

# SISTEM INFORMASI PENGARSIPAN SURAT MENGGUNAKAN METODE AGILE DI DEPARTEMEN MANAJEMEN HUTAN IPB UNIVERSITY

TEJO MULYONO



TEKNOLOGI REKAYASA PERANGKAT LUNAK  
SEKOLAH VOKASI  
INSTITUT PERTANIAN BOGOR  
BOGOR  
2025

@Hak cipta milik IPB University

IPB University





## PERNYATAAN MENGENAI LAPORAN PROYEK AKHIR DAN SUMBER INFORMASI SERTA PELIMPAHAN HAK CIPTA

Dengan ini saya menyatakan bahwa Laporan Proyek Akhir dengan judul “Sistem Informasi Pengarsipan Surat Menggunakan Metode Agile di Departemen Manajemen Hutan IPB University” adalah karya saya dengan arahan dari dosen pembimbing dan belum diajukan dalam bentuk apa pun kepada perguruan tinggi mana pun. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan maupun tidak diterbitkan dari penulis lain telah disebutkan dalam teks dan dicantumkan dalam Daftar Pustaka di bagian akhir Laporan Proyek Akhir ini.

Dengan ini saya melimpahkan hak cipta dari karya tulis saya kepada Institut Pertanian Bogor.

Bogor, Agustus 2025

Tejo Mulyono  
J0303211144

## ABSTRAK

TEJO MULYONO. Sistem Informasi Pengarsipan Surat Menggunakan Metode Agile di Departemen Manajemen Hutan IPB University. Dibimbing oleh ADITYA WICAKSONO.

Perkembangan teknologi informasi telah mendorong transformasi digital dalam pengelolaan arsip, termasuk surat menyurat di lingkungan instansi akademik. Departemen Manajemen Hutan IPB University saat ini masih mengelola surat masuk dan keluar secara semi-digital, yang menyulitkan pencarian dokumen dan berpotensi menghambat proses administrasi. Penelitian ini bertujuan untuk merancang dan mengevaluasi sistem informasi pengarsipan surat berbasis web guna mempermudah pencarian dokumen secara cepat, dan akurat. Sistem dikembangkan menggunakan metode agile yang melibatkan pengguna dalam setiap iterasi, dengan teknologi React.js untuk antarmuka, Express.js sebagai *backend*, PostgreSQL untuk basis data, serta Elasticsearch berbasis metode BM25 untuk pencarian dokumen. Pengujian sistem dilakukan dengan *blackbox testing* terhadap 28 skenario, dan seluruh fitur berjalan sesuai kebutuhan pengguna. Sistem juga telah dinilai kerentanannya dari celah keamanan melalui *Vulnerability Assessment*, dan seluruh celah dengan risiko sedang hingga tinggi berhasil diperbaiki. Hasil akhir menunjukkan bahwa sistem ini tidak hanya mampu menangani proses surat menyurat secara digital, tetapi juga siap diimplementasikan secara fungsional dan aman di lingkungan Departemen Manajemen Hutan IPB University.

Kata kunci: Agile, Elasticsearch, pengarsipan surat, sistem informasi

## ABSTRACT

TEJO MULYONO. Mail Archiving Information System Using the Agile Method in the Department of Forest Management IPB University. Supervised by ADITYA WICAKSONO.

Technological advancements have driven digital transformation in archive management, including correspondence in academic institutions. The Department of Forest Management at IPB University currently manages incoming and outgoing letters using a semi-digital approach, which complicates document retrieval and may hinder administrative processes. This study aims to design and evaluate a web-based letter archiving information system to facilitate faster, and more accurate document search. The system was developed using the Agile method, involving users in each iteration, with React.js as the front-end, Express.js for the back-end, PostgreSQL as the database, and Elasticsearch using the BM25 method for document search functionality. System testing was conducted through blackbox testing on 28 scenarios, all of which functioned according to user requirements. A Vulnerability Assessment was performed to identify potential security gaps, where all medium to high-risk vulnerabilities were resolved. The results indicate that the system not only handles correspondence processes digitally but is also functionally and securely ready for implementation within the Department of Forest Management at IPB University.

Keywords: Agile, Elasticsearch, information system, letter archiving



@Hak cipta milik IPB University

- Hak Cipta Dilindungi Undang-undang
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
    - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
    - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
  2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

© Hak Cipta milik IPB, tahun 2025  
Hak Cipta dilindungi Undang-Undang

*Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan atau menyebutkan sumbernya. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik, atau tinjauan suatu masalah, dan pengutip tersebut tidak merugikan kepentingan IPB.*

*Dilarang mengumumkan dan memperbanyak Sebagian atau seluruh karya tulis ini dalam bentuk apa pun tanpa izin IPB.*

# **SISTEM INFORMASI PENGARSIPAN SURAT MENGGUNAKAN METODE AGILE DI DEPARTEMEN MANAJEMEN HUTAN IPB UNIVERSITY**

**TEJO MULYONO**

Laporan Proyek Akhir  
Sebagai salah satu syarat untuk memperoleh gelar  
Sarjana Terapan pada  
Program Studi Teknologi Rekayasa Perangkat Lunak

**TEKNOLOGI REKAYASA PERANGKAT LUNAK  
SEKOLAH VOKASI  
INSTITUT PERTANIAN BOGOR  
BOGOR  
2025**

@Hak cipta milik IPB University

IPB University





## @Hak cipta milik IPB University

Hak Cipta Dilindungi Undang-undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
  - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.





Judul Proyek Akhir : Sistem Informasi Pengarsipan Surat Menggunakan Metode Agile di Departemen Manajemen Hutan IPB University

Nama : Tejo Mulyono  
NIM : J0303211144

@Hak cipta milik IPB University

Disetujui oleh

Pembimbing :  
Aditya Wicaksono, S.Komp., M.Kom.

Diketahui oleh

Ketua Program Studi:  
Medhanita Dewi Renanti, S.Kom., M.Kom.  
NPI 201807198305122001

Dekan Sekolah Vokasi:  
Dr. Ir. Aceng Hidayat, M.T.  
NIP 196607171992031003

Tanggal Ujian:  
13 Agustus 2025

Tanggal Lulus:



## PRAKATA

Puji dan syukur penulis panjatkan kepada Allah subhanaahu wa ta'ala atas segala karunia-Nya sehingga karya ilmiah ini berhasil diselesaikan. Tema yang dipilih dalam penelitian yang dilaksanakan sejak bulan November 2024 sampai bulan Juni 2025 ini ialah Sistem Informasi, dengan judul “Sistem Informasi Pengarsipan Surat Menggunakan Metode Agile di Departemen Manajemen Hutan IPB University”.

Terima kasih penulis ucapkan kepada para pembimbing, Aditya Wicaksono, S.Komp., M.Kom. yang telah membimbing dan banyak memberi saran. Ucapan terima kasih juga disampaikan kepada Departemen Manajemen Hutan IPB University, pembimbing lapang Dr. Soni Trison, S.Hut., M.Si. dan staf tenaga kependidikan Departemen Manajemen Hutan yang telah membantu selama pengumpulan data. Ungkapan terima kasih juga disampaikan kepada ibu, kakak, serta seluruh keluarga yang telah memberikan dukungan, doa, dan kasih sayangnya.

Semoga karya ilmiah ini bermanfaat bagi pihak yang membutuhkan dan bagi kemajuan ilmu pengetahuan.

Bogor, Agustus 2025

*Tejo Mulyono*



Hak Cipta Dilindungi Undang-undang  
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :  
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah  
b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.  
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

## DAFTAR ISI

DAFTAR TABEL	viii
DAFTAR GAMBAR	viii
DAFTAR LAMPIRAN	ix
I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan	2
1.4 Manfaat	2
1.5 Ruang Lingkup	2
II TINJAUAN PUSTAKA	3
2.1 Elasticsearch	3
2.2 ReactJS	3
2.3 ExpressJS	3
2.4 PostgreSQL	3
2.5 Metode Agile	4
2.6 <i>Vulnerability Assessment</i>	4
2.7 OWASP	4
III METODE	5
3.1 Lokasi dan Waktu	5
3.2 Teknik Pengumpulan Data dan Analisis Data	5
3.3 Prosedur Kerja	5
IV HASIL DAN PEMBAHASAN	8
4.1 <i>Requirements</i>	8
4.2 <i>Design</i>	13
4.3 <i>Development</i>	23
4.4 <i>Testing</i>	32
4.5 <i>Deployment</i>	35
4.6 <i>Review</i>	37
V KESIMPULAN DAN SARAN	38
5.1 Kesimpulan	38
5.2 Saran	38
DAFTAR PUSTAKA	39
RIWAYAT HIDUP	81



# DAFTAR TABEL

1 Kebutuhan Fungsional	12
2 Kebutuhan Fungsional (lanjutan)	13
3 Daftar <i>activity diagram</i> sistem pengarsipan surat	15
4 <i>Data Dictionary</i> Surat Masuk	16
5 <i>Data Dictionary</i> Surat Masuk (lanjutan)	17
6 <i>Data Dictionary</i> Surat Keluar	17
7 <i>Data Dictionary</i> Surat Keluar (lanjutan)	18
8 <i>Data Dictionary Document</i>	18
9 <i>Data Dictionary Disposisi</i>	19
10 <i>Data Dictionary Review</i>	19
11 <i>Data Dictionary User</i>	20
12 <i>Data Dictionary Recipient</i>	20
13 <i>Data Dictionary Log</i> Perubahan	21
14 <i>Data Dictionary</i> Klasifikasi Surat	21
15 <i>Data Dictionary</i> Tujuan Surat	22
16 Produk <i>Backlog</i>	24
17 <i>Sprint</i> 1	25
18 <i>Sprint</i> 2	26
19 <i>Sprint</i> 3	27
20 <i>Sprint</i> 3 (lanjutan)	28
21 <i>Sprint</i> 4	28
22 <i>Sprint</i> 4 (lanjutan)	29
23 <i>Sprint</i> 5	30
24 <i>Sprint</i> 6	31
25 Pengujian Alur Surat Masuk	32
26 Pengujian Alur Surat Masuk (lanjutan)	33
27 Pengujian Alur Surat Masuk	33
29 Pengujian Alur Surat Masuk (lanjutan)	34
30 Pengujian Alur Surat Masuk (lanjutan)	35

# DAFTAR GAMBAR

1 Metode Agile (Roby 2023)	6
2 <i>Flowchart</i> Surat Masuk Sistem Manual	8
3 <i>Flowchart</i> Surat Keluar Sistem Lama	9
4 <i>Flowchart</i> Surat Masuk Usulan	10
5 <i>Flowchart</i> Surat Keluar Usulan	11
6 <i>Use Case Diagram</i>	14
7 <i>Entity Relationship Diagram</i>	16
8 <i>Class Diagram</i>	23
9 Hasil Scan Awal	36
10 Hasil Scan setelah perbaikan	37

## DAFTAR LAMPIRAN

1 Buku Agenda surat masuk	42
2 <i>Activity Diagram Login</i>	42
3 <i>Activity Diagram Logout</i>	43
4 <i>Activity Diagram Input Surat Masuk</i>	44
5 <i>Activity Diagram Edit Surat Masuk</i>	45
6 <i>Activity Diagram Hapus Surat Masuk</i>	46
7 <i>Activity Diagram Arsipkan Surat Masuk</i>	47
8 <i>Activity Diagram Input Disposisi Surat Masuk</i>	48
9 <i>Activity Diagram Proses Disposisi Surat Masuk</i>	49
10 <i>Activity Diagram Selesaikan Surat Masuk</i>	50
11 <i>Activity Diagram Download Arsip Surat Masuk</i>	51
12 <i>Activity Diagram Filter Surat Masuk</i>	52
13 <i>Activity Diagram Search Surat Masuk</i>	53
14 <i>Activity Diagram Input Surat Keluar</i>	54
15 <i>Activity Diagram Edit Surat Keluar</i>	55
16 <i>Activity Diagram Hapus Surat Keluar</i>	56
17 <i>Activity Diagram Arsipkan Surat Keluar</i>	57
18 <i>Activity Diagram Generate Nomor Surat Keluar</i>	58
19 <i>Activity Diagram Menyetujui Surat Keluar</i>	59
20 <i>Activity Diagram Input Review Surat Keluar</i>	60
21 <i>Activity Diagram Proses Revisi Surat Keluar</i>	61
22 <i>Activity Diagram Download Arsip Surat Keluar</i>	62
23 <i>Activity Diagram Filter Surat Keluar</i>	63
24 <i>Activity Diagram Search Surat Keluar</i>	64
25 Struktur folder <i>frontend</i>	65
26 Struktur folder <i>backend</i>	65
27 <i>database postgresSQL</i>	66
28 Tampilan halaman <i>login</i>	66
29 Tampilan tombol <i>sign out</i>	67
30 Tampilan formulir surat masuk	67
31 Proses <i>upload</i> surat masuk	68
32 Tampilan halaman daftar surat masuk	68
33 Tampilan halaman detail surat masuk	69
34 Tampilan formulir <i>edit</i> surat masuk	69
35 Tampilan konfirmasi <i>alert</i> hapus surat masuk	70
36 Tampilan formulir surat keluar	70
37 Tampilan berhasil menyimpan surat masuk	71
38 Tampilan halaman daftar surat keluar	71
39 Tampilan halaman detail surat keluar	72
40 Tampilan formulir <i>edit</i> surat keluar	72
41 Tampilan konfirmasi <i>alert</i> hapus surat keluar	73
42 Tampilan formulir <i>input</i> disposisi	73
43 Tampilan aksi mengarsipkan surat masuk	74
44 Tampilan aksi <i>review</i> surat keluar	74
45 Tampilan aksi mengarsipkan surat keluar	75

Hak Cipta Dilindungi Undang-undang  
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :  
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah  
b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.

2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

46	Tampilan halaman <i>dashboard</i> ktu	75
47	Tampilan halaman <i>dashboard</i> administrasi	76
48	Tampilan pelacakan perubahan surat masuk	76
49	Tampilan pelacakan perubahan surat keluar	77
50	Potongan kode <i>indexing</i> surat masuk	77
51	Potongan kode <i>indexing</i> surat keluar	78
52	Tampilan hasil pencarian <i>query</i> surat masuk	78
53	Tampilan hasil pencarian <i>query</i> surat keluar	79
54	Tampilan tombol konfirmasi unduh rekap arsip surat masuk	79
55	Tampilan tombol konfirmasi unduh rekap surat keluar	80

## I PENDAHULUAN

### 1.1 Latar Belakang

Perkembangan teknologi telah mendorong transformasi dalam pengelolaan data dan informasi, termasuk di bidang administrasi. Salah satu tugas administrasi adalah melakukan pengelolaan surat masuk dan keluar, dengan adanya sistem informasi manajemen arsip surat dapat meningkatkan efisiensi dan mengurangi penggunaan kertas pada proses manajemen surat di sebuah instansi (Nilawati dan Widya 2023). Surat bukan hanya sarana komunikasi tetapi juga dapat menjadi dokumen resmi yang merekam jejak dilakukannya kegiatan pada sebuah instansi (Ridwan *et al.* 2022). Sehingga pengelolaan arsip surat menjadi bagian penting, guna memastikan kemudahan akses serta menjaga keutuhan informasi dokumen dari kerusakan maupun kehilangan (Wicaksono *et al.* 2021).

Departemen Manajemen Hutan IPB University merupakan salah satu departemen di Fakultas Kehutanan dan Lingkungan IPB University, yang memiliki fungsi penting dalam mendukung kelancaran kegiatan akademik. Di lingkungan akademik, kegiatan surat-menyurat menjadi bagian yang tidak terpisahkan, di mana seluruh aktivitas yang dilaksanakan umumnya memerlukan persetujuan dalam bentuk surat resmi. Pengelolaan data surat di Departemen Manajemen Hutan masih dilakukan dengan cara semi-digital, dengan cara mencatat surat masuk dan surat keluar pada buku agenda seperti pada lampiran 1 dan Microsoft Excel untuk dilakukan pengarsipan pada map arsip dan disimpan pada lemari. Proses pengelolaan surat saat ini menyulitkan pegawai pada saat mencari kembali surat-surat yang dibutuhkan di masa depan untuk kepentingan akreditasi atau administrasi lain. Ketergantungan pada sistem manual atau semi-digital menyebabkan proses pencarian dokumen memakan waktu lama, berpotensi menyebabkan tersebarnya data, dan menghambat pekerjaan.

Dibutuhkan sebuah sistem informasi yang dapat mengakomodir masalah di departemen manajemen hutan pada proses pengelolaan surat. Penggunaan teknologi berbasis web memberikan solusi praktis untuk mengelola data, seperti dokumen dan surat, dengan cara yang lebih terstruktur. Sistem manajemen surat dirancang untuk menyimpan dan mengatur informasi surat masuk dan keluar dalam format digital sehingga mempermudah pencarian dan pengolahan dengan cepat serta tepat saat diperlukan. Menurut Lembayung *et al.* (2024) dalam penelitiannya, Sistem informasi pengarsipan surat berbasis web dapat mengurangi risiko kehilangan data, mempercepat proses pencarian dokumen, serta meningkatkan efisiensi dan akurasi dalam pengelolaan arsip. Salah satu teknologi yang dapat digunakan untuk mendukung proses pencarian efektif adalah elasticsearch.

Elasticsearch adalah alat bantu yang merupakan mesin analisis dan pencarian teks yang memungkinkan pengguna menyimpan, mencari, dan menganalisis data dalam jumlah besar dengan cepat dan hampir *real-time* (Hakim 2024). Dalam penelitian ini, sistem akan dirancang menggunakan React.js sebagai antarmuka pengguna, Expressjs sebagai server *backend*, PostgreSQL sebagai basis data, dan Elasticsearch untuk mendukung pencarian dokumen berbasis BM25. Metode BM25 merupakan salah satu metode yang dapat digunakan untuk menghitung tingkat kemiripan dokumen (*similarity*) dari *query* yang diberikan terhadap



kumpulan dokumen yang tersedia. Kombinasi teknologi ini dirancang untuk menciptakan sistem arsip dokumen yang efisien dan relevan (Pranata *et al.* 2019).

Penelitian ini berfokus pada perancangan sistem informasi pengarsipan surat yang dikembangkan untuk mengatasi kendala dalam pengelolaan dokumen di Departemen Manajemen Hutan. Dengan menerapkan metode agile, penelitian ini mencakup seluruh tahapan, mulai dari *requirements, design, development, testing, deployment* dan *review*. Metode agile dipilih karena memberikan fleksibilitas dalam proses pengembangan, melakukan kolaborasi antara pengembang dan pengguna sistem untuk memastikan sistem yang dibangun sesuai dengan kebutuhan. Sistem yang dikembangkan diharapkan dapat mendukung proses pencarian, pengarsipan, dan monitoring surat secara efektif dan responsif terhadap kebutuhan operasional.

## 1.2 Rumusan Masalah

Berdasarkan uraian latar belakang di atas, maka dapat dirumuskan permasalahan dalam penelitian ini yaitu:

1. Bagaimana merancang sistem pengarsipan surat yang dapat mengurangi kesulitan dalam pencarian dokumen lama di Departemen Manajemen Hutan IPB University?
2. Apakah sistem pengarsipan surat berfungsi dengan baik dalam pencarian dokumen di Departemen Manajemen Hutan IPB University?

## 1.3 Tujuan

Berikut merupakan tujuan dari penelitian ini:

1. Merancang sistem pengarsipan surat untuk mempermudah pencarian dokumen surat di Departemen Manajemen Hutan IPB University secara cepat dan akurat.
2. Mengevaluasi fungsionalitas sistem pengarsipan surat yang dikembangkan dalam membantu pencarian dokumen di Departemen Manajemen Hutan IPB University.

## 1.4 Manfaat

Adapun manfaat yang ingin dicapai penulis dalam penelitian ini, yaitu mempermudah pihak tata usaha dalam melakukan pencatatan, pencarian, serta pengelolaan surat pada Departemen Manajemen Hutan IPB University.

## 1.5 Ruang Lingkup

Penelitian ini mencakup pengembangan sistem informasi pengarsipan surat berbasis web untuk mengelola surat masuk, surat keluar, disposisi, pencarian, dan pengarsipan digital di Departemen Manajemen Hutan IPB University. Sistem dirancang untuk digunakan oleh administrasi, kepala tata usaha, sekretaris departemen dan ketua departemen, dengan fitur yang sesuai peran masing-masing. Lingkup penelitian difokuskan pada perancangan, implementasi, dan pengujian fungsional sistem menggunakan *blackbox testing* serta penilaian kerentanan dengan *vulnerability assessment*. Penelitian tidak mencakup integrasi dengan sistem eksternal atau pengembangan modul di luar pengarsipan surat.

## II TINJAUAN PUSTAKA

### 2.1 Elasticsearch

Elasticsearch adalah mesin pencarian dan analitik berbasis RESTful yang dirancang untuk menangani berbagai jenis data, seperti teks, angka, data spasial, dan data terstruktur maupun tidak terstruktur. Dibangun di atas Apache Lucene, Elasticsearch memungkinkan pencarian dan analisis data dalam waktu nyata dengan skalabilitas tinggi, sehingga sering digunakan untuk analitik *log*, monitoring kinerja, dan pencarian berbasis teks. Teknologi ini mendukung fitur seperti pencarian penuh, agregasi data, pengindeksan cepat, dan skalabilitas horizontal, yang memungkinkan distribusi data di berbagai *node*. Dengan fleksibilitas dan efisiensinya, Elasticsearch mempermudah organisasi dalam mengolah dan memvisualisasikan data kompleks untuk mendukung pengambilan keputusan strategis (Atmaja dan Yulianto 2019).

### 2.2 ReactJS

ReactJS adalah *framework open-source* yang digunakan untuk membangun antarmuka pengguna secara efisien, terutama dalam aplikasi satu halaman (SPA). Teknologi ini memungkinkan pengembang untuk membuat aplikasi web yang besar dan dinamis dengan mengubah data tanpa memuat ulang seluruh halaman. Dengan algoritma cerdas untuk membandingkan perubahan di DOM, React hanya memperbaharui elemen yang relevan, sehingga mempercepat proses rendering. React juga mendukung pengembangan aplikasi modular melalui penggunaan komponen yang dapat digunakan kembali, membuat pengaturan UI lebih konsisten dan mudah dikelola. Selain itu, fitur seperti *lifecycle methods* dan *React Hooks* memberikan fleksibilitas tambahan, memungkinkan pengembang mengelola siklus hidup komponen dan memilih pustaka sesuai kebutuhan untuk menyelesaikan berbagai tugas secara optimal (Setiawan *et al.* 2024).

### 2.3 ExpressJS

Express.js adalah *framework* aplikasi berbasis web yang dibangun menggunakan core pemrograman Node.js, memanfaatkan modul Http dan Connect. Dikenal dengan kecepatannya, Express.js berjalan dalam mesin V8 Chrome, yang memungkinkan kinerja sangat cepat bersama dengan Node.js. *Framework* ini sangat populer karena dokumentasi yang lengkap, kemudahan penggunaan, dan kemampuan untuk mengembangkan berbagai produk, termasuk aplikasi web dan RESTful API (Widyoutomo *et al.* 2021).

### 2.4 PostgreSQL

PostgreSQL adalah sistem manajemen basis data relasional (RDBMS) *open-source* yang terkenal karena stabilitas, keandalan, dan kemampuannya dalam menangani data kompleks. PostgreSQL mendukung fitur-fitur canggih seperti integritas data, transaksi ACID, serta pemrosesan kueri yang sangat efisien. Sistem ini dirancang untuk menangani *query* yang kompleks, mendukung transaksi bersamaan, dan bekerja dengan volume data yang besar, menjadikannya cocok untuk aplikasi kecil maupun besar. PostgreSQL memiliki fleksibilitas tinggi dengan



kemampuan untuk menambahkan tipe data dan fungsi kustom, serta mendukung pengembangan aplikasi yang membutuhkan skalabilitas besar. PostgreSQL menggunakan skema terstruktur untuk penyimpanan data dan mendukung SQL, memastikan bahwa ia dapat mengelola hubungan antara berbagai entitas data dengan efisien (Fahreza 2024).

## 2.5 Metode Agile

Agile adalah metode pengembangan perangkat lunak yang mengutamakan kecepatan dalam merespon perubahan kebutuhan dalam waktu yang singkat. Penerapan metode agile dengan kolaborasi tim yang intensif dan fokus pada pengembangan aplikasi yang minim dokumentasi. Metode ini mengutamakan iterasi berulang atau literasi, di mana setiap perubahan ditangani secara fleksibel (Roby 2023).

## 2.6 Vulnerability Assessment

*Vulnerability Assessment* (VA) adalah proses evaluasi menyeluruh yang bertujuan untuk mengidentifikasi berbagai celah atau kelemahan dalam sistem keamanan, baik dari sisi jaringan, konfigurasi sistem, pengelolaan infrastruktur, hingga kesadaran pengguna terhadap keamanan. Penilaian ini juga mencakup aspek keamanan fisik dan bertujuan untuk menemukan potensi kerentanan kritis yang dapat membahayakan sistem. Dalam praktiknya, VA dilakukan melalui pengumpulan data, pendeteksian kerentanan, serta pemberian saran perbaikan yang relevan (Wahyudin *et al* 2024).

Seiring bertambahnya jumlah website yang dibangun, risiko serangan siber juga meningkat. Oleh karena itu, menjaga keamanan website menjadi sangat penting. VA menjadi salah satu langkah preventif untuk memastikan sistem yang dibangun tidak memiliki celah yang dapat dimanfaatkan oleh pihak tidak bertanggung jawab.

## 2.7 OWASP

OWASP (*Open Web Application Security Project*) adalah sebuah komunitas riset global yang bersifat nirlaba, terdiri dari para ilmuwan, peneliti, dan praktisi dari sektor swasta yang secara aktif mengembangkan serta mempublikasikan berbagai artikel, perangkat, serta dokumen keamanan aplikasi secara terbuka dan gratis. OWASP direkomendasikan penggunaannya karena menurut para ahli keamanan, OWASP merupakan perangkat lunak sumber terbuka yang dapat diakses dan dimanfaatkan secara bebas. Selain itu, OWASP menyediakan berbagai panduan penting yang membantu pengembang dalam menjaga keamanan aplikasi web. Dalam penelitian ini, terdapat dua dokumen utama dari OWASP yang dijadikan acuan sebagai parameter, yaitu *Developer Cheat Sheet Series* dan *OWASP Top 10* (Darojat *et al* 2022).

### III METODE

#### 3.1 Lokasi dan Waktu

Penelitian ini dilakukan selama proses magang di Departemen Manajemen Hutan IPB University yang berlokasi di Babakan, Kec. Dramaga, Kabupaten Bogor, Jawa Barat 16680. Kemudian, dilanjutkan dengan proses penelitian untuk memenuhi tugas akhir dari Januari hingga April 2025 di Kampus Sekolah Vokasi IPB University yang berlokasi di Jl. Kumbang No.14, Kecamatan Bogor Tengah, Kota Bogor, Jawa Barat 16128.

#### 3.2 Teknik Pengumpulan Data dan Analisis Data

Teknik dalam pengumpulan data yang dilakukan pada penelitian ini adalah sebagai berikut.

##### a. Studi Literatur

Studi literatur dilakukan dengan mengumpulkan informasi dari berbagai sumber tertulis seperti jurnal ilmiah, artikel, laporan penelitian sebelumnya, dan dokumen terkait mengenai pengarsipan surat dan metode pengembangan agile. Tujuan dari dilakukannya studi literatur agar memahami teori dan pendekatan yang relevan, serta mengetahui kelemahan dan kekuatan sistem informasi pengarsipan surat yang sudah ada.

##### b. Wawancara

Melakukan wawancara dengan pihak yang terlibat, yaitu staf tata usaha departemen, kepala departemen, dan pengguna sistem pegawai administrasi. Wawancara ini dilakukan guna mendapat informasi terkait kebutuhan sistem dan keluhan terhadap proses yang sedang berjalan. Berdasarkan hasil wawancara, didapatkan beberapa informasi yaitu alur proses pengelolaan surat masuk dan keluar di Departemen Manajemen Hutan dan masalah mengenai sulitnya menemukan dokumen atau surat lama karena sistem pencatatan yang terbagi antara buku agenda dan Microsoft Excel.

#### 3.3 Prosedur Kerja

Metode yang digunakan dalam penelitian ini adalah metode agile. *Agile Development* merupakan metode pengembangan perangkat lunak berupa iterasi atau perulangan, tujuannya untuk merespon dan mengatasi setiap perubahan secara fleksibel dan mencapai kepuasan pengguna. Metode agile dipilih karena cepat beradaptasi mengatasi perubahan yang diinginkan oleh pengguna pada saat pengembangan berlangsung (Hikmah *et al.* 2021). Tahapan yang dilakukan pada metode agile antara lain *Requirements, Design, Development, Testing, Deployment*, dan *Review*.



Gambar 1 Metode Agile (Roby 2023)

a. *Requirements*

Pada tahapan ini peneliti mengidentifikasi masalah dan mengumpulkan data yang dibutuhkan, dengan cara melakukan observasi dan wawancara kepada pengguna sistem yaitu kepala tata usaha, administrasi, dan ketua departemen. Setelah memahami proses pengarsipan surat yang berjalan saat ini, lalu dilakukan perancangan alur sistem untuk menjawab tantangan pengguna. Sehingga hasil dari tahapan ini dapat digunakan untuk proses perancangan sistem sesuai dengan kebutuhan fungsionalitas.

b. *Design*

Pada tahap ini dilakukan pembuatan *use case diagram* guna memberikan gambaran interaksi antara pengguna dan sistem. Perancangan model sistem dengan menggunakan *entity relationship diagram* (ERD) untuk memodelkan struktur basis data, serta *class diagram* untuk mendeskripsikan struktur dan hubungan antar kelas dalam sistem secara detail.

c. *Development*

Dalam tahap ini, proses pengembangan dilakukan pengkodean sistem yang telah dirancang ke dalam beberapa iterasi (sprint), di mana setiap sprint berfokus pada penyelesaian modul tertentu sesuai dengan pada tahapan desain. Pengembangan dilakukan menggunakan teknologi ReactJS untuk pengembangan antarmuka, NodeJs untuk *backend*, dan PostgreSQL untuk manajemen *database*.

d. *Testing*

Dalam tahap ini dilakukan pengujian untuk memastikan seluruh fungsionalitas sistem telah memenuhi harapan dan sesuai dengan kebutuhan. Pengujian dilakukan dengan menggunakan metode *blackbox testing* untuk mengidentifikasi *error* dan *bug*.

#### e. *Deployment*

Dalam tahap ini, rancangan sistem yang telah layak akan diunggah ke dalam web hosting agar dapat diakses oleh pengguna melalui internet (Suhari *et al.* 2022). Sistem diuji kembali guna memastikan semua stabilitas fungsi menggunakan *vulnerability assessment* sebelum diserahkan ke pengguna. Dengan tahapan ini, sistem manajemen surat diharapkan dapat mendukung operasional secara efektif dan meningkatkan efisiensi pengelolaan data.

#### f. *Review*

Dalam tahap ini, sistem akan dinilai apakah memenuhi tujuan yang diharapkan. Pengguna akan memberikan masukan terkait kemudahan penggunaan dan kesesuaian fungsi dengan kebutuhan operasional. Umpan balik tersebut menjadi dasar untuk melakukan revisi pada sistem, memastikan setiap fitur yang ada memberikan manfaat optimal.

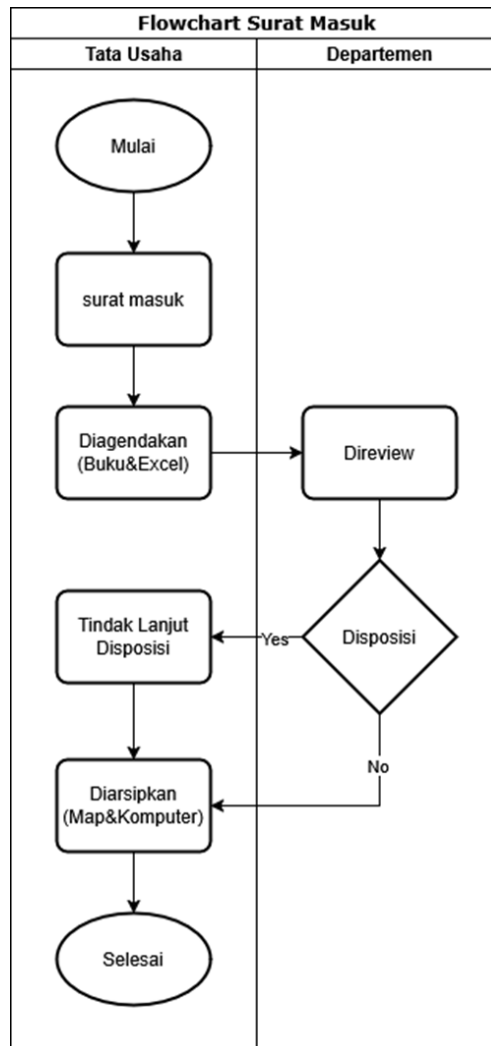


1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
  - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

## IV HASIL DAN PEMBAHASAN

### 4.1 Requirements

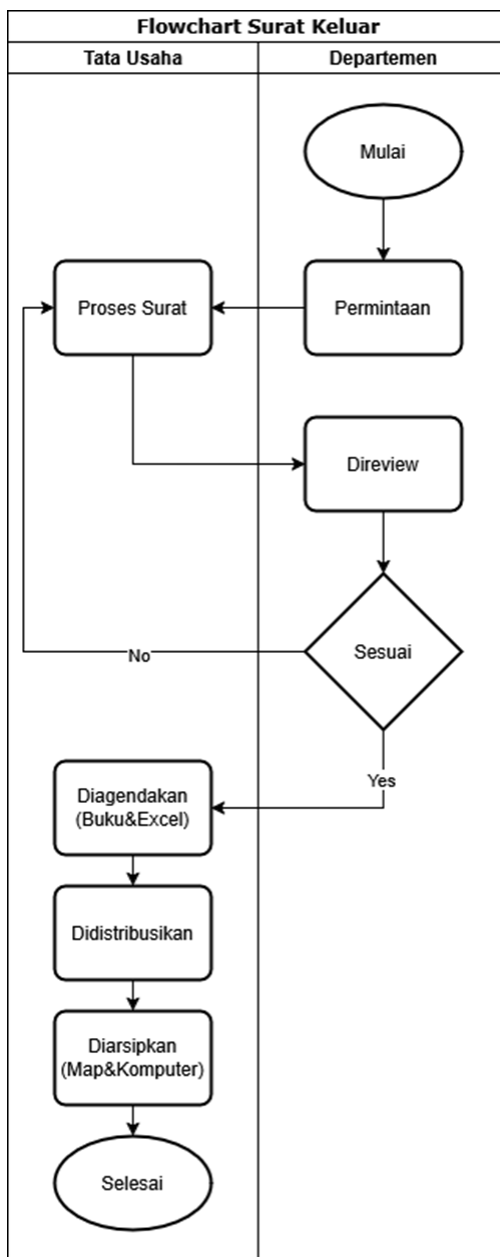
Tahap *requirements* dilakukan dengan pendekatan observasi dan wawancara langsung kepada pengguna sistem, yaitu bagian administrasi, kepala tata usaha serta ketua departemen. Dari hasil pengumpulan data tersebut, diperoleh proses bisnis saat ini yang digambarkan pada gambar 2 dan gambar 3. Berdasarkan analisis hasil wawancara di rancang proses bisnis dengan sistem informasi ditunjukkan pada gambar 4 dan gambar 5. Selanjutnya dilakukan pencatatan kebutuhan fungsional berupa fitur pengarsipan surat masuk dan keluar, unggah dokumen, disposisi, serta filter pencarian surat.



Gambar 2 Flowchart Surat Masuk Sistem Manual

Gambar 2 menunjukkan alur proses bisnis surat masuk yang masih berjalan secara manual. Proses dimulai dari penerimaan surat masuk oleh tata usaha. Setelah itu, surat di agendakan (dalam bentuk catatan di buku atau file Excel) untuk pencatatan awal. Surat kemudian diteruskan ke bagian terkait untuk *review*. Jika surat memerlukan disposisi, proses dilanjutkan dengan tindak lanjut disposisi.

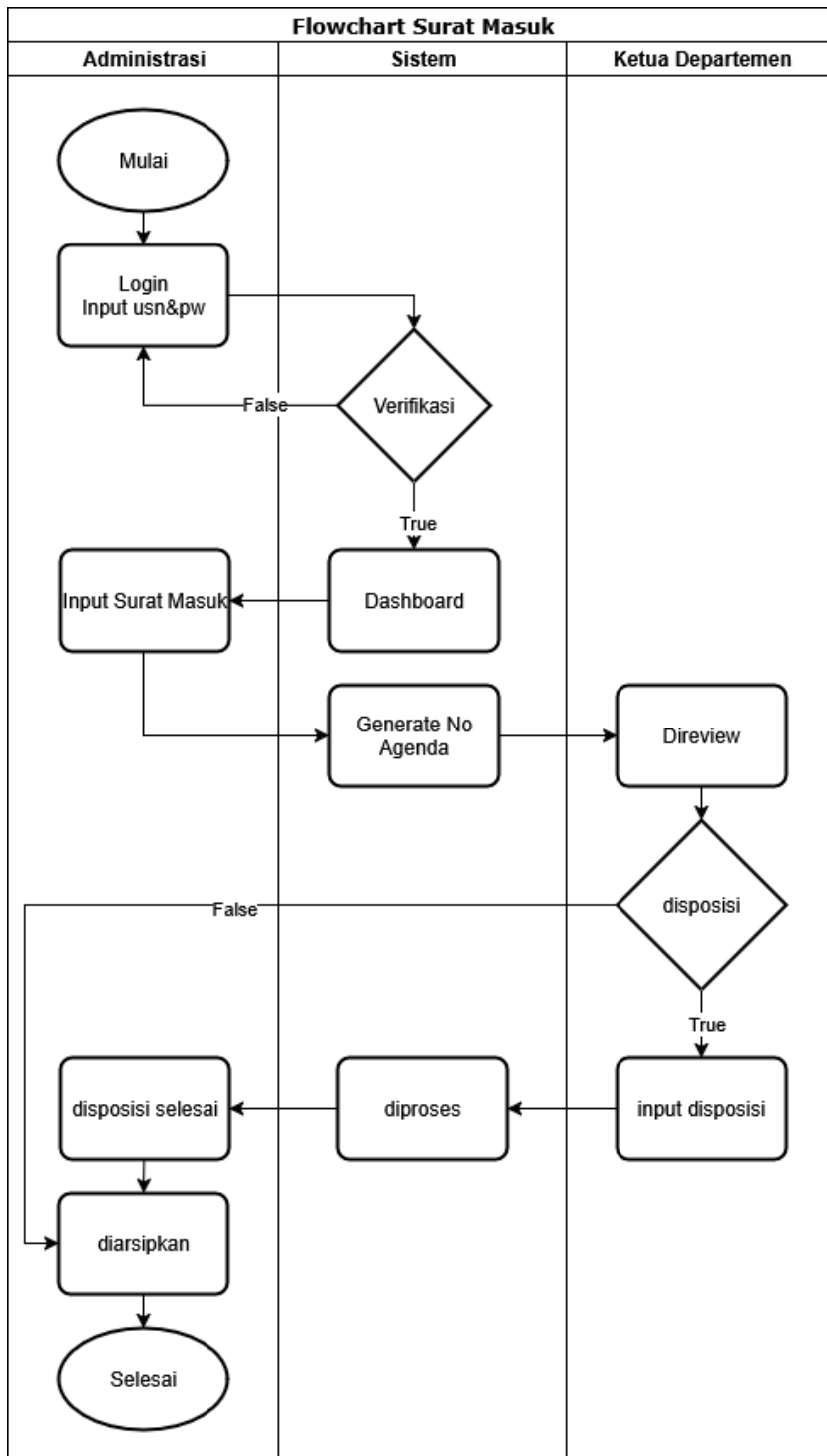
Apabila tidak ada disposisi atau disposisi telah selesai, surat akan diarsipkan secara manual (baik dalam map fisik maupun komputer). Proses berakhir ketika surat telah selesai diarsipkan.



Gambar 3 *Flowchart* Surat Keluar Sistem Manual

Gambar 3 ini mengilustrasikan alur surat keluar pada sistem lama, melibatkan departemen dan tata usaha. Proses dimulai dari permintaan surat oleh departemen, kemudian diproses oleh tata usaha, dan dikembalikan untuk direview. Jika tidak sesuai, surat akan kembali diproses ulang. Surat yang telah disetujui kemudian diagendakan secara manual (buku pada lampiran 1 dan Excel) dan didistribusikan. Terakhir, surat diarsipkan dalam bentuk fisik (map) dan digital (komputer).

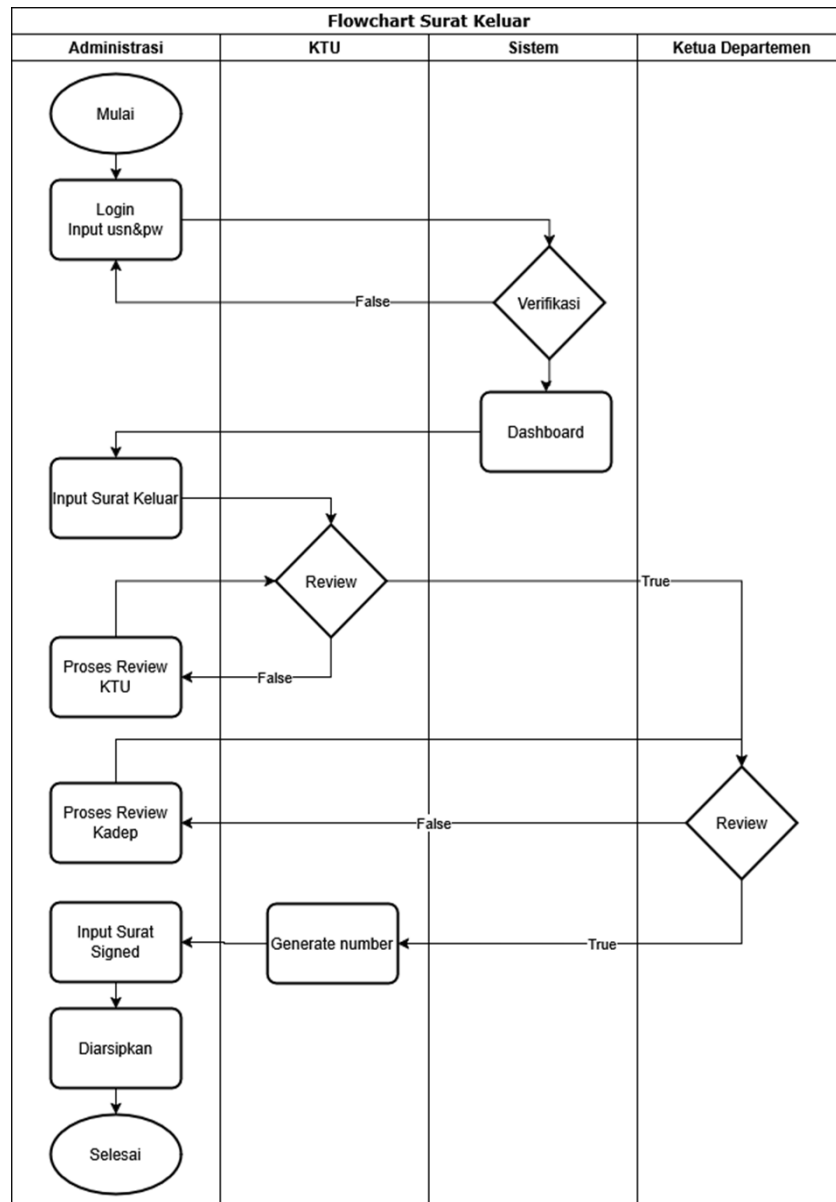




Gambar 4 *Flowchart* Surat Masuk Usulan

*Flowchart* Surat Masuk Usulan ini menguraikan alur proses surat masuk yang lebih terotomatisasi, melibatkan administrasi, sistem, dan ketua departemen. Dimulai dengan login dan verifikasi oleh administrasi ke sistem, kemudian surat masuk diinput dan secara otomatis mendapatkan nomor agenda. Surat tersebut lalu direview dan didisposisikan oleh ketua departemen langsung melalui sistem. Apabila disposisi sudah sesuai, surat akan diproses oleh sistem dan akhirnya diarsipkan secara digital oleh administrasi.





Gambar 5 Flowchart Surat Keluar Usulan

*Flowchart* Surat Keluar Usulan ini menggambarkan proses pengelolaan surat keluar yang terintegrasi dalam sistem baru, melibatkan administrasi, KTU (Kepala Tata Usaha), sistem, dan ketua departemen. Proses diawali dengan *login* dan verifikasi oleh administrasi ke sistem. Setelah berhasil masuk ke *dashboard*, Administrasi melakukan *input* data surat keluar. Surat tersebut kemudian melalui proses *review* berjenjang, pertama oleh KTU, dan jika disetujui, dilanjutkan oleh ketua departemen. Apabila ada ketidaksesuaian, surat akan dikembalikan untuk diproses *review* ulang. Setelah surat disetujui oleh kedua belah pihak, KTU akan membuat nomor untuk surat. Terakhir, administrasi menginput surat yang sudah ditandatangani ke sistem dan mengarsipkannya secara digital, menandai berakhirnya proses.

Berdasarkan analisis *flowchart* proses bisnis yang diusulkan dan hasil wawancara mendalam dengan pengguna, kebutuhan fungsional sistem diidentifikasi secara spesifik. Kebutuhan-kebutuhan ini merupakan fitur yang harus

ada pada sistem agar dapat mendukung alur kerja yang telah dirancang, dapat dilihat pada Tabel 1 (Daftar Kebutuhan Fungsional Sistem) merangkum seluruh *requirements* fungsional beserta deskripsinya.

Tabel 1 Kebutuhan Fungsional

No	Fungsionalitas	Aksi Pengguna	Penjelasan
1	<i>Login &amp; Logout</i>	Pengguna <i>login/logout</i> sesuai akun dan perannya	Sistem mengautentikasi pengguna dan memberikan akses sesuai peran. <i>Logout</i> digunakan untuk keluar dari sistem dengan aman.
2	<i>Input Surat Masuk</i>	Admin mengisi <i>form</i> surat masuk dan menyimpannya	<i>Form input</i> mencatat data surat seperti nomor, tanggal, pengirim, dan perihal.
3	<i>Upload Dokumen Surat Masuk</i>	Unggah file PDF surat	Sistem memvalidasi file agar hanya dokumen PDF yang bisa diunggah.
4	<i>Edit &amp; Hapus Surat Masuk</i>	Ubah atau hapus data surat masuk	Memungkinkan perbaikan data surat yang salah dan penghapusan jika tidak valid.
5	<i>Arsipkan Surat Masuk</i>	Klik tombol arsip dan isi nomor folder	Mengubah status surat menjadi "diarsipkan" agar tidak tampil sebagai surat aktif.
6	<i>Disposisi Surat Masuk</i>	Kepala Departemen memberikan disposisi ke <i>user</i> lain	Menugaskan pengguna lain untuk menindaklanjuti surat. Disposisi disimpan sebagai riwayat surat.
7	<i>Download Arsip Surat Masuk</i>	Klik tombol unduh arsip	Surat yang sudah diarsipkan dapat diunduh dalam format Excel untuk pelaporan.
8	<i>Filter &amp; Cari Surat Masuk</i>	Gunakan filter bulan/tahun atau <i>keyword</i> pencarian	Sistem menampilkan surat yang relevan sesuai filter atau kata kunci.
9	<i>Input Surat Keluar</i>	Admin mengisi <i>form</i> dan mengunggah file	Surat keluar didaftarkan dengan isi surat, penerima, dan dokumen PDF sebagai lampiran.
10	<i>Review &amp; Persetujuan Surat Keluar</i>	KTU dan ketua departemen memberi revisi atau menyetujui	Surat keluar harus ditinjau oleh beberapa pihak sebelum diarsipkan atau dikirim.

Tabel 2 Kebutuhan Fungsional (lanjutan)

No	Fungsionalitas	Aksi Pengguna	Penjelasan
11	<i>Edit &amp; Hapus</i> Surat Keluar	Ubah isi surat keluar atau hapus data	Surat keluar dapat diedit jika perlu atau dihapus jika tidak valid.
12	Arsipkan Surat Keluar	Klik arsip setelah disetujui	Surat keluar diarsipkan agar tidak lagi tampil di daftar surat aktif.
13	Download Arsip Surat Keluar	Unduh file surat keluar yang sudah diarsipkan	Dokumen surat keluar bisa diunduh untuk dokumentasi dan pelaporan.
14	Filter & Cari Surat Keluar	Gunakan filter atau cari berdasarkan kata kunci	Mempermudah pencarian dan penelusuran surat keluar yang diarsipkan.
15	<i>Role Akses</i> Berdasarkan Pengguna	Admin bisa menginput, ketua departemen hanya disposisi/ <i>review</i>	Hak akses sistem dibatasi berdasarkan peran untuk mencegah tindakan diluar kewenangan.

Tabel 1 menyajikan daftar kebutuhan fungsional sistem, yang mencakup berbagai aspek pengelolaan surat. Salah satunya, kebutuhan fungsional Input Surat Masuk (No. 2 pada tabel) mendefinisikan kemampuan sistem untuk menerima dan memvalidasi data surat masuk yang diinput oleh administrasi, sebagaimana direfleksikan dalam langkah Input Data Surat Masuk pada *flowchart* usulan surat masuk. Demikian pula, kebutuhan Pelacakan Perubahan Surat Masuk (No. 3 pada tabel) memastikan bahwa setiap modifikasi pada data surat tercatat, mendukung transparansi dan akuntabilitas. Secara keseluruhan, kebutuhan-kebutuhan ini dirancang untuk mewujudkan proses bisnis yang efisien dan terdokumentasi dengan baik, sebagaimana telah dipaparkan pada *flowchart* usulan.

Berdasarkan tabel 1 dan tabel 2 tersebut dapat dilihat seluruh kebutuhan fungsional yang diperlukan dalam sistem pengarsipan surat agar memudahkan proses bisnis yang sudah berjalan saat ini. Kebutuhan ini kemudian dituangkan ke dalam daftar backlog dan divalidasi secara informal bersama pengguna sebelum tahap pengembangan dimulai.

#### 4.2 Design

Pada tahap ini, dilakukan proses perancangan kebutuhan sistem yang menjadi dasar dalam pengembangan aplikasi atau sistem informasi. Perancangan ini bertujuan untuk menggambarkan secara jelas bagaimana sistem akan berjalan serta bagaimana komponen-komponen di dalamnya saling berinteraksi. Salah satu alat bantu yang digunakan dalam tahap ini adalah *use case diagram* yang berfungsi untuk memvisualisasikan hubungan antara aktor dan fungsionalitas sistem. Selain itu, perancangan basis data juga dilakukan dengan menyusun *entity relationship diagram* (ERD) dan *class diagram*, yang bertujuan untuk menggambarkan struktur

a. *Use Case Diagram*

*Use Case Diagram* sistem ini menunjukkan bahwa pengembangan fungsionalitas telah memperhatikan pembagian otorisasi dan peran pengguna yang berbeda. Dengan membedakan hak akses setiap aktor, sistem menjadi lebih aman dan sesuai dengan struktur organisasi. Diagram ini sekaligus menjadi dasar pembuatan modul-modul sistem yang dikembangkan pada tahap *development*.



Gambar 6 *Use Case Diagram*

b. *Activity Diagram*

*Activity Diagram* digunakan untuk menggambarkan alur aktivitas atau proses bisnis dalam sistem secara terstruktur. Diagram ini menunjukkan urutan langkah-langkah yang dilakukan aktor dan sistem, mulai dari awal hingga akhir proses, sehingga memudahkan analisis alur kerja. Berikut ini disajikan daftar activity diagram dalam bentuk tabel yang merangkum setiap proses utama yang dimodelkan, membantu memahami cakupan dan hubungan antar aktivitas secara lebih menyeluruh.

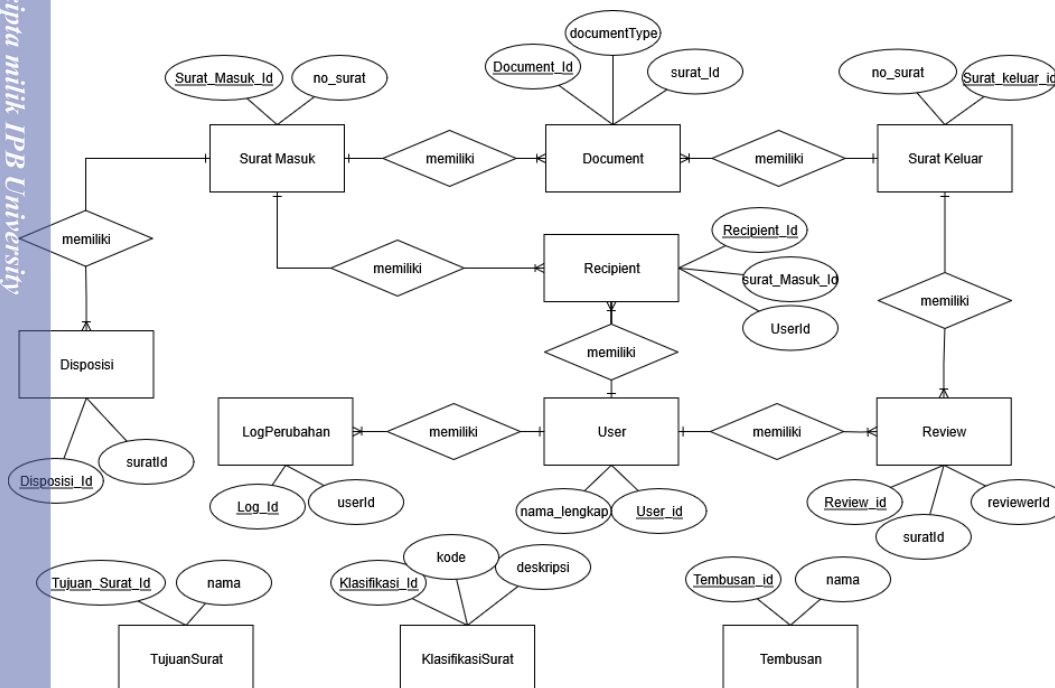
Tabel 3 Daftar *activity diagram* sistem pengarsipan surat

No	<i>Activity Diagram</i>	Gambar
1	<i>Login</i>	Lampiran 2
2	<i>Logout</i>	Lampiran 3
3	<i>Input</i> surat masuk	Lampiran 4
4	<i>Edit</i> surat masuk	Lampiran 5
5	Hapus surat masuk	Lampiran 6
6	Arsipkan surat masuk	Lampiran 7
7	<i>Input</i> disposisi	Lampiran 8
8	Proses disposisi	Lampiran 9
9	Selesaikan surat masuk	Lampiran 10
10	<i>Download</i> arsip surat masuk	Lampiran 11
11	<i>Filter</i> surat masuk	Lampiran 12
12	<i>Search</i> surat masuk	Lampiran 13
13	<i>Input</i> surat keluar	Lampiran 14
14	<i>Edit</i> surat keluar	Lampiran 15
15	Hapus surat keluar	Lampiran 16
16	Arsipkan surat keluar	Lampiran 17
17	<i>Input</i> revisi surat keluar	Lampiran 20
18	Proses revisi surat keluar	Lampiran 21
19	Setujui surat keluar	Lampiran 19
20	<i>Generate</i> nomor surat keluar	Lampiran 18
21	<i>Download</i> arsip surat keluar	Lampiran 22
22	<i>Filter</i> surat keluar	Lampiran 23
23	<i>Search</i> surat keluar	Lampiran 24



### c. Entity Relationship Diagram

ERD (*Entity Relationship Diagram*) ini menggambarkan struktur basis data sistem pengarsipan surat yang terdiri dari entitas utama seperti Surat Masuk, Surat Keluar, Disposisi, User, Document, dan log perubahan, serta entitas pendukung seperti Klasifikasi Surat, Tujuan Surat, dan Recipient. Hubungan antar entitas menunjukkan bagaimana data saling terhubung dan dikelola dalam sistem.



Gambar 7 Entity Relationship Diagram

Dalam perancangan basis data ini, struktur logis sistem direpresentasikan menggunakan *entity relationship diagram* (ERD), yang disajikan pada Gambar 7. ERD tersebut menggambarkan entitas-entitas utama dalam sistem beserta relasi dan kardinalitas antar entitas tersebut. Untuk memberikan deskripsi yang lebih komprehensif mengenai setiap atribut yang menyusun entitas-entitas tersebut, Kamus Data (*Data Dictionary*) disertakan dalam tabel dibawah ini. Kamus Data ini memuat informasi detail seperti nama atribut, tipe data, panjang karakter, serta batasan integritas data seperti properti *unique* dan *nullability*, yang esensial untuk implementasi dan pemeliharaan basis data yang akurat.

Tabel 4 Data Dictionary Surat Masuk

No	Nama Atribut	Tipe Data	Panjang	Unique	Null
1	Surat_Masuk_Id(PK)			Yes	No
2	no_agenda	STRING	255	Yes	No
3	tgl_terima	DATE			No
4	no_surat	STRING	255		No

Tabel 5 Data Dictionary Surat Masuk (lanjutan)

No	Nama Atribut	Tipe Data	Panjang	Unique	Null
5	tgl_surat	DATE			No
6	perihal	STRING	255		No
7	asal_surat	STRING	255		No
8	keterangan	STRING	255		No
9	status	STRING	255		No
10	sifat	STRING	255		No
11	lampiran	JSON			Yes
12	jenis	STRING	255		No
13	penerima	STRING	255		No
14	tembusan	STRING	255		Yes
15	no_folder	STRING	255		Yes

Tabel Surat Masuk digunakan untuk merekam seluruh informasi terkait surat masuk dalam sistem. Atribut utama Surat\_Masuk\_Id berfungsi sebagai *primary key* yang unik dan tidak boleh bernilai null. Atribut no\_agenda juga bersifat unik untuk membedakan setiap entri surat secara khusus. Informasi penting lainnya yang dicatat meliputi nomor dan tanggal surat (no\_surat, tgl\_surat), tanggal diterima (tgl\_terima), perihal, asal surat, penerima, keterangan, status, sifat, dan jenis surat, yang semuanya bertipe string dengan panjang maksimum 255 karakter. Atribut lampiran menggunakan tipe data JSON untuk menyimpan nama dokumen atau file terkait. Selain itu, tembusan dan no\_folder disiapkan untuk mendukung distribusi dan pengarsipan surat secara lebih tertata. Seluruh atribut dirancang agar mampu memenuhi kebutuhan pencatatan dan pelacakan surat masuk secara efektif dan terstruktur.

Tabel 6 Data Dictionary Surat Keluar

No	Nama Atribut	Tipe Data	Panjang	Unique	Null
1	Surat_Keluar_Id(PK)			Yes	No
2	no_surat	STRING	255	Yes	No
3	tgl_surat	DATE			No
4	perihal	STRING	255		No
5	ditujukan	STRING	255		No
6	keterangan	STRING	255		No
7	status	STRING	255		No



Tabel 7 Data Dictionary Surat Keluar (lanjutan)

No	Nama Atribut	Tipe Data	Panjang	Unique	Null
8	sifat	STRING	255		No
9	Lampiran	JSON			Yes
10	jenis	STRING	255		No
11	tembusan	STRING	255		Yes
12	no_folder	STRING	255		Yes

Tabel Surat Keluar merupakan struktur data yang digunakan untuk menyimpan informasi terkait surat keluar pada sistem. Tabel ini memiliki atribut utama Surat\_Keluar\_Id sebagai *primary key* yang bersifat unik dan tidak bernilai null. Atribut no\_surat juga bersifat unik, berisi nomor surat dengan panjang maksimum 255 karakter. Informasi lain yang dicatat meliputi tanggal surat (tgl\_surat), perihal, tujuan (ditujukan), keterangan, status, sifat, dan jenis surat, yang semuanya bertipe data string dengan panjang maksimum 255 karakter. Atribut lampiran menggunakan tipe data JSON untuk menyimpan informasi file yang terkait, sedangkan tembusan dan no\_folder dapat berisi null. Seluruh atribut dirancang agar mendukung kebutuhan pencatatan surat keluar secara sistematis.

Tabel 8 Data Dictionary Document

No	Nama Atribut	Tipe Data	Panjang	Unique	Null
1	Document_Id(PK)			Yes	No
2	documentType	STRING	255		No
3	suratId	STRING	255		No
4	name_doc	STRING	255		No
5	type_doc	STRING	255		No
6	path_doc	STRING	255		No

Tabel *Document* digunakan untuk menyimpan informasi terkait dokumen yang diunggah ke dalam sistem. Atribut Document\_Id berperan sebagai *primary key* dan bersifat unik serta tidak boleh bernilai null. Atribut lain seperti documentType, suratId, name\_doc, type\_doc, dan path\_doc bertipe string dengan panjang maksimal 255 karakter, berfungsi untuk mengidentifikasi jenis dokumen, ID referensinya, nama file, tipe file, dan jalur penyimpanan dokumen. Struktur tabel ini mendukung pengelolaan dokumen secara terorganisir dan terhubung dengan entitas lain dalam sistem.

Tabel 9 Data Dictionary Disposisi

No	Nama Atribut	Tipe Data	Panjang	Unique	Null
1	Disposisi_Id(PK)			Yes	No
2	suratId	STRING	255		No
3	type_surat	STRING	255		No
4	tindakan	STRING	255		No
5	dibuat	STRING	255		No
6	diteruskan	STRING	255		No
7	ket_disposisi	STRING	255		Yes
8	status	STRING	255		No
9	catatan_tindak_lanjut	STRING	255		Yes
10	waktu_selesai	DATE			Yes
11	urutan	INT			No
12	waktu_dibaca	DATE			Yes
13	waktu_diproses	DATE			Yes

Tabel Disposisi digunakan untuk mencatat alur tindak lanjut surat masuk yang didisposisikan oleh pihak berwenang. Atribut Disposisi\_Id berperan sebagai *primary key* yang bersifat unik dan wajib diisi. Tabel ini mencakup informasi seperti suratId, type\_surat, tindakan, dibuat, diteruskan, serta status, yang semuanya bertipe string dengan panjang maksimal 255 karakter. Terdapat pula atribut opsional seperti ket\_disposisi, catatan\_tindak\_lanjut, waktu\_selesai, waktu\_dibaca, dan waktu\_diproses untuk mendukung dokumentasi proses disposisi secara lebih rinci, serta urutan sebagai penanda tahapan proses dalam bentuk integer. Struktur tabel ini penting untuk memastikan transparansi dan akuntabilitas dalam penanganan surat.

Tabel 10 Data Dictionary Review

No	Nama Atribut	Tipe Data	Panjang	Unique	Null
1	review_Id(PK)			Yes	No
2	suratId	INT			No
3	reviewerId	INT			No
4	status	STRING	255		No
5	komentar	TEXT			Yes

Tabel *Review* digunakan untuk mencatat proses peninjauan surat oleh pihak yang berwenang, dalam bentuk setuju dan revisi. Atribut *review\_Id* berfungsi sebagai *primary key* dan harus unik serta tidak boleh kosong. Relasi dengan entitas surat dan pengguna dicerminkan melalui atribut *suratId* dan *reviewerId* yang bertipe integer. Atribut *status* menunjukkan hasil peninjauan, sedangkan *komentar* bersifat opsional dan digunakan untuk memberikan catatan. Struktur tabel ini bertujuan mendukung proses verifikasi surat secara sistematis dan terdokumentasi.

Tabel 11 *Data Dictionary User*

No	Nama Atribut	Tipe Data	Panjang	Unique	Null
1	User_Id(PK)			Yes	No
2	nama_lengkap	STRING	255		No
3	username	STRING	255		No
4	jabatan	STRING	255		No
5	email	STRING	255	Yes	No
6	password	STRING	255		No
7	role	STRING	255		No

Tabel *User* berfungsi untuk menyimpan data pengguna sistem yang memiliki peran tertentu dalam pengelolaan surat. Atribut *User\_Id* sebagai *primary key* bersifat unik dan wajib diisi, begitu pula dengan *nama\_lengkap*, *username*, *jabatan*, *email*, *password*, dan *role* yang mencerminkan identitas dan hak akses pengguna. *Email* ditetapkan sebagai atribut unik untuk menghindari duplikasi akun, sedangkan peran (*role*) digunakan untuk membedakan hak akses antar pengguna seperti admin, kepala bagian, atau staf. Struktur tabel ini dirancang untuk menjamin keamanan dan otorisasi dalam penggunaan sistem.

Tabel 12 *Data Dictionary Recipient*

No	Nama Atribut	Tipe Data	Panjang	Unique	Null
1	Recipient_Id(PK)			Yes	No
2	Surat_Masuk_Id	INT			No
3	UserId	INT			No

Tabel *Recipient* digunakan untuk mencatat data penerima surat masuk dalam sistem. Atribut *Recipient\_Id* berperan sebagai *primary key* untuk mengidentifikasi setiap entri secara unik. Atribut *Surat\_Masuk\_Id* merepresentasikan relasi ke surat masuk yang diterima, sedangkan *UserId* menunjukkan pengguna yang menjadi penerima surat tersebut. Tabel ini penting untuk mengelola distribusi surat masuk kepada pengguna yang relevan dalam sistem.

Tabel 13 *Data Dictionary* Log Perubahan

No	Nama Atribut	Tipe Data	Panjang	Unique	Null
1	Log_Id(PK)			Yes	No
2	suratId	STRING	255		No
3	jenisSurat	STRING	255		No
4	perubahan	STRING	255		No
5	userId	STRING	255		No
6	keterangan	STRING	255		No
7	isRead	STRING	255		No
8	aksi	STRING	255		No

Tabel Log digunakan untuk mencatat seluruh aktivitas perubahan yang terjadi pada data surat dalam sistem. Atribut Log\_Id berfungsi sebagai *primary key* dan wajib diisi untuk menjamin keunikan setiap catatan log. Setiap entri log mencakup informasi terkait suratId, jenisSurat, jenis perubahan yang dilakukan, userId pelaku perubahan, keterangan, status terbaca (*isRead*), dan jenis aksi. Struktur tabel ini penting untuk mendukung *audit trail* dan memastikan transparansi serta akuntabilitas dalam pengelolaan surat.

Tabel 14 *Data Dictionary* KlasifikasiSurat

No	Nama Atribut	Tipe Data	Panjang	Unique	Null
1	Klasifikasi_Id(PK)			Yes	No
2	kode	STRING	255		No
3	deskripsi	STRING	255		No

Tabel Klasifikasi berfungsi untuk menyimpan data pengelompokan atau kategori surat berdasarkan kode tertentu yang digunakan dalam komponen nomor surat keluar. Tabel ini memiliki atribut utama Klasifikasi\_Id sebagai *primary key* yang bersifat unik dan tidak boleh kosong, serta atribut kode dan deskripsi yang menyimpan kode klasifikasi dan penjelasan singkat mengenai klasifikasi tersebut. Penataan tipe data dan aturan null pada tabel ini bertujuan untuk memastikan setiap data klasifikasi tersimpan dengan konsisten dan dapat digunakan sebagai referensi dalam pengelolaan surat.

Tabel 15 *Data Dictionary* Tujuan Surat

No	Nama Atribut	Tipe Data	Panjang	Unique	Null
1	Tujuan_surat_Id(PK)			Yes	No
2	nama	STRING	255		No

Tabel Tujuan Surat digunakan untuk menyimpan data tujuan atau penerima surat dalam sistem. Tabel ini memiliki atribut utama Tujuan\_surat\_Id sebagai *primary key* yang unik dan tidak boleh kosong, serta atribut nama yang berfungsi menyimpan nama tujuan surat dengan tipe data string. Struktur tabel ini dirancang untuk memastikan pengelolaan data tujuan surat dapat dilakukan secara terstruktur dan konsisten.

#### d. Class Diagram

Setelah menyelesaikan tahap perancangan kebutuhan sistem, penyusunan *use case diagram*, serta pemodelan basis data melalui *entity relationship diagram* (ERD), langkah terakhir dalam proses desain adalah pembuatan *class diagram*. Diagram ini menyajikan representasi struktur kelas dalam sistem secara rinci, meliputi atribut, metode, serta relasi antar kelas yang mencerminkan fungsionalitas sistem secara keseluruhan. *Class diagram* berperan penting dalam memahami arsitektur sistem perangkat lunak berbasis objek, serta menunjukkan gambaran visual dari setiap kelas yang saling berinteraksi dan berelasi dalam membentuk sistem yang utuh.





*Class diagram* pada sistem informasi manajemen surat ini menggambarkan struktur dan hubungan antar komponen utama dalam sistem, seperti surat masuk, surat keluar, disposisi, pengguna, dan dokumen pendukung. Diagram ini dirancang untuk merepresentasikan bagaimana data saling berinteraksi di dalam sistem. Dengan menerapkan pendekatan berbasis objek dan arsitektur *Model View Controller* (MVC), sistem dikembangkan secara terstruktur, memisahkan antara logika aplikasi, pengelolaan data, dan tampilan antarmuka. Model ini membantu mempermudah pengembangan, pemeliharaan, serta memperjelas peran setiap bagian dalam sistem. *Class diagram* ini juga mendukung integrasi fitur-fitur penting yang menunjang proses administrasi surat secara efisien dan terorganisir.

Kebutuhan yang telah dikumpulkan pada tahap *requirement* kemudian diolah menjadi daftar *product backlog*. *Backlog* yang ada pada tabel 10 berisi fitur-fitur sistem yang akan dibangun secara bertahap menggunakan pendekatan iterasi. Masing-masing *backlog item* diberikan prioritas berdasarkan urgensi dan kompleksitas teknis, lalu dibagi ke dalam sprint agar pengembangan sistem lebih terstruktur dan terukur.

Tabel 16 Produk *Backlog*

No	Waktu	Fitur Utama	Prioritas	Keterangan
1	1-2 minggu	Setup proyek & struktur folder	Sangat Penting	Inisialisasi proyek ( <i>frontend &amp; backend</i> ), <i>setup database</i> , struktur API.
		Login & Logout	Sangat Penting	Autentikasi <i>user token handling</i>
2	2 minggu	CRUD Surat Masuk	Sangat Penting	Fitur inti untuk pencatatan dan pengelolaan surat masuk
		Upload Dokumen Surat Masuk	Sangat Penting	Upload dan simpan PDF ke server, validasi ekstensi/ukuran
3	2 minggu	CRUD Surat Keluar	Sangat Penting	Fitur inti untuk pencatatan dan pengelolaan surat keluar
		Upload Dokumen Surat Keluar	Sangat Penting	Upload dan simpan PDF ke server, validasi ekstensi/ukuran
4	2 minggu	Mengatur disposisi & review	Sangat Penting	Pengelolaan disposisi dan review
		Pengarsipan surat masuk & keluar	Sangat Penting	Mengubah status surat masuk dan keluar menjadi <i>archive</i>
5	2 minggu	Pembagian akses role	Sangat Penting	<i>Role-based access</i> untuk setiap surat sesuai statusnya
		Pelacakan perubahan surat masuk & keluar	Penting	Log siapa yang melakukan perubahan surat
6	2-3 minggu	Indexing surat masuk & keluar	Sangat Penting	Menyimpan metadata dan ekstrak konten kedalam index surat
		Search serta filter surat masuk & keluar	Sangat Penting	Filter pencarian <i>query</i> dari index
		Download surat masuk & keluar	Sangat Penting	Ekspor surat (Excel) dengan status arsip

Sprint dilakukan sebanyak 6 kali, masing-masing dengan waktu kurang lebih 2 minggu. Fitur-fitur dari *product backlog* dibagi berdasarkan prioritas, kompleksitas teknis, dan keterkaitan logis antar fitur. Setiap sprint dirancang untuk



menghasilkan fungsionalitas sistem yang dapat digunakan dan diuji langsung oleh pengguna.

Setiap sprint diakhiri dengan sesi *review* berupa uji coba fitur yang telah dibangun. *Review* ini dilakukan secara informal bersama pengguna sistem, yaitu administrasi departemen, serta melalui evaluasi pribadi oleh pengembang. Masukan yang diperoleh digunakan untuk perbaikan pada sprint berikutnya. Hal ini dilakukan untuk menjamin bahwa setiap iterasi menghasilkan sistem yang *usable* dan sesuai kebutuhan.

#### a. Sprint 1

Sprint pertama ini berfokus pada inisiasi fundamental proyek pengembangan sistem pengarsipan surat. Tujuan utamanya adalah membangun fondasi teknis yang kokoh dan mengimplementasikan fungsionalitas inti yang memungkinkan pengguna untuk memulai interaksi dengan sistem.

Tabel 17 Sprint 1

No	Backlog	Deskripsi	Output	Status
1.	Setup proyek, struktur folder & database	Sebagai seorang <i>developer</i> , saya ingin lingkungan pengembangan proyek terkonfigurasi dengan benar dan struktur folder serta database sudah terorganisir, agar dapat bekerja secara terstruktur.	Lampiran 25, 26, dan 27	Selesai
2.	Login	Sebagai pengguna, saya ingin dapat masuk ke sistem menggunakan email dan password sesuai peran saya, agar saya bisa mengakses sistem pengarsipan surat.	Lampiran 28	Selesai
3.	Logout	Sebagai pengguna, saya ingin dapat keluar dari sistem dan sesi saya dihentikan dengan aman, agar saya bisa mengakhiri penggunaan sistem pengarsipan surat.	Lampiran 29	Selesai

Pada awal sprint, peneliti memprioritaskan setup lingkungan pengembangan yang meliputi konfigurasi proyek, standarisasi struktur folder, dan inisialisasi skema basis data yang akan menjadi tulang punggung penyimpanan data. Setelah infrastruktur dasar siap, fokus beralih ke implementasi modul otentikasi pengguna yang krusial. Ini mencakup pengembangan fitur *login* yang memungkinkan pengguna untuk masuk ke sistem dengan aman berdasarkan kredensial mereka, serta fitur *logout* yang menjamin pengakhiran sesi pengguna secara aman.

b. Sprint 2

Sprint kedua ini berfokus pada pengembangan modul pengelolaan surat masuk, yang merupakan salah satu inti dari sistem pengarsipan surat. Tujuan utama sprint ini adalah untuk menyediakan fungsionalitas komprehensif yang memungkinkan administrasi untuk memasukkan, mengunggah, melihat, mengubah, dan menghapus data surat masuk dengan akurat.

Tabel 18 Sprint 2

No	Backlog	Deskripsi	Output	Status
1.	<i>Input</i> surat masuk	Sebagai administrasi, saya ingin dapat mengisi <i>form input</i> surat masuk dengan lengkap dan validasi yang akurat, agar data surat yang tersimpan benar dan sesuai.	Lampiran 30 dan 31	Selesai
2.	<i>Upload</i> surat masuk	Sebagai administrasi, saya ingin dapat mengunggah file surat masuk dalam format PDF, agar dokumen asli tersimpan dengan aman dan dapat diakses kapan saja.	Lampiran 30	Selesai
3.	Daftar surat masuk	Sebagai Pengguna, saya ingin dapat melihat daftar semua surat masuk dalam bentuk tabel atau list yang terorganisir, agar saya bisa dengan cepat meninjau dan mencari surat yang relevan.	Lampiran 32	Selesai
4.	Detail surat masuk	Sebagai pengguna, saya ingin dapat melihat detail lengkap dari setiap surat masuk, agar saya memiliki semua informasi yang saya butuhkan untuk memahami surat tersebut.	Lampiran 33	Selesai
5.	<i>Edit</i> surat masuk	Sebagai administrasi, saya ingin dapat mengubah data surat masuk yang sudah tersimpan, agar saya bisa mengoreksi kesalahan atau memperbarui informasi yang berubah.	Lampiran 34	Selesai
6.	Hapus surat masuk	Sebagai administrasi, saya ingin dapat menghapus data surat masuk dari sistem, agar mengurangi data yang tidak berguna.	Lampiran 35	Selesai

Dimulai dengan implementasi formulir *input* surat masuk yang dilengkapi validasi data untuk memastikan integritas informasi. Selanjutnya, kemampuan unggah file surat masuk dalam format PDF dikembangkan agar dokumen fisik dapat tersimpan secara digital dan terintegrasi dengan data di basis data. Untuk

memudahkan pengguna, fungsionalitas menampilkan daftar surat masuk dalam format tabel yang terorganisir, lengkap dengan fitur pencarian dan paginasi, menjadi prioritas. Selain itu, halaman detail surat masuk dikembangkan untuk menampilkan seluruh informasi relevan dari setiap surat. Terakhir, untuk memastikan fleksibilitas dan pemeliharaan data, fitur *edit* dan hapus surat masuk diimplementasikan, memungkinkan administrasi untuk mengoreksi atau menghapus data yang tidak lagi relevan.

### c. Sprint 3

Sprint ketiga difokuskan khusus pada pengembangan modul pengelolaan surat keluar, yang menjadi bagian penting dalam siklus dokumen pada sistem pengarsipan ini. Fokus utama sprint ini adalah mengimplementasikan fitur yang memungkinkan administrator mengelola data surat keluar secara menyeluruh, mulai dari proses penginputan hingga penghapusan data. Pengembangan diawali dengan pembuatan formulir input surat keluar yang lengkap, disertai dengan validasi data yang ketat untuk menjamin keakuratan dan keutuhan informasi yang tersimpan.

Tabel 19 Sprint 3

No	Backlog	Deskripsi	Output	Status
1.	<i>Input</i> surat keluar	Sebagai administrasi, saya ingin dapat mengisi <i>form input</i> surat keluar dengan lengkap dan validasi yang akurat, agar data yang akan dikirimkan benar dan sesuai standar.	Lampiran 36 dan 37	Selesai
2.	<i>Upload</i> surat keluar	Sebagai administrasi, saya ingin dapat mengunggah file surat keluar dalam format PDF, agar dokumen resmi tersimpan dengan aman dan dapat diakses kapan saja.	Lampiran 36	Selesai
3.	Daftar surat keluar	Sebagai pengguna, saya ingin dapat melihat daftar semua surat keluar dalam bentuk tabel atau list yang terorganisir, agar saya bisa dengan cepat meninjau dan mencari surat yang relevan.	Lampiran 38	Selesai
4.	Detail surat keluar	Sebagai pengguna, saya ingin dapat melihat detail lengkap dari surat keluar, agar saya memiliki semua informasi yang saya butuhkan.	Lampiran 39	Selesai

Tabel 20 Sprint 3 (*lanjutan*)

No	Backlog	Deskripsi	Output	Status
5.	Edit surat keluar	Sebagai administrasi, saya ingin dapat mengubah data surat keluar yang sudah tersimpan (sebelum status diarsipkan), agar saya bisa mengoreksi kesalahan atau memperbarui informasi yang berubah.	Lampiran 40	Selesai
6.	Hapus surat keluar	Sebagai administrasi, saya ingin dapat menghapus data surat keluar dari sistem (ketika surat sudah tidak relevan atau ada kesalahan input), agar mengurangi data yang tidak berguna.	Lampiran 41	Selesai

Selanjutnya, kapabilitas pengunggahan berkas surat keluar dalam format PDF diimplementasikan, memastikan bahwa dokumen resmi dapat tersimpan secara digital, aman, dan mudah diakses. Untuk memfasilitasi peninjauan dan pencarian, halaman daftar surat keluar dirancang dengan tampilan tabular yang terorganisir, didukung oleh fitur pencarian dan paginasi. Setiap surat juga dapat dilihat secara rinci melalui halaman detail surat keluar, yang menampilkan seluruh informasi relevan. Terakhir, untuk menjamin fleksibilitas dan pemeliharaan data, fungsionalitas modifikasi (*edit*) dan penghapusan surat keluar dikembangkan, memungkinkan administrasi untuk mengoreksi data atau menghapus entri yang tidak lagi relevan, dengan batasan yang sesuai (misalnya, sebelum status diarsipkan). Keberhasilan sprint ini menandai penyelesaian kapabilitas dasar untuk manajemen dokumen surat keluar, melengkapi modul surat masuk yang telah dikembangkan sebelumnya.

#### d. Sprint 4

Sprint keempat ini berfokus pada pengembangan kapabilitas lanjutan dalam manajemen dokumen, khususnya terkait dengan proses disposisi surat masuk dan siklus hidup pengarsipan dokumen surat masuk maupun surat keluar. Tujuan utama sprint ini adalah untuk memastikan efisiensi alur kerja dan ketersediaan dokumen yang telah melalui proses tindak lanjut atau persetujuan.

Tabel 21 Sprint 4

No	Backlog	Deskripsi	Output	Status
1.	Disposisi surat masuk	Sebagai pengguna, saya ingin dapat melakukan proses disposisi pada surat masuk, agar segera diteruskan dan di tindak lanjuti oleh pihak yang berwenang.	Lampiran 42	Selesai

Tabel 22 Sprint 4 (*lanjutan*)

No	Backlog	Deskripsi	Output	Status
2.	Arsipkan surat masuk	Sebagai administrasi, saya ingin dapat mengubah status surat masuk menjadi diarsipkan, agar dokumen yang sudah selesai ditindaklanjuti dapat tersimpan rapi dan mudah ditemukan kembali di kemudian hari.	Lampiran 43	Selesai
3.	Review surat keluar	Sebagai pengguna, saya ingin dapat melakukan proses <i>review</i> dan persetujuan terhadap surat keluar, agar surat yang dikirimkan sudah sesuai standar dan mendapat otorisasi resmi.	Lampiran 44	Selesai
4.	Arsipkan surat keluar	Sebagai administrasi, saya ingin dapat mengubah status surat keluar menjadi diarsipkan, agar surat yang sudah disetujui dapat tersimpan dengan baik dan mudah ditemukan kembali di kemudian hari.	Lampiran 45	Selesai

Pada modul surat masuk, fungsionalitas disposisi diimplementasikan untuk memungkinkan pengguna meneruskan surat kepada pihak yang berwenang dengan cepat, disertai catatan dan pembaruan status yang relevan. Setelah surat masuk selesai ditindaklanjuti, kemampuan untuk mengarsipkan surat masuk dikembangkan, memastikan dokumen tersimpan secara rapi dan tetap dapat diakses di masa mendatang melalui fitur pencarian arsip.

Sementara itu, untuk surat keluar, sprint ini mengedepankan proses review dan persetujuan. Fungsionalitas ini memungkinkan pengguna terkait untuk meninjau detail surat, memberikan persetujuan atau menolak dengan catatan revisi, serta memperbarui status surat keluar di basis data. Setelah surat keluar disetujui dan siap untuk penyimpanan jangka panjang, fitur pengarsipan surat keluar diimplementasikan, memastikan dokumen tersebut tersimpan dengan aman dan tetap mudah ditemukan melalui antarmuka arsip. Keberhasilan sprint ini esensial untuk melengkapi siklus manajemen dokumen, dari penerimaan hingga pengarsipan, serta mendukung kolaborasi antar pengguna dalam sistem.

#### e. Sprint 5

Sprint kelima ini difokuskan pada penguatan aspek keamanan dan akuntabilitas dalam sistem pengarsipan surat, melalui implementasi kontrol akses berbasis peran (RBAC) dan mekanisme pelacakan perubahan (*audit trail*). Tujuan utama adalah untuk memastikan integritas data, membatasi akses hanya kepada pihak yang berwenang, serta menyediakan transparansi penuh terhadap setiap modifikasi data.



Tabel 23 Sprint 5

No	Backlog	Deskripsi	Output	Status
1.	Role akses surat masuk	Sebagai pengguna, saya hanya ingin dapat mengakses fitur surat masuk sesuai hak akses <i>role</i> saya, agar data tidak diakses oleh pihak yang tidak berwenang.	Lampiran 46	Selesai
2.	Role akses surat keluar	Sebagai pengguna, saya hanya ingin dapat mengakses fitur surat keluar sesuai hak akses <i>role</i> saya, agar informasi tetap konsisten.	Lampiran 47	Selesai
3.	Pelacakan perubahan surat masuk	Sebagai pengguna, saya ingin dapat melihat riwayat perubahan yang terjadi pada proses surat masuk, untuk tujuan audit dan transparansi.	Lampiran 48	Selesai
4.	Pelacakan perubahan surat masuk	Sebagai pengguna, saya ingin dapat melihat riwayat perubahan yang terjadi pada proses surat keluar, untuk memantau alur dokumen.	Lampiran 49	Selesai

Sistem RBAC diterapkan pada modul surat masuk dan surat keluar untuk memastikan setiap pengguna hanya dapat mengakses fungsi dan data sesuai dengan peran atau hak akses mereka. Hal ini mencegah akses yang tidak sah serta menjaga konsistensi data. Implementasi mencakup pengecekan hak akses di sisi backend dan penyesuaian tampilan antarmuka pengguna agar sesuai dengan izin yang dimiliki oleh tiap *role*.

Selain itu, guna keperluan audit dan monitoring, dikembangkan fitur pelacakan perubahan (*audit trail*) pada surat masuk dan keluar. Sistem ini secara otomatis merekam setiap perubahan data, termasuk informasi tentang siapa yang melakukan perubahan, waktu pelaksanaan, serta detail perubahan yang terjadi. Riwayat perubahan tersebut dapat diakses melalui antarmuka khusus, memberikan transparansi dan akuntabilitas penting untuk pemantauan proses dokumen serta memastikan kepatuhan terhadap standar audit. Keberhasilan pengembangan fitur ini sangat penting agar sistem tidak hanya berfungsi dengan baik, tetapi juga aman, transparan, dan dapat dipertanggungjawabkan.

#### f. Sprint 6

Pada sprint ke enam, fokus pengembangan diarahkan pada peningkatan fitur pencarian dan pengelolaan arsip dalam sistem surat, dengan tujuan mempercepat akses informasi serta mendukung proses pelaporan. Tahapan ini diawali dengan penerapan mekanisme pengindeksan data untuk surat masuk dan keluar. Pengindeksan tersebut berperan penting dalam meningkatkan efisiensi pencarian,



sehingga pengguna dapat menemukan dokumen yang relevan secara cepat dan tepat, meskipun data yang dikelola cukup besar.

Tabel 24 Sprint 6

No	Backlog	Deskripsi	Output	Status
1.	Indexing surat masuk	Sebagai pengguna, saya ingin sistem dapat melakukan pengindeksan pada data surat masuk, agar pencarian di kemudian hari menjadi lebih cepat dan akurat.	Lampiran 50	Selesai
2.	Indexing surat keluar	Sebagai pengguna, saya ingin sistem dapat melakukan pengindeksan pada data surat keluar, agar pencarian surat keluar menjadi lebih cepat dan akurat.	Lampiran 51	Selesai
3.	Filter & cari surat masuk	Sebagai pengguna, saya ingin dapat mencari surat masuk berdasarkan kata kunci dan memfilter berdasarkan tanggal surat masuk. agar saya dapat menemukan surat yang relevan lebih mudah.	Lampiran 52	Selesai
4.	Filter & cari surat keluar	Sebagai pengguna, saya ingin dapat mencari surat keluar berdasarkan kata kunci dan memfilter berdasarkan tanggal surat, agar saya bisa menemukan surat keluar yang saya butuhkan.	Lampiran 53	Selesai
5.	Download arsip surat masuk	Sebagai pengguna, saya ingin dapat mengunduh file daftar arsip surat masuk, agar bisa digunakan untuk audit.	Lampiran 54	Selesai
6.	Download arsip surat keluar	Sebagai pengguna, saya ingin dapat mengunduh file daftar arsip surat keluar, agar saya bisa gunakan untuk audit.	Lampiran 55	Selesai

Pengembangan fitur pencarian dan filter lanjutan pada modul surat masuk dan keluar telah meningkatkan efektivitas dalam menemukan informasi. Sistem ini menggunakan teknologi Elasticsearch untuk melakukan pencarian teks penuh, sehingga memungkinkan pencarian berdasarkan kata kunci secara menyeluruh. Selain itu, penerapan filter khusus seperti rentang tanggal, tipe surat, dan status

membantu pengguna mempersempit hasil pencarian dengan lebih tepat, sehingga memudahkan dalam menemukan data yang relevan dengan cepat dan akurat.

Terakhir, untuk mendukung kebutuhan audit, analisis, dan penyimpanan data offline, fungsionalitas pengunduhan daftar arsip diimplementasikan untuk surat masuk dan surat keluar. Fitur ini memungkinkan pengguna untuk mengunduh ringkasan data arsip dalam format Excel, menyediakan rekaman historis yang mudah diakses dan memungkinkan analisis data di luar sistem. Keberhasilan sprint ini esensial dalam menjadikan sistem lebih efisien dan mendukung kebutuhan operasional serta analitis pengguna.

#### 4.4 Testing

Pengujian fungsional sistem dilakukan menggunakan metode *blackbox testing* untuk memastikan setiap fitur bekerja sesuai dengan kebutuhan pengguna. Pengujian difokuskan pada *input* dan *output* tanpa melihat struktur internal sistem. Tahapan ini bertujuan untuk memastikan sistem telah memenuhi fungsi sesuai rancangan awal.

Tabel 25 Pengujian Alur Surat Masuk

No	Nama Fitur	Test Case	Input	Expected Output	Status
1	Login	Login dengan kredensial valid	Email & password valid	Masuk ke dashboard sesuai peran pengguna	Berhasil
2	Login	Login dengan kredensial tidak valid	Email atau password salah	Tampil pesan “Email & password tidak valid”	Berhasil
3	Input surat masuk	Submit form dengan data lengkap	Semua field terisi dengan benar	Data tersimpan dan muncul pesan “Data surat masuk berhasil ditambahkan!”	Berhasil
4	Edit data surat masuk	Mengubah salah satu data surat masuk	Ubah data perihail	Data berhasil diperbarui, muncul pesan “data surat masuk berhasil diperbarui”	Berhasil
5	Input disposisi	Submit form disposisi	Mengisi semua field disposisi	Data disposisi disimpan dan muncul pesan “disposisi berhasil dibuat”	Berhasil
6	Read disposisi	Klik tombol “mark as read”	-	Muncul pesan “status disposisi diubah menjadi dibaca”	Berhasil

Tabel 26 Pengujian Alur Surat Masuk (*lanjutan*)

No	Nama Fitur	Test Case	Input	Expected Output	Status
7	Proses disposisi	Klik tombol “process”	-	Muncul pesan “status disposisi diubah menjadi diproses”	Berhasil
8	Selesaikan disposisi	Klik tombol “selesai”	Mengisi catatan tindak lanjut	Muncul pesan “status disposisi diubah menjadi selesai”	Berhasil
9	Selesaikan surat masuk	Klik tombol “tanda selesai”	-	Muncul pesan “surat masuk telah selesai diproses dan menunggu diarsipkan”	Berhasil
10	Arsipkan surat masuk	Klik tombol “arsipkan”	Mengisi nomor folder arsip	Muncul pesan “surat masuk berhasil diarsipkan”	Berhasil
11	Search surat masuk	Gunakan kolom pencarian surat masuk	Mengisi kata kunci “undangan”	Surat masuk terkait undangan ditampilkan	Berhasil
12	Unduh arsip surat masuk	Klik tombol “unduh surat masuk”	-	File excel berhasil diunduh	berhasil
13	Logout	Klik tombol “Sign Out”	-	Kembali ke halaman login	berhasil

Berdasarkan 13 skenario uji untuk proses surat masuk yang meliputi proses *login*, pengelolaan surat masuk, disposisi, pencarian, hingga *logout*, seluruh fitur berhasil berfungsi sesuai dengan harapan. Setiap input yang diberikan menghasilkan output yang tepat, seperti tampilan pesan keberhasilan, perubahan status, dan pengunduhan file, yang menunjukkan bahwa sistem telah bekerja sesuai kebutuhan pengguna dan siap untuk digunakan dalam lingkungan operasional.

Tabel 27 Pengujian Alur Surat Masuk

No	Nama fitur	Test case	Input	Expected output	Status
1	Login	Login dengan kredensial valid	Email & password valid	Masuk ke dashboard sesuai peran pengguna	Berhasil

Tabel 28 Pengujian Alur Surat Masuk (*lanjutan*)

No	Nama fitur	Test case	Input	Expected output	Status
2	Login	Login dengan kredensial tidak <i>valid</i>	Email atau password salah	Tampil pesan “Email & password tidak <i>valid</i> ”	Berhasil
3	Input surat keluar	Submit formulir surat keluar dengan data lengkap	Semua <i>field</i> terisi dengan benar	Data tersimpan, muncul pesan “data surat keluar berhasil disimpan”	Berhasil
4	Review KTU	Klik tombol “setujui”	-	Muncul alert konfirmasi persetujuan, pesan “surat keluar berhasil disetujui”	Berhasil
5	Review KTU	Klik tombol “revisi”	Mengisi catatan revisi surat keluar	Muncul pesan “revisi surat keluar berhasil dikirim”	Berhasil
6	Review kadep	Klik tombol “setujui”	-	Muncul <i>alert</i> konfirmasi persetujuan, pesan “surat keluar berhasil disetujui”	Berhasil
7	Review kadep	Klik tombol “revisi”	Mengisi catatan revisi surat keluar	Muncul pesan “revisi surat keluar berhasil dikirim”	Berhasil
8	Edit data surat keluar	Mengubah data surat keluar sesuai revisi	Ubah data tanggal surat keluar	Data berhasil diperbarui, muncul pesan “data surat keluar berhasil diperbarui”	Berhasil
9	Selesaikan revisi	Klik button “selesai revisi”	-	Muncul <i>alert</i> konfirmasi, pesan “revisi surat keluar telah diselesaikan”	Berhasil
10	Generate nomor surat keluar	Klik tombol “generate nomor”	-	Muncul pesan “nomor surat keluar berhasil dibuat”	Berhasil
11	Upload surat keluar final	Submit form unggah berkas	Mengisi <i>field</i> dengan file surat keluar	Muncul pesan “file final surat keluar berhasil ditambahkan”	Berhasil

@Hak cipta milik IPB University

IPB University

Hak Cipta Dilindungi Undang-undang  
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :  
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah  
b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.  
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

Tabel 29 Pengujian Alur Surat Masuk (*lanjutan*)

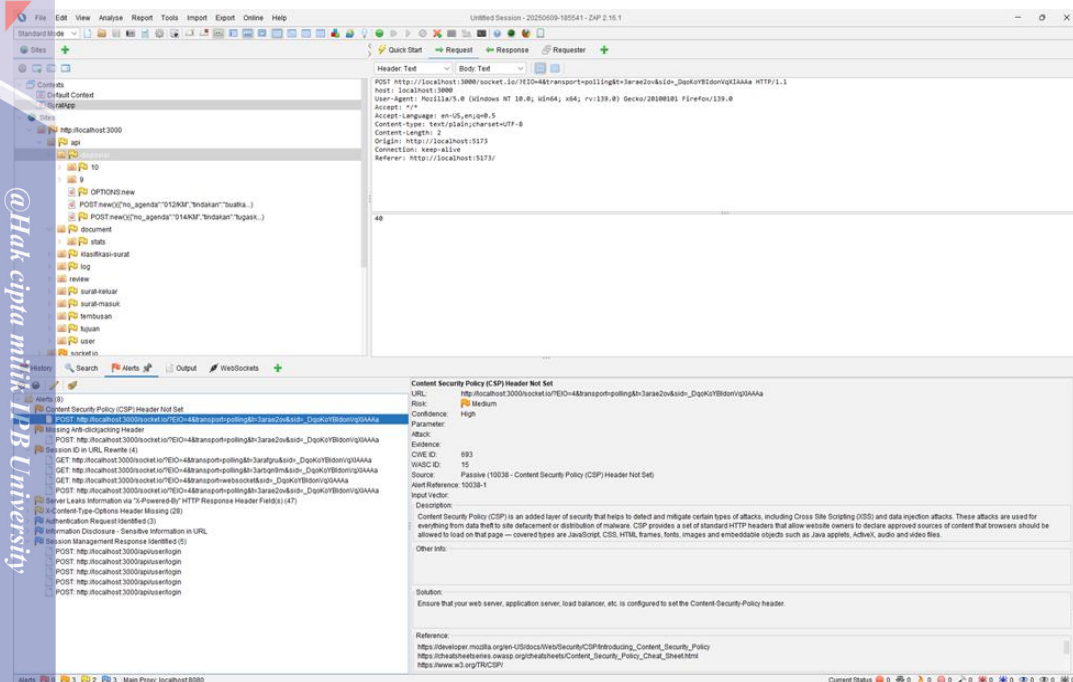
No	Nama fitur	Test case	Input	Expected output	Status
12	Arsipkan surat keluar	Klik tombol “arsipkan”	Mengisi nomor folder arsip	Muncul pesan “surat keluar berhasil diarsipkan”	Berhasil
13	Search surat keluar	Gunakan kolom pencarian surat keluar	Mengisi kata kunci “surat tugas”	Surat keluar terkait surat tugas ditampilkan	Berhasil
14	Unduh arsip surat keluar	Klik tombol “unduh arsip surat keluar”	-	File excel berhasil diunduh	Berhasil
15	Logout	Klik tombol “Sign Out”	-	Kembali ke halaman login	Berhasil

Berdasarkan hasil pengujian *blackbox testing* terhadap 15 skenario untuk proses surat keluar, seluruh fungsionalitas sistem menunjukkan berhasil berjalan sesuai dengan harapan. Mulai dari proses *login*, pengelolaan surat keluar, alur review oleh KTU dan kepala departemen, hingga fitur arsip dan pencarian, semua menghasilkan output yang sesuai dengan input yang diberikan. Validasi sistem menunjukkan bahwa sistem mampu merespons interaksi pengguna secara tepat, menampilkan pesan konfirmasi atau kesalahan yang informatif, dan memastikan data tersimpan maupun diproses dengan benar. Hal ini menunjukkan bahwa sistem telah memenuhi kebutuhan pengguna dan siap digunakan dalam operasional nyata.

#### 4.5 Deployment

Pada tahap *deployment*, meskipun sistem belum dilakukan proses unggah atau publikasi ke layanan web hosting secara resmi, peneliti telah melakukan simulasi deployment dengan menjalankan sistem pada lingkungan produksi lokal untuk memastikan sistem dapat berjalan secara optimal. Salah satu proses penting dalam tahap ini adalah penilaian kerentanan terhadap celah keamanan aplikasi, khususnya melalui *vulnerability assessment* menggunakan tool Zed Attack Proxy (ZAP) yang dikembangkan oleh OWASP.





Gambar 9 Hasil Scan Awal

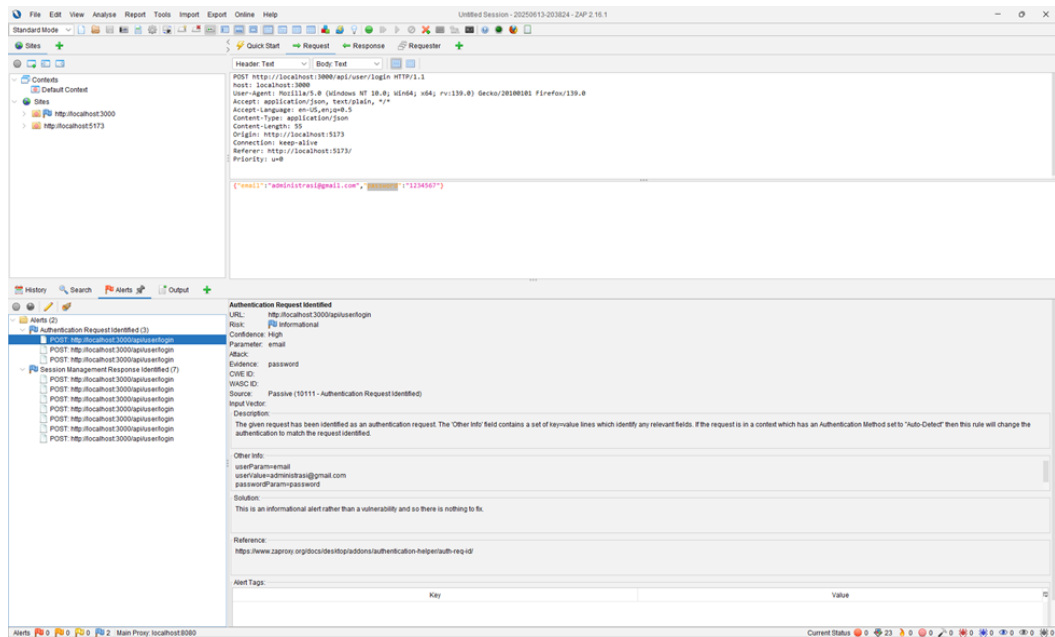
Hasil pemindaian awal yang dilakukan pada tanggal 9 Juni 2025 terhadap sistem menunjukkan adanya beberapa temuan kerentanan yang tergolong *low risk* namun tetap perlu ditindaklanjuti guna meningkatkan keamanan sistem. Beberapa alert yang terdeteksi antara lain:

- Tidak ditemukannya pengaturan *header Content-Security-Policy (CSP)* pada endpoint */socket.io/*, yang menyebabkan sistem berpotensi terbuka terhadap eksploitasi seperti *cross-site scripting (XSS)*.
- Tidak adanya header *X-Frame-Options* maupun direktif *frame-ancestors* dalam *Content-Security-Policy*, yang memungkinkan laman situs dapat di-*embed* dalam *iframe* sehingga berpotensi mengalami *clickjacking*.
- Ditemukannya *Session ID* di URL, terutama dalam komunikasi polling pada Socket.IO, yang dapat menyebabkan informasi tersebut terekam di *browser history* atau *log server*.
- Terbukanya informasi teknologi backend melalui *header X-Powered-By: Express*, yang dapat memberikan petunjuk kepada pihak tidak bertanggung jawab mengenai arsitektur sistem.
- Indikasi adanya informasi sensitif dalam URL, walaupun dalam konteks ini merupakan *session ID* socket yang tidak membahayakan secara langsung, tetapi tetap disarankan untuk ditangani lebih aman.

Seluruh temuan tersebut kemudian dianalisis dan ditindaklanjuti oleh peneliti dengan melakukan perbaikan konfigurasi pada sisi *backend*, termasuk penambahan header keamanan dan penyembunyian informasi yang tidak diperlukan pada *response server*. Setelah proses mitigasi selesai, sistem dipindai ulang menggunakan ZAP. Hasil pemindaian ulang yang dilakukan pada tanggal 13 Juni 2025 menunjukkan bahwa tidak ada lagi temuan kerentanan signifikan, dan hanya



terdapat *information-level alerts* yang bersifat informatif dan tidak berdampak langsung pada keamanan sistem.



Gambar 10 Hasil Scan setelah perbaikan

#### 4.6 Review

*Review* sistem dilakukan melalui sesi demonstrasi langsung kepada pengguna akhir yang terdiri dari administrasi, kepala tata usaha, dan ketua departemen. Pada sesi ini, pengguna diminta mencoba langsung setiap fitur sistem sambil mengamati alur kerja yang telah dirancang. Pengguna memberikan umpan balik secara lisan, mencakup aspek fungsionalitas, kemudahan penggunaan, tampilan antarmuka, serta efisiensi proses. Beberapa masukan yang diberikan meliputi penyesuaian tampilan alur proses dan penambahan proses input data. Hasil *review* menunjukkan bahwa secara keseluruhan sistem telah memenuhi kebutuhan pengguna, dan hanya terdapat catatan minor seperti penambahan *field* nama pegawai sebagai penerima tugas dari disposisi yang dapat di alokasikan pada *field* catatan disposisi dan diteruskan oleh kepala tata usaha kepada yang bersangkutan diluar sistem sehingga tidak berdampak signifikan terhadap fungsionalitas utama. Proses ini menunjukkan efektivitas pendekatan *agile*, yang memungkinkan tim pengembang untuk merespons perubahan kebutuhan dan masukan pengguna secara cepat dan iteratif.



## V KESIMPULAN DAN SARAN

### 5.1 Kesimpulan

Penelitian ini berhasil mengembangkan sebuah sistem informasi pengarsipan surat yang dibangun menggunakan metode agile dengan pendekatan iteratif dan fleksibel. Proses pengembangan sistem dilakukan melalui tahapan *requirement, design, development, testing*, dan *review*, dengan melibatkan pengguna secara aktif di setiap iterasi. Hasilnya, sistem mampu menangani seluruh proses surat masuk dan keluar, termasuk fitur *input* data, unggah dokumen, disposisi, pencarian, serta pengarsipan secara digital. Dari sisi fungsionalitas, pengujian menggunakan metode *blackbox testing* terhadap 13 skenario uji surat masuk dan 15 skenario uji surat keluar menunjukkan bahwa seluruh fitur berhasil memenuhi kebutuhan pengguna, serta mendukung pencarian surat secara cepat dan akurat. Proses *review* melalui demonstrasi sistem kepada pengguna akhir meliputi admin, kepala TU, dan ketua departemen menguatkan bahwa sistem telah memenuhi harapan pengguna. Selain itu, sistem telah melalui proses penilaian kerentanan dari celah keamanannya menggunakan metode *vulnerability assessment* (VA). Berdasarkan hasil pemindaian, seluruh celah dengan tingkat risiko sedang hingga tinggi telah diperbaiki, sehingga hanya menyisakan temuan informasional. Hal ini menunjukkan bahwa sistem tidak hanya siap digunakan secara fungsional, tetapi juga telah melalui verifikasi dari sisi keamanan untuk implementasi di lingkungan kerja.

### 5.2 Saran

Untuk pengembangan lebih lanjut, disarankan agar sistem segera dilakukan *deployment* ke server produksi agar dapat dimanfaatkan secara aktif oleh pengguna akhir dalam lingkungan nyata. Dengan demikian, performa sistem dapat diuji secara menyeluruh dan mendapatkan masukan langsung dari pengguna untuk perbaikan berkelanjutan. Selain itu, penambahan fitur notifikasi otomatis, penyesuaian log aktivitas secara *real-time*, serta integrasi dengan layanan email akan sangat mendukung efektivitas kerja, mempercepat proses tindak lanjut surat, dan meningkatkan kenyamanan serta produktivitas pengguna dalam pengelolaan arsip digital.

## DAFTAR PUSTAKA

- Atmaja AP, dan Yulianto SV. 2019. Pemanfaatan Elasticsearch untuk Temu Kembali Informasi Tugas Akhir. *Jurnal Nasional Teknologi dan Sistem Informasi*. 4(3):160–167.  
Doi:<https://doi.org/10.25077/TEKNOSI.v4i3.2018.160-167>.
- Darojat EZ, Sediyo E, dan Sembiring I. 2022. Vulnerability Assessment Website E-Government dengan NIST SP 800-115 dan OWASP Menggunakan Web Vulnerability Scanner. *Jurnal Sistem Informasi Bisnis*, 12(1), 36-44. <https://doi.org/10.21456/vol12iss1pp36-44>
- Fahreza M. 2024. Penerapan rest api menggunakan JSON web token (studi kasus: Aplikasi pembelajaran LearnPQI) [skripsi]. Jakarta. Fakultas Sains Teknologi UIN Syarif Hidayatullah Jakarta.
- Hakim F, Harahap NS, Handayani L, dan Afriyanti L. 2024. IMPLEMENTASI WEB SERVICE REST API UNTUK Pencarian Rujukan Islam Menggunakan Elasticsearch. *Jurnal Tekinkom (Teknik Informasi dan Komputer)*. 7(1):432-440.  
Doi:<https://doi.org/10.37600/tekinkom.v7i1.1314>.
- Hikmah N, Suradika A, dan Gunadi RAA. 2021. Metode Agile untuk Meningkatkan Kreativitas Guru Melalui Berbagi Pengetahuan (Knowledge Sharing). *Jurnal Instruksional (J. Ins)*. 3(1):030-039.  
Doi:<https://doi.org/10.24853/instruksional.3.1.30-39>.
- Lembayung ES, Subardin S, Findryani F, dan Safal S. 2024. PERANCANGAN SISTEM INFORMASI PENGARSIPAN SURAT (SIPS) BERBASIS WEB DI KANTOR KEPOLISIAN DAERAH SULAWESI TENGGARA. *Jurnal Kecerdasan Buatan dan Teknologi Informasi*. 3(1):17–25.  
Doi:<https://doi.org/10.69916/jkbt.v3i1.106>.
- Nilawati L, dan Widya SA. 2023. Penerapan Metode Scrum Pada Perancangan Sistem Informasi Manajemen Arsip Surat Berbasis Web. *Jurnal Teknologi Dan Sistem Informasi Bisnis*. 5(4):484-491.  
Doi:<https://doi.org/10.47233/jteksis.v5i4.1044>.
- Pranata AIB, Indriati I, dan Marji M. 2019. Klasifikasi Dokumen pada Laporan Kepolisian dengan Menggunakan Metode BM25 dan Improved K-Nearest Neighbor (IKNN). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*. 3(5):4434–4438. Doi:<https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/5226>.
- Ridwan NT, Sunoto A, dan Hendrawan. 2022. Perancangan Sistem Informasi Pengarsipan Surat Masuk dan Surat Keluar pada Badan SAR Nasional Jambi. *Jurnal Manajemen Teknologi Dan Sistem Informasi (JMS)*. 2(1):89–98.  
Doi:<https://doi.org/10.33998/jms.2022.2.1.42>.
- Roby M. 2023. Rancangan Aplikasi Deteksi Penyakit Diabetes Melitus Berbasis Mobile Menggunakan Metode Agile. *Jurnal SANTI - Sistem Informasi dan Teknik Informasi*, 2(3). Doi:<https://doi.org/10.58794/santi.v3i1.215>.
- Setiawan LOMI, Gunawan dan Tenriawaru A. 2024. RANCANG BANGUN SISTEM INFORMASI MONITORING BIMBINGAN TUGAS AKHIR BERBASIS WEB MENGGUNAKAN FRAMEWORK REACTJS. *AnoaTIK: Jurnal Teknologi Informasi dan Komputer*.

2(1):39–50. Doi:<https://doi.org/10.33772/anoatik.v2i1.25>.

Suhari S, Faqih A, dan Basysyar F. 2022. Sistem Informasi Kepegawaian Menggunakan Metode Agile Development di CV. Angkasa Raya. *Jurnal Teknologi dan Informasi*. 12(1):30-45. Doi:<https://doi.org/10.34010/jati.v12i1.6622>.

Wahyudin W, Kuswara H, Resti R, dan Dalis S. 2024. Metode Vulnerability Assesment Dalam Pengujian Kinerja Sistem Keamanan Website Points of Sales. *Computer Science (CO-SCIENCE)*, 4(1), 44-52. <https://doi.org/10.31294/coscience.v4i1.2978>

Wicaksono MA, Rudianto C, dan Tanaem PF. 2021. Rancang Bangun Sistem Informasi Arsip Surat Menggunakan Metode Prototype. *Jurnal Teknik Informatika dan Sistem Informasi*. 7(2):390–. Doi:<https://doi.org/10.28932/jutisi.v7i2.3664>

Widyoutomo F, Ajie H, dan Widodo. 2021. PENGEMBANGAN WEB SERVICE MODUL MAHASISWA PADA SISTEM INFORMASI AKADEMIK UNIVERSITAS NEGERI JAKARTA. *Jurnal Pendidikan Teknik Informatika dan Komputer (PINTER)*. 5(1). Doi: <https://doi.org/10.21009/pinter.5.1.9>.

Hak Cipta Dilindungi Undang-undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
  - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.