

BAB II

KAJIAN PUSTAKA

A. Kajian Empires

1. Sistem

Sistem merupakan suatu kesatuan yang terdiri atas komponen atau elemen yang saling berhubungan dan bekerja sama menjalankan fungsi guna mencapai tujuan yang telah ditetapkan secara efektif dan efisien (Deni Noperdiana, 2026). Penerapan sistem mampu meningkatkan produktivitas kerja, mengurangi biaya operasional, mempercepat proses pengambilan keputusan, serta meningkatkan respons organisasi terhadap perubahan lingkungan (Program et al., 2024). Dengan demikian penggunaan sistem mampu bertujuan menghasilkan informasi yang lengkap agar dapat menjadi bahan pertimbangan bagi seseorang dalam proses pengambilan keputusan (Nurhayati et al., 2023).

Berdasarkan berbagai pendapat tersebut, dapat disimpulkan bahwa sistem merupakan suatu elemen yang saling terintegrasi. Penerapan nya memberikan manfaat dalam meningkatkan kinerja dengan mengoptimalkan produktivitas, serta membantu organisasi beradaptasi terhadap perubahan lingkungan. Selain itu, sistem berperan dalam menghasilkan informasi yang lengkap dan bermanfaat sehingga dapat dijadikan sebagai dasar dalam mendukung pengambilan keputusan yang lebih tepat.

2. Monitoring

Monitoring merupakan kegiatan yang dilakukan secara sistematis dan berkelanjutan untuk mengumpulkan serta menganalisis informasi berdasarkan indikator yang telah ditetapkan, sehingga perkembangan suatu program dapat diketahui dan menjadi dasar dalam melakukan perbaikan apabila diperlukan (Suherman and Narmianti, 2022). Di samping itu, *monitoring* bisa dilihat sebagai bentuk pengawasan atas pelaksanaan kegiatan atau program supaya tetap sejalan dengan rencana dan tujuan awal (Nasril and Jupardi, 2024). Menurut (Hapsari et al., 2023) monitoring juga bertujuan untuk mengamati perkembangan pelaksanaan program, mengidentifikasi serta mengantisipasi permasalahan yang muncul sedini mungkin, sehingga dapat dilakukan tindakan perbaikan guna mencapai target yang telah ditetapkan.

Singkatnya, *monitoring* merupakan gabungan proses pemantauan dan evaluasi yang berjalan secara sistematis serta berkesinambungan, gunanya memastikan suatu kegiatan tetap sejalan dengan tujuan yang sudah ditetapkan sembari mengetahui bagaimana perkembangan pelaksanaannya dari waktu ke pelaksanaannya.

3. Target Penjualan

Target penjualan merupakan jumlah produk atau layanan yang ditetapkan perusahaan untuk dicapai dalam periode tertentu. Penentuan target tersebut dilakukan berdasarkan analisis strategi pemasaran yang

dinilai mampu mendukung pencapaian tujuan penjualan (Pratama and Rachman, 2023). Penerapan strategi pemasaran bertujuan mendukung perusahaan melalui penerapan konsep segmentasi, penentuan pasar sasaran (*targeting*), dan penetapan posisi produk (*positioning*) (Masan et al., 2023). Hal tersebut terjadi agar karyawan memiliki acuan yang pasti mengenai hasil yang harus dicapai, sehingga pekerjaan dapat diselesaikan dengan lebih efektif baik dari segi kualitas maupun kuantitas (Fajriah Mecca & Wiradendi Wolor, 2026) .

Jadi, bisa dibilang target penjualan itu sasaran yang mesti dikejar perusahaan dalam kurun waktu tertentu, sekaligus berfungsi jadi pedoman yang harus dicapai sehingga mampu mendukung perusahaan dalam menentukan pasar.

4. Sales

Penjualan (*sales*) atau *sales* pada dasarnya adalah karyawan yang memiliki tugas untuk memasarkan serta menjual produk atau layanan perusahaan kepada konsumen (Moch. Ubaydillah Ramadani, 2026). Tim sales dan marketing merupakan tim kunci dalam perusahaan dimana mereka menjadi garda depan dalam proses yang mendatangkan income dan profit bagi perusahaan (Melati Pandam Mandansari & Markus Idulfilastri, 2022). Selain itu sales juga berperan dalam membangun serta memperluas jaringan pelanggan guna meningkatkan kinerja penjualan dan memahami kebutuhan pelanggan dengan lebih baik (Eva Pradhita et al., 2025).

Dari semua penjelasan tadi, jelas bahwa bahwa sales merupakan bagian penting perusahaan yang bertugas memasarkan dan menjual produk kepada konsumen serta berperan dalam meningkatkan penjualan, keuntungan, dan hubungan dengan pelanggan.

5. *Android*

Android bisa dijelaskan sebagai sistem operasi berbasis Linux yang menyediakan software stack berupa sistem operasi, middleware, dan aplikasi utama dalam ekosistem open source (Rizki et al., 2026). *Android* menyediakan lingkungan pengembangan yang fleksibel dan didukung oleh komunitas yang besar, sehingga menjadi platform yang efektif untuk membangun aplikasi mobile yang mudah dioperasikan dan memiliki kinerja yang responsif (Harahap, 2026). Pemanfaatan aplikasi berbasis *Android* telah diterapkan pada berbagai bidang, termasuk e-commerce, pendidikan, dan administrasi. Kehadiran teknologi ini mampu membantu menyederhanakan proses kerja, meningkatkan efektivitas pelaksanaan aktivitas, serta memudahkan pengguna dalam memperoleh informasi dan layanan secara lebih cepat (Assaidah et al., 2026).

Ringkasnya, *Android* merupakan sistem operasi berbasis Linux yang bersifat open source dan banyak digunakan dalam pengembangan aplikasi mobile. *Android* juga mampu mendukung pembuatan aplikasi yang responsif dan bermanfaat dalam meningkatkan efektivitas, efisiensi, serta kemudahan akses informasi di berbagai bidang.

6. Flutter

Flutter adalah *framework open-source* garapan Google yang dipakai untuk membangun aplikasi *mobile* dengan tampilan antarmuka yang responsif. Berbekal satu basis kode saja (*single codebase*), Flutter bikin proses pengembangan aplikasi jadi lebih hemat waktu karena bisa langsung jalan di *platform* Android maupun Ios (Setiawan et al., 2025). Sebagai *framework* kekinian, Flutter dibekali banyak *widget* yang memudahkan *developer* merancang antarmuka pengguna yang interaktif, luwes, dan performanya tinggi. Popularitasnya juga terus naik berkat kemampuannya menghasilkan aplikasi responsif serta mendukung pengembangan lintas *platform* secara maksimal (Fau, 2024). Selain itu, Flutter pun bisa terintegrasi dengan layanan *backend*, seperti Firebase, sehingga pengelolaan data *real-time* bisa dilakukan lewat *library* khusus (Darma, 2026).

Jadi, dapat disimpulkan Flutter adalah *framework* yang mumpuni untuk pengembangan aplikasi *mobile*, sebab memungkinkan pembuatan aplikasi Android dan iOS cukup dengan satu basis kode (*single codebase*). Flutter menawarkan performa yang baik, tampilan yang responsif, sekaligus mendongkrak efisiensi proses pengembangan.

7. MySql

MySQL termasuk salah satu sistem manajemen basis data relasional (*Relational Database Management System/RDBMS*) yang

populer dipakai dalam pengembangan sistem informasi karena bersifat *open-source*, fleksibel, dan kinerjanya terbilang bagus. MySQL dipakai untuk menyimpan, mengelola, dan mengolah data secara terstruktur sehingga memudahkan pengguna mengakses informasi secara cepat dan akurat (Saputra & Fitri, 2026). Dalam penerapannya, MySQL kerap diintegrasikan dengan bahasa pemrograman seperti PHP untuk membangun aplikasi berbasis *web* yang dinamis dan interaktif. MySQL juga dikenal sebagai sistem basis data yang bisa diandalkan untuk menyimpan dan mengatur data secara rapi lewat tabel dan relasi antar data (Deda & Irawan, 2026). Pemakaian MySQL bikin pengolahan data lebih efisien, kesalahan pencatatan berkurang, dan informasi yang tersedia jadi lebih terintegrasi serta bisa diakses *real-time*.

Performa MySQL dapat ditingkatkan melalui penerapan teknik database tuning yang mencakup optimasi konfigurasi, struktur basis data, dan query. Penerapan teknik tersebut bertujuan untuk meningkatkan kinerja sistem, mempercepat akses data, mengurangi waktu respons, serta mengoptimalkan penggunaan sumber daya (Andriadi & Al-Faqih, 2026). Berkat teknik-teknik itu, sistem berbasis MySQL sanggup menangani beban pengguna yang tinggi sambil tetap menjaga performa tetap stabil.

Dari uraian di atas, MySQL bisa disimpulkan sebagai sistem manajemen basis data yang dipakai untuk menyimpan mengelola, dan

mengolah data secara terstruktur. Alasan MySQL banyak dipilih karena gampang dipakai, larinya cepat, serta mendukung pengembangan berbagai aplikasi berbasis *web* maupun *mobile*.

8. *System Usability Scale (SUS)*

System Usability Scale (SUS) merupakan ukur usability yang digunakan untuk mengevaluasi tingkat kemudahan dan kenyamanan penggunaan sistem berdasarkan sudut pandang pengguna sebagai pihak yang berinteraksi langsung dengan sistem (Abrory et al., 2025). Melalui metode ini, penilaian terhadap pengalaman dan persepsi pengguna dapat dilakukan untuk mengetahui tingkat usability sistem serta memastikan bahwa sistem yang dikembangkan mampu memberikan manfaat, kemudahan, dan kepuasan dalam penggunaannya (Halawa and Zebua, 2026). Rendahnya tingkat usability pada suatu sistem berpotensi menimbulkan kesulitan bagi pengguna, mulai dari tingginya tingkat kesalahan penggunaan hingga berkurangnya efisiensi dalam menyelesaikan tugas. Dampak tersebut dapat mengurangi kualitas sistem dan menghambat pencapaian tujuan pengembangannya. Oleh karena itu, evaluasi usability diperlukan untuk memastikan sistem dapat digunakan dengan mudah serta sesuai dengan kebutuhan pengguna (Al Qossam et al., 2026).

Kesimpulannya, berdasarkan uraian tersebut, *dapat* disimpulkan bahwa *System Usability Scale (SUS)* merupakan metode evaluasi usability yang sederhana, cepat, dan reliabel untuk mengukur

tingkat kemudahan penggunaan sistem serta memastikan kesesuaiannya dengan kebutuhan pengguna.

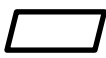
9. *FlowChart*

Flowchart atau bagan alir adalah bentuk visualisasi yang menunjukkan urutan langkah atau proses dalam suatu sistem untuk membantu memahami alur kerja secara lebih jelas dan terstruktur (Barutu & Susilawati, 2024). Pemahaman mengenai algoritma dan flowchart merupakan kompetensi penting yang harus dimiliki siswa karena keduanya menjadi landasan dalam mengembangkan kemampuan berpikir logis dan menyusun program komputer (Rochim et al., 2026). Di samping itu, *flowchart* bisa diartikan sebagai bagan yang memperlihatkan alur program atau prosedur sistem secara logis, sekaligus berfungsi sebagai alat komunikasi dan dokumentasi dalam perancangan sistem (Pramudya and Sondang, 2026).

Kesimpulannya, *flowchart* adalah diagram yang dipakai menggambarkan alur proses atau logika sistem secara sistematis. Kehadirannya memudahkan pemahaman atas tiap tahapan proses, berfungsi sebagai media dokumentasi, dan turut membantu dalam perancangan maupun pengembangan sistem.

Tabel 2.1 Keterangan *FlowChart*

Simbol	Nama	Keterangan
	<i>Terminator</i>	Menyatakan permulaan atau akhir suatu program

	<i>Process</i>	Menyatakan suatu tindakan (proses) yang dilakukan komputer
	<i>Document</i>	Dokumen atau laporan berupa print out
	<i>Decision</i>	Keputusan atau sub-point. Garis yang terhubung dengan bentuk decision merujuk pada situasi-situasi yang berbeda sesuai dengan keputusan yang digambarkan
	<i>Data</i>	Menyatakan masukan (input) dan keluaran (output) data di dalam proses atau sistem
	<i>On Page Connector</i>	Penghubung alur dalam halaman yang sama
	<i>Off Page Connector</i>	Penghubung alur dalam halaman yang berbeda
	<i>Flow</i>	Arah alur dalam prosedur atau sistem

Sumber: (Nazila, 2023)

10. UML (*Unified Modeling Language*)

Unified Modeling Language (UML) adalah media pemodelan sistem karena telah diakui sebagai standar dalam menggambarkan kebutuhan, alur proses, serta struktur suatu sistem secara terstruktur. (Simamora et al., 2026). UML telah diakui sebagai salah satu standar dalam pemodelan perangkat lunak karena mampu memberikan representasi sistem yang jelas dan mudah dipahami. Meskipun demikian, penggunaan UML tidak menutup peluang untuk melakukan pengembangan maupun penelitian terhadap metode pemodelan lainnya. (Wayahdi & Ruziq, 2023). Salah satu keunggulan UML adalah

kemampuannya menyajikan representasi visual yang mudah dipahami oleh pengembang sistem, sekaligus memiliki aturan formal yang mendukung proses implementasi serta simulasi perangkat lunak (Azmi et al., 2026). Tujuan pemakaian UML sendiri untuk menghasilkan rancangan sistem yang terstruktur, efektif, dan pas dengan kebutuhan penggunaanya.

Ada beberapa jenis diagram UML yang lazim dipakai dalam pengembangan perangkat lunak, di antaranya perangkat lunak mencakup berbagai jenis, diantaranya :

a) Use Case Diagram

Use Case Diagram adalah salah satu diagram UML yang dipakai untuk menggambarkan interaksi antara aktor (pengguna) dengan sistem. Setiap use case dapat disusun berdasarkan tingkat kepentingan, tingkat abstraksi, serta pengaruhnya terhadap kebutuhan pengguna (Hidayat & Noor, 2025). Diagram ini berguna menjelaskan fungsi-fungsi utama sistem sekaligus hubungan pengguna dengan tiap fitur yang ada (Johar & Soleman, 2026).

Perannya juga penting di tahap analisis kebutuhan, sebab memudahkan developer memahami dan merancang sistem sesuai apa yang dibutuhkan pengguna . Pada tahap perancangan, dibuat use case diagram untuk menggambarkan kebutuhan fungsional sistem, activity diagram untuk menjelaskan alur proses, serta class diagram untuk memodelkan struktur data dan hubungan antar kelas yang

menjadi dasar pembentukan sistem (Puspita Br Munthe et al., 2026).

Berikut gambar yang menjelaskan simbol dan keterangan pada use case diagram.

Tabel 2.2 Keterangan *Use Case Diagram*

Simbol	Nama	Keterangan
	<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan Use case.
	<i>Use Case</i>	Mewakili fitur atau layanan yang disediakan sistem untuk aktor.
	<i>Association</i>	Garis yang menghubungkan aktor (pengguna) dengan use case (fitur). Artinya: aktor itu berinteraksi langsung dengan fitur tersebut.
	<i>Include</i>	Menandakan bahwa use case A selalu menyertakan use case B. Artinya: proses A harus memanggil proses B.
	<i>Extend</i>	Menandakan bahwa use case B bisa jadi tambahan opsional untuk use case A. Artinya: proses B hanya dijalankan kalau kondisi tertentu terjadi.

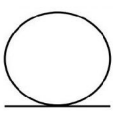
Sumber :(Harlina et al., 2025)

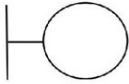
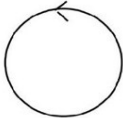
b) *Sequence Diagram*

Sequence diagram merupakan diagram UML yang digunakan untuk menggambarkan urutan interaksi antar objek dalam suatu

sistem (Siagian et al., 2026). Melalui sequence diagram, hubungan komunikasi antarobjek ditampilkan sesuai urutan waktu pelaksanaannya, sehingga mempermudah analisis alur kerja dan interaksi dalam sistem (Syam et al., 2024). Selain itu, sequence diagram juga berfungsi memvisualisasikan aliran pesan antar objek serta menunjukkan urutan pertukaran pesan yang terjadi pada setiap skenario proses dalam sistem (Ichsan Fadhlika Pangestu, 2024). Berikut ini gambar yang menjelaskan simbol dan keterangan pada Sequence Diagram.

Tabel 2. 1 Keterangan *Sequence Diagram*

Simbol	Nama	Keterangan
	<i>Actor</i>	<i>Actor</i> : Menggambarkan hubungan komunikasi antara objek dan sistem.
	<i>Entity Class</i>	<i>Entity</i> : simbol untuk menggambarkan suatu kegiatan yang akan dilakukan dalam diagram
	<i>Message</i>	<i>Message</i> : Simbol yang menggambarkan sebuah petunjuk pengiriman suatu pesan

	<i>Boundary</i>	<i>Boundary</i> : simbol yang menggambarkan alat untuk melakukan interaksi dalam sebuah sistem
	<i>Control Class</i>	Kelas yang berfungsi untuk mengatur alur proses, logika, dan koordinasi antara <i>boundary class</i> dan <i>entity class</i> dalam sistem.
	<i>Life Line</i>	<i>Lifeline</i> : menggambarkan aktivitas objek yang terdapat pada sebuah sistem.

Sumber: (Bantun et al., 2022)

c) *Activity Diagram*

Activity Diagram termasuk salah satu jenis diagram dalam Unified Modeling Language (UML) yang dipakai menggambarkan alur kerja atau proses bisnis dalam sistem secara sistematis (Widodo et al., 2026). Activity diagram digunakan untuk memodelkan urutan aktivitas dalam suatu sistem secara dinamis. Diagram ini merupakan bentuk khusus dari state diagram yang menampilkan aliran proses, dengan setiap state merepresentasikan suatu aksi dan transisi terjadi setelah aksi sebelumnya selesai

dieksekusi (Emah Huzzaemah, 2024). Selain digunakan untuk menggambarkan alur aktivitas dalam sistem, activity diagram juga digunakan untuk pengambilan keputusan, hingga proses berakhir (Gilliani et al., 2024). Setiap simbol memiliki fungsi yang berbeda dalam merepresentasikan alur proses. Berikut ini table yang menjelaskan simbol dan keterangan pada Activity Diagram.

Tabel 2. 2 Keterangan *Activity Diagram*

Simbol	Nama	Keterangan
	<i>Initial</i>	<i>Initial</i> , menunjukkan titik awal dimulainya alur aktivitas dalam sistem.
	<i>Action</i>	<i>Action</i> , menunjukkan aktivitas atau proses yang dilakukan dalam alur kerja sistem.
	<i>Decision</i>	<i>Decision Points</i> , menunjukkan percabangan atau pengambilan keputusan berdasarkan suatu kondisi.
	Swimlane	<i>Swimlane</i> mengelompokkan aktivitas berdasarkan aktor atau pihak yang bertanggung jawab.
	<i>Final</i>	<i>Final</i> , menunjukkan titik akhir atau selesainya alur aktivitas dalam sistem.

Sumber : (Suharni, 2023)

11. Pengujian *Blackbox*

Black Box Testing adalah metode pengujian perangkat lunak yang fokus menguji fungsi sistem tanpa perlu melihat struktur internal atau kode programnya (Suarmeddy et al., 2026). Caranya dengan memberi *input* tertentu ke sistem, lalu *output* yang keluar diamati dan dibandingkan dengan hasil yang seharusnya sesuai spesifikasi sistem (Zaky & Gunawan, 2026). Melalui pengujian Black Box, memungkinkan pendeteksian kesalahan pada fungsi sistem secara efektif, sehingga dapat meningkatkan kualitas, keandalan, serta keamanan perangkat lunak (Haqqoni et al., 2024).

Dari penjelasan tadi, bisa disimpulkan pengujian *Black Box* menunjukkan seluruh fitur sistem sudah berjalan sesuai fungsi dan kebutuhan yang diharapkan. *Input*, proses, sampai *output*-nya bekerja dengan baik, sehingga sistem ini layak dipakai.

B. Kajian Empiris

Bagian ini mengulas beberapa informasi dan penelitian sebelumnya sebagai bahan pembandingan, dengan menyoroti kekurangan dan kelebihan yang ada dari masing-masing penelitian tersebut.

Pertama, riset yang digarap Handono et al. menghasilkan sistem penilaian kinerja sales berdasarkan capaian target penjualan dengan mengandalkan metode Rapid Application Development (RAD). Sistem tersebut dirancang untuk menjawab persoalan evaluasi yang sebelumnya masih dilakukan secara manual. Melalui sistem ini,

pencapaian target bisa dihitung secara otomatis, kinerja sales bisa diklasifikasikan sesuai hasil penjualan, serta laporan bisa disajikan dalam bentuk visual maupun cetak. Penerapan metode RAD dipilih karena dinilai mampu mempercepat proses pengembangan lewat pendekatan iteratif yang melibatkan pengguna secara aktif. Hasil penelitian menunjukkan bahwa sistem ini membantu meningkatkan akurasi penilaian sekaligus mempermudah proses monitoring kinerja sales.

Kedua, studi yang digarap Kadek Dwi Mahesayasa membahas sistem monitoring kinerja salesman di CV Bharata Sport & Fashion. Masalah yang dihadapi mencakup pengolahan data penjualan, perhitungan bonus, dan pengawasan kinerja salesman yang masih dikerjakan manual. Kondisi ini bikin pelaporan jadi lambat, risiko kesalahan data naik, dan pengawasan kinerja antar cabang jadi ribet. Solusinya, dikembangkan aplikasi monitoring kinerja salesman berbasis web yang bisa memadukan pencatatan pesanan, laporan kunjungan, perhitungan bonus, sampai pemantauan kinerja secara real-time. Hasil implementasinya memperlihatkan sistem ini bisa mendongkrak efisiensi kerja, mempermudah pelaporan, dan membantu manajemen mengambil keputusan berbekal data yang lebih akurat dan tertata.

Ketiga, dalam jurnal berjudul “Web-Based Sales Force Automation Implementation Using Extreme Programming at CV.

Samudra Jaya”, Siti Farikha Jauhari Rahmadhani dan Dwi Fatrianto Suyatno menggarap sistem Sales Force Automation (SFA) berbasis web guna mendongkrak efisiensi proses penjualan. Sistem ini dikembangkan karena proses penjualan, seperti pencatatan pesanan dan laporan kunjungan harian, masih dilakukan secara manual sehingga berpotensi menimbulkan human error, keterlambatan penyampaian informasi, serta kesulitan dalam memantau kinerja salesman.

Keempat, masih dari peneliti yang sama, Siti Farikha Jauhari Rahmadhani dan Dwi Fatrianto Suyatno membangun sistem Sales Force Automation (SFA) berbasis web di CV. Samudra Jaya untuk mengatasi proses penjualan yang tadinya serba manual. Proses itu mencakup pencatatan pesanan dan laporan kunjungan harian yang rawan salah catat, informasi telat, dan kinerja salesman susah dipantau. Pengembangan sistemnya memakai metode Extreme Programming (XP) yang terdiri dari tahap planning, design, coding, dan testing. Hasilnya, sistem ini terbukti mampu mendongkrak efisiensi administrasi penjualan, mempermudah pemantauan kinerja salesman secara real-time, serta menyuguhkan informasi yang lebih akurat guna mendukung pengambilan keputusan manajemen.

Melihat keempat penelitian di atas, kelihatan jelas semuanya fokus memanfaatkan sistem informasi untuk memudahkan proses monitoring dan penilaian kinerja sales. Sistem-sistem yang dibangun terbukti bisa mendongkrak efisiensi, menekan kesalahan, dan

menyajikan informasi lebih cepat serta tertata dibanding cara manual. Bedanya dengan penelitian yang sedang dikerjakan ini terletak pada platform dan fokus sistemnya.

Penelitian ini menggunakan aplikasi berbasis Android sehingga lebih fleksibel dan gampang dipakai sales di lapangan. Selain itu, penelitian ini lebih menekankan pada monitoring target penjualan, sehingga sistem yang dibangun bukan hanya berfungsi untuk penilaian kinerja, tetapi juga untuk memantau pencapaian target secara langsung dan real-time.

C. Kerangka Berpikir

