



Sistem Informasi Pengarsipan Dokumen Berbasis Web di UPBJJ UT-Surabaya

¹Darmawan, ²Moh Noor Al Azam
^{1,2}Universitas Narotama

Alamat Surat

Email: darmawansby7@gmail.com , noor.azam@narotama.ac.id

Article History:

Diajukan: 25 Oktober 2023; Direvisi: 15 Maret 2024; Accepted: 22 April 2024

ABSTRAK

Saat ini di dalam perusahaan ataupun organisasi agar dapat bertahan dalam kondisi persaingan yang ketat harus melakukan efisiensi dalam segala hal, termasuk dalam hal biaya untuk penyimpanan arsip. Tujuannya adalah untuk memberikan solusi dalam membuat arsip dokumen menjadi lebih efisien dalam penyimpanan, pencarian, dan penggunaan. Penulis melakukan riset di UPBJJ-UT Surabaya, dalam riset tersebut penulis menemukan bahwa sistem penyimpanan dokumen di UPBJJ-UT Surabaya, masih menggunakan manual. Terjadi beberapa kendala ketika pencarian dokumen membutuhkan waktu cukup lama, selain itu semakin bertambahnya dokumen yang dikeluarkan setiap hari membuat kebutuhan penyimpanan dokumen *hardcopy* semakin besar dan menyebabkan kebutuhan area penyimpanan kabinet semakin luas. Oleh karena itu penulis menawarkan solusi untuk beralih dari sistem penyimpanan dokumen manual menjadi pengarsipan dokumen berbasis web. Dengan menggunakan metode *Waterfall* dimana tahapanya meliputi analisis kebutuhan, desain sistem, pengkodean, pengujian sistem dan implementasi. Aplikasi ini hanya bisa di akses di lingkungan UPBJJ UT-Surabaya.

Kata kunci: Arsip, Dokumen, Sistem Informasi, Penyimpanan Dokumen, Aplikasi Pengarsipan

ABSTRACT

Currently in a company or organization in order to survive in conditions of intense competition must make efficiency in all respects, including in terms of costs for archival storage. The aim is to provide solutions in making document archives more efficient in storage, search, and use. The author conducted research at UPBJJ-UT Surabaya, in this research the author found that the document storage system at UPBJJ-UT Surabaya still uses manuals. There were several problems when searching for documents took quite a long time, besides that the increasing number of documents issued every day made the need for hardcopy document storage even greater and caused the need for a wider cabinet storage area. Therefore the author offers a solution to switch from a manual document storage system to a web-based document archiving. By using the Waterfall method where the stages include needs analysis, system design, coding, system testing and implementation. This application can only be accessed in the UT-Surabaya UPBJJ environment.

Keywords: *Archiving Applications, Archives, Documents, Document Storage, Information Systems*

1. PENDAHULUAN

Perkembangan zaman saat ini, teknologi komunikasi berkembang dengan pesat, banyak bermunculan berbagai macam alat telekomunikasi yang canggih, seperti halnya *handphone*, televisi, radio, telegram dan lain sebagainya. tetapi sampai saat ini masih ada komunikasi tertulis dan masih

tetap dipakai seolah tidak bisa tergantikan oleh berbagai peralatan komunikasi yang canggih, komunikasi tersebut yaitu surat. namun saat ini masih banyak ditemukan dalam instansi yang melakukan dalam proses pengarsipan surat ataupun data-data. seperti halnya data atau surat tercecer bahkan rusak.

Pengarsipan dokumen yang ada di di UPBJJ UT-Surabaya masih menggunakan metode konservatif dalam hal penyimpanan berkas arsip, Sistem yang berjalan dinilai kurang efektif dan efisien karena belum memiliki prosedur yang jelas. arsip di UPBJJ UT-Surabaya di klasifikasikan menjadi beberapa bagian sesuai isi surat. Oleh karena itu, maka diperlukan sebuah sistem penyimpanan arsip berbasis web untuk menyimpan arsip surat secara terkomputerisasi sehingga memudahkan dalam pencarian berkas arsip.

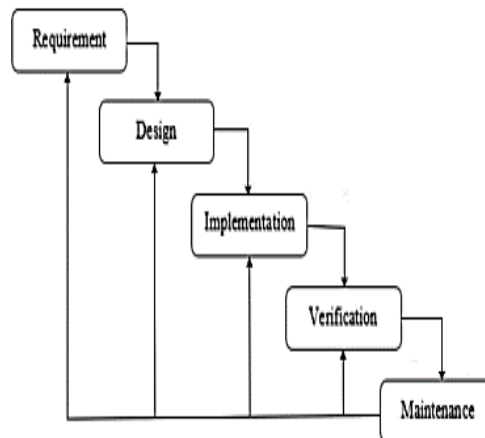
2. METODE PENELITIAN

Dalam penelitian ini menggunakan beberapa tahapan metode penelitian seperti:

1.1. Perencanaan Sistem

Metode yang digunakan pada pengembangan perangkat lunak yaitu waterfall. Menurut Rosa dan Shalahuddin (2013:28), metode *waterfall* adalah “metode yang menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisa, desain, pengkodean, pengujian dan pendukung (*support*)” (Suryadi, 2019).

Menurut Rosa dan Shalahuddin (2013:28), metode *waterfall* adalah “metode yang menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisa, desain, pengkodean, pengujian dan pendukung (*support*)”. Adapun tahapan dalam metode *waterfall* dapat dilihat pada gambar dibawah ini.



Gambar 1. Metode *Waterfall*

Sumber: (Pressman, 2012)

1.2. Requirement

Tahap ini pengembang sistem diperlukan komunikasi yang bertujuan untuk memahami perangkat lunak yang diharapkan oleh pengguna dan batasan perangkat lunak tersebut. Informasi dapat diperoleh melalui wawancara, diskusi atau survei langsung. Informasi dianalisis untuk mendapatkan data yang dibutuhkan oleh pengguna.

1.3. Desain

Pada tahap ini, pengembang membuat desain sistem yang dapat membantu menentukan perangkat keras (hardware) dan sistem persyaratan dan juga membantu dalam mendefinisikan arsitektur sistem secara keseluruhan.

a. *Flowchart* adalah penggambaran secara grafik dari langkah-langkah dan urutan prosedur dari suatu program.

b. *ERD (Entity Relationship Diagram)* merupakan salah satu model yang digunakan untuk mendesain *database* dengan tujuan menggambarkan data yang berelasi pada sebuah *database*.

1.4. Coding dan Pengetesan

Pengkodean adalah proses menerjemahkan model ke dalam bahasa yang dapat dikenali oleh komputer. Ini dilakukan oleh programmer yang menerjemahkan acara yang diminta oleh pengguna. Fase ini merupakan fase kerja sebenarnya dari sistem. Dalam artian penggunaan komputer dimaksimalkan pada tahap ini. Setelah pengkodean selesai dilakukan pengujian terhadap sistem yang telah dibangun sebelumnya.

1.5. Implementasi

Pada tahap ini, sistem pertama kali dikembangkan di program kecil yang disebut unit, yang terintegrasi dalam tahap selanjutnya. Setiap unit dikembangkan dan diuji untuk fungsionalitas yang disebut sebagai unit testing.

1.6. Verifikasi

Pada tahap ini, sistem dilakukan verifikasi dan pengujian apakah sistem sepenuhnya atau sebagian memenuhi persyaratan sistem, pengujian dapat dikategorikan ke dalam unit testing (dilakukan pada modul tertentu kode), sistem pengujian (untuk melihat bagaimana sistem bereaksi ketika semua modul yang terintegrasi) dan penerimaan pengujian (dilakukan dengan atau nama pelanggan untuk melihat apakah semua kebutuhan pelanggan puas).

1.7. Pemeliharaan

Ini adalah tahap akhir dari metode waterfall. Perangkat lunak yang sudah jadi dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya.

Terdapat beberapa pekerjaan rutin yang perlu dilakukan terhadap sistem informasi, antara lain:

- a. *Sistem maintenace*
- b. *Backup & recovery*
- c. *Data Archive*

1.8. Teknik Pengujian Sistem

Teknik pengujian yang digunakan adalah metode perangkat lunak, yaitu Metode *Black Box*. Dimana teknik pengujian ini memungkinkan perekayasa perangkat lunak menerima semua kondisi input yang sepenuhnya menggunakan persyaratan fungsional untuk suatu program. Pengujian *Black Box* memungkinkan pengembang web untuk membuat sesuatu satu set kondisi input yang melatih semua persyaratan fungsional dari program.

3. TINJAUAN PUSTAKA

Dasar Teori yang relevan digunakan dalam penelitian ini antara lain:

1. Pengertian Arsip

Arsip Merupakan dokumen yang digunakan untuk membuat keputusan yang tepat dalam berkomunikasi dengan orang lain berdasarkan memori yang telah ada. Dengan demikian, pengelolaan arsip harus mudah ditemukan, sederhana, dan aman (Ni'matussholihah, 2014).

2. Pengertian sistem informasi

Sistem informasi adalah kumpulan dari subsistem apapun baik fisik ataupun non-fisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan yaitu mengolah data menjadi informasi yang memiliki arti dan berguna (Riefnaldi, Aranta, & Muaidi, 2021).

3. Pengertian website

Menjelaskan bahwa "Website adalah halaman informasi yang disediakan melalui jalur internet sehingga bisa diakses diseluruh dunia, selama terkoneksi dengan jaringan internet (Suryadi & Zulaikhah, 2019).

4. Pengertian *Unified Modelling Language* (UML)

Unified Modelling Language adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek (Nirsal, Rusmala, & Syafriadi, 2020).

4. HASIL DAN PEMBAHASAN

Proses pengarsipan dokumen yang saat sedang berjalan yaitu staf arsip di beri tugas membuat surat sesuai arahan dari atasan, kemudian petugas menuliskan nomor surat di buku khusus untuk menerbitkan nomor surat sesuai urutan, kemudian petugas membuat konsep surat menyerahkan ke manager sesuai dengan isi bidang surat untuk di lakukan verifikasi dan memberi paraf konsep surat. selanjutnya konsep surat tersebut diserahkan kepada Pimpinan UPBJJ UT-Surabaya untuk di tanda tangan.

4.1 *Workshop Design*

Dalam fase ini menjelaskan rancangan proses pengarsipan dokumen pada UPBJJ UT-Surabaya.

4.2 *Unified Modelling Language* (UML)

Adapun langkah-langkah dalam membuat *Usecase diagram* adalah sebagai berikut:

a. Identifikasi Aktor

Tabel 1. Identifikasi Aktor

No	Aktor	Deskripsi
1	Staf Arsip	User yang mengelola (<i>input</i> dan <i>update</i>) dokumen
2	admin	User yang melakukan pengecekan pada hasil <i>input</i> dari staf arsip
3	Klien	User yang dapat mencetak, melihat dokumen sesuai kebutuhan

b. Identifikasi *Usecase*

Tabel 2. Identifikasi *Usecase*

No	<i>Usecase</i>	Deskripsi
1	<i>Login</i>	User masuk ke dalama sistem dengan cara memasukkan Nama atau NIP serta <i>Password</i> agar dapat mengakses sistem
2	Kelola dokumen	Dokumen yang baru diterima diinputkan ke dalam sistem oleh staff arsip.
3	Mengecek dokumen	Dokumen yang telah diinput akan dicek, memastikan tidak ada kesalahan dalam proses <i>input</i> .
4	Mengarsipkan dokumen	Dokumen yang di tanda tangan / divalidasi akan di arsipkan.
5	Mencetak laporan dokumen	Data dokumen yang telah diinput akan disatukan menjadi laporan dokumen dan dicetak dengan melakukan klik cetak
6	Melihat data dokumen	Melihat kumpulan data-data dokumen yang telah diinput.
7	<i>Log Out</i>	User keluar dari sistem

4.3 *Usecase Diagram*

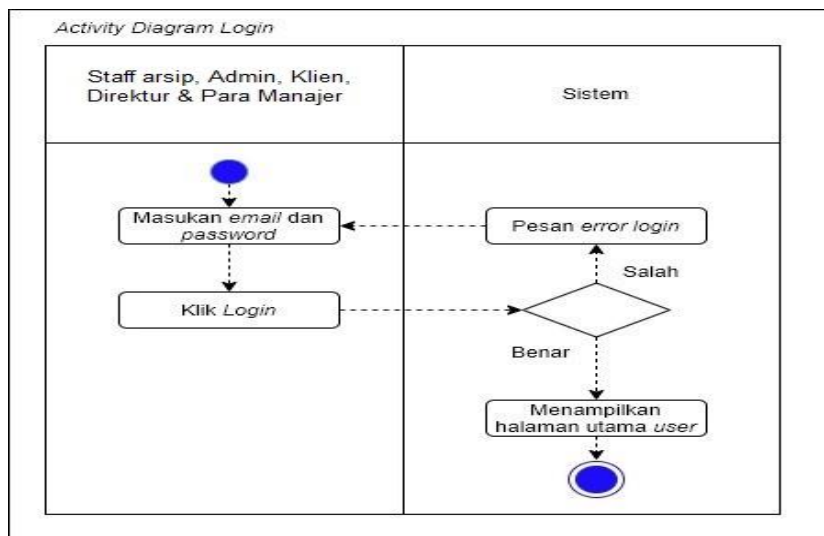


Gambar 1. Usecase Diagram

4.4 *Activity Diagram*

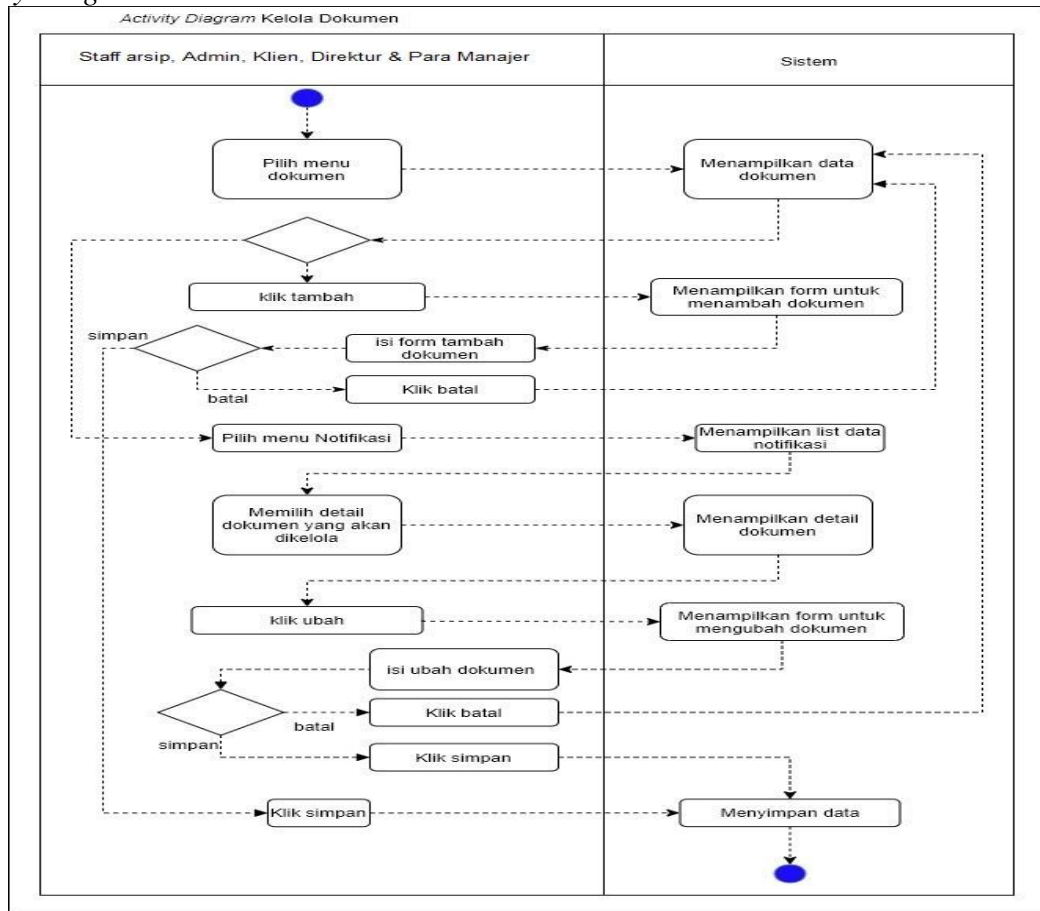
Activity diagram sistem informasi menggambarkan aktifitas-aktifitas yang terjadi dalam sistem dan *user*. Berikut ini adalah *activity diagram* yang terdapat pada sistem informasi UPBJJ UT-Surabaya:

a. *Activity Diagram Login*



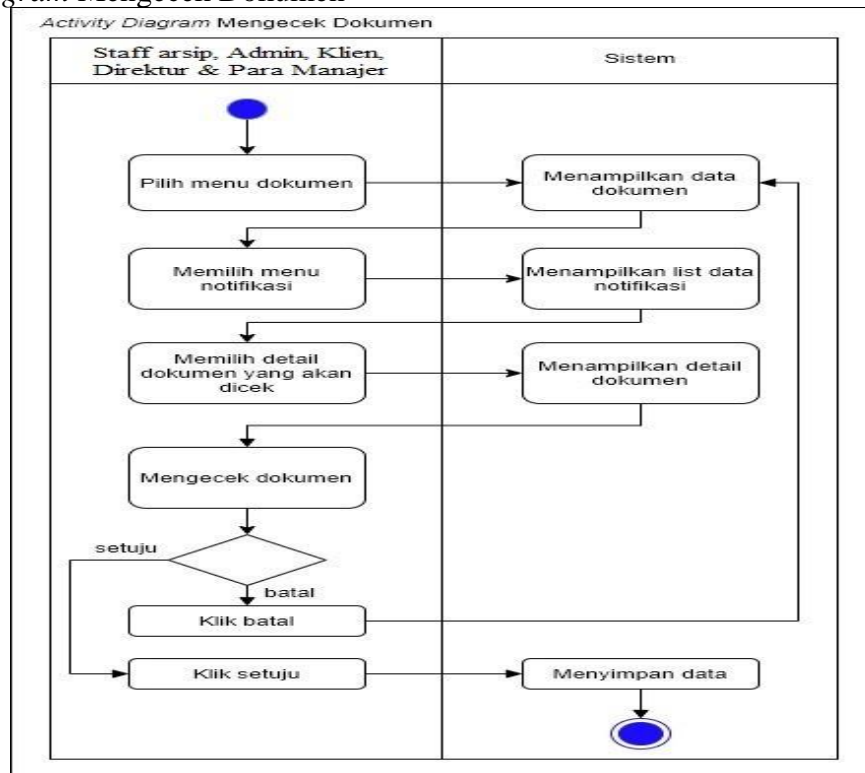
Gambar 2. Activity Diagram Login

b. Activity Diagram Kelola Dokumen



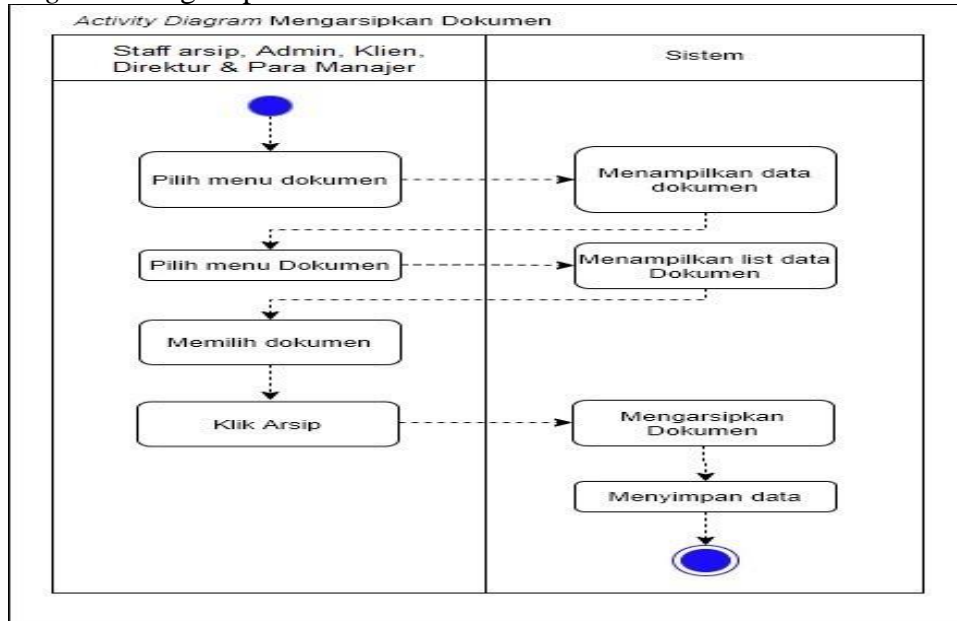
Gambar 3. Activity Diagram Kelola Dokumen

c. Activity Diagram Mengecek Dokumen



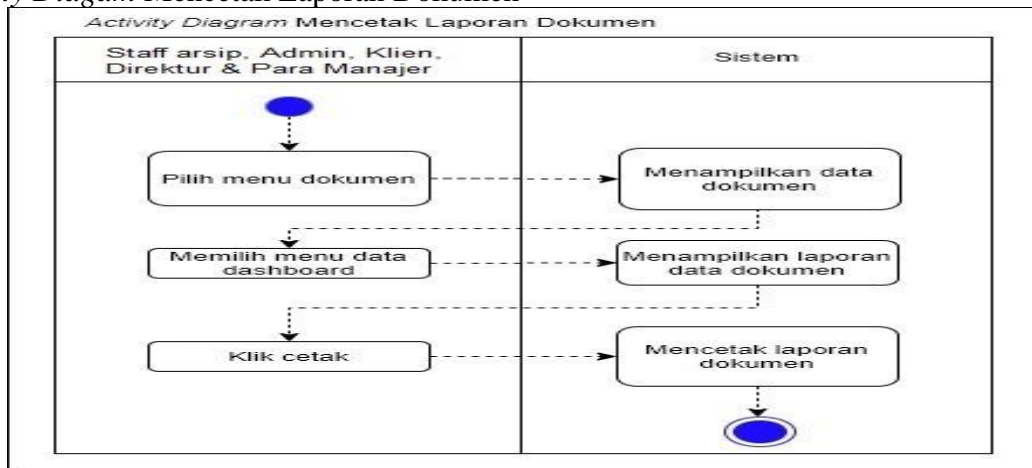
Gambar 4. Activity Diagram Mengecek Dokumen

d. *Activity Diagram* Mengarsipkan Dokumen



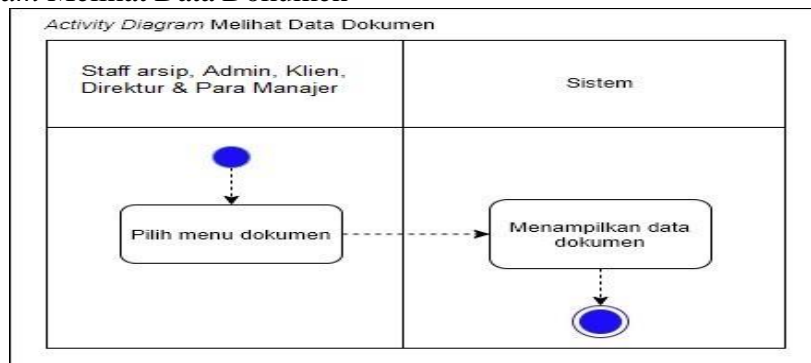
Gambar 5. Activity Diagram Mengarsipkan Dokumen

e. *Activity Diagram* Mencetak Laporan Dokumen



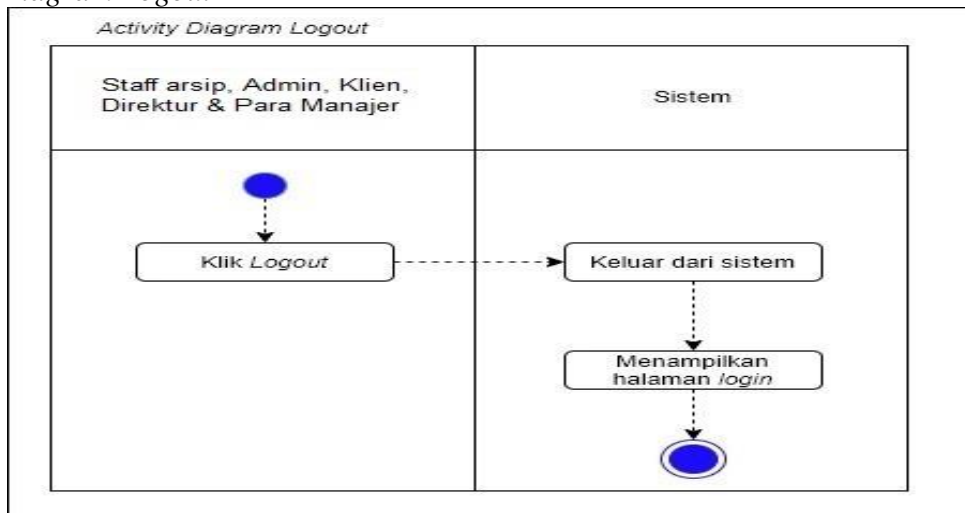
Gambar 6. Activity Diagram Mencetak Laporan Dokumen

f. *Activity Diagram* Melihat Data Dokumen



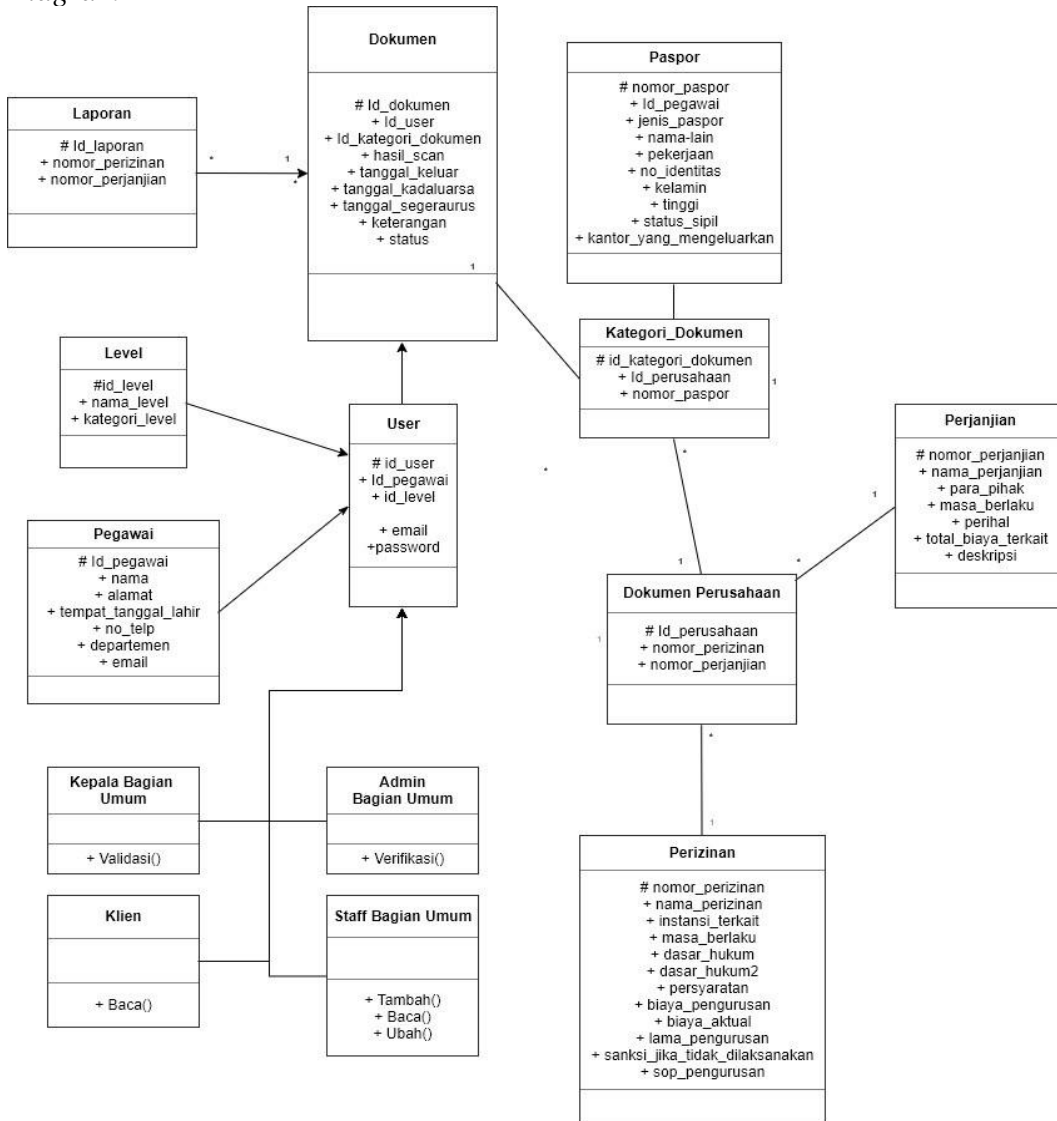
Gambar 7. Activity Diagram Melihat Data Dokumen

g. Activity Diagram Logout



Gambar 8. Activity Diagram Logout

4.5 Class Diagram



Gambar 9. Class Diagram

4.6 Perancangan *Database*

a. Pegawai

Tabel 3. Pegawai

Nama Field	TipeData	Ukuran	Keterangan
id_pegawai	<i>Int</i>	10	NIP
nama	<i>String</i>	25	Nama
departemen	<i>Varchar</i>	10	Departemen
jabatan	<i>Varchar</i>	10	Jabatan

b. User

Tabel 4. User

Nama Field	Tipe Data	Ukuran	Keterangan
id_user	<i>Int</i>	5	Kode User
NIP	<i>String</i>	25	NIP
Password	<i>String</i>	15	Password

c. Dokumen

Tabel 5. Dokumen

Nama Field	Tipe Data	Ukuran	Keterangan
id_dokumen	<i>Int</i>	10	id Dokumen
hasil_scan	<i>Text</i>	-	Hasil Scan
tanggal_keluar	<i>Date</i>	-	Tanggal Keluar
tanggal_kadaluarsa	<i>Date</i>	-	Tanggal Kadaluarsa
Keterangan	<i>text</i>	-	Keterangan

4.7 Perancangan Antar Muka

a. Rancangan Menu *Login*

Gambar 10. Rancangan Menu *Login*

b. Rancangan Menu *Home*

Gambar 11. Rancangan Menu *Home*

c. Rancangan *Form* Tambah Dokumen

Gambar 12. Rancangan *Form* Tambah Dokumen

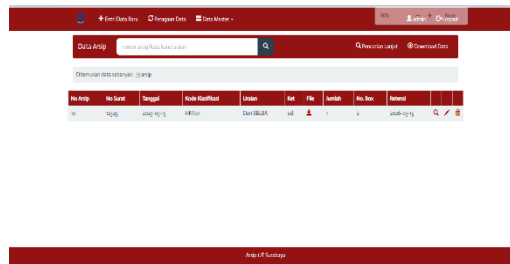
4.8 Tampilan Antar Muka

a. Menu *Login*



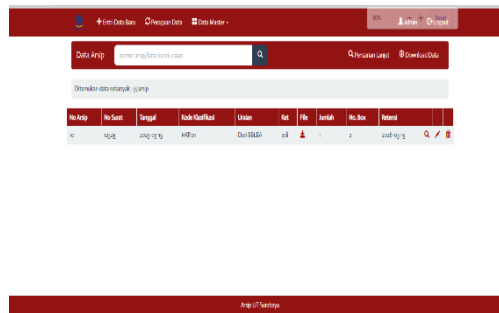
Gambar 13. Menu *Login*

b. Menu *Home*



Gambar 14. Menu *Home*

c. Menu Tambah Dokumen



Gambar 15. Menu Tambah Dokumen

5. SIMPULAN DAN SARAN

Kesimpulan yang didapat adalah sebagai berikut:

1. Dengan adanya fitur pencarian dokumen pada aplikasi ini, maka aktivitas pencarian dokumen menjadi lebih mudah, hal ini dibuktikan dengan hasil wawancara dengan *user*, yang sebelumnya memerlukan waktu sekitar 10 menit untuk mencari sebuah dokumen, menjadi sekitar 1 menit dengan menggunakan aplikasi ini.

2. Dengan menggunakan *database SQL server*, maka penyimpanan dokumen dapat diintegrasikan ke dalam *server* perusahaan sehingga tidak perlu khawatir apabila nantinya dibutuhkan untuk menyimpan banyak dokumen.
3. Aplikasi yang telah dirancang telah mengelompokkan dokumen, dengan begitu dokumen dapat diidentifikasi sesuai dengan kelompoknya

Adapun saran yang dapat diberikan pada penelitian ini adalah:

1. Diharapkan pada penelitian selanjutnya mengkhususkan untuk membangun *security* atau sistem keamanan untuk sistem pengarsipan ini guna mengamankan data lembaga.
2. Menambah fitur *chatting* antar pegawai agar para pegawai bisa berinteraksi lewat aplikasi tanpa harus melalui aplikasi lain agar tidak terjadi *miss communication* dalam mengelola dokumen.

6. DAFTAR PUSTAKA

- Ni'matussholiha, A. (2014). *Google drive for storing archives*. 11-22.
- Nirsal, Rusmala, & Syafriadi. (2020). Desain Dan Implementasi Sistem Pembelajaran Berbasis *E-Learning* Pada Sekolah Menengah Pertama Negeri 1 Pakue Tengah. *Journal Ilmiah d'Computare*, 10, 30-37.
- Riefnaldi, A. R., Aranta, A., & Muaidi, M. (2021). Pembuatan Sistem Informasi Pengarsipan Surat Pada Kantor Desa Sandik Berbasis Website. *Jurnal Begawe Teknologi Informasi (JBegaTI)*, 2(2), 191–202.
- Suryadi, A., & Zulaikhah, Y. S. (2019). Rancang Bangun Sistem Pengelolaan Arsip Surat Berbasis Web Menggunakan Metode *Waterfall*. *Jurnal Khatulistiwa Informatika*, 7(1), 13–21.