BAB II
TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Beberapa penelitian yang telah ditelaah dengan peneliti, ada beberapa yang memiliki keterkaitan dengan penelitian yang dilakukan.

Penelitian pertama adalah penelitian yang dilakukan oleh Adjat, Rahdian, Rino dan Ispandi (2022) dengan judul "Penerapan Konsep MVC Pada Perancangan Aplikasi Sistem Informasi Persediaan Alat Kebersihan". Tujuan penelitian ini ialah untuk mengkaji penerapan konsep *Model View Controller* (MVC) dalam merancang aplikasi sistem informasi persediaan alat kebersihan. Fokus utamanya adalah memahami bagaimana konsep MVC dapat diterapkan secara efektif dalam konteks manajemen persediaan alat kebersihan, dengan penggunaan *Framework CodeIgniter* dan Metode *Agile*.

Penelitian kedua menjadi referensi peneliti adalah jurnal dengan judul "Pengembangan Sistem Informasi Presensi Menggunakan Metode *Waterfall*" (Tri, Supriyanta, Husni, 2021), yang memiliki tujuan untuk mengeksplorasi pengembangan sistem informasi presensi dengan menerapkan Metode *Waterfall*. Pada konteks ini, penelitian bertujuan menciptakan solusi yang lebih fleksibel dalam menghadapi tantangan, terutama selama pandemi *Covid-19* dan menciptakan aplikasi yang *user-friendly*, memiliki tata letak komponen yang baik, bermanfaat, dan mempermudah user dalam administrasi presensi.

Pada jurnal penelitian ketiga juga menerapkan Metode *Waterfall* yaitu "Website Monitoring *Cleaning Service* pada Bank X Palembang" (Dimas, 2022).

Penelitian ini mencoba menerapkan Metode *Waterfall* untuk solusi dalam pengembangan berbasis web saat memonitoring kebersihan di bank. Tujuannya adalah memberikan kontribusi dalam meningkatkan efisiensi dan efektivitas monitoring kebersihan di lingkungan bank tersebut.

Berikut cara membedakan antara penelitian terdahulu dengan penelitian yang akan dilakukan yaitu dengan tabel berikut.

Tabel 2.1 Tabel Penelitian 1

Judul Penelitian	**Peneliti & tahun Penelitian**
Penerapan Konsep MVC Pada Perancangan Aplikasi Sistem Informasi Persediaan Alat Kebersihan	Adjat, Rahdian, Rino dan Ispandi (2022)
Tujuan	**Ruang lingkup**
Tujuan penelitian ini ialah untuk mengkaji penerapan konsep *Model View Controller (MVC)* dalam merancang aplikasi sistem informasi persediaan alat kebersihan. Fokus utamanya adalah memahami bagaimana konsep MVC dapat diterapkan secara efektif dalam konteks manajemen persediaan alat kebersihan, dengan penggunaan *framework CodeIgniter* dan metode *agile.*	Memiliki ruang lingkup yang mencakup penelitian tentang penerapan konsep *Model View Controller* (MVC) pada perancangan aplikasi sistem informasi persediaan alat kebersihan. Menggunakan bahasa pemrograman *PHP, HTML, CSS, JQuery*, dan *MySQLi* sebagai basis data.
Hasil Penelitian	
Penelitian ini mengembangkan sebuah sistem informasi untuk perusahaan *cleaning service* guna mengelola persediaan alat kebersihan, menggunakan metode *agile* dan konsep MVC, dengan implementasi aplikasi berbasis *PHP, HTML, CSS, JQuery,* dan *MySQLi,* dilengkapi dengan pengujian *blackbox testing* dan dokumentasi UML, untuk memenuhi kebutuhan efisiensi dan efektivitas manajemen persediaan alat kebersihan. Kelebihan: 1. Meningkatkan efisiensi dan efektivitas manajemen persediaan alat kebersihan. 2. Mempercepat proses pembuatan laporan dan pengelolaan stok barang.	

3. Meminimalkan kesalahan manusia dalam pengelolaan data. 4. Memudahkan akses dan pengelolaan data secara online. 5. Menerapkan konsep *MVC* untuk meningkatkan modularitas dan reusabilitas sistem. Kekurangan: 1. Memerlukan pemeliharaan berkala untuk mencegah kerusakan sistem. 3. Tidak menyediakan aplikasi berbasis Android untuk kemudahan akses bagi pimpinan perusahaan.
Perbedaan dengan penelitian yang akan di lakukan
Meskipun penelitian yang telah dilakukan dengan penelitian yang akan dilakukan menggunakan konsep yang sama, yaitu MVC. Terdapat perbedaan dalam penggunaan *framework* dan penyediaan aplikasi. Penelitian yang telah dilakukan menggunakan *framework CodeIgniter* dan hanya dapat diakses melalui website. Sementara itu, penelitian yang akan dilakukan menggunakan *framework Laravel* dan menyediakan akses website serta akses berbasis Android.

Tabel 2.2 Tabel Penelitian 2

Judul Penelitian	**Peneliti & tahun Penelitian**
Pengembang Sistem Informasi Presensi Menggunakan Metode *Waterfall*	Tri Wahyudi, Supriyanta, Husni Faqih (2021)
Tujuan	**Ruang lingkup**
Menciptakan solusi yang lebih fleksibel dalam menghadapi tantangan, terutama selama pandemi *Covid-19* dan menciptakan aplikasi yang *user-friendly*, memiliki tata letak komponen yang baik, bermanfaat, dan mempermudah user dalam administrasi presensi dengan Metode *Waterfall.*	Memiliki ruang lingkup yang terfokus pada pengembangan sistem informasi presensi dengan menggunakan metode pengembangan perangkat lunak *Waterfall*. Dalam proses pengembangannya, jurnal ini menjelaskan secara rinci tahapan-tahapan metode *Waterfall*, mulai dari analisis kebutuhan, desain perangkat lunak, implementasi, pengujian unit, integrasi, hingga pemeliharaan. Sistem yang dikembangkan dapat diaplikasikan baik di kantor *(Working From Office/WFO)* maupun di rumah *(Working From Home/WFH)*, menekankan fleksibilitas presensi sesuai jadwal yang ditentukan. Teknologi yang digunakan mencakup pengembangan berbasis web dengan *PHP, framework Codeigniter*, dan *DBMS MariaDB.*

Hasil Penelitian
Menghasilkan aplikasi Sistem Informasi Presensi yang *user-friendly*, dengan tata letak komponen yang baik, bermanfaat, dan mempermudah pengguna dalam administrasi presensi. Hasil pengujian beta menunjukkan nilai prosentase sebesar 96.5%, menunjukkan bahwa aplikasi telah sesuai dengan tujuan pengembangan. Kelebihan : • *User Friendly*: Aplikasi dinilai mudah digunakan oleh pengguna. • Tampilan Menarik: Aplikasi memiliki tampilan yang menarik. • Kemudahan Navigasi: Pengguna merasakan kemudahan navigasi dalam aplikasi. • Penyajian Informasi: Aplikasi memberikan penyajian informasi yang sesuai dengan kebutuhan pengguna. Kekurangan : • Terdapat kebutuhan admin untuk dapat mengedit jadwal kerja karyawan, namun pada desain *Entity-Relationship Diagram* (ERD), tidak ada atribut waktu yang menunjukkan jadwal kerja. Sebaliknya pada desain tampilan, terdapat sebuah formulir yang menyediakan opsi untuk mengedit waktu jam kerja. Hal ini menciptakan ketidaksesuaian antara kebutuhan fungsional dan representasi struktural dalam desain sistem. Untuk meningkatkan konsistensi, disarankan untuk memasukkan elemen waktu atau jam kerja ke dalam ERD agar sesuai dengan kemampuan sistem dalam mengelola dan menampilkan informasi jadwal kerja karyawan.
Perbedaan dengan penelitian yang akan di lakukan
Meskipun penelitian yang telah dilakukan dan penelitian yang akan dilakukan menggunakan metode yang sama, yaitu Metode *Waterfall*, terdapat perbedaan signifikan dalam fokus objek penelitian. Penelitian yang telah dilakukan difokuskan pada pengembangan sistem informasi presensi untuk pegawai, pelajar, atau mahasiswa. Sementara itu, penelitian yang akan dilakukan memiliki fokus yang berbeda, yakni pada pengembangan sistem informasi kinerja *cleaning service* di PT. Mega Karya Bersinar.

Tabel 2.3 Tabel Penelitian 3

Judul Penelitian	Peneliti & tahun Penelitian
Website Monitoring *Cleaning Service* pada Bank X Palembang	Dimas, 2022
Tujuan	**Ruang lingkup**
Tujuannya adalah memberikan kontribusi dalam meningkatkan efisiensi dan efektivitas monitoring kebersihan di lingkungan bank tersebut.	1. Penelitian dilakukan di Bank X Palembang

	2. Pembuatan web monitoring pegawai *cleaning service* dikelola oleh admin, pegawai *cleaning service*, dan *supervisor.* 3. Proses yang dapat dilakukan oleh masing-masing peran: - Pegawai *cleaning service* dapat melakukan input kegiatan harian, menambahkan foto, serta menginput jika ada barang yang habis. - Admin dapat menambah, menghapus data pegawai, serta mengupdate data dan menambahkan stock barang yang habis. - *Supervisor* dapat melihat detail kegiatan harian yang telah dilakukan oleh admin dan *cleaning service.* 4. Alat pengembangan yang digunakan adalah *Data Flow Diagram (DFD), Entity Relationship Diagram (ERD),* dan *Flowchart*. 5. Aplikasi berbasis website ini dikembangkan menggunakan Bahasa pemrograman PHP dan database *MYSQL*.
Hasil Penelitian	
Menghasilkan sebuah website monitoring *cleaning service* pada Bank X, hal tersebut memudahkan *cleaning service* dalam mengirimkan laporan monitoring pekerjaan, mempermudah admin dalam rekap laporan pekerjaan, serta memudahkan *supervisor* dalam memantau pekerjaan. Kelebihan : • Meningkatkan efisiensi dan efektifitas kinerja *cleaning service* pada Bank X • Dengan menggunakan alat pengembangan seperti *Data Flow Diagram (DFD), Entity Relationship Diagram (ERD),* dan *Flowchart*, pengelolaan data dapat dilakukan secara terstruktur dan sistematis, memudahkan dalam analisis dan pemeliharaan sistem. • Penggunaan web monitoring dapat meningkatkan transparansi dalam pengelolaan kebersihan dan persediaan barang-barang kebersihan di Bank X Palembang. Hal ini dapat mengurangi potensi kesalahan atau kecurangan dalam pencatatan dan pelaporan. Kekurangan : • Pegawai *cleaning service*, admin, dan *supervisor* mungkin memerlukan waktu untuk beradaptasi dengan sistem baru. Selain itu, mereka juga mungkin memerlukan pelatihan untuk menggunakan sistem ini secara efektif. • Pembangunan dan pemeliharaan aplikasi berbasis website memerlukan biaya yang signifikan, terutama dalam hal pengembangan awal, pemeliharaan rutin,	

dan peningkatan sistem sesuai kebutuhan yang mungkin timbul dari waktu ke waktu. • Dalam jurnal ini dijelaskan bahwa peneliti hanya menggunakan *PHP* dan *MYSQL,* tanpa menggunakan *framework.* Hal tersebut cukup berpengaruh untuk kedepannya. Disaat sistem hendak dikembangkan, terkadang struktur kode yang dibangun tanpa *framework* dapat menjadi sulit untuk dikelola dan diperluas.
Perbedaan dengan penelitian yang akan di lakukan
Penelitian yang telah dilakukan dan penelitian yang akan dilakukan menggunakan website dengan tema yang sama yaitu mengenai monitoring kinerja *cleaning service.* Ada perbedaan pada sistem yang akan dibuat, pada penelitian yang telah dilakukan hanya menggunakan bahasa pemrograman *PHP* tanpa *framework.* Sementara itu, penelitian yang akan dilakukan menggunakan *Framework Laravel.*

2.2 Sistem Informasi

Menurut Widianti dalam Utomo dkk (2017:77) menyimpulkan bahwa sistem informasi adalah suatu sistem di dalam sebuah organisasi untuk keperluan mengelola transaksi harian, mendukung operasi, bersifat eksekutif dan rencana kegiatan dari organisasi serta menyediakan laporan-laporan yang diperlukan untuk pihak luar.

Hutahaean (2015:13) menjelaskan bahwa sistem informasi adalah sebuah sistem dalam organisasi yang mengintegrasikan pengelolaan transaksi harian, mendukung operasi, manajerial, dan kegiatan strategis organisasi, serta menyediakan laporan-laporan untuk pihak luar yang membutuhkan.

Pandangan kedua penulis ini menunjukkan kesamaan konsep dalam menggambarkan peran sistem informasi di dalam suatu organisasi. Keduanya menggarisbawahi pentingnya sistem informasi dalam mendukung berbagai aspek kegiatan organisasi, baik yang bersifat operasional, manajerial, eksekutif, maupun strategis. Oleh karena itu, dapat disimpulkan bahwa sistem informasi memiliki

peran yang sangat integral dalam mengelola informasi, mendukung pengambilan keputusan, dan menyediakan informasi yang dibutuhkan oleh pihak luar untuk keperluan tertentu.

2.3 Sistem Informasi *Cleaning Service*

Sistem Informasi *Cleaning Service* adalah platform atau aplikasi perangkat lunak yang dirancang khusus untuk membantu dalam pengelolaan dan pemantauan operasional layanan kebersihan atau *cleaning service*. Tujuan utama dari sistem ini ialah untuk mempermudah pembuatan laporan.

2.4 Efisiensi dan Efektivitas

Efektivitas merupakan suatu istilah yang berasal dari kata "efektif". Tim Prima Pena (2001), mengemukakan bahwa *effective* adalah berhasil, manjur, tepat.

Pengertian tersebut sejalan dengan rumusan yang dikemukan oleh The Liang Gie (2001) yaitu Efektivitas merupakan keadaan yang mengandung pengertian mengenai terjadinya sesuatu efek maupun akibat yang dikehendaki. Apabila seseorang melakukan suatu perbuatan dengan tujuan tertentu yang memang dikehendaki, maka orang tersebut dikatakan efektif apabila menimbulkan suatu akibat.

Ulum (2008) menyatakan bahwa efektivitas berkaitan erat dengan pencapaian tujuan atau target kebijakan (hasil guna). Efektivitas merupakan hubungan antara keluaran dan tujuan yang ingin dicapai. Suatu kegiatan operasional dianggap efektif jika dalam prosesnya berhasil mencapai tujuan akhir kebijakan *(spending wisely).*

Efisiensi dapat diartikan sebagai rasio antara suatu aktivitas dan hasilnya. Efisiensi adalah indikator keberhasilan yang dinilai berdasarkan besarnya biaya yang diperlukan untuk mencapai hasil dari aktivitas yang dilakukan.

Pengertian efisiensi menurut Mulyamah (1987), efisiensi merupakan ukuran dalam membandingkan suatu rencana penggunaan masukan dengan penggunaan yang terealisasikan.

Menurut S.P. Hasibuan (1984), dalam sebuah kutipan yang mengacu pada H. Emerson, efisiensi adalah tentang membandingkan input (masukan) dan output (luaran). Konsep ini menekankan pentingnya melakukan sesuatu dengan optimal, menghasilkan hasil terbaik tanpa membuang waktu yang berlebihan dalam prosesnya. Penting untuk diingat bahwa sebuah kegiatan yang efektif belum tentu efisien, demikian pula sebaliknya.

2.5 UML *(Unifield Modeling Language)*

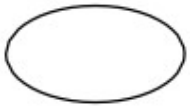
UML adalah alat/ model yang digunakan untuk merancang pengembangan perangkat lunak berbasis objek. UML menyediakan standar untuk penulisan *blueprint* sistem, yang mencakup konsep proses bisnis, definisi kelas-kelas dalam bahasa pemrograman tertentu, skema database, dan komponen-komponen yang diperlukan dalam sistem perangkat lunak (Sonata, 2019).

2.5.1 *Use Case*

Peneliti menggunakan *Diagram Use Case* yang merupakan pemodelan untuk perilaku (*behavior*) sistem informasi yang akan dibuat. *Use Case* menjelaskan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi

yang akan dibuat. Dalam arti lain, *Use Case* dipergunakan untuk mengetahui fungsi apa saja yang berhak menggunakan fungsi-fungsi tersebut (Hakam, dkk, 2022). Simbol-simbol yang ada pada *use case* dapat dilihat pada tabel 2.1.

Tabel 2.4 Tabel simbol *Use Case*

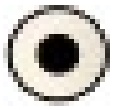
Simbol	Nama	Keterangan
	*Use Case*	Berfungsi menggambarkan fungsionalitas dari sistem sebagai unit yang bisa bertukar pesan antar aktor dan unit
	Aktor	Merupakan pengguna dari sistem yang harus ditentukan berdasarkan pembagian tugas dalam peranan di sistem yang akan dibuat. Aktor tersebut dapat berinteraksi dengan *use case* yang ada.
	Association	Menghubungkan antara objek yang satu dengan lainnya.
	Generalization	Hubungan dimana objek anak

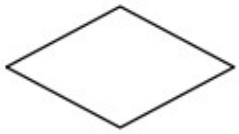
		(*descendent)* berbagai perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor).*
	*Include*	Relasi *use case* tambahan ke sebuah *use case* terjadi ketika *use case* tambahan memerlukan *use case* tersebut untuk menjalankan fungsionalitasnya atau sebagai syarat agar *use case* tersebut dapat dijalankan.
	*Extend*	Relasi use case tambahan dengan sebuah use case terjadi ketika use case tersebut dapat berdiri sendiri meskipun tanpa use case tambahan.

2.5.2 *Activity Diagram*

Activity diagram menggambarkan sebuah *workflow* (aliran kerja) atau aktivitas dari suatu sistem maupun menu yang ada pada perangkat lunak, atau bisa diartikan bahwa diagram aktivitas menggambarkan aktivitas sistem bukan yang dilakukan aktor, dan aktivitas tersebut dilakukan oleh sistem (Hakam, dkk, 2022). Berikut merupakan simbol-simbol yang digunakan dalam *activity diagram* :

Tabel 2.5 Tabel simbol *Activity Diagram*

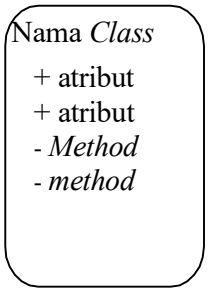
Simbol	Nama	Keterangan
	Start	Status awal aktivitas sistem.
	End	End adalah akhir dari suatu aktivitas, biasanya berada di bagian bawah.
	Activity	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
	*Swimlane*	Bertujuan memberikan informasi siapa saja yang sedang mengakses aktivitas.

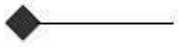
	*Decision*	Digunakan untuk penggambaran pilihan pengambilan *true/false.*

2.5.3 *Class Diagram*

Class diagram merupakan suatu diagram yang menggambarkan struktur sistem dari segi pendefenisian kelas-kelas yang dibuat untuk membangun sistem. *Class* adalah kumpulan objek-objek dengan dan yang mempunyai struktur umum, behavior umum, relasi umum, dan semantic/kata yang umum. *Class-class* ditentukan/ditemukan dengan cara memeriksa objek-objek dalam *sequence* diagram dan *collaboration* diagram (Hakam, dkk, 2022).

Tabel 2.6 Tabel simbol *Class Diagram*

Nama	Keterangan	Simbol
Dependency	Penggunaan *dependency* digunakan untuk menunjukkan operasi pada suatu *class* yang menggunakan *class* yang lain.	
Class	*Class* merupakan elemen dasar dalam pemrograman berorientasi objek. *Class* dapat digambarkan sebagai sebuah kotak yang terbagi menjadi tiga bagian. Bagian atas mencantumkan nama *class*, bagian tengah berisi atribut-atribut *class*, dan bagian bawah mendefinisikan metode-metode yang dimiliki oleh *class* tersebut.	

Association	Sebuah asosiasi merupakan hubungan paling umum antara dua kelas dan dilambangkan oleh garis yang menghubungkan keduanya. Garis ini bisa mewakili berbagai jenis hubungan dan juga dapat menampilkan aturan *multiplicity* dalam sebuah hubungan.	
Composition	Apabila sebuah *class* tidak bisa berdiri sendiri dan harus merupakan bagian dari *class* yang lain, maka *class* tersebut memiliki relasi *Composition* terhadap *class* tempat dia bergantung tersebut.	

2.6 Database

Dalam era perkembangan teknologi, media menjadi kunci utama untuk memberikan informasi dengan cepat dan menyeluruh melalui perantara jaringan *internet.* Seiring dengan hal tersebut, konsep website menjadi sangat relevan sebagai media penyampaian informasi. Menurut Rahmadi (2013), website atau situs adalah kumpulan halaman yang saling terkait, seringkali dilengkapi dengan berbagai jenis berkas seperti gambar, video, dan lainnya. Ippho Santoso juga mengklasifikasikan website menjadi 2 kelompok utama: Website Statis dan Website Dinamis.

1. Website Statis

   Web yang memiliki tampilan halaman tidak berubah-ubah.

2. Website Dinamis

   Web dengan struktur yang sengaja ditujukan untuk melakukan update sesering mungkin.

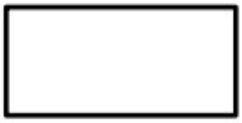
2.6.1 ERD

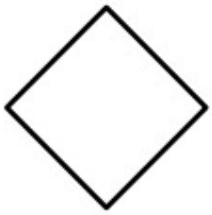
ERD *(Entity Relationship Diagram)* adalah suatu diagram yang digunakan untuk menunjukkan sebuah struktur data pada sistem informasi atau basis data. ERD menyajikan entitas (objek atau konsep yang memiliki data yang disimpan) dan hubungan antara entitas tersebut. ERD digunakan sebagai alat visual untuk memodelkan dan merancang struktur data dalam sebuah proyek pengembangan perangkat lunak atau basis data.

Menurut Pulungan, dkk, (2023) Entitas adalah sesuatu yang bisa dijelaskan dengan data dan mewakili objek nyata atau konsep lain. Dalam sistem manajemen basis data, entitas yang kuat dan saling bergantung akan dihubungkan dengan garis untuk menunjukkan hubungan mereka. Contohnya adalah keanggotaan, yang bergantung pada entitas lain dan menunjukkan bagaimana data tersebut saling terkait.

Di dalam ERD terdapat 3 elemen dasar, yaitu entitas, atribut, dan relasi. Berikut penjelasan elemen dasar tersebut.

Tabel 2.7 Tabel *ERD*

Simbol	Nama	Keterangan
	Entitas	Entitas adalah objek dalam database yang bisa berupa manusia, tempat, benda, atau kondisi yang terkait dengan

		data yang diperlukan.
	Atribut	Sebuah entitas harus memiliki *primary key* sebagai identifikasi unik dan atribut deskriptif. Atribut biasanya terletak di dalam tabel entitas, tetapi juga bisa berada di tabel yang terpisah.
	Relasi	Relasi dalam ERD *(Entity-Relationship Diagram)* adalah keterkaitan antara dua atau lebih entitas. Ada beberapa jenis relasi dalam ERD: Satu ke Satu *(One to On*e): Satu anggota dari satu entitas terhubung dengan satu anggota dari entitas lain. Satu ke Banyak *(One to Many):* Satu anggota dari satu

		entitas terhubung dengan beberapa anggota dari entitas lain. Banyak ke Banyak *(Many to Many):* Beberapa anggota dari satu entitas terhubung dengan beberapa anggota dari entitas lain

2.6.2 DBMS

DBMS atau *Database Management System* (Sistem Manajemen Basis Data) adalah perangkat lunak yang telah dirancang untuk keperluan pengelolaan, menyimpan, mengorganisir, dan menyediakan akses terhadap data dalam suatu basis data. Tujuan utama dari DBMS adalah menyediakan cara efisien untuk menyimpan, mengambil, dan memanipulasi data, serta menjaga integritas data dan keamanan akses.

MySQL adalah perangkat lunak yang termasuk dalam kategori DBMS (*Database Management System*) dan memiliki sifat *Open Source*. Keberadaan sifat *Open Source* pada *MySQL* menyiratkan bahwa perangkat lunak ini dilengkapi dengan *source code* atau kode sumber yang digunakan untuk membuat *MySQL*. Dengan kata lain, pengguna memiliki akses terhadap kode-kode yang membentuk struktur dan fungsionalitas *MySQL* (Jayanti, 2022).

Dalam konteks *Open Source*, ini tidak hanya berarti pengguna dapat melihat dan memahami cara *MySQL* beroperasi, tetapi juga dapat memodifikasi dan mendistribusikan kembali kode tersebut sesuai dengan lisensi *Open Source* yang diterapkan. Dengan adanya *source code* terbuka, komunitas pengembang dapat berkontribusi pada pengembangan dan peningkatan *MySQL*.

2.7 Bahasa Pemrograman

PHP (akronim dari PHP: *Hypertext Preprocessor*) adalah bahasa pemrograman yang digunakan untuk mengembangkan situs web dinamis dan aplikasi web. Berbeda dengan HTML yang hanya menampilkan konten statis, PHP dapat berinteraksi dengan database, file, dan folder, sehingga memungkinkan untuk menampilkan konten dinamis di situs web. Contoh aplikasi web yang bisa dibuat dengan PHP termasuk blog, toko online, CMS, forum, dan situs jejaring sosial. Program PHP ditulis dalam file teks biasa dengan ekstensi *".php"* (Yuliano, 2007).

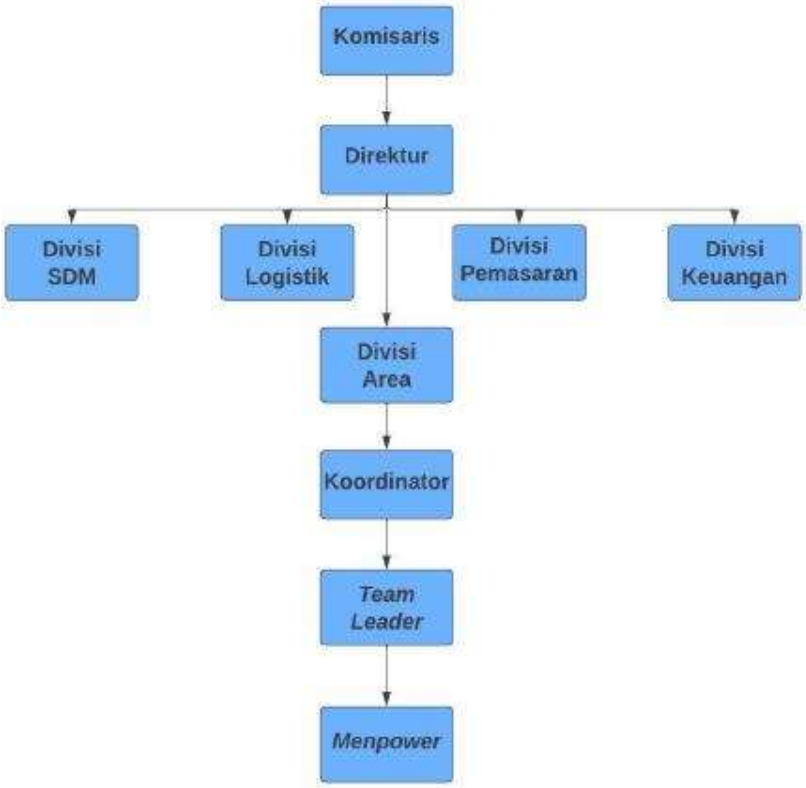
2.8 Gambaran Umum Obyek Penelitian

PT. Mega Karya Bersinar merupakan perusahaan yang bergerak di bidang jasa outsourcing, khususnya dalam penyediaan tenaga kerja. Di tengah era globalisasi yang berkembang pesat saat ini, perusahaan ini memusatkan perhatian pada kegiatan inti dan mengandalkan outsourcing untuk mengelola kegiatan pendukung. PT. Mega Karya Bersinar menawarkan layanan seperti Cleaning Service, Suplier, Konstruksi, dan Transportasi, dengan fokus utama pada Cleaning Service di daerah Malang.

Perusahaan telah berhasil memenangkan sejumlah tender di berbagai kota, termasuk Malang, Surabaya, Sidoarjo, Nganjuk, Kediri, Madiun, dan Tulungagung. Mereka memiliki izin operasional dari Dinas Tenaga Kerja, Transmigrasi, dan Kependudukan Provinsi Jawa Timur dengan nomor: 560/1958/106.04/2014. Kantor pusat PT. Mega Karya Bersinar berlokasi di Danau Sentani Utara VIII H3/H8 Kota Malang, sementara untuk operasional di area Malang, mereka berada di Poltekkes Malang dan Universitas Brawijaya.

Dengan moto "MKB, Semangat Kerja Bersih Yes", PT. Mega Karya Bersinar terus membangun eksistensi mereka di daerah Malang dengan komitmen untuk menjaga kualitas kinerja karyawan mereka.Ada sekitar 300 orang yang bekerja menjadi *cleaning service* di PT. Mega Karya Bersinar.

- Struktur organisasi perusahaan sebagai berikut.

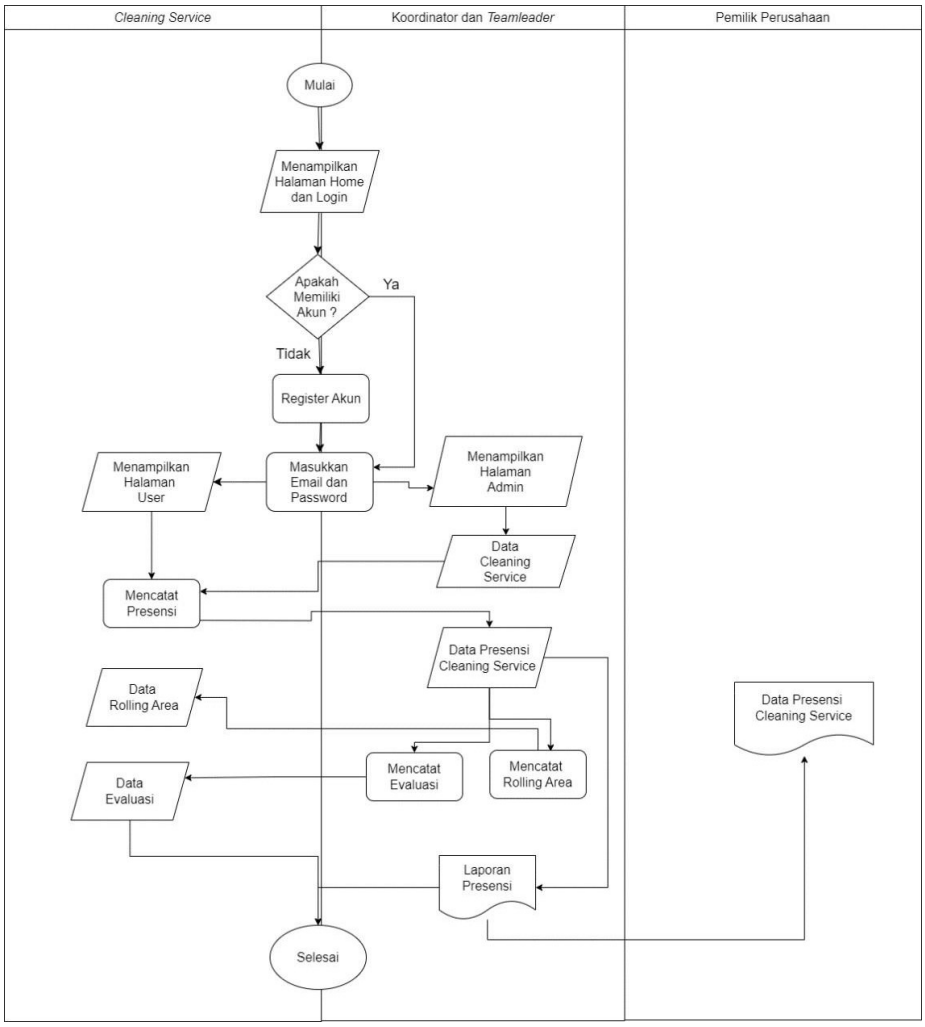


Gambar 2.1 Struktur Organisasi Perusahaan

- Sistem yang berjalan saat ini di perusahaan

  Sebelumnya, perusahaan menerapkan sistem manual untuk proses manajemen presensi dan evaluasi kinerja.

Sebagai hasil dari Penelitian Kerja Lapangan (PKL) yang peneliti jalani di perusahaan tersebut, terjadi transformasi signifikan dalam pengelolaan sistem menjadi sebuah sistem berbasis website. Berikut desain proses sistem hasil dari PKL.



Gambar 2.2 Alur Proses Sistem Penelitian Sebelumnya

Pada alur di atas, sebelum masuk ke halaman user *(cleaning service)* atau admin (Koordinator dan *Team* Leader), perlu melakukan *login* terlebih dahulu. Apabila belum memiliki akun maka perlu dilakukan registrasi.

Registrasi akun dapat dilakukan oleh Koordinator, *Team Leader,* dan *cleaning service.*

Di halaman *cleaning service,* terdapat fitur untuk presensi, melihat data *rolling* area dan evaluasi. Sedangkan di halaman admin (Koordinator dan *Team Leader*) terdapat fitur mengenai data *cleaning service*, melihat presensi (rekapan presensi), fitur mencatat evaluasi dan *rolling* area, serta untuk fitur untuk mencetak laporan presensi. Laporan presensi tersebut akan diserahkan kepada pemilik perusahaan PT. Mega Karya Bersinar dalam bentuk *hardfile*.

Pada bagian evaluasi kinerja terdapat peringatan atas pelanggaran yang dilakukan *cleaning service.* Dimana *cleaning service* melakukan kesalahan/ mendapat evaluasi, namun evaluasi tersebut tidak dikerjakan maksimal 3 hari setelah evaluasi dikeluarkan, maka *cleaning service* mendapat peringatan secara lisan maupun tertulis. Apabila selama sebulan *cleaning service* mendapat 2x evaluasi dan tidak dikerjakan maksimal 3 hari setelah evaluasi dikeluarkan, maka *cleaning service* mendapat SP 1. Begitu pun seterusnya hingga SP 3, *cleaning service* akan dimusyawarahkan dan perusahaan mengeluarkan *cleaning service* tersebut.

Perubahan dari sistem manual ke sistem website ini, melibatkan pengembangan sistem yang mencakup aspek presensi hingga evaluasi kinerja karyawan. Implementasi sistem berbasis website memberikan kemudahan akses dan pemantauan secara efisien, membantu meningkatkan

produktivitas, dan meminimalkan risiko kesalahan manusia yang mungkin terjadi pada sistem manual sebelumnya.

Dalam tugas akhir ini, fokus akan diberikan pada pengembangan lebih lanjut terhadap sistem yang telah ada. Tujuan utama pengembangan ini adalah memberikan kemudahan pada pembuatan laporan, sejalan dengan kebutuhan perusahaan yang terus berkembang.