

BAB II

KAJIAN PUSTAKA

2.1 Teori-teori terkait bahasan Penelitian (*Relevan Theoris*)

Pada bab ini, Peneliti akan memaparkan sejumlah landasan teori yang digunakan dalam penelitian berjudul **“Implementasi Algoritma *Lempel–Ziv–Welch* Untuk Efisiensi Penyimpanan Dokumen Berbasis Foto Pada Sistem PPDB Di SMPN 1 Cilimus”** landasan teori tersebut mencakup penelitian terkait, definisi, serta teori yang diambil dari jurnal, artikel, maupun sumber *online*.

2.1.1 Impelementasi

Implementasi merupakan proses penerapan suatu konsep atau sistem ke dalam praktik nyata yang melibatkan berbagai komponen agar tujuan yang telah direncanakan dapat tercapai secara efektif (Maninggar et al., 2023). Dalam konteks sistem informasi, implementasi merupakan bagian dari proses pengembangan yang bertujuan untuk menerapkan hasil perancangan ke dalam sistem yang dapat dijalankan dan digunakan oleh pengguna (Algiffary & Sutabri, 2023).

Dari pengertian di atas dapat disimpulkan bahwa implementasi adalah proses penerapan suatu sistem, metode, atau rancangan ke dalam bentuk nyata sehingga dapat digunakan sesuai dengan tujuan yang telah ditetapkan. Dalam pengembangan sistem informasi, implementasi merupakan tahap lanjutan setelah perancangan yang mencakup proses pembangunan sistem, pengujian, serta penerapan kepada pengguna. Tahapan ini juga merupakan momen krusial dimana sistem mulai diooperasikan dalam situasi sebenarnya. Sehingga dapat dievaluasi apakah sistem tersebut mampu mencapai tujuan yang diharapkan. Dengan demikian, implementasi tidak hanya sebatas pelaksanaan, tetapi juga mencakup pengujian keberhasilan dalam memenuhi kebutuhan yang telah

di rancang sebelumnya.

2.1.2 Algoritma

Algoritma merupakan urutan langkah-langkah logis dan sistematis yang digunakan untuk menyelesaikan suatu permasalahan tertentu. Algoritma disusun secara terstruktur agar dapat menghasilkan solusi yang tepat dan efisien dalam proses pengolahan data.

Algoritma adalah serangkaian langkah yang dirancang secara sistematis untuk memproses input menjadi output yang diinginkan dalam suatu sistem komputasi (Fauzan et al., 2022). Selain itu, algoritma merupakan prosedur penyelesaian masalah yang memiliki tahapan yang jelas, teratur, dan dapat diimplementasikan ke dalam program computer (Yuliastuti et al., 2024).

Dalam bidang teknologi informasi, algoritma memiliki peran yang sangat penting sebagai dasar dalam pengembangan perangkat lunak, khususnya dalam proses pengolahan data dan pengambilan keputusan secara otomatis. Algoritma yang baik harus memenuhi beberapa karakteristik, di antaranya memiliki input dan output yang jelas, langkah-langkah yang terbatas tidak ambigu, serta efektif dalam menghasilkan solusi.

Dapat disimpulkan bahwa algoritma merupakan suatu prosedur sistematis yang digunakan untuk menyelesaikan permasalahan melalui langkah-langkah yang terstruktur, logis, dan berurutan, serta dapat diimplementasikan baik secara manual maupun melalui sistem komputer untuk menghasilkan solusi yang efektif dan efisien.

2.1.3 Algoritma *Lempel Ziv Welch*

Algoritma *Lempel–Ziv–Welch* (LZW) merupakan salah satu algoritma kompresi data yang termasuk dalam kategori *lossless compression*, yaitu teknik kompresi yang tidak menghilangkan informasi asli dari data. Algoritma ini dikembangkan oleh Terry Welch sebagai pengembangan dari algoritma LZ78, dengan tujuan meningkatkan efisiensi

dalam proses kompresi data.

Algoritma LZW bekerja dengan prinsip *dictionary-based compression*, yaitu menggunakan kamus (*dictionary*) untuk menyimpan pola-pola karakter yang muncul dalam data. Berbeda dengan metode kompresi berbasis statistik, algoritma ini tidak menghitung probabilitas kemunculan karakter, melainkan mengidentifikasi pola berulang dalam data dan menggantinya dengan kode tertentu. Pendekatan ini membuat LZW mampu melakukan kompresi dengan cepat dan efisien (Prabiantissa et al., 2023).

Secara umum, proses kompresi menggunakan algoritma LZW dimulai dengan inisialisasi *dictionary* yang berisi karakter dasar. Selanjutnya, algoritma membaca data input secara berurutan dan membentuk kombinasi karakter. Jika kombinasi tersebut sudah terdapat dalam *dictionary*, maka proses dilanjutkan dengan menambah karakter berikutnya. Namun, jika kombinasi belum tersedia, maka algoritma akan menghasilkan output berupa kode dari kombinasi sebelumnya, kemudian menambahkan kombinasi baru tersebut ke dalam *dictionary*. Proses ini berlangsung secara berulang hingga seluruh data selesai diproses.

Kelebihan utama algoritma LZW adalah kemampuannya dalam melakukan kompresi data dengan kecepatan tinggi serta mempertahankan keutuhan data karena bersifat *lossless*. Selain itu, algoritma ini tidak memerlukan proses perhitungan yang kompleks sehingga relatif mudah untuk diimplementasikan. Namun, tingkat kompresi yang dihasilkan sangat bergantung pada banyaknya pola berulang dalam data, sehingga pada data dengan variasi tinggi, hasil kompresi cenderung kurang optimal (Fauzan et al., 2022).

2.1.4 Efisiensi

Efisiensi merupakan konsep penting dalam pengelolaan sistem informasi, khususnya dalam kaitannya dengan penggunaan sumber daya secara optimal. Efisiensi dalam pengelolaan data digital dapat dilihat dari

kemampuan sistem dalam menyimpan, mengakses, dan mengolah data secara cepat tanpa memerlukan kapasitas penyimpanan yang berlebihan (Simbolon et al., 2024). Selain itu, efisiensi penyimpanan data juga berpengaruh terhadap kinerja sistem, terutama dalam hal kecepatan akses data dan penggunaan *bandwidth* (Tantowi & Wijayanti, 2023).

Berdasarkan pendapat tersebut, efisiensi dapat dipahami sebagai kemampuan suatu sistem dalam mencapai hasil yang optimal dengan penggunaan sumber daya seminimal mungkin, baik dari segi waktu, kapasitas penyimpanan, maupun kinerja sistem secara keseluruhan. Dalam konteks sistem informasi, efisiensi tidak hanya berkaitan dengan penghematan sumber daya, tetapi juga mencakup peningkatan kecepatan proses dan kemudahan akses data.

Dalam pengolahan data digital, efisiensi sering dikaitkan dengan proses kompresi data, yaitu upaya untuk mengurangi ukuran file tanpa mengurangi fungsi utama dari data tersebut. Dengan ukuran file yang lebih kecil, sistem dapat menghemat ruang penyimpanan serta mempercepat proses transfer data, seperti pada proses unggah dan unduh dokumen. Oleh karena itu, semakin besar tingkat penghematan yang dihasilkan tanpa mengurangi kualitas data, maka semakin tinggi tingkat efisiensi yang dicapai dalam suatu sistem.

2.1.5 Penyimpanan

Penyimpanan data digital, khususnya berbasis *cloud storage*, memiliki peran penting dalam mendukung pengelolaan dokumen secara fleksibel dan efisien. Namun, penggunaan penyimpanan digital juga menghadapi berbagai tantangan, seperti keterbatasan kapasitas, keamanan data, serta kebutuhan *bandwidth* yang tinggi ketika terjadi akses data secara bersamaan (Tantowi & Wijayanti, 2023). Selain itu, pengelolaan penyimpanan data yang baik dapat meningkatkan efisiensi operasional sistem, terutama dalam hal pengarsipan dokumen digital, kemudahan akses, serta penghematan ruang penyimpanan (Simbolon et al., 2024).

Berdasarkan pendapat tersebut, penyimpanan data dapat dipahami sebagai suatu proses atau mekanisme dalam sistem informasi yang berfungsi untuk menyimpan data agar dapat diakses, dikelola, dan digunakan kembali sesuai kebutuhan. Dalam sistem digital, penyimpanan tidak hanya berkaitan dengan kapasitas ruang, tetapi juga mencakup kecepatan akses, keamanan data, serta efisiensi dalam pengelolaan data.

Dalam implementasinya, pengelolaan penyimpanan data yang kurang optimal dapat menyebabkan berbagai permasalahan, seperti keterbatasan ruang penyimpanan, lambatnya proses akses data, serta meningkatnya beban sistem. Oleh karena itu, diperlukan strategi yang tepat dalam mengelola penyimpanan data agar sistem dapat berjalan secara optimal.

Salah satu upaya yang dapat dilakukan untuk mengoptimalkan penyimpanan data adalah dengan menerapkan teknik kompresi data. Kompresi data memungkinkan ukuran file menjadi lebih kecil tanpa mengurangi fungsi utama dari data tersebut, sehingga dapat menghemat kapasitas penyimpanan serta mempercepat proses transfer data.

Berdasarkan uraian tersebut, dapat disimpulkan bahwa penyimpanan data merupakan aspek penting dalam sistem informasi yang tidak hanya berfokus pada kapasitas, tetapi juga pada efisiensi, keamanan, dan kinerja sistem secara keseluruhan.

2.1.6 Dokumen

Dokumen merupakan salah satu komponen penting dalam sistem informasi yang berfungsi sebagai media penyimpanan informasi dalam berbagai bentuk, baik fisik maupun digital. Dokumen digital merupakan bentuk pengembangan dari dokumen konvensional yang disimpan dalam format elektronik sehingga lebih mudah diakses, dikelola, dan didistribusikan (Simbolon et al., 2024). Selain itu, dokumen digital memiliki keunggulan dalam hal efisiensi penyimpanan, kemudahan akses, serta fleksibilitas dalam pengelolaan data dibandingkan dengan dokumen

fisik. Namun, dokumen digital juga memiliki tantangan, seperti kebutuhan kapasitas penyimpanan yang besar serta risiko keamanan data apabila tidak dikelola dengan baik (Tantowi & Wijayanti, 2023).

Dalam konteks sistem informasi, dokumen dapat dipahami sebagai kumpulan informasi yang direkam dalam suatu media, baik berupa teks, gambar, maupun bentuk lainnya, yang memiliki nilai informasi dan digunakan sebagai arsip, referensi, serta pendukung proses administrasi.

Pada sistem berbasis digital, dokumen umumnya disimpan dalam berbagai format, seperti gambar (*image*), PDF, maupun dokumen teks. Dokumen berbasis gambar, seperti hasil pemindaian kartu keluarga, kartu KIP, screenshot jarak serta sertifikat kejuaraan cenderung memiliki ukuran file yang besar sehingga memerlukan pengelolaan penyimpanan yang optimal.

Berdasarkan uraian tersebut, dapat disimpulkan bahwa dokumen merupakan media penyimpanan informasi yang memiliki peran penting dalam sistem informasi, sehingga memerlukan pengelolaan yang efektif dan efisien.

2.1.7 Foto (Citra Digital)

Foto merupakan hasil tangkapan gambar dari suatu objek yang direkam menggunakan perangkat digital dan disimpan dalam bentuk file. Dalam teknologi informasi, foto termasuk ke dalam citra digital, yaitu representasi visual yang tersusun atas kumpulan piksel yang membentuk suatu gambar (Fauzan et al., 2022).

Citra digital banyak digunakan dalam sistem informasi karena mampu menyajikan informasi secara visual. Dalam praktiknya, foto sering digunakan sebagai media penyimpanan dokumen, seperti hasil pemindaian kartu keluarga, kartu KIP, screenshot jarak serta sertifikat kejuaraan (Fauzan et al., 2022).

Foto sebagai citra digital memiliki ukuran file yang relatif besar, terutama jika memiliki resolusi tinggi, sehingga berdampak pada kebutuhan penyimpanan dan kecepatan transfer data. Hal ini menjadi kendala dalam

sistem yang mengelola banyak dokumen.

2.1.8 Sistem

Sistem merupakan sekumpulan elemen atau komponen yang saling berinteraksi dan bekerja sama untuk mencapai suatu tujuan tertentu. Dalam konteks teknologi informasi, sistem digunakan untuk mengelola data dan menghasilkan informasi yang bermanfaat bagi pengguna (Algiffary & Sutabri, 2023).

Sistem dalam bidang informasi umumnya terdiri dari beberapa komponen utama, seperti input, proses, dan output. Input merupakan data yang dimasukkan ke dalam sistem, kemudian diproses untuk menghasilkan output berupa informasi yang dapat digunakan dalam pengambilan keputusan. Selain itu, sistem juga melibatkan komponen lain seperti penyimpanan data dan mekanisme pengendalian untuk memastikan sistem berjalan dengan baik.

Dalam penerapannya, sistem informasi berperan penting dalam meningkatkan efisiensi dan efektivitas pengelolaan data, terutama dalam organisasi atau lembaga pendidikan. Dengan adanya sistem yang terintegrasi, proses pengolahan data dapat dilakukan secara lebih cepat, akurat, dan terstruktur.

Berdasarkan uraian tersebut, dapat disimpulkan bahwa sistem merupakan suatu kesatuan komponen yang saling terhubung dan bekerja secara terstruktur untuk mengolah data menjadi informasi yang berguna, sehingga dapat mendukung berbagai aktivitas dalam suatu organisasi.

2.1.9 Penerimaan Peserta Didik Baru (PPDB)

Penerimaan Peserta Didik Baru (PPDB) merupakan proses seleksi dan administrasi yang dilakukan oleh satuan pendidikan untuk menerima calon peserta didik baru pada setiap tahun ajaran. Proses ini bertujuan untuk menjaring calon siswa sesuai dengan ketentuan yang berlaku serta memastikan pemerataan akses Pendidikan (Renisa Indriyanti Ramadhani & Eko Darmanto, 2025).

Dalam pelaksanaannya, PPDB melibatkan berbagai tahapan, seperti pendaftaran, pengumpulan berkas persyaratan, verifikasi data, hingga pengumuman hasil seleksi. Seiring dengan perkembangan teknologi informasi, proses PPDB mulai beralih dari sistem konvensional ke sistem berbasis digital atau *online*, sehingga dapat meningkatkan efisiensi, transparansi, dan kemudahan akses bagi calon peserta didik.

Sistem PPDB berbasis *online* memungkinkan proses pendaftaran dilakukan secara daring, sehingga dapat mengurangi penggunaan kertas dan mempercepat proses administrasi. Namun, dalam implementasinya masih terdapat beberapa kendala, seperti pengelolaan dokumen digital yang berukuran besar serta keterbatasan kapasitas penyimpanan sistem.

Berdasarkan uraian tersebut, dapat disimpulkan bahwa PPDB merupakan proses penting dalam sistem pendidikan yang memerlukan dukungan sistem informasi yang efektif dan efisien agar dapat berjalan secara optimal.

2.1.10 Tools Perancangan Sistem

2.1.10.1 HTML

HTML (*HyperText Markup Language*) merupakan bahasa markup yang digunakan untuk menyusun struktur dasar halaman web. HTML berfungsi untuk menyusun struktur tampilan sebelum dikombinasikan dengan CSS untuk pengaturan desain visual. Hal ini menunjukkan bahwa HTML memiliki peran penting sebagai fondasi dalam pembangunan antarmuka sistem berbasis web (Sari et al., 2022).

HTML merupakan salah satu teknologi dasar dalam front-end development yang digunakan untuk membangun struktur halaman web sehingga dapat diakses dan berinteraksi dengan pengguna melalui browser (Abdullayev, 2024).

Berdasarkan uraian tersebut, dapat disimpulkan bahwa HTML merupakan bahasa dasar dalam pengembangan web yang berfungsi untuk membangun struktur halaman serta menyajikan informasi secara terorganisir agar dapat ditampilkan dengan baik kepada pengguna.

2.1.10.2 PHP

PHP (*Hypertext Preprocessor*) merupakan bahasa pemrograman yang digunakan untuk mengembangkan aplikasi berbasis web pada sisi server (*server-side scripting*). PHP berfungsi untuk mengolah data, menerima input pengguna, serta menghasilkan output dalam bentuk halaman web yang dinamis (Anaking et al., 2023). Selain itu, PHP juga digunakan dalam pengembangan sistem informasi berbasis web untuk mengelola proses pengolahan data serta mendukung integrasi antara antarmuka pengguna dengan basis data (Assani, 2024).

Berdasarkan uraian tersebut, dapat disimpulkan bahwa PHP adalah bahasa pemrograman *server-side* yang bersifat *open source* yang dapat terintegrasi dengan HTML dan berperan dalam mengelola logika aplikasi, memproses data, serta membangun sistem web yang dinamis dan terintegrasi dengan basis data.

2.1.10.3 CSS

CSS (*Cascading Style Sheets*) merupakan bahasa yang digunakan untuk mempercantik tampilan halaman website yang sebelumnya disusun menggunakan HTML, sehingga menghasilkan desain yang lebih menarik dan mudah dipahami oleh pengguna (Sari et al., 2022).

Sejalan dengan itu, CSS merupakan bagian penting dalam *front-end development* yang berfungsi untuk mengontrol tampilan dan layout halaman web agar lebih responsif dan konsisten di berbagai perangkat (Abdullayev, 2024).

Dapat disimpulkan bahwa CSS merupakan bahasa yang digunakan untuk mengatur tampilan visual halaman web sehingga menghasilkan desain yang menarik, terstruktur, dan responsif, serta meningkatkan kenyamanan pengguna dalam berinteraksi dengan sistem.

2.1.10.4 JavaScript

JavaScript merupakan bahasa pemrograman yang digunakan untuk menambahkan interaktivitas pada halaman web. JavaScript memungkinkan halaman web tidak hanya bersifat statis, tetapi dapat merespon tindakan pengguna seperti klik, input data, serta perubahan tampilan secara dinamis. JavaScript merupakan salah satu teknologi utama dalam *front-end development* yang digunakan untuk mengembangkan fitur interaktif pada halaman web sehingga meningkatkan pengalaman pengguna (*user experience*) (Prasetyo et al., 2022).

Selain itu, JavaScript juga dapat digunakan untuk memanipulasi elemen HTML dan CSS secara langsung melalui browser, sehingga memungkinkan perubahan tampilan tanpa harus memuat ulang halaman (*reload*).

Dapat disimpulkan bahwa JavaScript merupakan bahasa pemrograman yang digunakan untuk menambahkan interaktivitas, mengelola perilaku halaman web, serta meningkatkan pengalaman pengguna dalam berinteraksi dengan sistem berbasis web.

2.1.10.5 MYSQL

MySQL merupakan sistem manajemen basis data (*Database Management System/DBMS*) yang digunakan untuk menyimpan, mengelola, dan mengolah data dalam suatu sistem informasi. MySQL bersifat *open source* dan banyak digunakan dalam pengembangan aplikasi berbasis web karena kemampuannya dalam menangani data secara cepat dan terstruktur.

Menurut (Siregar et al., 2024), MySQL merupakan DBMS yang digunakan untuk menyimpan dan mengelola data secara terstruktur, serta mendukung proses pengolahan data melalui relasi antar tabel sehingga dapat meningkatkan efisiensi dan performa sistem.

Berdasarkan uraian tersebut, dapat disimpulkan bahwa MySQL merupakan sistem manajemen basis data yang berperan dalam penyimpanan, pengelolaan, dan pengolahan data secara terstruktur, serta mendukung kinerja sistem informasi agar lebih efisien dan terintegrasi.

2.1.10.6 Laravel

Laravel merupakan salah satu *framework* PHP yang digunakan untuk mengembangkan aplikasi berbasis web dengan konsep *Model-View-Controller* (MVC). Laravel dirancang untuk mempermudah proses pengembangan aplikasi dengan menyediakan berbagai fitur seperti *routing*, autentikasi, manajemen database, serta keamanan sistem.

Laravel merupakan *framework* PHP yang digunakan dalam pengembangan aplikasi web karena mampu menyediakan struktur yang terorganisir dan efisien sehingga mempermudah proses pembangunan system(Sinlae et al., 2024).

Dapat disimpulkan bahwa Laravel merupakan *framework* PHP yang digunakan untuk membangun aplikasi web secara terstruktur dengan konsep MVC, serta membantu meningkatkan efisiensi, kemudahan, dan keamanan dalam pengembangan sistem informasi.

2.1.10.7 Bootstrap

Bootstrap merupakan *framework* CSS yang digunakan untuk mempermudah pengembangan tampilan (*user interface*) pada aplikasi berbasis web. Bootstrap menyediakan berbagai komponen

siap pakai seperti tombol, form, navigasi, serta sistem grid yang membantu dalam menyusun *layout* halaman web agar lebih rapi dan responsif. *Framework* ini berbentuk template HTML dan CSS untuk kebutuhan seperti tombol, tipografi, dan komponen yang lainnya. Tidak hanya itu, bootstrap juga memiliki fitur library javascript (Prasetyo et al., 2022).

Selain itu, Bootstrap juga mendukung konsep *responsive design*, sehingga tampilan website dapat menyesuaikan secara otomatis dengan berbagai ukuran layar, baik pada perangkat desktop, tablet, maupun smartphone.

Dapat disimpulkan bahwa Bootstrap merupakan *framework* CSS yang digunakan untuk membangun tampilan web yang responsif, terstruktur, dan menarik, serta membantu meningkatkan efisiensi dalam proses pengembangan antarmuka sistem.

2.1.11 Tools Perangkat Lunak

2.1.11.1 Visual Studio Code

Visual Studio Code merupakan perangkat lunak *code editor* yang digunakan untuk menulis, mengedit, dan mengelola kode program dalam pengembangan aplikasi. penggunaan *code editor* dalam pengembangan aplikasi berbasis web dapat membantu proses penulisan dan pengelolaan kode program sehingga menjadi lebih terstruktur dan mudah dikembangkan (Kherina, 2022).

Salah satu *code editor* yang banyak digunakan adalah Visual Studio Code yang dikembangkan oleh Microsoft. Visual Studio Code bersifat *open source* dan mendukung berbagai bahasa pemrograman seperti HTML, CSS, JavaScript, dan PHP. Selain itu, Visual Studio Code dilengkapi dengan fitur seperti *syntax highlighting*, *auto completion*, *debugging*, serta dukungan *extension* yang dapat meningkatkan produktivitas dalam pengembangan sistem.

2.1.11.2 Laragon

Laragon merupakan perangkat lunak yang digunakan sebagai *local development environment* dalam pengembangan aplikasi berbasis web. Laragon menyediakan berbagai layanan seperti web server (Apache/Nginx), PHP, serta database MySQL yang terintegrasi dalam satu paket, sehingga mempermudah proses pengembangan dan pengujian aplikasi secara lokal.

Penggunaan *local server* dalam pengembangan sistem informasi sangat membantu dalam proses pengujian dan implementasi aplikasi sebelum dipublikasikan secara *online*. Hal ini menunjukkan bahwa keberadaan *development environment* seperti Laragon memiliki peran penting dalam mendukung proses pengembangan system (Andarsyah, R., 2022).

Laragon juga memiliki keunggulan dalam kemudahan instalasi, konfigurasi yang sederhana, serta performa yang ringan, sehingga banyak digunakan oleh pengembang dalam membangun aplikasi berbasis web. Selain itu, Laragon mendukung berbagai bahasa pemrograman serta integrasi dengan database, sehingga mempermudah pengelolaan data dalam sistem.

2.1.11.3 Draw.io

Draw.io merupakan perangkat lunak berbasis web yang digunakan untuk membuat berbagai jenis diagram seperti *flowchart*, *Entity Relationship Diagram* (ERD), dan diagram lainnya dalam proses perancangan sistem. Penggunaan tools ini membantu dalam memvisualisasikan alur proses serta struktur sistem secara lebih jelas dan terstruktur. Pemanfaatan Draw.io dalam pembuatan *flowchart* dapat membantu pengguna dalam memahami alur proses secara visual serta mempermudah dalam penyusunan langkah-langkah sistem secara sistematis (Firdaus et al., 2025).

Selain itu, Draw.io memiliki keunggulan berupa kemudahan

penggunaan, berbasis web tanpa memerlukan instalasi, serta menyediakan berbagai template diagram yang dapat digunakan sesuai kebutuhan. Dengan demikian, proses pembuatan diagram menjadi lebih efisien dan mudah dipahami.

2.1.11.4 Figma

Figma adalah salah satu tools atau aplikasi desain yang dapat digunakan pada windows untuk membuat prototype aplikasi serta berbagai desain lainnya. Umumnya Figma banyak digunakan oleh seseorang yang bekerja dibidang UI/UX, web design dan bidang lainnya yang sejenis (Suparman et al., 2023).

Figma merupakan perangkat lunak berbasis web yang digunakan untuk merancang desain antarmuka (*user interface*) dan pengalaman pengguna (*user experience*) pada aplikasi berbasis digital. Figma memungkinkan pengguna untuk membuat desain tampilan secara visual, mulai dari *layout*, warna, hingga komponen interaktif sebelum sistem diimplementasikan.

Selain itu, Figma memiliki keunggulan berupa akses berbasis web, fitur kolaborasi secara *real-time*, serta kemudahan dalam pembuatan *prototype* tampilan aplikasi. Dengan fitur tersebut, proses perancangan desain menjadi lebih efisien dan terstruktur.

2.1.12 Tools Desain Perancangan

2.1.12.1 Rich Picture

Rich Picture merupakan salah satu teknik dalam analisis sistem yang digunakan untuk menggambarkan kondisi suatu sistem secara menyeluruh dalam bentuk visual. Rich Picture digunakan untuk merepresentasikan berbagai elemen dalam sistem seperti aktor, alur proses, serta permasalahan yang terjadi, sehingga memberikan gambaran umum terhadap sistem yang sedang dianalisis.

Dalam proses perancangan sistem informasi, tahap analisis awal sangat penting untuk memahami kebutuhan sistem dan permasalahan yang terjadi. Salah satu cara yang dapat digunakan adalah dengan menggambarkan kondisi sistem secara visual sehingga memudahkan dalam memahami alur dan interaksi antar komponen sistem. Hal ini menunjukkan bahwa penggunaan Rich Picture dapat membantu dalam mengidentifikasi permasalahan serta menggambarkan sistem secara lebih jelas (Zulfa, 2020).

Rich Picture tidak memiliki aturan baku dalam penggunaan simbol, sehingga dapat digambarkan secara bebas sesuai dengan kebutuhan analisis sistem. Hal ini memungkinkan pengembang untuk menampilkan hubungan antar aktor, alur informasi, serta kendala yang terjadi dalam sistem secara lebih fleksibel dan mudah dipahami.

2.1.12.2 UML

Unified Modelling Language (UML) merupakan bahasa pemodelan yang digunakan untuk merancang dan menggambarkan sistem dalam bentuk diagram secara terstruktur. UML digunakan sebagai alat bantu dalam perancangan sistem informasi berbasis web untuk menggambarkan kebutuhan sistem serta hubungan antar komponen yang terlibat di dalamnya. Penggunaan UML membantu pengembang dalam memahami alur sistem serta mempermudah proses implementasi (Hidayati et al., 2023).

Dengan menggunakan UML, proses perancangan sistem menjadi lebih terstruktur dan mudah dipahami, sehingga dapat mengurangi kesalahan dalam tahap pengembangan serta meningkatkan kualitas sistem yang dibangun. Ada beberapa diagram UML yang sering digunakan, diantaranya:

1. *Use Case Diagram*

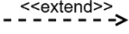
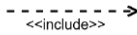
Use Case Diagram merupakan salah satu diagram dalam UML yang digunakan untuk menggambarkan interaksi antara aktor dengan sistem. Diagram ini menunjukkan fungsi-fungsi yang tersedia dalam sistem serta hubungan antara pengguna dengan sistem yang dikembangkan.

Use Case Diagram digunakan dalam perancangan sistem informasi berbasis web untuk menggambarkan kebutuhan fungsional sistem serta interaksi antara aktor dengan sistem (Voutama & Novalia, 2022).

Use Case Diagram menggunakan simbol-simbol standar UML yang dijelaskan dalam tabel simbol berikut :

Tabel 2.1 Use Case Diagram (Zulfa, 2020)

No	Simbol	Nama	Keterangan
1		<i>Actor</i>	Pihak di luar sistem yang berinteraksi dengan sistem informasi. Aktor dapat berupa orang, proses, atau sistem lain dan tidak selalu mempresentasikan manusia meskipun digambarkan dengan simbol orang.
2		<i>Use case</i>	Fungsi atau layanan yang disediakan oleh sistem untuk menggambarkan interaksi antara aktor dan sistem.

3		<i>Association</i>	Hubungan antara aktor dengan use case yang menunjukkan adanya interaksi atau keterlibatan aktor dalam menjalankan fungsi tertentu pada sistem.
4		<i>Extend</i>	Hubungan antar use case yang menunjukkan adanya perluasan atau tambahan perilaku pada use case tertentu yang bersifat opsional dan terjadi pada kondisi tertentu.
5		<i>Generalization</i>	Hubungan yang menggambarkan pewarisan dari use case atau aktor yang bersifat umum ke use case atau aktor yang bersifat umum ke use case atau aktor yang lebih spesifik.
6		<i>Include</i>	Hubungan antar use case yang menyatakan bahwa suatu use case selalu melibatkan use case lain sebagai bagian dari prosesnya.

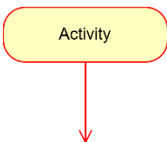
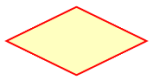
2. Activity Diagram

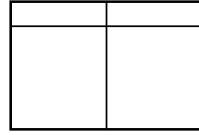
Activity Diagram merupakan salah satu diagram dalam UML yang digunakan untuk menggambarkan alur aktivitas atau proses kerja sistem secara berurutan dari awal hingga akhir. Diagram ini menunjukkan bagaimana suatu proses dijalankan serta menggambarkan aliran kontrol dalam sistem.

Activity Diagram digunakan dalam perancangan sistem informasi berbasis web untuk menggambarkan alur proses bisnis serta aktivitas yang terjadi dalam sistem (Voutama & Novalia, 2022). Dengan adanya diagram ini, pengembang dapat memahami urutan proses yang terjadi serta kondisi yang mungkin muncul dalam sistem.

Activity Diagram menggunakan simbol-simbol standar UML yang dijelaskan dalam tabel berikut:

Tabel 2.2 Activity Diagram (Zulfa, 2020)

No.	Simbol	Nama	Keterangan
1		<i>Start Point</i>	Menunjukkan titik awal dimulainya alur aktivitas dalam sistem.
2		<i>Activity</i>	Menggambarkan kegiatan atau proses yang dijalankan oleh sistem, umumnya dituliskan dalam bentuk kata kerja.
3		<i>Decision</i>	Titik dimana menunjukan beberapa kondisi atau pencabangan, sehingga alur

			memiliki akan memiliki sistem percabangan.
4		<i>Join</i>	Menunjukkan proses penggabungan beberapa alur aktivitas menjadi satu alur yang sama.
5		<i>End State</i>	Menandai akhir dari seluruh rangkaian aktivitas yang dilakukan oleh sistem dalam activity diagram.
6		<i>Swimlane</i>	Digunakan untuk memisahkan aktivitas berdasarkan aktor atau pihak yang bertanggung jawab.

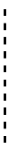
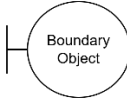
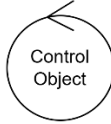
3. *Sequence Diagram*

Sequence Diagram merupakan salah satu diagram dalam UML yang digunakan untuk menggambarkan interaksi antar objek dalam sistem berdasarkan urutan waktu. Diagram ini menunjukkan bagaimana pesan atau komunikasi dikirimkan dari satu objek ke objek lain dalam suatu proses.

Sequence Diagram digunakan dalam perancangan sistem informasi berbasis web untuk menggambarkan alur komunikasi antar komponen sistem secara berurutan sesuai dengan proses yang terjadi (Voutama & Novalia, 2022). Dengan adanya diagram ini, pengembang dapat memahami bagaimana sistem merespon setiap permintaan yang dilakukan oleh pengguna.

Sequence Diagram menggunakan simbol-simbol standar UML yang dijelaskan dalam tabel berikut:

Tabel 2.3 *Sequence Diagram* (Zulfa, 2020)

No	Gambar	Nama	Deskripsi
1		<i>Actor</i>	Pihak yang berinteraksi dengan sistem.
2		Garis hidup (<i>lifeline</i>)	Menunjukkan keberadaan objek selama proses berlangsung.
3		<i>Entity</i>	Gambaran sistem dalam landasan menyusun basis data.
4		<i>Boundary</i>	Menangani komunikasi lingkungan ke sistem
5		<i>Control</i>	Bertanggung jawab atas kelas-kelas serta objek yang berisi logika.
6		Waktu Aktif	Menunjukkan periode objek sedang menjalankan proses.

7	1:nama_metode() 	<i>Call</i>	Pesan untuk memanggil proses atau fungsi.
8	1:masukan 	<i>Send</i>	Pesan pengiriman data antar objek.
9	1:keluarkan 	<i>Return</i>	Pesan balasan hasil dari suatu proses.

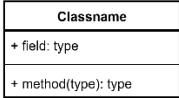
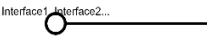
4. *Class Diagram*

Class Diagram merupakan salah satu diagram dalam UML yang digunakan untuk menggambarkan struktur sistem secara statis. Diagram ini menunjukkan kelas-kelas yang ada dalam sistem, beserta atribut, metode, serta hubungan antar kelas.

Class Diagram digunakan dalam perancangan sistem informasi berbasis web untuk menggambarkan struktur data serta relasi antar komponen dalam sistem (Voutama & Novalia, 2022). Dengan adanya diagram ini, pengembang dapat memahami bagaimana data disusun dan saling berhubungan dalam sistem yang dibangun.

Class Diagram menggunakan simbol-simbol standar UML yang dijelaskan dalam tabel berikut:

Tabel 2.4 Class Diagram (Zulfa, 2020)

No	Gambar	Nama	Keterangan
1		<i>Class</i>	Merupakan representasi kelas dalam struktur sistem.
2		<i>Interface</i>	Menggambarkan konsep antarmuka seperti pada pemrograman berorientasi objek.
3		<i>Association</i>	Menunjukkan hubungan keterkaitan antara satu kelas dengan kelas lainnya.
4		<i>Directed association</i>	Hubungan antar kelas dengan arah tertentu.
5		<i>Generalization</i>	Hubungan pewarisan antar kelas.
6		<i>Dependency</i>	Hubungan ketergantungan antar kelas.
7		<i>Aggregation</i>	Hubungan antara kelas yang menunjukkan kepemilikan sebagian

2.1.13 Tools Pengujian Perangkat lunak

2.1.13.1 White Box Testing

White Box Testing merupakan metode pengujian perangkat lunak yang menitikberatkan pada analisis struktur internal dan logika kode program untuk memastikan setiap jalur eksekusi berjalan sesuai dengan perancangan sistem. Pengujian ini dilakukan dengan menganalisis alur program dan jalur independen guna menemukan kesalahan logika yang tidak terlihat dari sisi fungsional system (Putra & Nuryasin, 2024). Metode ini digunakan untuk memastikan bahwa seluruh struktur kode program telah berjalan dengan benar serta meminimalisir kesalahan pada logika pemrograman.

2.1.13.2 Black Box Testing

Black Box Testing adalah metode pengujian perangkat lunak yang menilai fungsi sistem berdasarkan spesifikasi tanpa melihat struktur internal program. Pengujian ini bertujuan memastikan output sistem sesuai dengan input dan kebutuhan pengguna (Fahrullah et al., 2025). Metode ini digunakan untuk menguji fungsi sistem dari sisi pengguna sehingga dapat diketahui apakah fitur yang dibangun telah berjalan sesuai dengan kebutuhan.

2.1.13.3 User Acceptance Testing (UAT)

User Acceptance Testing (UAT) merupakan tahap pengujian akhir yang dilakukan oleh pengguna untuk memastikan bahwa sistem telah memenuhi kebutuhan fungsional dan siap digunakan dalam lingkungan operasional sebenarnya. UAT berfokus pada pengujian berdasarkan skenario penggunaan nyata sebagai dasar penerimaan system (Fahrullah et al., 2025). Pengujian ini dilakukan untuk memastikan sistem yang dikembangkan telah sesuai dengan harapan pengguna sebelum sistem diimplementasikan secara penuh.

2.2 Penelitian Sebelumnya (*Previous Work*)

Berikut adalah beberapa penelitian yang berkaitan untuk mendukung penelitian ini:

Tabel 2.5 Tabel SOTA

No	Peneliti (Judul, Penulis, Tahun Penelitian)	Masalah Penelitian	Tujuan Penelitian	Metodologi Penelitian/Pemecahan Masalah	Jumlah data/Sampel Data Penelitian	Hasil Penelitian
1	<i>Discrete Shearlet Transform and Lempel-Ziv Welch Coding for Lossless Fingerprint Image Compression</i> Nahidah A. Kadim, Shawkat K. Guirguis, dan Hend A. Elsayed (2024)	- Ukuran file citra sidik jari besar - Membutuhkan penyimpanan tinggi - Perlu metode kompresi tanpa kehilangan data	- Menghasilkan kompresi citra yang efisien - Menjaga kualitas citra (<i>lossless</i>) - Mengoptimalkan penyimpanan data citra	- Discrete Shearlet Transform - Algoritma LZW - Proses encoding & decoding citra	Dataset citra sidik jari	- Ukuran file berkurang signifikan - Kualitas citra tetap terjaga - Efisiensi penyimpanan meningkat
2	<i>Comparison of Huffman Algorithm and Lempel-Ziv Welch Algorithm in Text File Compression</i> Muhammad Alif, Mohamad Nurkamal Fauzan, dan Cahyo	- Belum diketahui algoritma terbaik untuk kompresi teks - Perbedaan efisiensi antar algoritma	- Membandingkan algoritma Huffman dan LZW - Menentukan algoritma paling efisien	- Perbandingan algoritma - Pengujian kompresi file teks - Analisis rasio kompresi & waktu	File teks dengan berbagai pola	- LZW lebih optimal pada data berpola berulang - Proses kompresi lebih cepat - Efisiensi ukuran file

	Prianto (2022)					lebih baik pada kondisi tertentu
3	Penerapan Algoritma Lempel-Ziv Welch (LZW) Untuk Kompresi Data. Ichwan Arizki, Anggra Triawan, dan Farhan Zayid (2024)	- Ukuran data digital besar - Penyimpanan tidak efisien - Kebutuhan kompresi data	- Mengimplementasikan LZW - Mengurangi ukuran data - Meningkatkan efisiensi penyimpanan	- Implementasi algoritma LZW - Teknik dictionary-based compression - Proses kompresi & dekompresi	Data digital (teks/file)	- Ukuran data berhasil dikurangi - Tidak terjadi kehilangan data (lossless) - Proses berjalan efektif
4	<i>Implementation of Rapid Application Development (RAD) Method in the Design of Research Partner Recommendation System in Higher Education</i> Purnama Anaking, Mochamad Nizar Palefi Ma'ady, dan 'Ainatul Fathiyah Abdul Rahim (2023)	- Pengembangan sistem memakan waktu lama - Kurang fleksibel terhadap perubahan kebutuhan	- Mempercepat pengembangan sistem - Meningkatkan fleksibilitas sistem	- Metode RAD (Rapid Application Development) - Tahapan: planning, design, implementation - Iterasi cepat & evaluasi berulang	Data sistem rekomendasi	- Waktu pengembangan lebih cepat - Sistem mudah disesuaikan - Efisiensi proses pembangunan meningkat
5	Implementasi Algoritma Lempel-	- Ukuran file teks besar	- Mengoptimalkan kompresi teks	- Algoritma LZW - Teknik dictionary	File teks	- Ukuran file berkurang

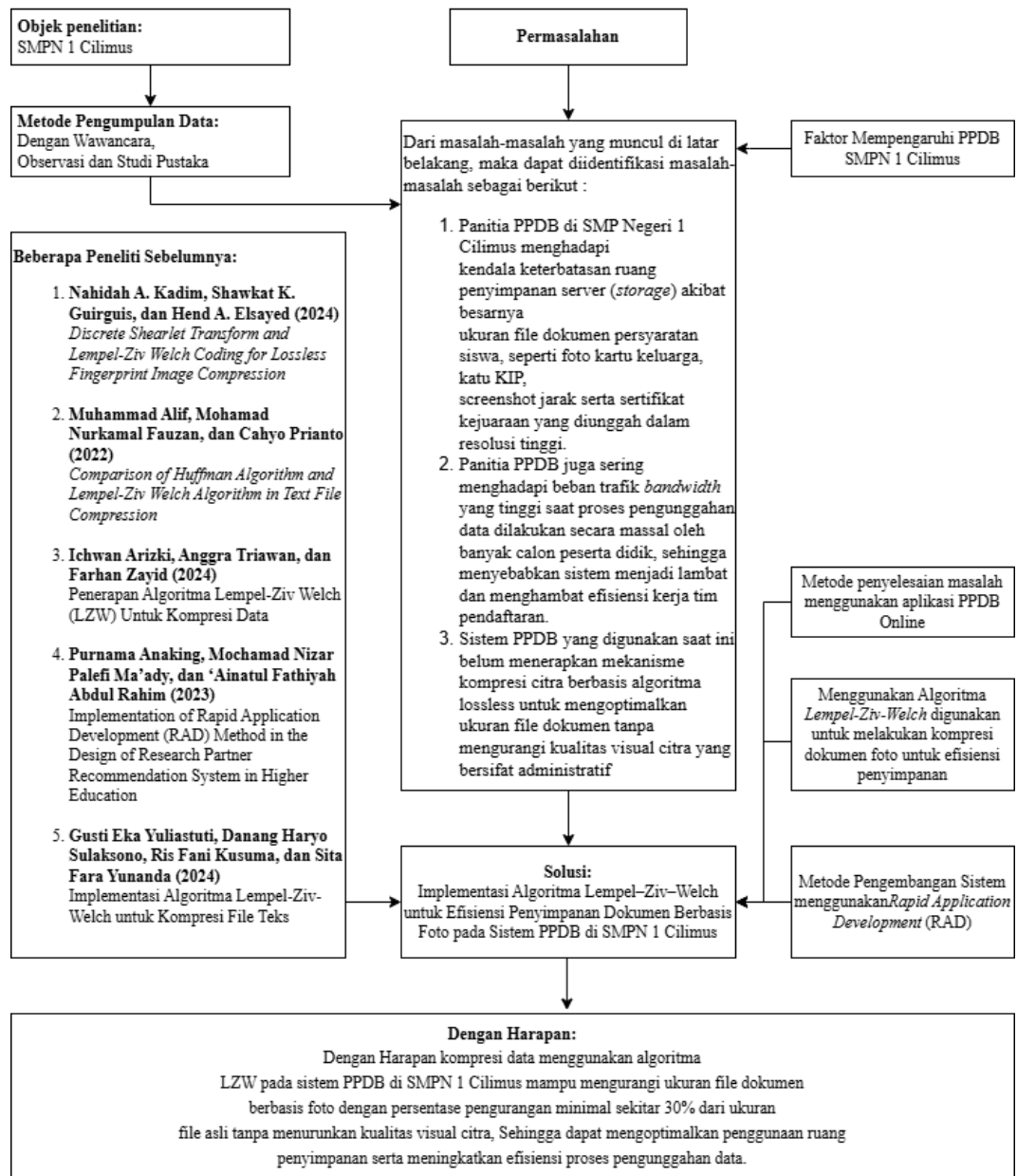
Ziv-Welch untuk Kompresi File Teks. Gusti Eka Yulastuti, Danang Haryo Sulaksono, Ris Fani Kusuma, dan Sita Fara Yunanda (2024)	- Pemborosan ruang penyimpanan - Kurangnya efisiensi data	- Mengurangi ukuran file - Menjaga keutuhan data	(kamus dinamis) - Proses encoding & decoding		- signifikan - Proses kompresi cepat - Data tetap utuh tanpa kehilangan informasi
---	--	---	---	--	---

Penelitian berjudul “Implementasi Algoritma *Lempel–Ziv–Welch* untuk Efisiensi Penyimpanan Dokumen Berbasis Foto pada Sistem PPDB di SMPN 1 Cilimus” bertujuan untuk mengatasi permasalahan pengelolaan dokumen pendaftaran yang memiliki ukuran file besar sehingga membutuhkan kapasitas penyimpanan tinggi. Selain itu, proses kompresi manual oleh pengguna seringkali kurang efisien dan menyulitkan.

Dalam penelitian ini, diterapkan algoritma *Lempel–Ziv–Welch* (LZW) sebagai metode kompresi data yang diintegrasikan ke dalam sistem PPDB berbasis web, sehingga proses kompresi dapat dilakukan secara otomatis saat pengunggahan dokumen tanpa menghilangkan informasi (lossless).

Berdasarkan penelitian sebelumnya, algoritma LZW terbukti efektif dalam mengurangi ukuran file baik pada data teks maupun citra. Namun, penelitian ini berfokus pada penerapan LZW dalam sistem PPDB berbasis web untuk mengoptimalkan penyimpanan dokumen digital. Dengan demikian, penelitian ini diharapkan dapat membantu SMPN 1 Cilimus dalam meningkatkan efisiensi penyimpanan serta pengelolaan data pendaftaran.

2.3 Kerangka Teoritis (*Theoretical Framework*)



Gambar 2.1 Kerangka Teoritis

Gambar 2.1 menunjukkan kerangka teoritis penelitian yang menggambarkan alur permasalahan hingga solusi yang diusulkan dalam pengembangan sistem PPDB di SMPN 1 Cilimus. Penelitian ini diawali dengan penentuan objek penelitian yaitu SMPN 1 Cilimus, serta metode pengumpulan data yang dilakukan melalui wawancara, observasi, dan studi pustaka untuk memperoleh informasi yang relevan.

Berdasarkan hasil analisis, ditemukan beberapa permasalahan utama, yaitu keterbatasan kapasitas penyimpanan server akibat ukuran file dokumen yang besar, tingginya penggunaan bandwidth saat proses unggah data secara bersamaan, serta belum adanya mekanisme kompresi data berbasis lossless pada sistem PPDB yang digunakan. Permasalahan tersebut dipengaruhi oleh beberapa faktor dalam pelaksanaan PPDB yang memerlukan pengelolaan data secara efektif dan efisien.

Sebagai solusi, penelitian ini mengusulkan pengembangan aplikasi PPDB berbasis web yang dilengkapi dengan penerapan algoritma *Lempel-Ziv-Welch* (LZW) untuk melakukan kompresi dokumen berbasis foto. Algoritma ini digunakan untuk mengurangi ukuran file tanpa menghilangkan kualitas data, sehingga dapat mengoptimalkan penggunaan ruang penyimpanan. Selain itu, metode pengembangan sistem yang digunakan adalah *Rapid Application Development* (RAD) untuk mempercepat proses pembangunan sistem dan menyesuaikan kebutuhan pengguna.

Dengan penerapan solusi tersebut, diharapkan sistem PPDB dapat meningkatkan efisiensi penyimpanan data, mengurangi beban bandwidth, serta mempermudah proses pengunggahan dokumen oleh pengguna. Target yang ingin dicapai adalah pengurangan ukuran file minimal sekitar 30% tanpa menurunkan kualitas visual, sehingga sistem dapat berjalan lebih optimal dan efektif.