

BAB I PENDAHULUAN

1.1 Latar Belakang

Penggunaan Teknologi Informasi memegang peranan penting dalam pengelolaan dokumen di institusi pemerintahan. Untuk mengelola arsip secara efektif, baik arsip digital maupun manual merupakan dua metode yang sering digunakan dalam melakukan penyimpanan dokumen. Namun, saat ini banyak sistem pengarsipan yang masih menggunakan metode manual, yaitu dengan menyimpan dokumen dalam bentuk fisik di lemari atau rak sehingga proses pengarsipan menimbulkan kendala, seperti jumlah dokumen yang terus bertambah sementara ruangan penyimpanan terbatas, proses pencarian dokumen menjadi lebih sulit, kerusakan pada dokumen yang disebabkan oleh suhu dan kelembaban di dalam ruang penyimpanan. Akibatnya, diperlukan ruang penyimpanan yang lebih luas dengan suhu ruangan yang stabil untuk menjaga keutuhan dokumen (Yaqin et al., 2023).

Arsip digital adalah data yang disimpan secara sistematis, dan dapat ditemukan kembali dengan cepat. Data-data yang dapat diolah dalam bentuk digital dapat berupa gambar dan dokumen yang dapat dijadikan sebagai sebuah data dalam bentuk biner, sehingga dapat diolah dalam program komputasi dan disimpan dalam penyimpanan data digital. Kelebihan arsip digital adalah pencarian informasi yang lebih cepat dan salinan arsip elektronik yang mudah diakses. Ini karena arsip diproses menggunakan teknologi yang disimpan di media penyimpanan elektronik dan didaftarkan menggunakan sistem informasi. Saat ini banyak layanan penyimpanan online, juga dikenal sebagai *cloud*, tetapi banyak yang memiliki kelemahan terutama pada pencarian data, karena pencarian data biasanya hanya menemukan nama file (Terkini, 2021).

Jaringan Syaraf Tiruan (*Artificial Neural Network*) memiliki kemampuan dalam mengenali pola dan mengklasifikasikan data secara otomatis. Dalam konteks pengarsipan dokumen digital, teknologi ini dapat diterapkan untuk mengidentifikasi dan mengelompokkan berbagai jenis surat berdasarkan isi atau formatnya. Dengan integrasi *Optical Character Recognition* (OCR), sistem dapat membaca teks dari dokumen fisik dan mengubahnya menjadi data digital yang dapat diproses lebih lanjut, sehingga mempercepat pencarian dan pengelolaan arsip secara efisien. Jaringan Syaraf Tiruan adalah pemodelan data yang kuat yang dapat menangkap dan mewakili hubungan input-output yang kompleks, memiliki ketahanan terhadap input data, kecepatan eksekusi, dan kemampuan untuk menginisialisasikan sistem yang lebih kompleks (Sudarsono, 2016). Jaringan Syaraf Tiruan menggunakan teknologi yang diprogram komputer untuk meniru sistem syaraf manusia. Salah satu fungsi otak manusia adalah mengenali objek. Dengan menggunakan Jaringan Syaraf Tiruan, komputer akan dilatih untuk mengenali pola sampai akhirnya dapat mengenali objek (Khairina & Khoiruddin Harahap, n.d.).

Seiring kemajuan teknologi, metode pengenalan citra digital yang lebih sederhana dan efektif diperlukan. Untuk menyelesaikan masalah ini, penulis menggunakan teknologi Jaringan Syaraf Tiruan (*Artificial Neural Network*), yaitu *Optical Character Recognition* (OCR). Sistem OCR dapat membaca karakter yang berasal dari tulisan tangan atau dari mesin cetak (*printer*) yang memungkinkan pengguna memasukkan data ke dalam sistem dengan lebih mudah (Muhtadii & Tawakal, 2016). Sistem OCR telah berkembang menjadi banyak jenis seperti data entry untuk dokumen bisnis, pengenalan plat nomor otomatis, dan konversi tulisan tangan. Manusia bisa dengan mudah untuk mengerti dan membaca isi konten berupa text pada gambar atau dokumen hanya dengan melihat. Sedangkan, komputer tidak bekerja dengan cara yang sama. Terkadang sangat sulit untuk ekstraksi teks dari sebuah gambar atau dokumen hanya karena berbeda ukuran, jenis, orientasi, kompleksitas gambar latar dari sebuah gambar atau dokumen, dan lain-lain (Susanty & Nugroho, 2020).

Penulis menggunakan library *Tesseract*, merupakan alat OCR berbasis *Open-Source* yang menggunakan bahasa pemrograman Python, dalam penelitian ini. *Tesseract* adalah platform yang fleksibel dengan fitur analisis pengenalan teks yang sedang dikembangkan oleh Google. Tingkat akurasi ketepatan tinggi adalah parameter yang akan dianalisis yang menunjukkan keberhasilan penelitian ini dengan menggunakan metode *Tesseract* pada dokumen digital.

Beberapa penelitian terkait dengan perancangan sistem untuk aplikasi arsip digital dilakukan oleh (Amalia & Huda, 2020) dengan judul "Sistem Informasi Pengarsipan Surat Masuk dan Surat Keluar Pada Dinas Tenaga Kerja dan Transmigrasi Kabupaten Musi" dan penelitian yang dilakukan oleh (Rozana & Musfikar, 2020) dengan judul "Analisis dan Perancangan Sistem Informasi Pengarsipan Surat Berbasis Web Pada Kantor Lurah Desa Dayah Tuha". Kedua penelitian ini bertujuan untuk membantu kinerja pegawai dalam proses pengarsipan dan arsip yang dimiliki dapat tersimpan dengan baik dan aman. Dalam hal implementasi sistem OCR ke dalam aplikasi website, terdapat beberapa penelitian seperti yang dilakukan oleh (Rizal Toha & Triayudi, 2022) dengan judul "Penerapan Membaca Tulisan di dalam Gambar Menggunakan Metode OCR Berbasis Website pada e-KTP". Hasil dari penelitian ini adalah pihak yang memerlukan dan menggunakan e-KTP tidak perlu *input* data secara manual dan hanya tinggal *upload* foto e-KTP ke dalam website.

Penelitian mengenai konversi file berformat image menjadi file teks yang dilakukan oleh (Napitupulu et al., 2023) dengan tujuan memudahkan pemilik dokumen gambar saat proses pengetikan ulang dokumen atau menyalin teks dari dokumen fisik ke bentuk digital. Melalui penelitian ini, aplikasi yang dirancang dapat mengkonversi file teks dari gambar menjadi teks digital yang dapat di akses dan disunting di komputer atau perangkat digital lainnya. Penelitian yang dilakukan oleh (Nurhaliza et al., 2022) dengan tujuan untuk melakukan pengkajian kinerja OCR dalam mengenali karakter dokumen dengan menggunakan jenis huruf Arial, Bahnschrift, Condensed, Georgia, Lucida Sans Unicode, Roman, Segoe UI Semibold, dan Times New Roman. Berdasarkan hasil penelitian ini, metode OCR mampu mengenali dengan akurasi 100% untuk karakter jenis huruf Georgia, Lucida Sans Unicode, Roman dan Segoe UI Semibold dengan akurasi Terendah 71.21% pada karakter jenis huruf Bahnschrift Condensed.

Keberadaan Teknologi Pengenalan Karakter memungkinkan munculnya peluang dalam menghadapi permasalahan untuk melakukan pengarsipan secara digital di institusi pemerintahan. Dengan melakukan digitalisasi, pegawai yang memiliki peran dalam hal melakukan pengarsipan dapat dengan mudah untuk mengarsipkan dokumen-dokumen fisik menjadi bentuk digital yang dapat mempercepat waktu pencarian apabila ada dokumen atau surat yang diperlukan dan mengurangi penggunaan ruangan yang sebelumnya digunakan untuk menyimpan dokumen-dokumen fisik. Hal ini mendorong penulis untuk melakukan penelitian dalam bentuk tulisan skripsi dengan judul "Rancang Bangun Aplikasi Web Cerdas Untuk Arsip Digital Surat Menggunakan Jaringan Syaraf Tiruan Multi-Layer Perceptron".

1.2 Rumusan Masalah

1. Bagaimana membangun dan mengintegrasikan model klasifikasi dokumen surat berbasis algoritma Naïve Bayes dan Artificial Neural Network Multi-Layer Perceptron (ANN-MLP) secara efektif?
2. Bagaimana merancang dan membangun aplikasi web cerdas untuk pengelolaan arsip digital surat yang terintegrasi dengan sistem klasifikasi berbasis OCR, Naïve Bayes, dan ANN-MLP?
3. Bagaimana hasil pengujian sistem aplikasi arsip digital surat menggunakan metode User Acceptance Test (UAT) dalam mengevaluasi tingkat kepuasan dan kemudahan penggunaan sistem bagi pengguna?

1.3 Batasan Masalah

1. Fokus pada penerapan Jaringan Syaraf Tiruan *Multi-Layer Perceptron* (ANN-MLP) dan algoritma *Naïve Bayes* sebagai metode utama untuk klasifikasi data surat.
2. Aplikasi web cerdas yang dibangun hanya mencakup fitur dasar seperti pengarsipan, pencarian, klasifikasi, dan laporan dokumen surat.
3. Pendekatan pengembangan sistem menggunakan metode SDLC dengan *model Waterfall*.
4. Pengujian sistem akan menggunakan metode *black-box testing*.
5. Evaluasi kinerja model ANN-MLP dan OCR dilakukan menggunakan metrik akurasi, presisi, recall, dan F1-score.

1.4 Tujuan Penelitian

1. Untuk membangun dan mengintegrasikan model klasifikasi dokumen surat menggunakan algoritma Naïve Bayes dan Artificial Neural Network Multi-Layer Perceptron (ANN-MLP) secara efektif, serta mengevaluasi performanya dalam proses klasifikasi jenis surat.
2. Untuk merancang dan mengembangkan sebuah aplikasi web cerdas yang dapat mengelola arsip digital surat secara otomatis, dengan integrasi teknologi OCR (Optical Character Recognition) serta model klasifikasi berbasis Naïve Bayes dan ANN-MLP.
3. Untuk mengukur tingkat penerimaan dan kepuasan pengguna terhadap aplikasi yang telah dikembangkan melalui pengujian User Acceptance Test (UAT), sebagai evaluasi terhadap kemudahan penggunaan, efektivitas, dan kelayakan sistem.

1.5 Manfaat Penelitian

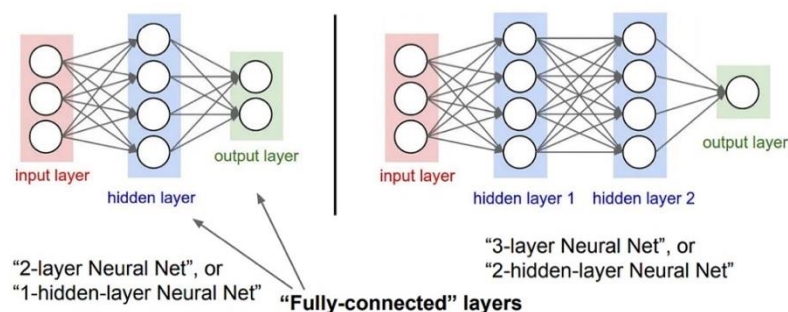
1. Hasil penelitian ini dapat dijadikan referensi bagi penelitian selanjutnya yang berkaitan dengan pengembangan aplikasi cerdas untuk pengelolaan arsip digital surat.
2. Menghasilkan sebuah aplikasi web cerdas yang dapat membantu organisasi dalam mengelola arsip citra digital surat secara lebih efektif dan efisien.
3. Menghasilkan model ANN-MLP yang dapat digunakan sebagai dasar untuk pengembangan sistem klasifikasi dokumen lainnya.

1.6 Landasan Teori

1.6.1 Jaringan Syaraf Tiruan

Paradigma pengolahan informasi Jaringan Syaraf Tiruan (ANN) terinspirasi dari sistem kerja syaraf biologis, seperti kinerja otak yang memproses data. (Mubarak et al., 2020). Jaringan informasi yang mengandung sejumlah komponen pemrosesan yang terhubung disebut *neuron* adalah elemen kunci. *Neuron* bekerja bersama-sama untuk menyelesaikan suatu pekerjaan khusus. Jaringan saraf tiruan belajar dengan mencontohi cara kerja manusia. Komponen pemrosesan yang berisi *neuron-neuron* yang saling terhubung disebut dengan Bobot (Niko Surya Atmaja, 2021).

Multi-Layer Perceptron (MLP) merupakan teknik umum untuk pengenalan pola yang terdiri dari kumpulan *neuron* yang disusun menjadi suatu lapisan: lapisan input, lapisan tersembunyi (*Hidden Layer*), dan lapisan output. *Feed Forward Network* adalah kategori utama dalam MLP. Kinerja dari teknik ini sangat tergantung pada pengaturan setiap tingkatannya, sehingga pembangunan MLP memerlukan perhatian yang ekstra. Elemen MLP meliputi algoritma pembelajaran, struktur jaringan, dan aktivasi fungsi (Danil, 2024).



Gambar 1. Arsitektur Neural Network

Gambar berikut menunjukkan fungsi matematis perceptron. Keluaran ($\hat{y}$) dari perceptron merupakan bias (w_0) dari perceptron ditambah dengan jumlah setiap input (x_i) dikali dengan bobot masing-masing (w_i). Setelah itu dimasukkan ke dalam fungsi aktivasi (g)

$$\hat{y} = g \left(w_0 + \sum_{i=1}^m x_i w_i \right)$$

Keterangan :

$\hat{y}$: Output

g : Fungsi Aktivasi

w_0 : Bias

w_i : Bobot

x_i : Input

Fungsi aktivasi pada perceptron bertugas untuk membuat jaringan saraf mampu menyesuaikan pola pada data non linier

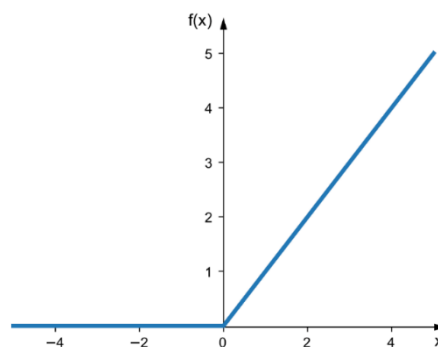
1.6.2 Fungsi Aktivasi

Fungsi Aktivasi merupakan sebuah fungsi non-linear yang melatih neural network dengan menentukan output neuron dari set input tertentu. Di neural network, fungsi aktivasi bekerja dengan memberikan input ke network dari lapisan input. Di lapisan input berikutnya, jumlah bobot input dihitung dan biasanya ditambahkan ke jumlah tersebut. Selanjutnya, jumlah ini melalui fungsi aktivasi(Suahati et al., 2022).

1. *ReLU Activation*

Fungsi aktivasi yang paling umum untuk *deep convolutional neural network* (CNN) adalah Rectified Linear Unit (*ReLU*). Kurva fungsi ini akan membuat nilai output sama dengan nilai input jika inputnya kurang atau sama dengan 0. Jika inputnya sama dengan 0, fungsi ini akan memaksa outputnya menjadi 0.(Li & Yuan, 2017).

$$f(x) = \max(0, x)$$

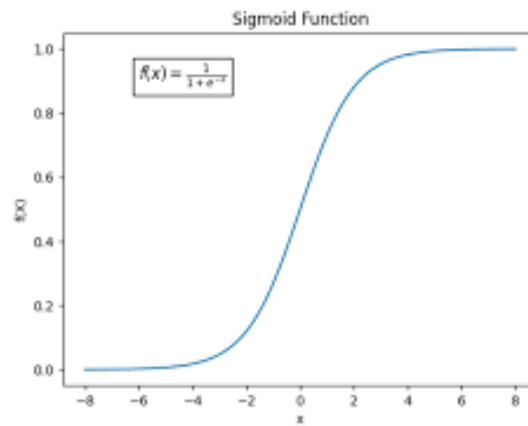


Gambar 2. *ReLU Activation*

2. *Sigmoid Activation*

Dalam OCR, fungsi Sigmoid banyak digunakan karena dapat memetakan bilangan menjadi nilai mulai dari 0 hingga 1. Ini cocok untuk tugas klasifikasi biner.

$$f(x) = \frac{1}{1 + e^{-x}}$$

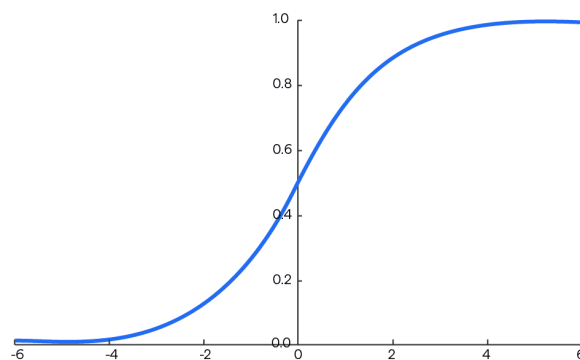


3. Softmax Activation

Fungsi Softmax membuat output menjadi distribusi probabilitas di seluruh kelas dengan mengubah skornya menjadi probabilitas yang berjumlah satu untuk setiap kelas.

$$S(z) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$$

Softmax Function



Gambar 4. Softmax Activation

1.6.3 Naïve Bayes

Naïve Bayes adalah salah satu metode klasifikasi yang paling sederhana dan umum digunakan. Metode ini menghitung probabilitas kelas berdasarkan distribusi kata-kata yang ada dalam dokumen. *Naïve Bayes* memiliki

Gambar 3. Sigmoid Activation

beberapa keunggulan seperti sederhana, cepat, dan akurasi yang tinggi (Renyut et al., 2022) . menurut (Riyanto, 2019) *Naïve Bayes* merupakan metode prediksi sederhana yang memanfaatkan probabilitas, dengan menggunakan teorema Bayes dan mengasumsikan independensi kuat antar fitur. Fitur pada data dianggap tidak saling terkait satu sama lain.

Menurut (Indrayuni, 2019) *Naïve Bayes* memanfaatkan teori probabilitas dalam matematika untuk menemukan peluang tertinggi dari berbagai kemungkinan klasifikasi dengan menganalisis frekuensi masing-masing klasifikasi dalam data mining. Metode klasifikasi *Naïve Bayes* adalah teknik statistik yang digunakan untuk memperkirakan probabilitas keanggotaan kelas tertentu.

Berdasarkan kutipan diatas, peneliti menyimpulkan bahwa *Naïve Bayes* adalah metode klasifikasi yang paling sederhana dan umum digunakan. Metode ini menghitung probabilitas kelas berdasarkan distribusi kata-kata dalam dokumen. *Naïve Bayes* memiliki beberapa keunggulan, seperti sederhana, cepat, dan akurasi yang tinggi. Metode ini memanfaatkan teorema Bayes dan mengasumsikan independensi kuat antar fitur, sehingga fitur pada data dianggap tidak saling terkait satu sama lain. *Naïve Bayes* juga memanfaatkan teori probabilitas dalam matematika untuk menemukan peluang tertinggi dari berbagai kemungkinan klasifikasi dengan menganalisis frekuensi masing-masing klasifikasi dalam data mining.

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

Keterangan:

1. A, B = Event
2. $P(A|B)$ = probabilitas bersyarat A yang diberikan oleh B
3. $P(B|A)$ = probabilitas bersyarat B yang diberikan oleh A
4. $P(A)$ = probabilitas kejadian A
5. $P(B)$ = probabilitas kejadian B

1.6.4 Website

Menurut (Elgamar, 2020) Website merupakan suatu media dengan sejumlah halaman yang saling berhubungan (*hyperlink*) dengan kemampuan memberikan atau menggabungkan informasi dalam bentuk teks, gambar, video, audio, dan animasi. Website adalah kumpulan dari halaman web yang berhubungan satu sama lain. Halaman awal merupakan halaman yang dilihat pertama kali oleh pengunjung situs web tersebut (Nugroho, 2011).

Website atau situs juga dapat diartikan sebagai kumpulan halaman-halaman yang menampilkan informasi, data teks, data gambar, data animasi, suara, video, serta kombinasi halaman statis dan dinamis yang masing-masing membentuk rangkaian bangunan yang saling berhubungan melalui satu sama lain yang disebut Halaman atau hyperlink terhubung (Maharani et al., 2021).

Berdasarkan berbagai kutipan diatas, peneliti menyimpulkan bahwa Website adalah media yang terdiri dari kumpulan halaman yang saling terhubung melalui hyperlink, menampilkan berbagai informasi dalam bentuk teks, gambar, video, audio, dan animasi, serta memiliki halaman awal yang dilihat pertama kali oleh pengunjung. Halaman-halaman ini dapat berupa kombinasi dari konten statis dan dinamis, membentuk suatu rangkaian informasi yang saling terkait.

1.6.5 PHP

Menurut (Elgamar, 2020) PHP awalnya dikembangkan oleh Rasmus Lerdorf pada akhir tahun 1994 dan kini telah diakuisisi oleh PHP Group. PHP awalnya merupakan singkatan dari *Personal Home Page*, namun dalam perkembangannya diubah menjadi PHP: *Hypertext Preprocessor*.

Menurut (Shofura, 2023) PHP (PHP: *Hypertext Preprocessor*) adalah bahasa *scripting* sisi server dimana pengolahan data terjadi di sisi server. Sederhananya, server menerjemahkan skrip program dan mengirimkan hasilnya ke klien yang membuat permintaan.

PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman web yang banyak digunakan dalam proses pengembangan web untuk membuat situs web yang dinamis dan interaktif. Artinya, ketika browser klien membuat permintaan, server memproses sintaks dan perintah PHP dan kemudian mengirimkan hasilnya ke browser klien dalam bentuk HTML biasa. Sintaks PHP untuk dokumen HTML tersedia dalam bagian kedua. Hal ini terletak di antara tag HTML yang membentuk dokumen HTML (Azimah, 2021).

Berdasarkan sumber kutipan diatas, peneliti menyimpulkan bahwa PHP adalah bahasa skrip sisi server yang dikembangkan oleh Rasmus Lerdorf pada akhir tahun 1994 dan sekarang diakuisisi oleh Grup PHP. PHP memungkinkan pemrosesan data di sisi server, memodifikasinya menggunakan skrip terprogram, dan mengirimkan hasilnya ke klien yang meminta dalam format HTML.

1.6.6 Framework

Pengertian Framework menurut (Rio Renaldo Prasena, 2020) Adalah Kerangka kerja yang dirancang untuk mempercepat dan menyederhanakan proses pembuatan aplikasi berbasis web. Kerangka kerja ini biasanya terdiri dari berbagai fungsi dan pustaka yang memfasilitasi berbagai aspek pengembangan bagi pengembang, mulai dari organisasi kode, pengujian, hingga pemeliharaan aplikasi. Kerangka kerja juga memungkinkan pengembang untuk bekerja dengan cara yang lebih sistematis dan terstruktur, yang pada akhirnya meningkatkan efisiensi dan kualitas aplikasi yang mereka kembangkan.

Menurut (Haniefardy et al., 2019) Framework adalah kerangka kerja yang dibuat untuk mempercepat kinerja pengembangan web. Kerangka kerja memungkinkan sistem web yang Anda buat menjadi lebih terorganisir dan terstruktur dengan lebih jelas. Kerangka kerja ini memiliki beragam fitur yang dapat digunakan pengembang untuk mempercepat proses pembuatan web, membuat kode lebih mudah dibaca, dan membantu proses perencanaan, pembangunan, pengujian, dan pemeliharaan.

Berdasarkan sumber kutipan diatas, peneliti menyimpulkan bahwa Framework adalah suatu struktur yang dibuat untuk mempercepat dan mempermudah proses pengembangan aplikasi web. Kerangka kerja menawarkan berbagai alat dan koleksi sumber daya yang mendukung pengembang dalam berbagai tahap pengembangan aplikasi, seperti struktur kode, pengujian, dan perawatan. Framework memfasilitasi pengembang agar dapat bekerja dengan lebih teratur dan terstruktur, sehingga pada akhirnya dapat meningkatkan efisiensi dan kualitas aplikasi yang dibuat. Kerangka kerja juga mempermudah pembuatan sistem web yang terstruktur dan mudah dimengerti, memudahkan proses perencanaan, pembangunan, pengujian, dan pemeliharaan.

1.6.7 JavaScript

JavaScript awalnya dibuat untuk memvalidasi form pada situs web, tugas yang sebelumnya dipegang oleh bahasa server seperti Perl. Namun, dengan perkembangan waktu, JavaScript berkembang menjadi komponen utama dalam pengembangan teknologi situs web (Shofura, 2023). Menurut (Wicaksana, 2016) JavaScript adalah bahasa pemrograman yang didasarkan pada paradigma pemrograman berorientasi objek. Selain itu, JavaScript juga sering disebut sebagai bahasa pemrograman skrip, karena berbeda dengan bahasa pemrograman umumnya. Bahasa pemrograman skrip adalah bahasa yang diinterpretasi secara khas dan dapat diketik langsung, serta dapat dieksekusi tanpa perlu compile.

Menurut (Ardiansyah, 2021) JavaScript adalah salah satu bahasa pemrograman yang sering digunakan untuk mengembangkan aplikasi client-side. Bahasa ini memungkinkan pembuatan website yang lebih interaktif dan dinamis. Awalnya, JavaScript hanya berjalan di sisi client, tetapi dengan perkembangan teknologi, sekarang JavaScript juga dapat berjalan di sisi server dengan bantuan Node.js.

Berdasarkan sumber kutipan diatas, peneliti menyimpulkan bahwa JavaScript awalnya diciptakan untuk memvalidasi form pada situs web, tetapi seiring waktu, bahasa ini berkembang menjadi komponen penting dalam pengembangan teknologi situs web. JavaScript adalah bahasa pemrograman berorientasi objek

yang dapat dieksekusi langsung tanpa perlu dicompile, dan sering digunakan untuk mengembangkan aplikasi client-side. Selain itu, JavaScript juga dapat berjalan di sisi server dengan bantuan Node.js, sehingga memungkinkan pembuatan website yang lebih interaktif dan dinamis.

1.6.8 MySQL

MySQL adalah salah satu perangkat lunak database yang populer dan sering diintegrasikan dengan web server Apache. MySQL memiliki keunggulan karena dapat melakukan operasi multithread, yang memungkinkan pekerjaan dibagi menjadi beberapa proses yang dapat dikerjakan secara bersamaan dan dapat beroperasi pada berbagai platform sistem operasi. MySQL menggunakan bahasa SQL (Structured Query Language) sebagai standar untuk mengakses server database. Bahasa SQL awalnya dikembangkan oleh IBM, tetapi sekarang telah diadopsi sebagai standar industri (Azimah, 2021).

Menurut (Shofura, 2023) database adalah server yang bertanggung jawab untuk mengelola database. Database adalah suatu sistem pengorganisasian data yang dirancang untuk memudahkan penyimpanan dan pengaksesan data. Database menyimpan data dan memungkinkan akses yang mudah dan cepat. MySQL merupakan salah satu contoh dari sistem manajemen database relasional (RDBMS). Dalam model ini, data dinyatakan dalam dua dimensi, yang disebut sebagai tabel.

Menurut (Praba & Safitri, 2020) MySQL diluncurkan pada tahun 2000 dengan lisensi ganda yang memungkinkan penggunaan gratis oleh publik di bawah lisensi GNU GPL (*General Public License*), yang menyebabkan popularitasnya meningkat drastis. MySQL mampu mengelola puluhan ribu tabel dan miliaran baris data dengan cepat dan lancar. Menurut (Hidayat, 2023) MySQL adalah sistem manajemen database relasional (RDBMS) yang didistribusikan secara gratis, di mana setiap orang bebas untuk menggunakannya. Namun, penggunaan MySQL tidak boleh dijadikan produk turunan yang bersifat komersial. MySQL adalah turunan dari salah satu konsep utama dalam database sejak lama, yaitu SQL (Structured Query Language).

Berdasarkan kutipan di atas, peneliti menyimpulkan bahwa MySQL adalah perangkat lunak database yang populer dan sering diintegrasikan dengan web server apache. MySQL memiliki keunggulan karena dapat melakukan operasi multithread, memungkinkan pekerjaan dibagi menjadi beberapa proses yang dapat dikerjakan bersamaan, dan dapat beroperasi pada berbagai platform sistem operasi. MySQL diluncurkan pada tahun 2000 dengan lisensi ganda yang memungkinkan penggunaan gratis oleh publik.

1.6.9 UML

Menurut (Alfina & Harahap, 2019) UML (*Unified Modeling Language*) adalah sebuah set konvensi pemodelan yang digunakan untuk menggambarkan dan menentukan sistem software yang berkaitan dengan objek. UML adalah alat bantu yang sangat efektif dalam pengembangan sistem berorientasi objek karena menyediakan bahasa pemodelan visual yang memungkinkan pengembang sistem untuk membuat rancangan visi mereka dalam bentuk yang standar. UML berfungsi sebagai penghubung dalam mengkomunikasikan aspek sistem melalui elemen grafis yang dapat digabungkan menjadi diagram.

Menurut (Krisna, 2020) *Unified Modeling Language* (UML) adalah salah satu standar bahasa yang paling umum digunakan dalam industri untuk mendefinisikan persyaratan, melakukan analisis desain, dan menggambarkan arsitektur pemrograman berorientasi objek. Dengan perkembangan teknologi pemrograman berorientasi objek, standar bahasa pemodelan untuk pengembangan perangkat lunak telah muncul. UML dibuat menggunakan teknologi pemrograman berorientasi objek, yang telah berkembang seiring waktu.

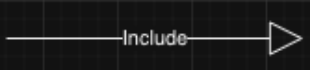
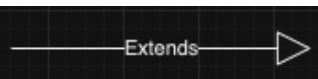
UML (*Unified Modeling Language*) adalah bahasa pemodelan yang digunakan untuk sistem atau perangkat lunak yang berorientasi objek. Konsep dasar UML dapat dijabarkan menjadi tiga kategori utama: klasifikasi struktural, perilaku dinamis, dan manajemen model. Ketika membuat diagram dan view, kita dapat memahami konsep utama sebagai istilah yang akan muncul. Diagram-diagram yang didefinisikan oleh UML termasuk Use Case Diagram, Class Diagram, Statechart Diagram, Activity Diagram, Sequence Diagram, Collaboration Diagram, Component Diagram, dan Deployment Diagram (Yahya, 2022).

Berdasarkan kutipan diatas, peneliti menyimpulkan bahwa UML adalah sebuah set konvensi pemodelan yang digunakan untuk menggambarkan dan menentukan sistem software yang berkaitan dengan objek. Dengan menggunakan bahasa pemodelan visual yang efektif, UML membantu pengembang sistem dalam membuat rancangan visi mereka dalam bentuk yang standar. Selain itu, UML juga berfungsi sebagai alat komunikasi yang efektif dalam mengkomunikasikan aspek sistem melalui elemen grafis yang dapat digabungkan menjadi diagram.

1. Use Case Diagram

Use Case Diagram (UC) adalah salah satu diagram yang penting dalam menggambarkan kebutuhan sistem, yang menunjukkan interaksi antara aktor dengan sistem secara visual. Setiap use case menggambarkan spesifikasi perilaku dari sistem yang diperlukan oleh aktor untuk mencapai tujuannya (Sutejo, 2016). Berdasarkan sebuah tulisan (Wulandari, 2023), Komponen-komponen pada use case diagram di antaranya sebagai berikut:

Tabel 1. Simbol Use Case Diagram

Simbol	Keterangan
	Aktor : Simbol ini berfungsi sebagai representasi dari entitas yang berkomunikasi dengan sistem, baik secara langsung maupun tidak langsung. Simbol ini merepresentasikan peran orang, sistem lain, atau alat ketika mereka berinteraksi dengan use case.
	Use Case : Simbol ini menggambarkan tindakan aktor dengan tujuan tertentu yang ingin dicapai. Simbol ini merupakan abstraksi dari interaksi antara sistem dan aktor.
	Association : Simbol ini merupakan abstraksi penghubung antara actor dan use case
	Generalization : Simbol ini menunjukkan spesialisasi actor untuk bisa berpartisipasi dalam use case.
	Include : Simbol ini menjelaskan bahwa suatu use case merupakan fungsionalitas dari use case lainnya.
	Extend : Simbol ini menjelaskan bahwa suatu use case merupakan tambahan fungsional dari use case lainnya jika suatu kondisi terpenuhi.

2. Activity Diagram

Activity Diagram menggambarkan kegiatan yang dilakukan oleh sistem, bukan oleh aktor, sehingga fokusnya adalah pada tindakan yang dapat dilakukan oleh sistem. Satu aktivitas dapat diwakili oleh satu atau lebih use case. Aktivitas ini mencerminkan proses yang berlangsung, sedangkan use case menunjukkan cara aktor berinteraksi dengan sistem untuk melaksanakan aktivitas tersebut (A. D. Putra et al., 2018). Komponen-komponen pada *activity diagram* adalah sebagai berikut (Dicoding, n.d.).

Tabel 2. Komponen-komponen Activity Diagram

Simbol	Keterangan
	<i>Initial node</i> : menggambarkan titik awal aktivitas
	<i>Activity final node</i> : menggambarkan akhir dari semua aliran control aktivitas
	<i>Activity</i> : menggambarkan aktivitas yang sedang berlangsung
	<i>Decision</i> : menggambarkan percabangan dimana ada pilihan aktivitas lebih dari satu
	<i>Join</i> : menggambarkan penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu

3. Class Diagram

Class Diagram adalah Representasi dari sistem tersebut, baik dalam bentuk program maupun menu pilihan, disajikan melalui class diagram. Diagram ini digambarkan sebagai sekumpulan kotak kecil yang berisi kata kunci beserta detail terkait yang saling berhubungan. Dalam proses perancangannya, diperlukan ketelitian dan pemikiran mendalam agar class diagram dapat diimplementasikan secara efektif ke dalam sistem yang akan dikembangkan.

Tabel 3. Komponen-komponen Class diagram

Simbol	Keterangan
	Class : Himpunan dari objek yang berbagi atribut serta operasi yang sama.
	Realization : operasi yang benar-benar dilakukan oleh objek.
	Dependency : merupakan hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (independent) akan memengaruhi elemen yang bergantung pada elemen yang tidak mandiri..
	Association: apa yang menghubungkan antara objek satu dengan objek lainnya.

1.6.10 SDLC

Metode SDLC (*System Development Life Cycle*) dalam konteks rekayasa sistem dan perangkat lunak adalah rangkaian langkah yang digunakan untuk merancang dan memodifikasi sistem, serta mencakup model dan metodologi yang diterapkan dalam pengembangan sistem tersebut. Istilah ini biasanya

berkaitan dengan sistem komputer atau informasi. Selain itu, SDLC juga dapat dipahami sebagai kerangka kerja yang digunakan untuk proses pengembangan perangkat lunak (Sugiarto et al., 2021).

Menurut (Puspa & Wulandari, 2021) metode SDLC (*System Development Life Cycle*), setiap fase harus diselesaikan sepenuhnya sebelum beralih ke fase berikutnya. Hal ini berarti bahwa perhatian dapat difokuskan secara maksimal pada setiap fase, karena tidak ada pekerjaan yang dilakukan secara bersamaan. Menurut (Banin, 2021) *System Development Life Cycle* (SDLC) adalah penerapan pendekatan sistematis dalam merancang dan memanfaatkan sistem berbasis komputer. Metode ini mencakup langkah-langkah awal dalam pengembangan sistem informasi yang dilakukan oleh analis sistem dan programmer untuk menciptakan sebuah sistem informasi.

SDLC adalah proses yang digunakan untuk mengembangkan atau memodifikasi sistem perangkat lunak dengan menerapkan berbagai model dan metodologi yang diterapkan oleh para profesional dalam pengembangan perangkat lunak. Selain itu, SDLC juga berfungsi sebagai kerangka acuan dalam proses pengembangan perangkat lunak. Dalam bidang rekayasa perangkat lunak, prinsip-prinsip SDLC menjadi dasar bagi berbagai metodologi pengembangan. Metodologi ini menyediakan struktur untuk perencanaan dan pengelolaan pembuatan sistem informasi, yang merupakan bagian dari proses pengembangan perangkat lunak (Hasanah, 2020).

Berdasarkan kutipan diatas, peneliti menyimpulkan bahwa Metode *System Development Life Cycle* (SDLC) adalah pendekatan sistematis dalam rekayasa perangkat lunak yang mencakup langkah-langkah penting untuk merancang, mengembangkan, dan memodifikasi sistem informasi. Setiap fase dalam SDLC harus diselesaikan sepenuhnya sebelum melanjutkan ke fase berikutnya, memungkinkan fokus maksimal pada setiap tahap. SDLC berfungsi sebagai kerangka acuan yang mendasari berbagai metodologi pengembangan perangkat lunak, memberikan struktur untuk perencanaan dan pengelolaan proyek. Dengan demikian, SDLC tidak hanya memandu proses pengembangan, tetapi juga memastikan bahwa setiap langkah dilakukan dengan cermat, menghasilkan produk akhir yang berkualitas tinggi.

1.6.11 Black-box Testing

Black-box Testing adalah metode pengujian perangkat lunak yang fokus pada fungsionalitas aplikasi tanpa mempertimbangkan struktur internal atau cara kerjanya. Pengujian ini tidak memerlukan pengetahuan mendalam tentang kode atau struktur aplikasi, serta pemahaman tentang pemrograman secara umum. Sebagai gantinya, pengujian ini menggunakan deskripsi eksternal perangkat lunak, seperti spesifikasi, persyaratan, dan desain, untuk merumuskan kasus uji. Tes ini dapat bersifat fungsional maupun non-fungsional, meskipun biasanya lebih condong ke fungsional. Dalam perancangan uji, dilakukan pemilihan input yang valid dan tidak valid serta penentuan output yang diharapkan (A. P. Putra et al., 2020).

Menurut (Rahadi & Vikasari, 2020) *Black-box Testing* adalah teknik pengujian perangkat lunak yang menekankan pada spesifikasi fungsi-fungsi dalam perangkat lunak yang sedang dikembangkan. Metode ini cenderung efektif dalam mengidentifikasi berbagai masalah, seperti fungsionalitas yang tidak berjalan dengan baik atau hilang, kesalahan dalam struktur data, masalah akses basis data, kesalahan pada antarmuka, serta isu terkait performa dan proses inisialisasi atau terminasi.

Menurut (Pramudita, 2020) Metode *Black-box testing* adalah salah satu teknik yang sederhana untuk diterapkan, karena hanya memerlukan batas bawah dan batas atas dari data yang diharapkan. Estimasi jumlah data uji dapat dihitung berdasarkan jumlah field data entri yang akan diuji, aturan entri yang harus dipenuhi, serta kasus batas atas dan batas bawah yang relevan.

Berdasarkan kutipan diatas, peneliti menyimpulkan bahwa *Black-box Testing* merupakan metode pengujian perangkat lunak yang berfokus pada pengujian fungsionalitas aplikasi tanpa mempertimbangkan struktur internalnya. Metode ini tidak memerlukan pemahaman mendalam tentang kode atau pemrograman, melainkan mengandalkan deskripsi eksternal seperti spesifikasi dan persyaratan untuk menyusun kasus uji. Pengujian ini dapat mencakup aspek fungsional dan non-fungsional, meskipun lebih sering digunakan untuk menguji fungsionalitas. *Black-box Testing* efektif dalam mendeteksi berbagai masalah, termasuk kesalahan fungsional, masalah struktur data, dan isu performa. Metode ini juga dikenal karena kesederhanaannya, yang memungkinkan pengujian dilakukan dengan menentukan batas bawah dan atas dari data yang diharapkan serta menghitung jumlah data uji berdasarkan aturan entri yang

relevan. Secara keseluruhan, *Black-box Testing* adalah teknik yang penting dalam memastikan kualitas dan keandalan perangkat lunak dengan menilai apakah aplikasi berfungsi sesuai dengan yang diharapkan.

1.6.12 User Acceptance Testing (UAT)

Menurut (Chamida et al., 2021) Pengujian User Acceptance Testing bertujuan untuk memverifikasi kesesuaian sistem dengan kebutuhan pengguna, mengukur pengalaman kenyamanan pengguna, serta menyelesaikan permasalahan yang ditemukan selama penggunaan. Sedangkan menurut penelitian yang dilakukan oleh (Suabdinegara et al., 2021) Pengujian *User Acceptance Testing* dilakukan untuk mengevaluasi keberhasilan penerapan sistem yang dirancang berdasarkan preferensi dan kebutuhan pengguna.

Berdasarkan jurnal diatas, maka peneliti menyimpulkan bahwa User Acceptance Testing berfungsi sebagai metode untuk memastikan sistem bekerja sesuai dengan harapan pengguna, baik dari segi kenyamanan, kesesuaian kebutuhan, maupun penyelesaian masalah dalam penerapannya.

BAB II

METODE PENELITIAN

2.1 Pendekatan Penelitian

Pendekatan penelitian yang akan digunakan dalam penelitian ini adalah pendekatan kualitatif. Pendekatan kualitatif adalah suatu metode penelitian yang memerlukan pemahaman teori untuk menjalankan kajiannya. Dalam penelitian kualitatif, penting untuk memahami teori-teori yang ada agar dapat dihubungkan dengan temuan terbaru. Penelitian ini tidak selalu bertujuan untuk menguji teori-teori yang telah ada sebelumnya; teori-teori tersebut dapat berfungsi sebagai pertimbangan dalam menilai validitas temuan yang dihasilkan oleh peneliti (Yusanto, 2020).

2.2 Metodologi Penelitian

Metode penelitian yang digunakan dalam penelitian ini adalah metode waterfall. Metode Waterfall adalah salah satu model dalam SDLC yang banyak diterapkan dalam pengembangan sistem informasi atau perangkat lunak. Model ini mengikuti pendekatan yang terstruktur dan berurutan. Proses dalam model ini dimulai dari tahap perencanaan dan berlanjut hingga tahap pengelolaan, dilakukan secara bertahap. Para pengembang perlu memahami lebih dalam tentang proses pengembangan sistem yang menggunakan model waterfall serta karakteristik yang melekat pada model ini (Abdul Wahid, 2020).

2.3 Metode Pengumpulan Data

Data dan informasi yang lengkap sangat krusial sebagai dasar dalam pengembangan sebuah sistem. Oleh karena itu, penting untuk mengumpulkan data yang diperlukan untuk proses pengembangan tersebut. Dalam penelitian ini, pengumpulan data dilakukan melalui studi literatur dan observasi.

1. Studi Literatur adalah serangkaian kegiatan yang berkaitan dengan pengumpulan data pustaka, membaca dan mencatat, serta metode pengolahan bahan penelitian (Kartiningrum, 2015). Pada penelitian ini, studi literatur dilakukan dengan mencari informasi dari sumber terpercaya seperti jurnal, buku, dan situs web yang relevan dengan penelitian.
2. Observasi adalah serangkaian kegiatan yang dilakukan untuk pengumpulan data dengan melakukan kegiatan pengamatan, mencatat fenomena yang muncul, dan mempertimbangkan hubungan antar aspek dalam fenomena tersebut (RAHMAWATI, 2023).
3. Wawancara merupakan salah satu metode pengumpulan data yang dapat kita gunakan dalam mendapatkan data melalui tatap muka dan tanya jawab langsung antara peneliti dan narasumber.

2.4 Waktu dan Lokasi Penelitian

Penelitian dilakukan mulai dari bulan April sampai dengan Agustus 2024 bertempat di Kantor Kecamatan Rappocini, Kecamatan Rappocini, Kota Makassar, Sulawesi Selatan..

Tabel 4. Jadwal Penelitian

No.	Kegiatan		2024															
			April				Mei				Juni				Juli			
			1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Studi Literatur																	
2	Pengumpulan data																	
3	Pengembangan Sistem	Requirements																
4		Design																
5		Implementation																
6		Testing																

2.5 Dataset

Dataset yang digunakan dalam penelitian ini berasal dari berbagai sumber, dataset ini terdiri dari gambar Surat Edaran, Surat Perintah, Surat Undangan, Surat Keputusan, Surat Pengantar, Surat Permohonan, dan Surat Kuasa. Total gambar dalam dataset ini berjumlah 190 gambar.

2.6 Pre-processing

Dalam penelitian ini, dilakukan *pre-processing* data pada dataset yang sebelumnya dikumpulkan untuk tujuan klasifikasi dokumen surat. *Dataset* awal yang tersimpan dalam satu folder, dibagi menjadi dua subset: data train dan data test.

2.7 Pelatihan Model

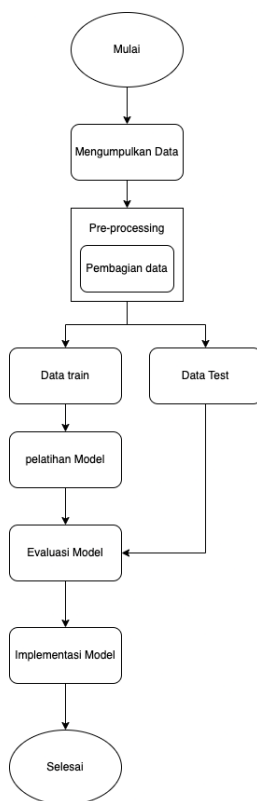
Pada pelatihan ini, model dilatih dengan menggunakan library *Tesseract-OCR* untuk mengekstrak text dari gambar kebentuk text digital. Setelah itu, hasil ekstraksi akan diolah kembali menggunakan algoritma Naïve Bayes untuk klasifikasi data text dan mendapatkan prediksi dan probabilitasnya. Hasil dari algoritma Naïve Bayes akan diolah lagi menggunakan algoritma *Artificial Neural Network Multi-Layer Perceptron* (ANN-MLP) dengan aktivasi *ReLU*, *Sigmoid*, dan *Softmax* untuk klasifikasi data citra berdasarkan jenis suratnya.

2.8 Evaluasi Model

Model yang dihasilkan dari proses pelatihan akan menghasilkan sejumlah metrik kinerja yang kemudian akan dievaluasi, termasuk melihat *confusion matrix*, akurasi, presisi, *recall*, dan *f1-score*.

2.9 Implementasi Model

Setelah proses evaluasi model, tahap selanjutnya adalah mengimplementasikan model tersebut agar dapat diintegrasikan ke dalam aplikasi web cerdas arsip digital surat.



Gambar 5. Alur Pengembangan Model

2.10 Metode Pengembangan Sistem

Pengembangan sistem pada penelitian kali ini menggunakan metode waterfall dengan tahapan sebagai berikut :

1. Requirements

Pada tahap awal penelitian, peneliti melakukan Analisa masalah yang ada. Setelah itu, peneliti melakukan analisis kebutuhan sistem yang diperlukan untuk menentukan komponen-komponen yang harus ada dalam sistem. Setelah melakukan analisis ini, peneliti dapat mengetahui apa saja komponen yang diperlukan untuk menciptakan sistem yang efektif dan efisien.

2. Design (Perancangan)

Pada tahap ini, peneliti merancang diagram use case dan diagram aktivitas untuk menggambarkan hubungan antara sistem dan pengguna. Selanjutnya, peneliti membuat desain antarmuka pengguna sebagai representasi visual dari fitur-fitur yang akan ada dalam sistem, yang akan membantu pengguna memahami sistem dengan lebih baik.

3. Implementation (Pengembangan sistem)

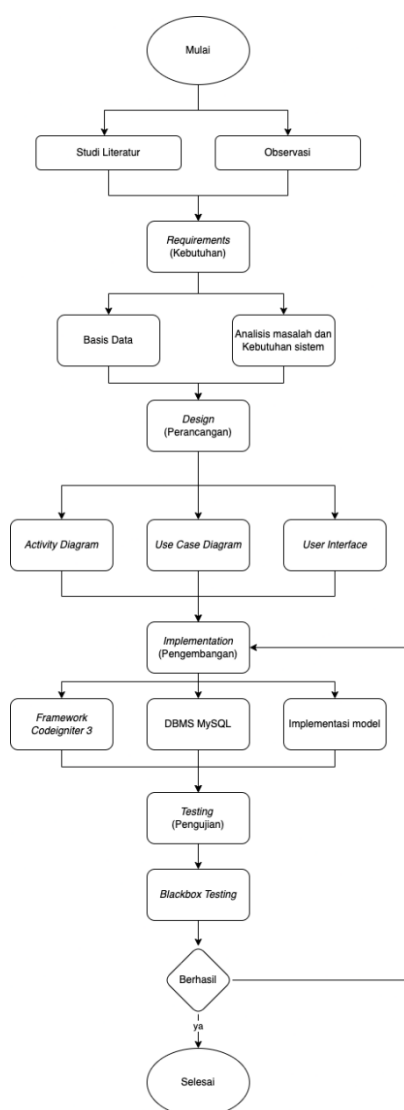
Pada tahap ini, peneliti menerapkan rancangan yang ada menjadi sebuah aplikasi website. Peneliti menggunakan Bahasa pemrograman PHP dengan kerangka kerja CodeIgniter 3 dalam pembangunan sistem. Dalam pembangunan sistem pengenalan dokumen digital surat, peneliti menggunakan algoritma *Naïve Bayes* dan *Artificial Neural Network-Multi Layer Perceptron* (ANN-MLP) yang diintegrasikan dengan library *Tesseract-OCR* menggunakan Bahasa *Python*. Manajemen basis data dilakukan dengan menggunakan MySQL.

4. Testing (Pengujian)

Pada tahap terakhir ini, pengujian website dilakukan dengan menggunakan metode black-box testing untuk memastikan sistem berfungsi secara optimal.

2.11 Pelaksanaan Penelitian

Penelitian ini dimulai dengan mengumpulkan data melalui observasi dan tinjauan literatur. Setelah data terkumpul, proses pengembangan sistem dimulai. Sistem ini dirancang dengan mengikuti pendekatan SDLC (Siklus Hidup Pengembangan Sistem) dengan menggunakan model waterfall. Proses pengembangan dimulai dengan tahap perancangan aplikasi, termasuk diagram use case, diagram aktivitas, dan antarmuka pengguna. Setelah perancangan selesai, tahap selanjutnya adalah implementasi dari pengembangan sistem. Sistem ini dikembangkan dengan menggunakan kerangka kerja CodeIgniter 3 dan MySQL sebagai manajemen basis data. Tahapan terakhir adalah tahap pengujian, yang dilakukan dengan menggunakan metode black-box testing. Ketika sistem memenuhi kebutuhan, penelitian dianggap selesai. Namun, jika terdapat ketidaksesuaian dengan kebutuhan, tahapan harus diulang dari awal, dimulai dari analisis kebutuhan.



Gambar 6. Alur Penelitian

2.12 Analisis Pengembangan Sistem

Pada penelitian ini, analisis pengembangan sistem mencakup analisis masalah dan analisis kebutuhan sistem.

2.12.1 Analisis Masalah

Berdasarkan hasil observasi yang dilakukan di kantor Kecamatan Rappocini, permasalahan yang ditemukan adalah proses pengarsipan surat yang masih menggunakan metode arsip secara fisik, dari hal ini mengakibatkan terjadinya penumpukan berkas-berkas surat yang membuat tempat penyimpanan surat yaitu lemari menjadi penuh, setelah itu pendataan surat yang masuk dan keluar di Kantor Kecamatan Rappocini masih dicatat menggunakan buku dengan tulis tangan sehingga hal ini membutuhkan waktu yang cukup lama. Selain itu, pencarian berkas surat memerlukan waktu yang lama karena surat-surat menumpuk tanpa ada pembatasan khusus seperti waktu surat masuk dan tujuan surat. Oleh karena itu, diperlukan upaya untuk meningkatkan efisiensi pengolahan arsip surat agar dapat lebih mudah dicari dan diakses oleh karyawan atau institusi terkait.

2.12.2 Analisis Kebutuhan Sistem

Perangkat lunak dan perangkat keras merupakan elemen yang sangat dibutuhkan sebagai factor pendukung penelitian, kebutuhan sistem dalam penelitian ini adalah sebagai berikut:

1. Perangkat lunak

Perangkat lunak yang dibutuhkan dalam penelitian ini adalah Visual Studio Code, Codeigniter, PhpMyAdmin, MariaDB, Draw.io.

2. Perangkat keras

Peneliti menggunakan laptop Macbook Air 2015 13inch dengan spesifikasi processor Intel Core i5, Ram 8gb, dan SSD 128 GB.

3. Pengguna Sistem

Terdapat 2 kategori pengguna pada aplikasi E-Arsip : Admin dan Super Admin. Admin adalah staf Kantor di Kantor Kecamatan Rappocini. Admin memiliki akses untuk mencetak laporan surat masuk dan surat keluar. Super Admin adalah staf Bidang umum yang sudah ditunjuk langsung oleh Camat di kantor Kecamatan Rappocini. Super Admin memiliki kewenangan untuk menginput data surat masuk dan surat keluar, melakukan perubahan data surat apabila akan dilakukan disposisi, melakukan registrasi akun admin. Menambah dan mengedit opsi perihal.

2.13 Rancangan Sistem

Rancangan sistem melibatkan perencanaan yang mendalam dan penetapan struktur serta fungsionalitas sistem yang akan dikembangkan. Hal ini melibatkan analisis yang mendalam mengenai operasi sistem, interaksi antara komponen-komponennya, dan penerapan kebutuhan fungsional dan non-fungsional. Berikut adalah use case diagram dari desain sistem informasi pada penelitian ini:



Gambar 7. Use Case Diagram

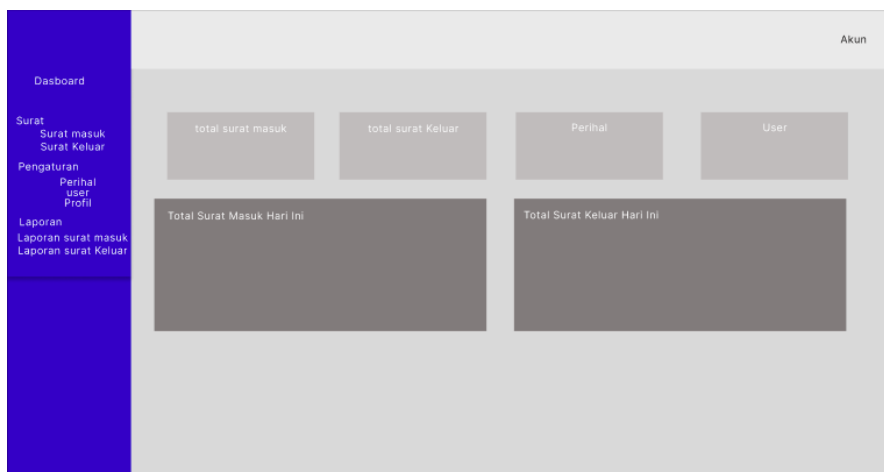
Gambar 5 mengilustrasikan bahwa pada aplikasi ini memiliki 2 tingkatan pengguna yaitu Admin dan Super Admin. Admin perlu melakukan Login Terlebih Dahulu untuk mencetak laporan surat masuk dan Laporan surat keluar. Super Admin harus login terlebih dahulu untuk menambahkan surat masuk dan surat keluar, menghapus surat masuk dan surat keluar, mengedit surat masuk dan surat keluar, menambahkan, menghapus dan mengedit opsi perihal, membuat akun untuk Admin, serta mencetak Laporan surat masuk dan surat keluar.

2.14 Desain Sistem

Peneliti merancang desain sistem aplikasi berupa Sketsa sebagai representasi awal dari antarmuka pengguna dan fungsionalitas sistem yang akan dibangun. Tujuan dari sketsa ini adalah untuk memberikan gambaran visual awal tentang susunan visual dalam pengembangan sistem yang akan dilakukan.

2.14.1 Desain Super Admin

Desain halaman beranda sistem untuk Super Admin pada penelitian ini digambarkan pada sketsa berikut :

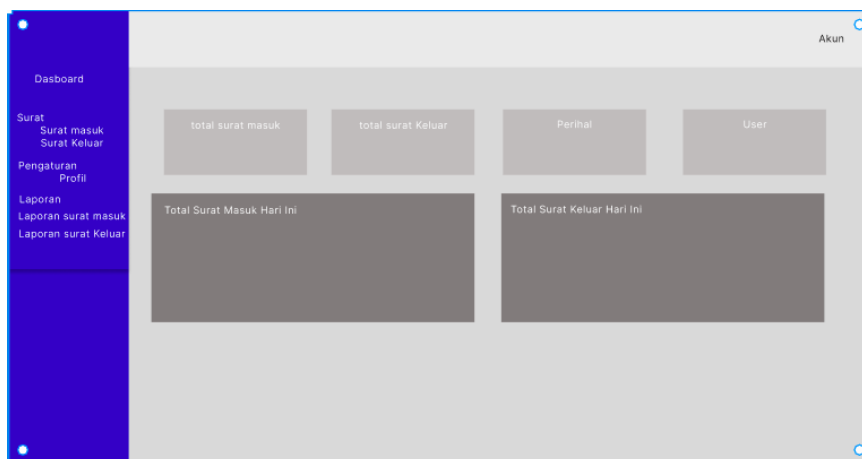


Gambar 8. Desain Super Admin

Bagian ini terdapat elemen *navigation bar* sebagai navigasi disebelah kiri yang mengarahkan pengguna ke fitur yang diinginkan. Semua fitur ini dapat diakses oleh Super Admin.

2.14.2 Desain Admin

Desain halaman beranda sistem untuk Admin pada penelitian ini digambarkan pada sketsa berikut :



Gambar 9. Desain Admin

Bagian ini terdapat elemen *navigation bar* sebagai navigasi disebelah kiri yang mengarahkan admin ke fitur yang diinginkan tetapi tidak dapat menambah, menghapus, mengedit surat masuk dan surat keluar. Admin hanya bisa mencetak laporan surat masuk dan surat keluar saja.