

BAB I. PENDAHULUAN

1.1 Latar Belakang

Situs *web* memiliki beberapa komponen, salah satunya adalah yang menyuguhkan tampilan kepada pengguna atau biasa disebut sebagai *front-end*. *Front-end developer* harus memastikan bahwa tampilannya efisien dan efektif yang merupakan prioritas utama yang harus dipenuhi (Rohman, 2023). Seorang *front-end developer* bertanggung jawab dalam menciptakan tampilan situs *web* atau aplikasi dengan menggunakan HTML, CSS, dan *JavaScript*, yang merupakan dasar yang wajib dikuasai oleh seorang *front-end developer* dalam pengetahuannya. HTML dianggap sebagai suatu keharusan karena merupakan struktur utama atau tumpuan dasar dalam pembuatan situs *web*. Peran CSS dalam pembuatan situs *web* sangat penting, karena CSS berfungsi sebagai pengontrol tampilan halaman pada struktur HTML. CSS berperan dalam mengatur berbagai elemen seperti *font*, warna, gambar latar belakang, dan komponen lainnya untuk mencapai tampilan yang diinginkan. Selain HTML dan CSS, *JavaScript* juga memegang peranan penting dalam meningkatkan performa fitur-fitur serta memberikan animasi, sehingga mampu menciptakan situs *web* yang interaktif dan dinamis bagi pengguna (Basri, 2020).

PT. Telkom Akses merupakan anak perusahaan dari PT. Telekomunikasi Indonesia Tbk. yang fokus pada konstruksi dan pembangunan infrastruktur jaringan, serta menyediakan layanan manajemen. PT. Telkom Akses secara aktif terlibat dalam pekerjaan jasa konstruksi instalasi jaringan akses *broadband* dan juga berperan sebagai penyedia *Network Terminal Equipment* (NTE), selain itu, mereka juga menawarkan layanan jasa manajemen operasional dan pemeliharaan jaringan akses *broadband* (Rinaldi, 2022). ODP (*Optical Distribution Point*) adalah ruang manajemen serat optik yang dilengkapi dengan perangkat-perangkat seperti *optical pigtail*, *connector adaptor*, dan ruangan *splitter*. ODP memiliki fungsi sebagai titik awal tempat penggunaan *drop cable* sebelum kabel tersebut masuk ke rumah pelanggan, dengan menyediakan tempat untuk jalur masuk dan keluar kabel terminasi (Muliandhi et al., 2020).

ODP berperan sebagai tempat instalasi yang menghubungkan kabel *fiber* optik distribusi dan kabel drop, khususnya untuk sambungan jaringan optik *single mode* (Nasution et al., 2022). Kegiatan rutin yang dilakukan oleh teknisi yang berkolaborasi dengan *team leader* adalah *monitoring* ODP. *Monitoring* ini dapat berupa perbaikan, pembersihan, ataupun pergantian perangkat. *Monitoring* ODP dilakukan agar jaringan yang tersalur pada pelanggan dapat berjalan dengan lancar dan tidak memiliki kendala. Teknisi akan memiliki target bulanan untuk jumlah ODP yang harus dilakukan pengecekan, yang kemudian dapat dikerjakan dengan target harian agar target bulanan tersebut dapat terpenuhi.

Teknologi informasi berperan penting dalam kehidupan masyarakat, baik dalam segi bisnis maupun individu. Peran penting teknologi dalam kehidupan bisnis tentunya dimanfaatkan oleh PT. Telkom Akses dalam rangka mempermudah dan meningkatkan produktivitas kerja, maka dibuatlah bot telegram untuk mencari titik lokasi dengan *output* kode *latitude* dan *longitude*. Teknisi dapat mengakses bot telegram tersebut ketika akan melakukan *monitoring* ODP. Misi PT. Telkom Akses yang dikutip dari *website* resminya yaitu “*Providing Excellent Fiber Network Deployment and Managed Service to Deliver Best Value for the Stakeholder*” yang berarti ingin menjadi penyedia layanan unggul dan memiliki nilai terbaik bagi penggunanya. Bot telegram ternyata masih belum cukup untuk untuk mengejar misi tersebut karena dinilai masih memakan banyak waktu sebab seringnya terjadi *error* atau kesalahan pemberian titik bahkan tidak berjalan sama sekali, sehingga teknisi tidak dapat dengan cepat melakukan pengecekan ODP.

Solusi selanjutnya untuk menuju misi menjadi penyedia layanan dan memiliki nilai terbaik tentunya produktivitas dan efektivitas kerja harus ditingkatkan, untuk menunjang efektivitas dalam memantau ODP adalah dengan membuat Aplikasi KYRA Sistem *Monitoring* ODP yang berfokus pada titik lokasi ODP dan tanggal terakhir ODP telah diperbaiki atau *maintenance* oleh teknisi. Sistem ini terdiri dari UI/UX, *front-end*, *back-end*, dan *testing*. Tugas Akhir ini ditulis dengan memusatkan perhatian dan fokusnya pada *front-end* yang memberikan hasil terjemahan dari desain *User Interface* agar aplikasi dapat

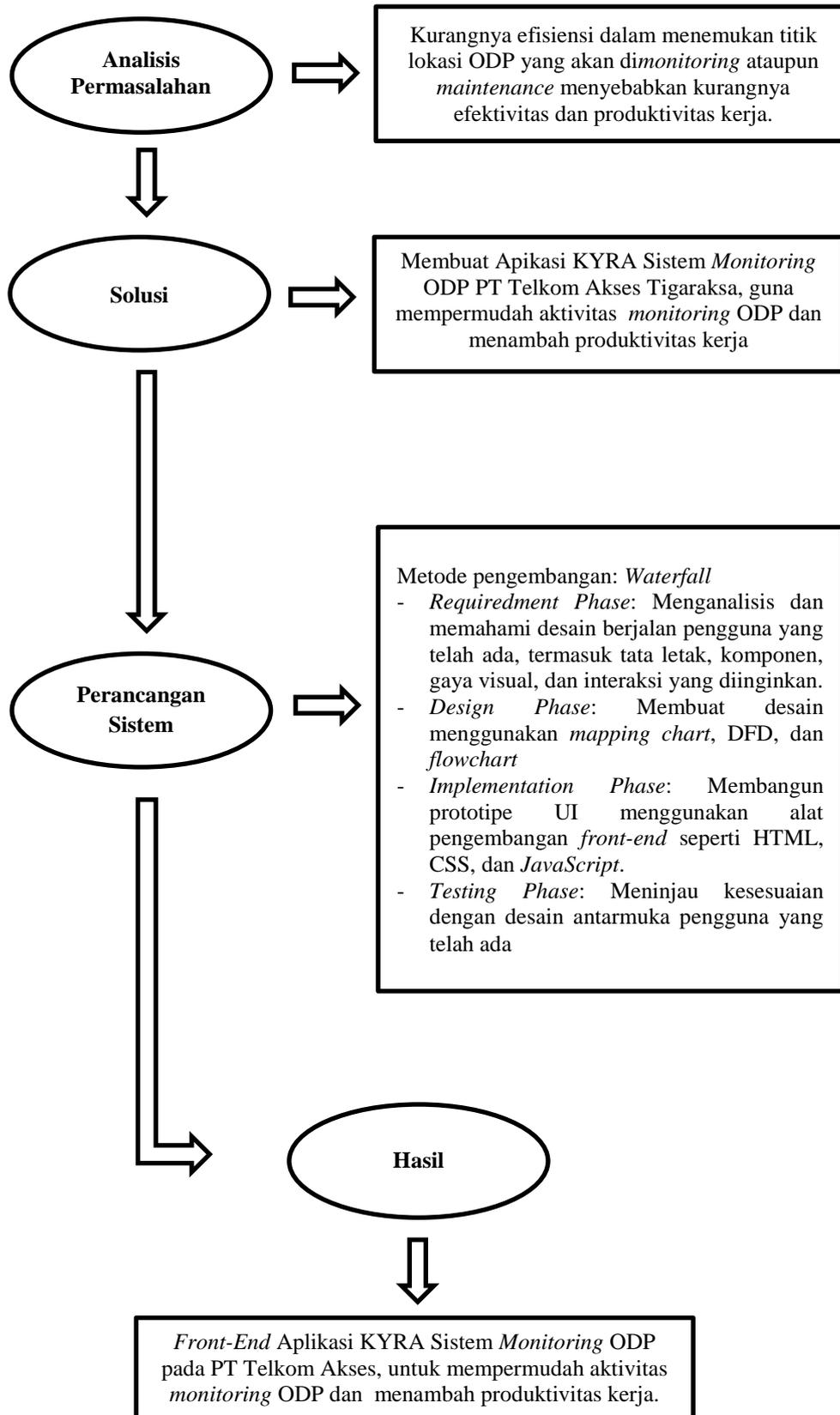
digunakan dalam berbasis *web* dan dapat digunakan oleh staf telkom terutama teknisi. Sistem ini akan memudahkan staf telkom dalam *monitoring* ODP.

1.2 Tujuan

Tujuan dari pembuatan tugas akhir ini adalah untuk merancang tampilan aplikasi yang dapat memberikan kemudahan dalam menjalankan tugas-tugas sehari-hari, meningkatkan efektivitas dan produktivitas, serta memudahkan akses ke informasi dan data terkait *monitoring* ODP.

1.3 Kerangka Pemikiran

Pencarian data lokasi ODP pada PT. Telkom Akses STO Tigaraksa saat ini masih perlu peningkatan untuk menunjang efektivitas dan produktivitas kerja. Pencarian data lokasi menggunakan bot telegram masih sering terjadi *error* atau kesalahan sistem bahkan tidak berjalan sama sekali maka dibutuhkan solusi yang dapat membantu teknisi maupun staf untuk mencari titik lokasi ODP yaitu dengan membuat aplikasi KYRA sistem *monitoring* ODP. Metode yang digunakan adalah metode *waterfall*. Kerangka pemikiran disajikan pada Gambar 1.



Gambar 1. Kerangka Pemikiran

1.4 Kontribusi

Pengaruh yang diharapkan dapat terlaksana setelah dibuatnya *front-end* Aplikasi KYRA Sistem *Monitoring* ODP diantaranya adalah:

1. Menghasilkan implementasi *front-end* aplikasi yang mudah digunakan oleh staf Telkom.
2. Memberikan kemudahan tampilan penggunaan aplikasi KYRA.

BAB II. TINJAUAN PUSTAKA

2.1 Implementasi

Implementasi adalah tahapan yang melibatkan pelaksanaan ide-ide kreatif, rangkaian aktivitas inovatif, atau program-program baru dengan tujuan mendorong penerimaan dari orang lain dan mampu mengubah pola pikir serta perilaku mereka sesuai dengan upaya yang telah dijalankan (Diatias, 2023). Proses implementasi ini berperan penting dalam menghadirkan perubahan nyata dan berkelanjutan, menghubungkan ide-ide inspiratif menjadi aksi yang dapat dijalankan oleh individu, kelompok, atau organisasi, sehingga memungkinkan tercapainya tujuan yang telah ditetapkan dengan lebih efektif dan efisien.

2.2 Aplikasi Berbasis *Web* dan *Front-end*

Aplikasi adalah *subclass* perangkat lunak yang dirancang khusus untuk memanfaatkan kemampuan komputer secara langsung, dengan tujuan menyediakan fungsi yang sesuai untuk memenuhi kebutuhan pengguna secara efisien dan efektif (Setyawan & Munari, 2020). Pengertian lain dari aplikasi adalah perangkat lunak atau program yang dibuat dan dikembangkan khusus untuk melaksanakan tugas-tugas spesifik di perangkat komputer, laptop, dan ponsel pintar, dengan tujuan memberikan solusi yang tepat dan efisien bagi penggunanya (Rohman, 2023). Aplikasi memiliki peran yang sangat signifikan dalam kehidupan manusia, mencakup berbagai bidang mulai dari fungsi dasar seperti alarm bangun pagi hingga peran penting dalam berbagi informasi politik dan ekonomi yang relevan.

Web mencakup semua komponen di internet yang dapat dijangkau dan diakses melalui perangkat lunak *web* browser. *Web* awalnya merupakan ruang informasi di dalam internet yang memanfaatkan teknologi *hypertext*. Pengguna dapat menemukan informasi dengan mengikuti tautan yang disediakan dalam dokumen *web* yang ditampilkan melalui *web browser* (Firmansyah, 2019).

Halaman *web* memiliki beberapa bagian, salah satunya adalah *front-end*. *Front-end* dikembangkan dengan menggunakan HTML, CSS, dan *JavaScript*. Ini memungkinkan URL dapat berfungsi dengan baik dan menampilkan situs *web* secara optimal (Dharma, 2021). Pengembangan *front-end* memungkinkan

pengguna untuk mengelola data dengan efektif dan mengubahnya menjadi tampilan aplikasi atau situs *web* yang menarik secara visual. Tugas *front-end developer* yakni menerapkan desain *interface* dan *experience* yang telah dibuat oleh *UI/UX Designer*.

2.3 Sistem

Sistem dapat didefinisikan sebagai gabungan elemen-elemen, komponen-komponen, atau variabel-variabel yang saling berhubungan dan terintegrasi satu sama lain untuk mencapai tujuan tertentu. Komponen tersebut tentunya saling berinteraksi secara terorganisir dan bergantung satu sama lain secara terpadu (Ayu et al., 2017). Dapat diartikan sistem merupakan sekumpulan objek, unsur, atau komponen yang saling berinteraksi dan saling mempengaruhi satu sama lain, serta memiliki keterhubungan yang erat dalam upaya mencapai tujuan tertentu di dalam lingkungan yang kompleks.

2.4 Monitoring

Monitoring atau pemantauan adalah proses pengawasan dan pemantauan sistem, proses, atau aktivitas tertentu yang terintegrasi untuk memastikan kelancaran proses agar sesuai dengan rencana yang telah ditetapkan. Pemantauan dapat memberikan data, laporan, atau pemahaman yang diperlukan mengenai status, kinerja, atau perkembangan yang sedang berlangsung untuk menerapkan langkah-langkah yang mengarah ke perbaikan berkelanjutan. Umumnya, pemantauan digunakan untuk memeriksa perbandingan antara kinerja aktual dengan target yang telah ditetapkan (Febriani et al., 2020). Tujuan utama dari *monitoring* adalah untuk mengumpulkan data yang relevan, menganalisisnya, dan mengambil tindakan yang sesuai berdasarkan hasil pemantauan tersebut.

2.5 Optical Distribution Point (ODP)

ODP (*Optical Distribution Point*) merupakan peranti terminasi yang berfungsi sebagai titik akhir dari jaringan serat optik dan merupakan titik awal bagi penggunaan *drop cable* sebelum masuk ke rumah pelanggan. Terdapat tiga jenis ODP yang umum digunakan, yaitu ODP *Pedestal*, ODP *Pole*, dan ODP *Closure* (Muliandhi et al., 2020).

ODP biasanya terletak di sepanjang rute jaringan serat optik, baik di bawah tanah maupun di atas tiang atau tiang listrik. ODP umumnya dilengkapi dengan perlengkapan dan perangkat keras yang diperlukan untuk menghubungkan serat optik, seperti splitter optik (*optical splitter*) atau penghubung serat optik (*fiber optic connector*). Ini memungkinkan serat optik yang berasal dari pusat distribusi serat untuk dipecah dan dihubungkan ke pelanggan individual melalui serat optik yang terhubung ke ODP.

2.6 Hypertext Pre-processor (PHP)

PHP adalah bahasa pemrograman skrip *server* yang digunakan secara luas dalam pengembangan aplikasi *web* interaktif dan dinamis. Penggunaan PHP dipadukan dengan HTML dan dieksekusi di *server* sebelum hasilnya dikirim ke komputer klien, memungkinkan sintaks dan perintah dieksekusi sepenuhnya sebelum ditampilkan kepada pengguna (Firmansyah, 2019). Menurut penelitian oleh Supono & Putratama pada tahun 2018, PHP merupakan bahasa pemrograman yang berfungsi sebagai alat untuk menerjemahkan basis kode program menjadi kode mesin yang dapat dipahami oleh komputer. PHP memiliki kekhususan sebagai bahasa pemrograman sisi *server* (*server-side*) yang dapat diintegrasikan dengan HTML.

Dikembangkan pada tahun 1994 oleh Rasmus Lerdorf. Awalnya, PHP merupakan akronim dari "*Personal Home Page Tools*" yang menunjukkan fokus awal bahasa pemrograman tersebut pada pembuatan halaman *web* pribadi, namun sekarang diartikan sebagai "*PHP: Hypertext Pre-processor*". PHP menggunakan sintaksis mirip dengan bahasa pemrograman C dan menyediakan berbagai fitur yang memudahkan pengembangan aplikasi *web*. PHP dapat berinteraksi dengan basis data, menghasilkan *output* HTML, mengelola formulir, mengirim dan menerima *cookie*, serta berkomunikasi dengan *server web* melalui protokol HTTP.

2.7 Cascading Style Sheet (CSS)

CSS (*Cascading Style Sheet*) merupakan bahasa yang berperan dalam mengontrol dan mengatur tampilan serta gaya dari elemen-elemen yang ada dalam markup bahasa halaman *web* seperti HTML. Fungsi utama CSS adalah memisahkan antara konten dengan presentasinya, sehingga memungkinkan pengaturan visual yang terpisah dan fleksibel pada situs *web* (Susanti et al., 2019).

Penggunaan CSS memiliki benefit bagi pengembang *web* untuk dapat mengontrol presentasi visual dan tata letak halaman *web*, termasuk warna, ukuran teks, jenis *font*, jarak antara elemen, dan banyak lagi. CSS bekerja sama dengan HTML dan *JavaScript* untuk mengembangkan halaman *web* yang menarik, responsif, dan interaktif. Pemisahan gaya dari struktur dan konten, CSS memberikan fleksibilitas dan kontrol yang lebih besar dalam merancang tampilan visual yang konsisten dan menarik untuk pengguna.

2.8 *JavaScript Object Notation (JSON)*

JSON (*JavaScript Object Notation*) merupakan format yang digunakan untuk pertukaran data secara ringan dan efisien antara *server* dan aplikasi *web*. Penggunaan JSON memungkinkan penyimpanan dan pengiriman data dengan efisiensi tinggi, memfasilitasi komunikasi yang cepat dan efektif antara komponen-komponen yang terlibat. JSON menggunakan sintaksis yang sederhana dan mudah dibaca oleh manusia, serta mudah diolah oleh mesin. *JavaScript Object Notation (JSON)* dibuat berdasarkan bahasa pemrograman *JavaScript*. JSON merupakan sebuah format yang digunakan untuk berbagi data antara *server* dan *client*. Format ini dapat diterapkan pada bahasa pemrograman lainnya seperti *Ruby*, *Python*, *PHP*, *Java*. Struktur JSON ditulis dengan format *key* (kata kunci) yang diikuti dengan *value* (nilai) dan mendukung beberapa tipe data seperti *array*, *object*, *string*, *number*, *boolean*, *null* (Hasanudin, 2020).

2.9 *JavaScript*

JavaScript merupakan bahasa pemrograman yang secara khusus dirancang untuk memberikan kehidupan pada halaman *web*, dengan memungkinkan interaksi dan responsivitas yang dinamis untuk meningkatkan pengalaman pengguna. *JavaScript* memungkinkan pengembang *web* untuk menambahkan fungsionalitas interaktif ke dalam halaman *web*. Dua suku kata yang menyusun *JavaScript* adalah *Java* dan *Script* yang masing-masing memiliki arti. *Java* adalah bahasa pemrograman yang berbasis objek yang digunakan untuk mengembangkan aplikasi yang kompleks, sementara *Script* merujuk pada serangkaian instruksi program yang lebih ringkas dan lebih sederhana (I Ardiansyah & J Nursalim, 2021). Penggunaan *JavaScript* memiliki keuntungan yaitu pengguna dapat berinteraksi dengan elemen-elemen dalam halaman, seperti tombol, formulir, dan

tautan, serta merespons peristiwa seperti klik *mouse*, pengisian formulir, dan navigasi. *JavaScript* memungkinkan pengembang untuk membuat halaman *web* yang interaktif, responsif, dan dinamis, serta mengelola data dan mengontrol alur eksekusi aplikasi *web*.

2.10 Framework Bootstrap

Bootstrap merupakan salah satu *framework* pemrograman CSS (*Cascading Style Sheet*) yang memiliki tujuan untuk menyederhanakan proses pengembangan desain situs *web* (I Ardiansyah & J Nursalim, 2021). *Bootstrap* dirancang dengan fokus pada responsivitas. Artinya, *interface* yang dibangun dengan menggunakan *Bootstrap* dapat menyesuaikan tampilannya secara otomatis dengan perangkat dan ukuran layar yang berbeda. Ini membantu memastikan bahwa halaman *web* tetap terlihat dan berfungsi dengan baik di berbagai perangkat, mulai dari desktop hingga ponsel pintar. Keuntungan menggunakan *bootstrap*, *developer* dapat memanfaatkan kerangka kerja yang telah ada untuk membangun *user interface* yang menarik dan konsisten dengan waktu yang lebih efisien.

2.11 XAMPP

XAMPP adalah sebuah perangkat yang menggabungkan beberapa perangkat lunak menjadi satu paket yang lengkap (Fikri, 2019). XAMPP menggabungkan singkatan "X" yang merepresentasikan pada berbagai sistem operasi, *Apache* sebagai *server web*, *MySQL* sebagai sistem pengelolaan *database*, *PHP* sebagai bahasa pemrograman yang berjalan di *server*, serta *Perl* sebagai bahasa pemrograman *skrip-oriented* ke dalam satu paket perangkat lunak. Lokal *web server* dikomputer dapat dibuat dengan menggunakan XAMPP yang dapat memudahkan dalam mengembangkan, mendesain, dan keperluan *testing website*. XAMPP menyediakan cara yang mudah dan terintegrasi untuk mengatur dan menjalankan lingkungan pengembangan *web* lokal. Ini memungkinkan pengembang untuk menguji dan mengembangkan aplikasi *web* secara lokal sebelum mempublikasikannya secara *online*. XAMPP sangat berguna bagi mereka yang ingin mengembangkan dan menguji proyek *web* di lingkungan lokal sebelum mengunggahnya ke *server web* yang sebenarnya.

2.12 Perancangan dan Desain Sistem

Perancangan adalah proses mendefinisikan dengan sebuah teknik yang berbeda, termasuk detail komponen dan deskripsi arsitektur, serta kendala yang dihadapi selama proses tersebut. Untuk menanggapi kebutuhan pengguna dan memberikan tampilan yang jelas dan lengkap merupakan tujuan perancang sistem (Aziz, N., Pribadi, G., & Nurcahya, MS., 2020). Perancangan bisa digambarkan dalam bentuk diagram alir yang menampilkan proses secara urut.

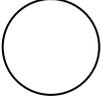
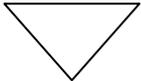
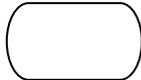
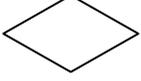
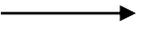
Desain sistem merupakan sekumpulan elemen yang berkaitan kemudian diklasifikasikan menggunakan standarisasi yang terpadu pada suatu sistem. Desain sistem bisa digunakan secara berulang oleh pengembang dan disesuaikan dengan kebutuhan organisasi. Pola pikir gambaran besar dalam merancang sistem dianggap penting karena diperlukan untuk mengambil keputusan (Kurniawan & Adiwijaya, 2021).

Perancangan dan desain sistem merujuk pada proses merencanakan dan merancang struktur, komponen, dan fungsionalitas suatu sistem komputer. Perancangan dan desain sistem yang baik mempertimbangkan aspek-aspek seperti skalabilitas, keamanan, kinerja, ketersediaan, dan pengalaman pengguna. Tujuannya adalah untuk menghasilkan sistem yang efektif, efisien, mudah digunakan, dan dapat diandalkan dalam memenuhi kebutuhan pengguna dan tujuan bisnis yang ditetapkan.

2.12.1 *Mapping chart*

Mapping chart bermakna sebagai diagram arus/alir yang mempunyai proses untuk menampilkan langkah dalam bentuk simbol-simbol grafis dan membantu dalam menganalisis penelitian untuk memecahkan suatu masalah. Fungsi *mapping chart* adalah untuk menyederhanakan dan menggambarkan proses agar lebih mudah dilihat dan dipahami (Mukodimah, S., Muslihudin, M., & Trisnawati, 2019). Data *mapping chart* diwakili oleh elemen-elemen grafis seperti bar, garis, lingkaran, atau area, yang membantu memvisualisasikan pola, tren, atau hubungan antara data secara lebih jelas. Simbol-simbol *mapping chart* disajikan pada Tabel 1.

Tabel 1. Simbol-simbol *mapping chart*

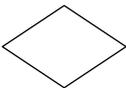
Simbol	Nama	Fungsi
	Dokumen	Menyatakan <i>penginputan</i> dokumen yang dicetak
	Dokumen Rangkap	Menyatakan dokumen yang lebih dari satu
	<i>Connector</i>	Menyatakan sambungan satu dengan yang lainnya pada proses berikutnya
	Proses	Proses pengolahan yang ditugaskan oleh komputer
	Proses Manual	Proses pengolahan yang dilakukan secara manual
	<i>Display</i>	Menyatakan <i>output</i> yang digunakan
	Arsip	Mengarsipkan data di dalam program
	Terminal	Awalan dan akhiran pada program
	<i>Decision</i>	Menyatakan kondisi yang menentukan 2 kemungkinan (iya atau tidak)
	<i>Connecting Line</i>	Menghubungkan simbol dengan simbol lainnya dengan menyatakan suatu alur proses

Sumber: (Fitriyana & Susianto, 2018)

2.12.2 Flowchart

Flowchart merupakan bagan untuk holistik sistem termasuk aktivitas-aktivitas manual dan genre atau arus dokumen yang digunakan dalam sistem untuk mendeskripsikan proses aktivitas dalam suatu organisasi. *Flowchart* memiliki simbol-simbol untuk menggambarkan pekerjaan yang saling terhubung, membentuk bagan proses yang jelas dan terstruktur. Standar simbol-simbol tersebut dipengaruhi oleh Institut Standar Nasional Amerika Serikat (*American National Standards Institute Inc*), yang mengatur dan menetapkan standar yang digunakan dalam *flowchart* (Abdurahman, 2019). Simbol-simbol *flowchart* disajikan pada Tabel 2.

Tabel 2. Simbol-simbol *flowchart*

Simbol	Nama	Fungsi
	<i>Procces</i>	Proses Perhitungan/proses pengolahan data
	<i>Decision</i>	Menyatakan kondisi yang menentukan 2 kemungkinan seperti iya atau tidak.
	<i>Predefined Process (Sub Program)</i>	Sub program
	<i>Input/Output Data</i>	Menghasilkan masukan/keluaran pada data
	<i>Terminator</i>	Permulaan/akhir program
	<i>On Page Connector</i>	Penghubung pada bagian <i>flowchart</i>
	<i>Flow Line</i>	Arah aliran program
	<i>Off Page Connector</i>	Penghubung pada bagian <i>flowchart</i> yang letaknya di halaman yang berbeda

Sumber: (Fitriyana & Susianto, 2018)

2.12.3 Data Flow Diagram (DFD)

Data Flow Diagram (DFD) merupakan gambaran grafis yang memvisualisasikan aliran informasi dan proses transformasi data dari *input* ke *output*. DFD digunakan untuk menggambarkan bagaimana data mengalir dalam sistem dan bagaimana data tersebut diubah dan diproses dalam setiap langkahnya (Sukanto, 2019). Data flow diagram dapat digunakan sebagai alat representasi yang fleksibel untuk menggambarkan sistem dalam berbagai level detail yang lebih mendalam. DFD memungkinkan pemodelan sistem mulai dari tingkat yang lebih tinggi hingga tingkat yang lebih rinci, sehingga memungkinkan pemahaman yang komprehensif tentang aliran data dan proses yang terjadi dalam sistem. Simbol-simbol DFD disajikan pada Tabel 3.

Tabel 3. Simbol-simbol data flow diagram

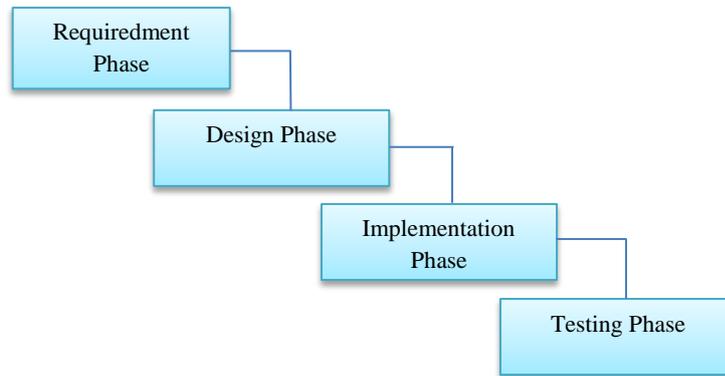
Simbol	Nama	Keterangan
	<i>External Entity</i>	Merupakan sumber atau tujuan yang dapat berupa orang/unit dari aliran data dari/ke sistem.
	Arus Data (<i>Data Flow</i>)	Aliran data dengan arah khusus dari sumber ke tujuan.
	Proses	Proses atau fungsi yang mempergunakan atau melakukan transformasi data.
	<i>Data Store</i>	Komponen penyimpanan data

Sumber: (Rosa, A., & Shalahudin, 2018)

2.13 Metode Waterfall

Waterfall merupakan metode siklus hidup pengembangan perangkat lunak yang terdiri dari serangkaian tahapan penting yang krusial dalam eksistensi perangkat lunak, dilihat dari perspektif pengembangan (Febriani et al., 2020). Metode ini menggambarkan siklus tahapan pengembangan *software* dengan urutan tahap yang jelas dan linear, mirip dengan aliran air yang terjun (*waterfall*). Fokus dari *waterfall* adalah pada perencanaan dan pengaturan jadwal dengan tenggat waktu dan anggaran yang telah ditentukan (Firdaus et al., 2022). Empat fase *waterfall* yang digunakan pada Tugas Akhir ini yaitu *requirement phase*,

design phase, implementation phase, dan testing phase. Aliran empat fase tersebut dapat dilihat pada Gambar 2.



Gambar 2. Metode *Waterfall*

Sumber: (Firdaus et al., 2020)

2.14 Artikel Ilmiah Terkait

Perancangan tugas akhir ini penulis memiliki beberapa artikel ilmiah yang menjadi bahan referensi pembuatan sistem.

1. Farabi et al., 2023. Berjudul "Perancangan Dan Implementasi *Front-end* Aplikasi Pemesanan Makanan Berbasis *Web* Di Kafe Angkringan Cahsaiki". Karya ilmiah ini dibuat karena melihat kafe angkringan Cahsaiki yang sudah banyak dikenal pada area Bekasi dan berhasil memperoleh pelanggan yang signifikan. Namun, penggunaan sistem pembayaran yang dinilai belum efisien, mengakibatkan kerumunan pelanggan menjadi sangat panjang dan menghabiskan waktu yang tidak sebentar saat pelanggan ingin memesan makanan atau minuman. Hasil dari penelitian ini yaitu sebuah aplikasi pemesanan makanan berbasis *web* yang memiliki solusi untuk meningkatkan efektivitas sistem transaksi di kafe angkringan Cahsaiki. Metode yang digunakan adalah metode SDLC.
2. R. Rinaldi et al., 2022. Berjudul "Aplikasi *Monitoring Manage Operation Project* Digitalisasi SPBU Pada PT. Telkom Akses Banjarmasin Berbasis *Web*". Karya ilmiah ini dibuat karena adanya *project* kerja sama modernisasi digital SPBU Telkom dan Pertamina dalam memantau

distribusi bahan bakar minyak (BBM) di seluruh wilayah Indonesia. PT.Telkom Akses sebagai penyelenggara di lapangan pada proyek tersebut membutuhkan manajemen data yang terstruktur. Pengelolaan data tersebut saat ini, terus-menerus dilakukan secara konvensional dengan mencatat perkembangan proyek menggunakan *Microsoft Office* dan pengisian data juga dilakukan secara konvensional berdasarkan laporan dari lapangan. Hasil dari penelitian ini adalah Aplikasi *Monitoring Manage Operation Project* Digitalisasi SPBU di PT. Telkom akses Banjarmasin Berbasis *Web* yang memiliki solusi guna mempersingkat proses pengelolaan dan pencarian data *project* disetiap spbu yang berupa catatan status, perangkat, data spbu untuk seluruh orang yang diarea unit ini. Metode yang digunakan adalah metode *Waterfall*.

3. I Ardiansyah & J Nursalim, 2021. Berjudul "Rancang Bangun Sistem *Monitoring* Pada Mesin *Extruder* DTE-4 Berbasis *Web*". Penelitian ini dilakukan karena menyadari perlunya penggunaan sistem baru dalam bidang manufaktur agar dapat mengikuti perkembangan teknologi yang terus berlanjut. Untuk menjaga hasil produksi sesuai target dan mencapai hasil yang diinginkan, penting untuk memantau parameter-parameter seperti Kecepatan Garis dan Waktu Operasi pada mesin DTE-4. Apabila terjadi variasi parameter yang seharusnya sesuai dengan spesifikasi produk, intervensi dapat dilakukan segera untuk mencegah terjadinya kesalahan. Namun, karena mesin DTE-4 berada pada jarak yang jauh dari ruangan tersebut, memantau parameter produksi memakan waktu karena harus pergi ke mesin terlebih dahulu. Hasil penelitian ini adalah sistem *monitoring* berbasis *web* dapat dimanfaatkan untuk memantau mesin.