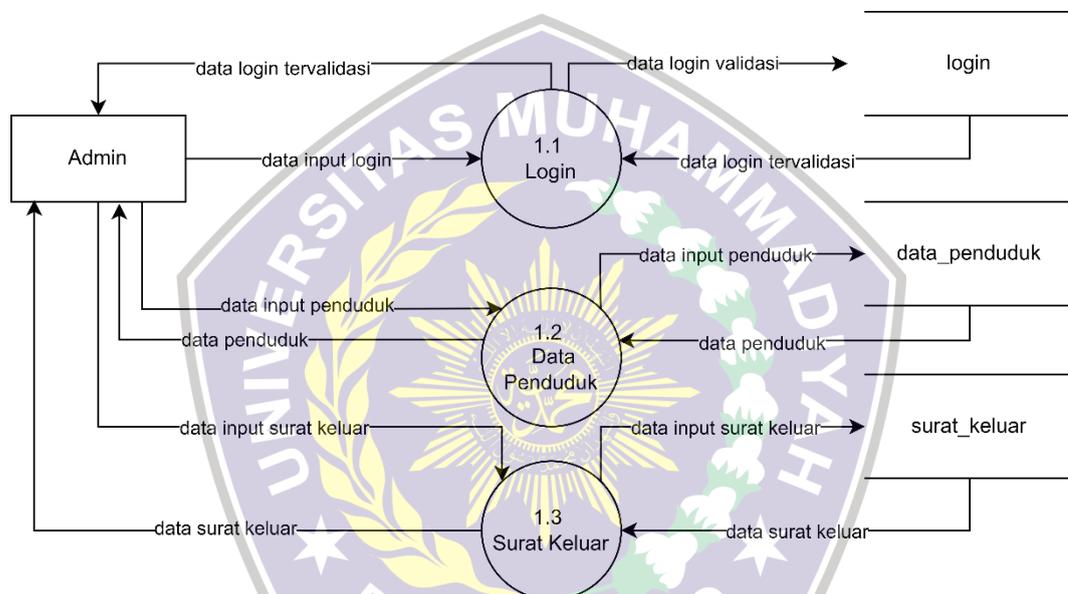


Gambar 3.4 Diagram Konteks

Gambar 3.4 menggambarkan proses interaksi yang dilakukan oleh admin dalam sistem informasi surat keluar di Kelurahan Bulukerto. Pada awalnya, admin memasuki sistem, yang menandai langkah awal dari proses. Kemudian, admin melakukan tindakan untuk menentukan

template surat yang akan digunakan, sekaligus memilih format dan struktur yang sesuai dengan kebutuhan untuk surat yang akan dibuat. Selain itu, admin juga memiliki kemampuan untuk mengedit data penduduk yang terintegrasi dalam sistem. Hal ini memungkinkan admin untuk melakukan perubahan atau pembaruan data penduduk yang relevan, yang dapat memengaruhi konten atau informasi yang akan disertakan dalam surat keluar.

d) DFD level 1



Gambar 3.5 DFD level 1

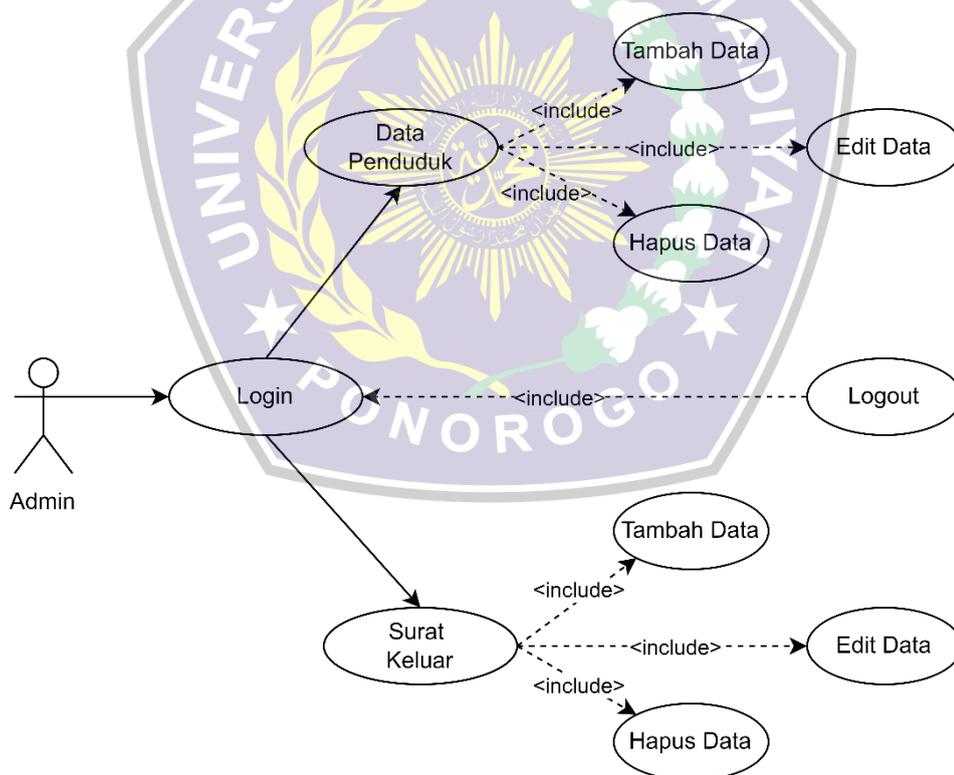
DFD level 1 memberikan penjelasan terperinci tentang diagram konteks, yang menggambarkan arus data dari sistem penyimpanan ke database. Gambar 3.5 menampilkan proses-proses dengan urutan berikut :

- Admin : Admin bertanggung jawab atas manajemen sistem informasi surat keluar di Kelurahan Bulukerto, dengan hak akses penuh untuk tugas administratif termasuk pengelolaan data penduduk dan surat keluar.
- Login : Langkah awal untuk admin masuk ke dalam sistem, memerlukan informasi login valid seperti nama

pengguna dan kata sandi untuk autentikasi sebelum mendapatkan akses penuh ke sistem.

- Data Penduduk : Mencakup informasi tentang penduduk di Kelurahan Bulukerto seperti nama, alamat, nomor identitas, dan lainnya, penting untuk pembuatan surat keluar dan pemutakhiran data.
- Surat Keluar : Proses surat keluar di Kelurahan Bulukerto meliputi pembuatan dokumen resmi oleh pihak berwenang, penyimpanan dalam arsip, dan pengiriman kepada penerima di luar kelurahan. Langkah-langkah ini penting untuk memastikan kelancaran komunikasi dan administrasi antara kelurahan dan penerima surat keluar.

e) *Usecase Diagram*

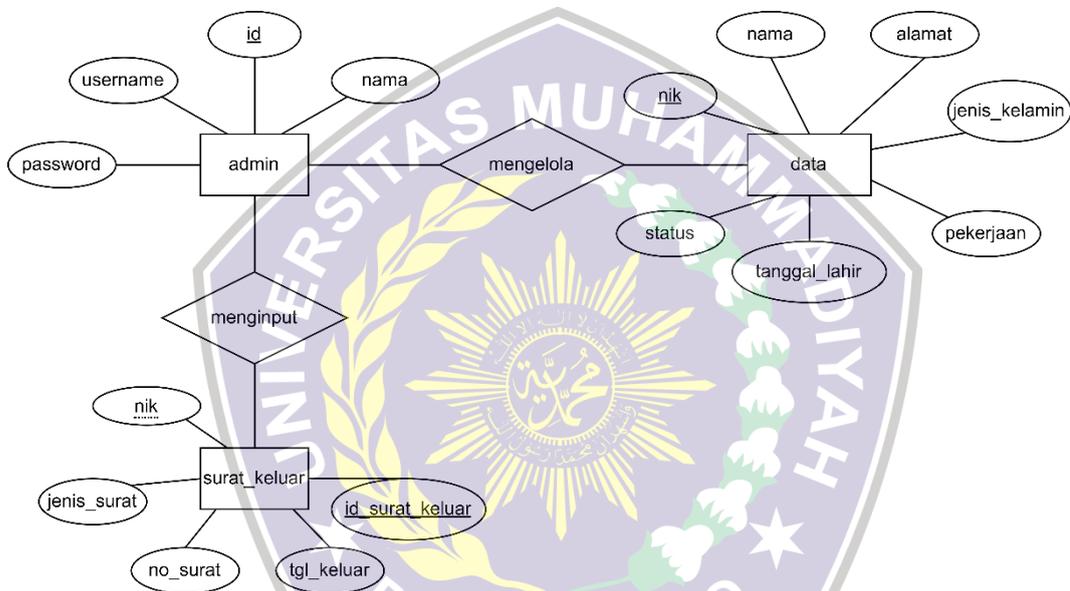


Gambar 3.6 *Usecase Diagram*

Gambar 3.6 menggambarkan peran admin dalam sistem informasi surat keluar di Kelurahan Bulukerto, di mana admin memiliki

kewenangan mulai dari login dengan memasukkan nama pengguna dan kata sandi untuk mengakses sistem, mengelola data penduduk dengan menambah, memperbarui, dan mencari informasi seperti nama, alamat, dan NIK, hingga menangani surat keluar dengan pembuatan, penyimpanan, dan pengiriman dokumen resmi kepada penerima di luar kelurahan, serta *logout* dari sistem setelah menyelesaikan tugas untuk menjaga keamanan data.

f) ERD



Gambar 3.7 Entity Relationship Diagram

Pada gambar 3.7, menjelaskan entitas, atribut dan proses pada sistem pendukung keputusan pemilihan mahasiswa lulusan terbaik. Yang mana terdapat admin, data penduduk dan surat keluar.

g) Perancangan Tabel

Perancangan tabel untuk Sistem Informasi Surat Keluar di Kelurahan Bulukerto dengan menggunakan Algoritma *Selection Sort*, sebagai berikut :

(1) Tabel Admin

Fungsi : menyimpan data admin

Primary key : id\_admin

Total : 70 byte

Tabel 3.1 tabel Admin

<i>Field</i>	<i>Type</i>	<i>Length</i>	<i>Keterangan</i>
id_admin	integer	5	Primary key
username	varchar	20	
nama	varchar	30	
password	varchar	15	

(2) Tabel Data Penduduk

Fungsi : menyimpan data penduduk

Primary key : NIK

Total : 61 byte

Tabel 3.2 tabel data penduduk

<i>Field</i>	<i>Type</i>	<i>Length</i>	<i>Keterangan</i>
NIK	integer	16	Primary key
nama	varchar	30	
alamat	text		
jenis_kelamin	enum		
pekerjaan	varchar	15	
tgl_lahir	date		
status	enum		

(3) Tabel Surat Keluar

Fungsi : menyimpan data surat keluar

Primary key : id\_surat\_keluar

Foregin key : NIK

Total : 62 byte

Tabel 3.3 tabel surat keluar

<i>Field</i>	<i>Type</i>	<i>Length</i>	<i>Keterangan</i>
id_surat_keluar	<i>integer</i>	6	<i>Primary key</i>
NIK	<i>varchar</i>	16	<i>Foregin key</i>
Jenis_surat	<i>varchar</i>	20	
no_surat	<i>integer</i>	20	
tgl_keluar	<i>date</i>		

h) *User Interface*

(1) Login

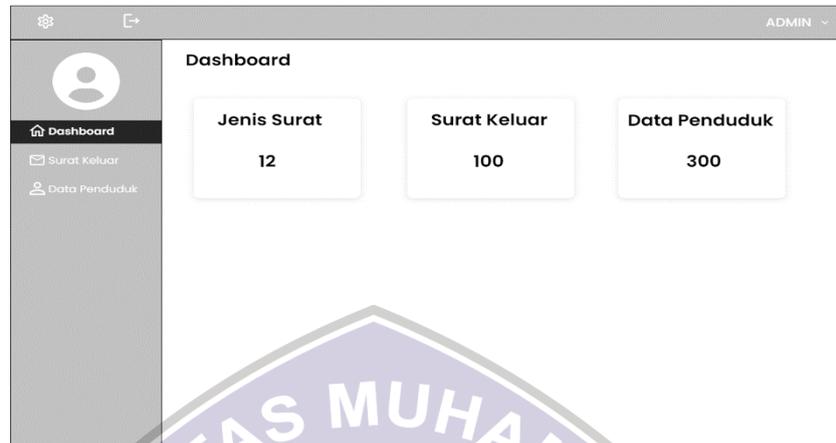


Gambar 3.8 *Login*

Pada gambar 3.8, merupakan desain antarmuka login, yang mana *user* dan admin diminta untuk memasukkan informasi identifikasi yaitu *username* dan *password*. Terdapat juga sebuah

tombol "Masuk" yang bertujuan untuk mengizinkan akses ke menu *dashboard* setelah *user* atau admin login ke dalam sistem.

## (2) Dashboard Admin



Gambar 3.9 Dashboard Admin

Pada gambar 3.9, merupakan desain antarmuka *Dashboard* Admin setelah melakukan *login*. Halaman ini dirancang untuk memberikan tampilan awal admin ketika berhasil *login* ke dalam sistem yang mana pada bagian *sidebar* terdapat data penduduk, dan surat keluar.

## (3) Surat Keluar

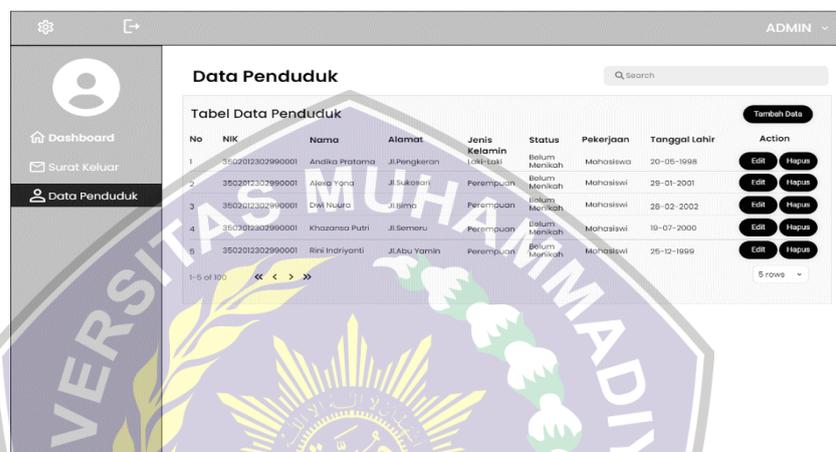


Gambar 3.10 Surat Keluar

Pada gambar 3.10, tampilan desain antarmuka surat keluar dalam sistem informasi surat keluar di Kelurahan Bulukerto.

Halaman ini dirancang khusus untuk admin dan berfungsi sebagai tempat admin memilih template surat yang akan dibuat. Antarmuka ini memungkinkan admin untuk dengan mudah mengakses berbagai jenis template surat resmi yang diperlukan, memudahkan proses pembuatan surat dengan format yang konsisten dan sesuai standar.

#### (4) Data Penduduk



No	NIK	Nama	Alamat	Jenis Kelamin	Status	Pekerjaan	Tanggal Lahir	Action
1	3502012302990001	Andika Pratama	Jl. Pengkeran	Laki-laki	Belum Menikah	Mahasiswa	20-05-1998	Tambah Data Edit Hapus
2	3502012302990001	Alexa Yana	Jl. Sukarini	Perempuan	Belum Menikah	Mahasiswa	29-01-2001	Edit Hapus
3	3502012302990001	Dwi Nuara	Jl. Bima	Perempuan	Belum Menikah	Mahasiswa	28-02-2002	Edit Hapus
4	3502012302990001	Khazama Putri	Jl. Semeru	Perempuan	Belum Menikah	Mahasiswa	10-07-2000	Edit Hapus
5	3502012302990001	Rini Indriyanti	Jl. Abu Yamin	Perempuan	Belum Menikah	Mahasiswa	25-12-1999	Edit Hapus

Gambar 3.11 Data Penduduk

Pada gambar 3.11, tampilan desain antarmuka data penduduk dalam sistem informasi surat keluar di Kelurahan Bulukerto. Halaman ini dirancang untuk memungkinkan admin mengelola informasi penduduk dengan mudah dan efisien. Antarmuka ini mencakup fitur-fitur seperti penambahan, pengeditan, dan penghapusan data penduduk. Admin dapat melihat daftar penduduk yang berisi informasi penting seperti nama, alamat, nomor identitas, dan data relevan lainnya. Desain yang user-friendly mempermudah admin dalam mencari dan memperbarui informasi penduduk, memastikan data selalu akurat dan up-to-date untuk keperluan administrasi kelurahan.

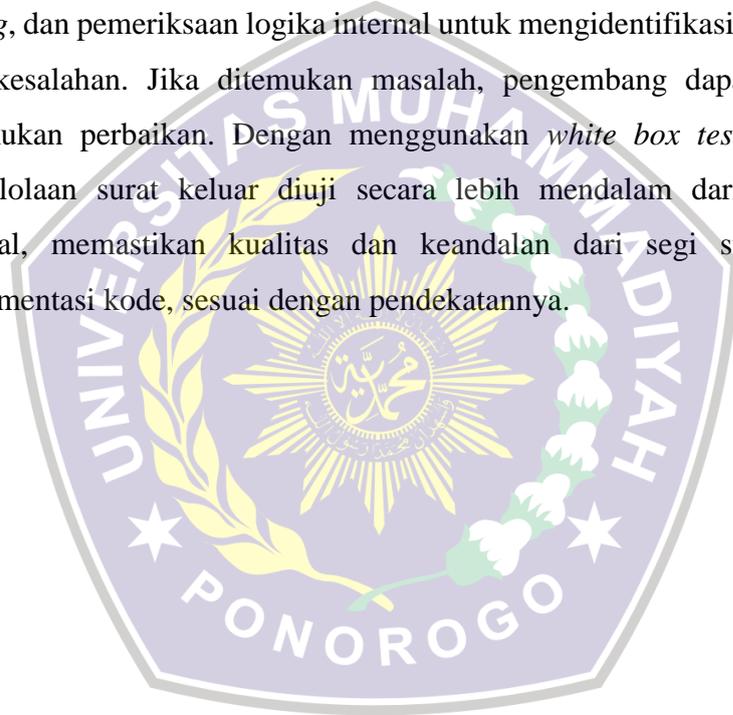
### 3.1.6 Implementation

Tahap implementasi melibatkan pengembangan solusi perangkat lunak berdasarkan rancangan yang telah disepakati sebelumnya. Pada tahap ini, pengembangan sistem pengelolaan surat keluar di Kelurahan Bulukerto dilakukan dengan memanfaatkan teknologi yang sesuai, seperti PHP dan MySQL. Solusi perangkat lunak yang dibangun dirancang untuk memenuhi persyaratan yang telah ditetapkan dalam tahap analisis dan desain sebelumnya, termasuk pengembangan antarmuka pengguna, fungsi pencarian dan pengurutan surat, serta fitur keamanan yang sesuai.

Proses implementasi ini mengikuti langkah-langkah yang terstruktur sesuai dengan metodologi *Waterfall*, dengan fokus pada pemenuhan deliverables yang telah ditentukan sebelumnya seperti *Flowchart* proses pengelolaan surat keluar, diagram konteks, dan desain database dengan *Entity Relationship Diagram* (ERD) serta relasi tabel yang diperlukan. Salah satu fungsi penting yang dikembangkan adalah pengurutan surat keluar, yang menggunakan algoritma *Selection Sort*. Sistem memulai dengan daftar surat keluar yang belum diurutkan, kemudian untuk setiap posisi dalam daftar, algoritma mencari elemen terkecil dari posisi tersebut hingga akhir daftar, dan elemen tersebut ditukar dengan elemen di posisi awal yang sedang diproses. Langkah-langkah ini diulang untuk setiap posisi dalam daftar hingga seluruh daftar terurut. Implementasi *Selection Sort* menunjukkan bagaimana algoritma ini bekerja dalam konteks data surat keluar. Selama tahap implementasi, kolaborasi terus berlangsung antara tim pengembang dan pihak terkait di Kelurahan Bulukerto untuk memastikan bahwa solusi yang dihasilkan sesuai dengan kebutuhan dan harapan. Hasil akhir dari tahap implementasi ini adalah sistem pengelolaan surat keluar yang terstruktur dan efisien, termasuk fitur pengurutan yang memanfaatkan algoritma *Selection Sort* untuk membantu pengelolaan surat dengan lebih baik.

### 3.1.7 *Testing*

Setelah tahap implementasi, pengujian perangkat lunak untuk sistem pengelolaan surat keluar dilakukan melalui metode *white box testing*. Metode ini melibatkan pemeriksaan internal dan detail implementasi kode oleh pengembang atau pihak yang memiliki akses ke struktur internal perangkat lunak. Fokus utama dalam metode ini adalah memastikan bahwa setiap komponen dan alur logika dalam perangkat lunak berfungsi dengan benar sesuai spesifikasi yang telah ditetapkan dalam tahap analisis dan desain sebelumnya. Pengujian mencakup analisis struktur kode, *coverage testing*, dan pemeriksaan logika internal untuk mengidentifikasi potensi bug atau kesalahan. Jika ditemukan masalah, pengembang dapat langsung melakukan perbaikan. Dengan menggunakan *white box testing*, sistem pengelolaan surat keluar diuji secara lebih mendalam dari perspektif internal, memastikan kualitas dan keandalan dari segi struktur dan implementasi kode, sesuai dengan pendekatannya.



## BAB 4

### HASIL DAN PEMBAHASAN

#### 4.1 Implementasi Sistem

Pada tahap implementasi sistem terdapat dua bagian utama, yaitu implementasi algoritma dan implementasi antarmuka pengguna (*interface*). Implementasi algoritma *Selection Sort* digunakan untuk mengurutkan data surat keluar berdasarkan kriteria tertentu seperti nomor surat atau tanggal, sehingga memudahkan pencarian dan pengelolaan data. Di sisi lain, implementasi antarmuka pengguna bertujuan untuk merancang tampilan web yang mudah digunakan oleh petugas kelurahan, memungkinkan mereka untuk memasukkan, mengelola, dan mengakses data surat keluar dengan lebih efisien melalui desain yang sederhana dan responsif. Kedua bagian ini bekerja bersama untuk memastikan sistem dapat berfungsi dengan baik dalam meningkatkan pengelolaan data surat keluar.

##### 4.1.1 Implementasi Algoritma *Selection Sort*

```
$domisilis = Domisili::all();
$usahas = Usaha::all();
$bedanamas = Bedanama::all();
$hakpakais = HakPakai::all();
$ketpengantars = KetPengantar::all();

$allSurats = collect()
    ->merge($domisilis)
    ->merge($usahas)
    ->merge($bedanamas)
    ->merge($hakpakais)
    ->merge($ketpengantars)
    ->all();

$sortedSurats = $allSurats;
$n = count($sortedSurats);
```

```

for ($i = 0; $i < $n - 1; $i++) {
    $minIndex = $i;

    for ($j = $i + 1; $j < $n; $j++) {
        if($sortedSurats[$j]->noSurat< $sortedSurats
[$minIndex] ->noSurat) {
            $minIndex = $j;
        }
    }

    if ($minIndex != $i) {
        $temp = $sortedSurats[$i];
        $sortedSurats[$i] =
$sortedSurats[$minIndex];
        $sortedSurats[$minIndex] =
$temp;
    }
}

```

Gambar 4.1 Implementasi Algoritma *Selection Sort*

Gambar 4.1 menunjukkan implementasi algoritma *Selection Sort* untuk mengurutkan data surat keluar berdasarkan atribut `noSurat`. Data surat diambil dari lima tabel berbeda, yaitu `Domisili`, `Usaha`, `Bedanama`, `HakPakai`, dan `KetPengantar`, kemudian digabungkan menjadi satu koleksi menggunakan fungsi `collect()` dan `merge()`, lalu disimpan dalam variabel `\$allSurats`. Proses pengurutan dilakukan dengan mencari elemen terkecil pada setiap iterasi dan menukarnya dengan elemen di posisi awal yang sesuai. Panjang data dihitung menggunakan `count()` untuk menentukan jumlah iterasi, lalu dilakukan pencarian elemen terkecil dari indeks tertentu hingga akhir data, dan jika ditemukan elemen yang lebih kecil, dilakukan penukaran elemen tersebut. Setelah semua iterasi selesai, data dalam `\$sortedSurats` akan tersusun berdasarkan nilai `noSurat` dari kecil ke besar, sehingga mempermudah pengelolaan dan pencarian data dalam sistem.

#### 4.1.2 Implementasi Koneksi Database

```
APP_NAME=Surat Keluar
APP_ENV=local
APP_KEY=base64:Xt3MqZU9TAqDyUnNCHdoZHTQsYldHHzcxl4KJ1y9
iUY=
APP_DEBUG=true
APP_URL=http://localhost

LOG_CHANNEL=stack
LOG_DEPRECATED_CHANNEL=null
LOG_LEVEL=debug

DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=db_surat
DB_USERNAME=root
DB_PASSWORD=
BROADCAST_DRIVER=log
CACHE_DRIVER=file
FILESYSTEM_DISK=local
QUEUE_CONNECTION=sync
SESSION_DRIVER=file
SESSION_LIFETIME=120

MEMCACHED_HOST=127.0.0.1

REDIS_HOST=127.0.0.1
REDIS_PASSWORD=null
REDIS_PORT=6379

MAIL_MAILER=smt
MAIL_HOST=mailhog
MAIL_PORT=1025
MAIL_USERNAME=null
MAIL_PASSWORD=null
MAIL_ENCRYPTION=null

AWS_ACCESS_KEY_ID=
AWS_SECRET_ACCESS_KEY=
AWS_DEFAULT_REGION=us-east-1
AWS_BUCKET=
AWS_USE_PATH_STYLE_ENDPOINT=false

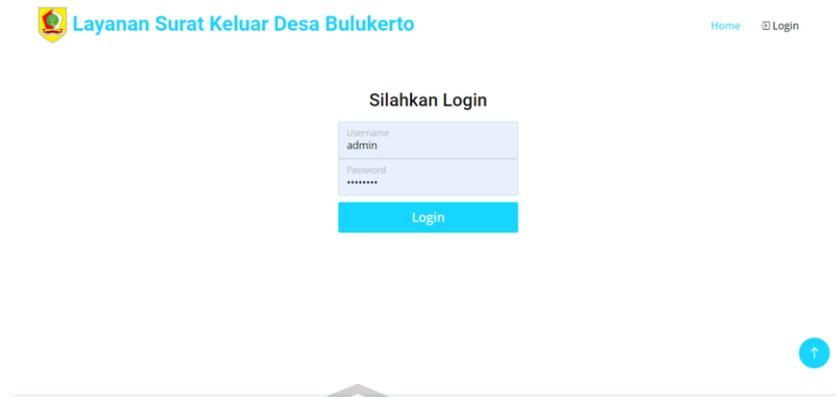
PUSHER_APP_ID=
PUSHER_APP_KEY=
PUSHER_APP_SECRET=
PUSHER_HOST=
PUSHER_PORT=443
PUSHER_SCHEME=https
PUSHER_APP_CLUSTER=mt1
```

Gambar 4.2 Koneksi Database

Pada gambar 4.2 menampilkan konfigurasi file `.env`, yaitu file konfigurasi utama dalam pengembangan aplikasi berbasis Laravel, yang berisi berbagai pengaturan lingkungan sistem untuk memastikan aplikasi berjalan dengan optimal. File ini mencakup pengaturan variabel `APP_NAME`, yang menentukan nama aplikasi, yaitu *Surat Keluar*, serta `APP_ENV`, yang menunjukkan bahwa aplikasi berjalan dalam mode *local* untuk keperluan pengembangan. Selain itu, file `.env` juga mengatur koneksi database dengan variabel `DB_CONNECTION=mysql` sebagai jenis koneksi yang digunakan, `DB_HOST=127.0.0.1` sebagai alamat host, `DB_DATABASE=db_surat` sebagai nama database, `DB_USERNAME=root` sebagai nama pengguna, dan password yang sesuai untuk menghubungkan aplikasi dengan database MySQL. Variabel tambahan, seperti `MAIL_MAILER`, digunakan untuk mengatur layanan pengiriman email, `CACHE_DRIVER` untuk menentukan metode caching yang digunakan, dan `SESSION_DRIVER` untuk mengelola penyimpanan sesi pengguna. Dengan konfigurasi ini, aplikasi dapat diatur sesuai kebutuhan, baik saat pengembangan lokal maupun untuk penerapan di lingkungan produksi, sehingga memberikan fleksibilitas dan keandalan dalam pengelolaan sistem.

### 4.1.3 Implementasi *Interface*

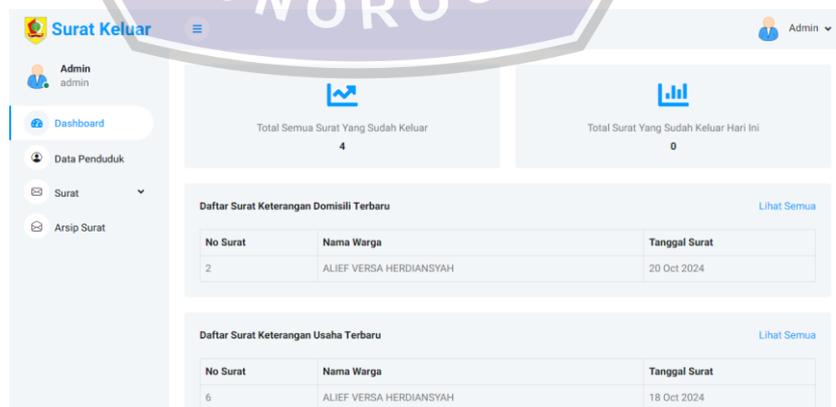
#### 1) *Login*



Gambar 4.3 *Login*

Pada gambar 4.3 ditampilkan halaman login dari aplikasi "Layanan Surat Keluar Desa Bulukerto". Halaman ini digunakan untuk mengautentikasi pengguna sebelum mengakses sistem. Tampilan halaman login terdiri dari logo aplikasi di bagian kiri atas, judul aplikasi "Layanan Surat Keluar Desa Bulukerto" dengan warna biru, dan form login yang terletak di tengah halaman. Form ini memiliki dua input utama, yaitu *username* dan *password*, serta tombol login berwarna biru di bawahnya.

#### 2) *Dashboard Admin*

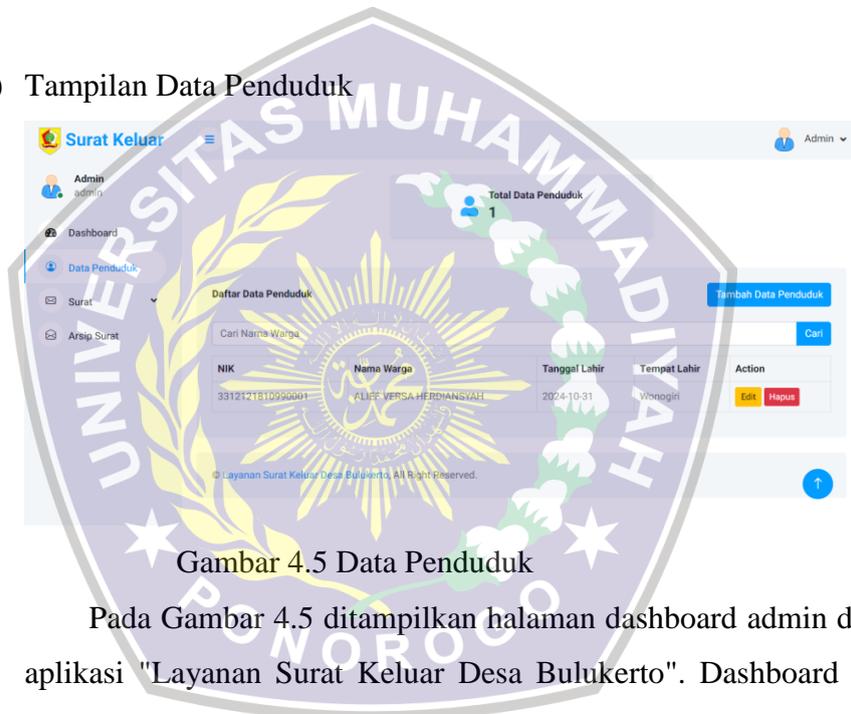


Gambar 4.4 *Dashboard Admin*

Pada gambar 4.4 ditampilkan halaman *dashboard* admin dari aplikasi "Layanan Surat Keluar Desa Bulukerto". *Dashboard* ini

berfungsi sebagai pusat informasi bagi admin untuk memantau data surat keluar. Pada bagian atas halaman terdapat dua kotak informasi utama yang menampilkan total semua surat yang sudah keluar dan total surat yang keluar pada hari itu. Di bawahnya terdapat dua tabel yang masing-masing menampilkan daftar surat keterangan domisili terbaru dan daftar surat keterangan usaha terbaru, dengan informasi berupa nomor surat, nama warga, dan tanggal surat. Di sebelah kiri, terdapat menu navigasi berupa *dashboard*, data penduduk, surat, dan arsip surat untuk memudahkan akses ke fitur-fitur aplikasi.

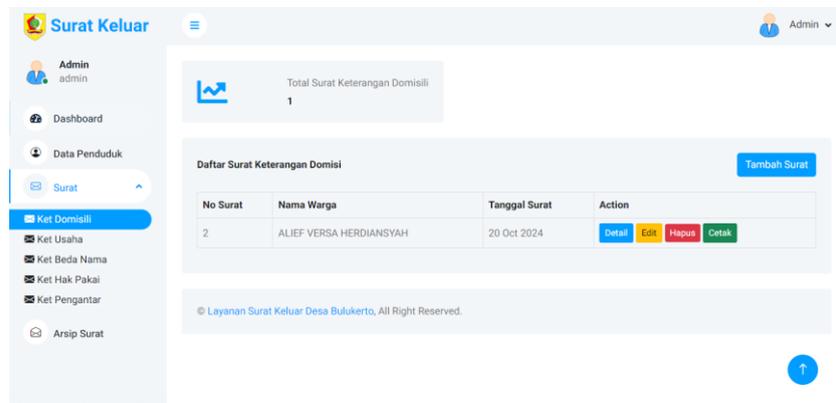
### 3) Tampilan Data Penduduk



Gambar 4.5 Data Penduduk

Pada Gambar 4.5 ditampilkan halaman dashboard admin dari aplikasi "Layanan Surat Keluar Desa Bulukerto". Dashboard ini berfungsi sebagai pusat informasi bagi admin untuk memantau data surat keluar. Pada bagian atas halaman terdapat dua kotak informasi utama yang menampilkan total semua surat yang sudah keluar dan total surat yang keluar pada hari itu. Di bawahnya terdapat dua tabel yang masing-masing menampilkan daftar surat keterangan domisili terbaru dan daftar surat keterangan usaha terbaru, dengan informasi berupa nomor surat, nama warga, dan tanggal surat. Di sebelah kiri, terdapat menu navigasi berupa dashboard, data penduduk, surat, dan arsip surat untuk memudahkan akses ke fitur-fitur aplikasi.

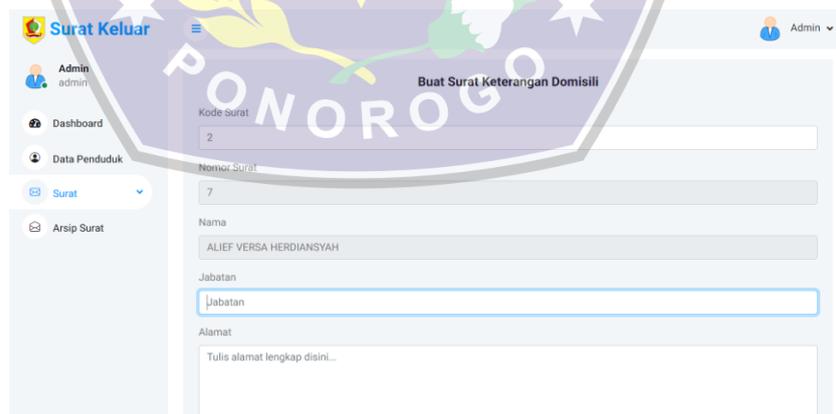
#### 4) Tampilan Surat Keterangan Domisili



Gambar 4.6 Tampilan Surat Keterangan Domisili

Pada gambar 4.6, tampilan menunjukkan halaman Surat Keterangan Domisili dalam aplikasi layanan surat keluar desa, dengan informasi total surat yang telah dibuat sebanyak 1. Di bawahnya terdapat tabel daftar surat yang mencakup kolom No Surat, Nama Warga, Tanggal Surat, dan Action. Sebagai contoh, terdapat data surat dengan No Surat 2, Nama Warga ALIEF VERSA HERDIANSYAH, dan Tanggal Surat 20 Oktober 2024.

#### 5) Tampilan Buat Surat



Gambar 4.7 Tampilan Buat Surat

Pada gambar 4.7, tampilan menunjukkan halaman Buat Surat dalam aplikasi layanan surat keluar desa. Halaman ini menyediakan formulir untuk mengisi data yang diperlukan dalam pembuatan

surat. Beberapa kolom yang tersedia meliputi kode surat, nomor surat, nama, jabatan, dan alamat.

#### 6) Tampilan Arsip Surat Keluar



Gambar 4.8 Tampilan Arsip Surat Keluar

Pada gambar 4.8, tampilan menunjukkan halaman Arsip Surat Keluar dalam aplikasi layanan surat keluar desa. Halaman ini menampilkan informasi mengenai surat-surat yang telah dibuat dan diarsipkan. Di bagian atas terdapat dua panel yang menunjukkan total semua surat yang sudah keluar, yaitu 4, dan total surat yang sudah keluar hari ini, yaitu 0.

## 4.2 Pengujian Sistem

Pengujian sistem adalah tahap pengujian perangkat lunak di mana seluruh sistem atau aplikasi diuji sebagai satu kesatuan yang terintegrasi. Tujuan dari pengujian sistem adalah untuk memastikan bahwa semua komponen dan modul dari perangkat lunak berfungsi dengan baik bersama-sama dan memenuhi spesifikasi serta kebutuhan yang telah ditentukan. Berikut adalah penjelasan lebih rinci mengenai pengujian sistem :

### 1) Pengujian *Whitebox*

Pengujian unit adalah proses pengujian perangkat lunak yang berfokus pada pengujian masing-masing unit atau komponen terkecil dari suatu aplikasi secara terpisah untuk memastikan bahwa setiap unit berfungsi

sesuai dengan logika yang diharapkan. Dalam pengujian ini, *flow graph* digunakan untuk memvisualisasikan alur logika program, mempermudah identifikasi jalur eksekusi, dan memastikan bahwa setiap cabang kode telah diuji secara menyeluruh. Pengujian ini bertujuan untuk memverifikasi bahwa setiap unit perangkat lunak memenuhi kebutuhan spesifiknya sebelum diintegrasikan dengan unit lain, sehingga kesalahan dapat dideteksi lebih awal dalam proses pengembangan.

a) Pengujian Unit

Pengujian unit pada proses pengurutan data surat keluar menggunakan algoritma *Selection Sort* melibatkan beberapa langkah pengujian. Pertama, data surat dari tabel `Domisili`, `Usaha`, `Bedanama`, `HakPakai`, dan `KetPengantar` digabungkan menjadi satu koleksi menggunakan metode `collect()` dan `merge()`. Selanjutnya, pengujian memastikan bahwa proses pengurutan berjalan sesuai logika *Selection Sort*, dengan memeriksa setiap iterasi. Pengujian ini meliputi langkah-langkah seperti menetapkan variabel `\$minIndex`, membandingkan elemen berdasarkan atribut `noSurat`, dan memastikan bahwa elemen dengan nilai `noSurat` terkecil dipindahkan ke posisi yang sesuai. Setiap jalur logika dalam proses ini, termasuk kondisi saat tidak ada pertukaran yang diperlukan atau saat elemen ditukar, diuji secara menyeluruh. Melalui pengujian ini, diharapkan algoritma dapat mengurutkan data dengan benar, menghasilkan daftar surat yang terurut sesuai dengan nilai `noSurat`.

Tabel 4.1 *Pseudocode Method* “\$allSurats = collect()” dari *Class* “Surat”

Pseudocode	Node
\$allSurats = collect()	1
->merge(\$domisilis)	1
->merge(\$usahas)	1
->merge(\$bedanamas)	1
->merge(\$hakpakais)	1
->merge(\$ketpengantars)	1
->all();	1
\$sortedSurats = \$allSurats;	1

---

```

$N = count($sortedSurats);           1
for ($i = 0; $i < $N - 1; $i++)      2
{                                       2
    $minIndex = $i;                   3
for ($j = $i + 1; $j < $N; $j++)     4
{                                       4
    if ($sortedSurats[$j]->noSurat <  5
        $sortedSurats[$minIndex]->noSurat) 5
    {                                       5
        $minIndex = $j;                 5
        if ($minIndex != $i)           6
        {                                       7
            $temp = $sortedSurats[$i];  7
            $sortedSurats[$i] = $sortedSurats[$minIndex]; 7
            $sortedSurats[$minIndex] = $temp; 7
        }                                       7
    }                                       7
}                                       7
}

```

---

b) *Flow graph*

Flow graph untuk pengujian unit dengan *white box testing*, seperti yang ditunjukkan pada Gambar 4.9, adalah representasi visual dari semua jalur logika dalam algoritma *Selection Sort* untuk proses pengurutan data surat keluar. *Flow graph* ini terdiri dari *node* dan *edges*, di mana setiap node mewakili satu atau lebih pernyataan dalam kode, seperti inisialisasi variabel, perbandingan elemen, atau pertukaran elemen. Setiap edge merepresentasikan aliran kontrol dari satu langkah logika ke langkah berikutnya. Dalam pengujian white box menggunakan teknik *basis path testing*, *flow graph* membantu mengidentifikasi semua jalur independen, seperti jalur ketika elemen terkecil ditemukan, jalur saat elemen perlu ditukar, atau jalur tanpa pertukaran. Dengan memastikan setiap jalur eksekusi dalam algoritma *Selection Sort* diuji, pengujian ini memberikan jaminan bahwa semua kondisi logis dan kontrol dalam kode telah terverifikasi, sehingga memperkecil risiko kesalahan atau jalur yang tidak teruji.

- *Cyclomatic Complexity* ( $V(G)$ )

*Cyclomatic complexity* adalah metrik yang digunakan untuk mengukur kompleksitas sebuah program dengan menghitung jumlah jalur independen dalam kode. Rumus yang digunakan adalah sebagai berikut:

*Cyclomatic complexity* dihitung dengan rumus:

- $V(G) = \text{Jumlah region} = 2$
- $V(G) = E - N + 2 = 8 - 7 + 2 = 3$

Keterangan :

E = Jumlah *Edge*

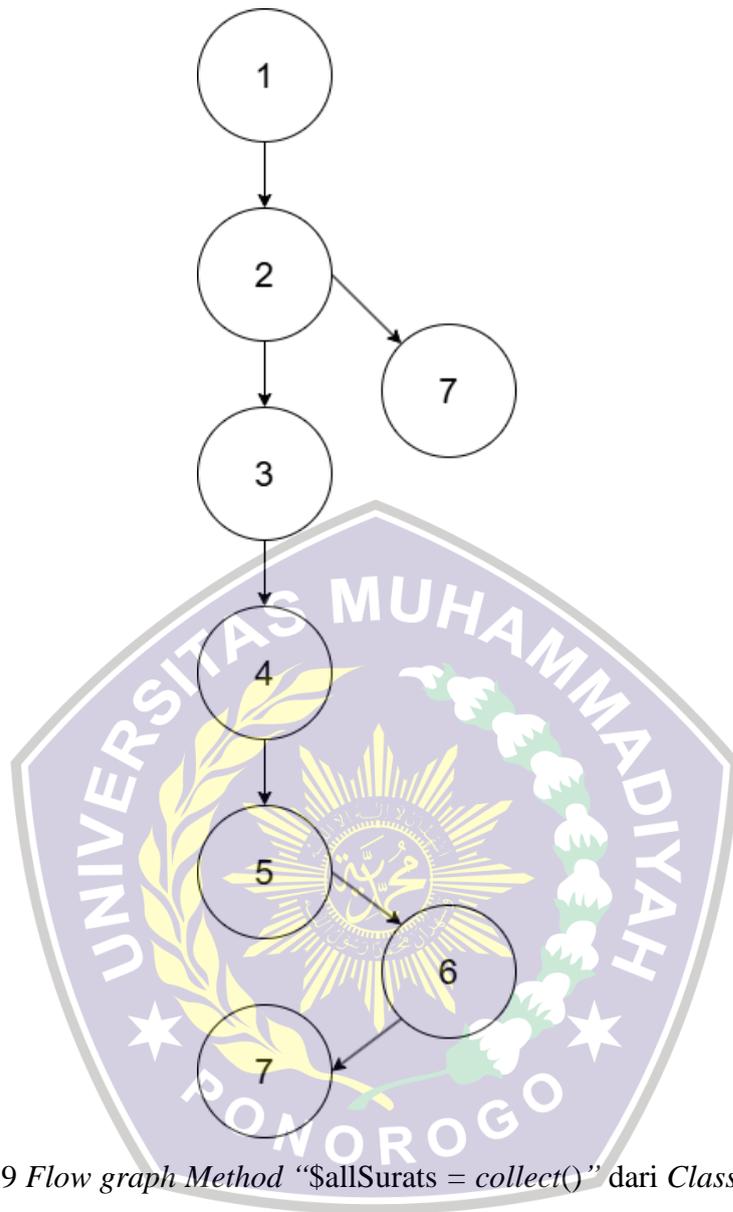
N = Jumlah *Node*

- $V(G) = P + 1 = 2 + 1 = 3$

Semua metode memberikan hasil yang konsisten yaitu  $V(G) = 3$ . *Cyclomatic complexity* sebesar 2 menunjukkan bahwa ada dua jalur independen yang perlu diuji untuk cakupan penuh dari metode tersebut.

- *Independent Path*

- Jalur 1:  $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 7$
- Jalur 2:  $1 \rightarrow 2 \rightarrow 7$
- Jalur 3:  $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7$



Gambar 4.9 *Flow graph Method “\$allSurats = collect()” dari Class “Surat”*

## **BAB 5**

### **PENUTUP**

#### **5.1 Kesimpulan**

Kesimpulan Implementasi *Selection Sort* pada Aplikasi Surat Keluar di Kelurahan Bulukerto Berbasis Web :

1. Penerapan Algoritma *Selection Sort* digunakan untuk mengelola dan mengurutkan data surat secara sistematis dengan memilih elemen terkecil dari daftar yang belum terurut dan menempatkannya di posisi yang sesuai. Proses ini memungkinkan data surat diurutkan berdasarkan kriteria tertentu, seperti tanggal, nomor, atau jenis surat. Metode ini efektif untuk volume data sebanyak 7 hingga 10 surat per harinya dan memberikan pengelolaan data yang lebih terstruktur.
2. Pengujian *white box* memastikan logika algoritma *Selection Sort* berjalan dengan benar, terutama dalam pengurutan atribut *noSurat*. Data dari berbagai tabel, seperti *Domisili*, *Usaha*, *Bedanama*, *HakPakai*, dan *KetPengantar*, digabungkan menggunakan metode *collect()* dan *merge()* sebelum diuji. Hasil pengujian menunjukkan algoritma mampu mengurutkan data surat secara akurat dan meningkatkan keandalan sistem dalam mengelola informasi administrasi.
3. Keunggulan dan Manfaat Sistem Berbasis Web Sistem ini membantu staf kelurahan mengelola data surat keluar secara efisien dengan fitur pengurutan yang mempermudah penyusunan laporan administrasi secara profesional dan akurat. Dengan sifatnya yang berbasis web, sistem dapat diakses kapan saja tanpa bergantung pada koneksi internet, sehingga sangat bermanfaat bagi daerah dengan akses internet terbatas. Hal ini mendukung kebutuhan administrasi yang terorganisir dan fleksibel.

#### **5.2 Saran**

Berdasarkan kesimpulan di atas, saran untuk penelitian selanjutnya adalah sebagai berikut :

- 1) Pengembangan sistem dengan menambahkan fitur *multi-user* untuk memungkinkan lebih dari satu pengguna mengakses sistem secara bersamaan, sehingga meningkatkan kolaborasi dalam pengelolaan surat keluar
- 2) Pengembangan sistem agar dapat diakses secara *online*, memperluas jangkauan penggunaan tanpa batasan koneksi internet
- 3) Eksplorasi algoritma pengurutan lain seperti *Quick Sort* atau *Merge Sort* dapat dilakukan untuk membandingkan performa dalam hal kecepatan dan efisiensi, terutama pada volume data yang lebih besar



## DAFTAR PUSTAKA

- [1] L. Andraini and C. Bella, “Pengelolaan Surat Menyurat Dengan Sistem Informasi ( Studi Kasus : Kelurahan Gunung Terang),” *J. Portal Data*, vol. 2, no. 1, pp. 1–11, 2022, [Online]. Available: <http://portaldata.org/index.php/portaldata/article/view/71>
- [2] R. Simanungkalit and T. A. Agustina, “Algoritma Selection Sort Untuk Pengelolaan Berkas Surat Berbasis Web (Studi Kasus: Kantor Notaris PPAT),” *J. armada Inform.*, no. 88, pp. 1–6, 2022.
- [3] D. U. P. dan P. H. P. K. Negara and B. P. Keuangan, “Peraturan Bupati (Perbup) Kabupaten Wonogiri Nomor 25 Tahun 2021 tentang Tata Naskah Dinas di Lingkungan Pemerintah Kabupaten Wonogiri,” *PERATURAN DATABASE*, 2021. [https://peraturan.bpk.go.id/Details/215526/perbup-kab-wonogiri-no-25-tahun-2021?utm\\_source=chatgpt.com](https://peraturan.bpk.go.id/Details/215526/perbup-kab-wonogiri-no-25-tahun-2021?utm_source=chatgpt.com) (accessed Feb. 05, 2025).
- [4] J. Banjarnahor, D. Bawamenewi, C. Tanoto, and M. NK Nababan, “Implementasi Metode Selection Sort Dalam Sistem Repository Skripsi,” *J. Sist. Inf. dan Ilmu Komput. Prima (JUSIKOM PRIMA)*, vol. 5, no. 2, pp. 107–113, 2022, doi: 10.34012/jurnalsisteminformasidanilmukomputer.v5i2.2389.
- [5] Susanto, “Rancang Bangun Aplikasi Pengelolaan Arsip Inaktif Dinas Kearsipan Dan Perpustakaan Daerah Kabupaten Grobogan Menggunakan Metode Selection Sort,” Universitas Semarang, 2023.
- [6] N. Haming, S. Lestanti, and S. Nur Budiman, “Aplikasi Pengelolaan Surat Keluar Menggunakan Sequential Search Dan Selection Sort Pada Kpu Kota Blitar,” *JATI (Jurnal Mhs. Tek. Inform.*, vol. 6, no. 1, pp. 17–25, 2021, doi: 10.36040/jati.v6i1.4291.
- [7] Lasriana and A. Gunaryati, “Sistem Informasi Apotek Berbasis Web Menggunakan Algoritma Sequential Search Dan Selection Sort,” *JUPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.*, vol. 7, no. 2, pp. 392–401, 2022, doi: 10.29100/jipi.v7i2.2709.

- [8] A. Syahputra, "Implementasi Algoritma Selection Sort Untuk Pengurutan Nilai IPK Mahasiswa Universitas Potensi Utama," *J. Tek. Inform. Kaputama*, vol. 6, no. 2, pp. 390–398, 2022.
- [9] A. Kuswantoro, R. B. Maremitha Ungu, W. D. Rahmahwati, and F. D. Rahmawati, "Manajemen Surat Masuk Dan Surat Keluar Di Universitas Negeri Semarang Melalui Siradi (Sistem Informasi Surat Dinas)," *J. Pustaka Budaya*, vol. 9, no. 1, pp. 42–49, 2022, doi: 10.31849/pb.v9i1.7716.
- [10] J. Hutagalung, H. Winata, and H. Jaya, "Perancangan Dan Implementasi E-Learning Berbasis Web Pada SMA Negeri 1 Siantar," *J-SISKO TECH*, vol. 62, no. 1, pp. 62–68, 2019, [Online]. Available: <https://ojs.trigunadharma.ac.id/index.php/jsk/article/viewFile/90/43>
- [11] R. T. Septian, "Aplikasi Pembelajaran Interaktif Untuk Anak Paud Menggunakan Html 5," *J. Teknol. Pint.*, vol. 3, no. 3, pp. 1–16, 2023.
- [12] Muhammad Saed Novendri, A. Saputra, and C. E. Firman, "Aplikasi Inventaris Barang Pada Mts Nurul Islam Dumai Menggunakan Php Dan Mysql," *Lentera Dumai*, vol. 10, no. 2, pp. 46–57, 2019.
- [13] R. Hermiati, A. Asnawati, and I. Kanedi, "Pembuatan E-Commerce Pada Raja Komputer Menggunakan Bahasa Pemrograman Php Dan Database Mysql," *J. Media Infotama*, vol. 17, no. 1, pp. 54–66, 2021, doi: 10.37676/jmi.v17i1.1317.
- [14] F. Soufitri, "Perancangan Data Flow Diagram Untuk Sistem Informasi Sekolah (Studi Kasus Pada Smp Plus Terpadu)," *Ready Star*, vol. 2, no. 1, pp. 240–246, 2019.
- [15] R. Nurmasari, S. Pinem, and U. Nurkhalifah, "Perancangan Pengelolaan Data Pelabuhan Perikanan Nusantara (PPN) Pelabuhan Ratu Menggunakan Entity Relationship Diagram (ERD)," *J. Ilm. Rekayasa dan Manaj. Sist. Inf.*, vol. 9, no. 1, pp. 52–57, 2023.
- [16] I. Budiman, S. Saori, R. N. Anwar, Fitriani, and M. Y. Pangestu, "Analisis Pengendalian Mutu Di Bidang Industri Makanan (Studi Kasus: UMKM Mochi Kaswari Lampion Kota Sukabumi)," *J. Inov. Penelit.*, vol. 1, no. 10, pp. 2185–2190, 2021.

- [17] M. F. Londjo, “Implementasi White Box Testing Dengan Teknik Basis Path Pada Pengujian Form Login,” *J. Siliwaangi*, vol. 7, no. 2, pp. 35–40, 2021.

