

BAB II

LANDASAN TEORI

2.1. Konsep Dasar Sistem Informasi

Sistem informasi merupakan suatu kesatuan komponen yang saling berinteraksi untuk mengumpulkan, mengolah, menyimpan, dan menyajikan data menjadi informasi yang berguna bagi penggunanya. Komponen utama dalam sistem informasi meliputi perangkat keras, perangkat lunak, basis data, prosedur, serta sumber daya manusia yang bekerja secara terintegrasi. Sistem informasi dirancang untuk mendukung kegiatan operasional, manajerial, dan pengambilan keputusan agar berjalan lebih efektif dan efisien. Dengan adanya sistem informasi, proses yang sebelumnya dilakukan secara manual dapat diotomatisasi sehingga mampu mengurangi kesalahan, mempercepat alur kerja, serta meningkatkan akurasi dan konsistensi data yang dihasilkan.

Penerapan konsep dasar sistem informasi sangat membantu dalam pengelolaan data obat karena seluruh data dapat diolah secara terpusat dan terintegrasi dalam satu sistem berbasis komputer. Proses pencatatan stok, pengelolaan data obat, pemantauan keluar-masuk barang, hingga penyajian laporan dapat dilakukan secara sistematis dan real-time. Integrasi antara basis data, antarmuka pengguna, serta logika pemrosesan data memungkinkan informasi stok tersedia dengan cepat dan akurat untuk mendukung aktivitas operasional apotek. Pemanfaatan sistem informasi juga mendorong peningkatan efisiensi kerja, meminimalkan kesalahan pencatatan, serta membantu pengambilan keputusan terkait pengendalian persediaan obat secara lebih terstruktur.

2.1.1. Pengertian Sistem

Sistem dapat dipahami sebagai suatu kesatuan yang terdiri dari berbagai elemen yang saling berhubungan dan bekerja sama untuk mencapai tujuan tertentu. Elemen-elemen tersebut dapat berupa komponen fisik, prosedur, data, manusia, maupun perangkat lunak yang terintegrasi sehingga mampu menjalankan fungsi tertentu secara teratur dan terstruktur. Sebuah sistem memiliki input, proses, dan output yang saling berkaitan, serta membutuhkan mekanisme pengendalian agar dapat beroperasi dengan efektif. Konsep sistem juga menekankan adanya keteraturan dan keterpaduan, sehingga setiap komponen memiliki peran dan hubungan yang jelas dalam mendukung keseluruhan aktivitas yang dijalankan.

Penerapan konsep sistem sangat penting ketika mengelola aktivitas yang membutuhkan koordinasi dan akurasi tinggi, seperti pengelolaan persediaan obat di sebuah apotek. Penggunaan sistem yang terkomputerisasi memungkinkan setiap komponen mulai dari pencatatan stok, pemantauan obat masuk dan keluar, hingga penyajian laporan dapat berjalan secara terstruktur dan saling terhubung. Sistem yang dibangun dengan baik mampu mengurangi kesalahan manusia, mempercepat proses kerja, serta menyediakan informasi yang real-time untuk mendukung pengambilan keputusan. Melalui pendekatan sistem, seluruh aktivitas pengelolaan dapat dijalankan lebih efisien, transparan, dan dapat dipertanggungjawabkan karena alur kerja diarahkan untuk bekerja secara terpadu dan saling melengkapi.

2.1.2. Pengertian Informasi

Informasi merupakan salah satu sarana untuk memperkenalkan suatu perusahaan atau organisasi, sangat erat hubungannya dengan perkembangan organisasi yang masih dalam tahap perkembangan, dengan tidak adanya informasi maka suatu organisasi tidak akan pernah dapat cepat berkembang seperti apa yang diinginkan (Kristania et al., 2025). Informasi yang baik harus memiliki karakteristik akurat, relevan, tepat waktu, dan mudah dipahami agar mampu memberikan nilai tambah bagi penggunanya. Proses pembentukan informasi biasanya melibatkan pengumpulan data, pengolahan, penyimpanan, hingga penyajian kembali dalam bentuk yang lebih terstruktur. Informasi menjadi elemen penting dalam berbagai kegiatan karena keberadaannya membantu memecahkan masalah, mengurangi ketidakpastian, serta meningkatkan efisiensi dan efektivitas suatu aktivitas atau organisasi.

Informasi merupakan hasil dari pengolahan data sehingga menjadi bentuk yang penting bagi penerimanya dan mempunyai kegunaan sebagai dasar dalam pengambilan keputusan yang dapat dirasakan akibatnya secara langsung saat itu juga atau secara tidak langsung pada saat mendatang (Afriansyah et al., 2022). Keberadaan informasi yang berkualitas sangat diperlukan untuk pengelolaan persediaan obat yang menuntut ketelitian dan keakuratan tinggi. Sistem yang mampu menyajikan informasi stok secara real-time, termasuk jumlah ketersediaan obat, riwayat keluar-masuk barang, serta masa kedaluwarsa, akan membantu pihak apotek dalam membuat keputusan yang tepat terkait pemesanan dan pengendalian persediaan. Informasi yang tersusun rapi dan mudah diakses juga meminimalkan risiko kesalahan pencatatan yang sering terjadi pada pengelolaan

manual. Dengan dukungan informasi yang akurat dan terintegrasi, aktivitas operasional apotek dapat berjalan lebih efisien, responsif, dan sesuai kebutuhan pelayanan masyarakat.

2.1.3. Pengertian Sistem Informasi

Sistem informasi adalah kumpulan data yang didistribusikan untuk memecahkan masalah dan pengambilan Keputusan (Aqli, 2023). Keberadaan sistem informasi memungkinkan suatu organisasi menjalankan proses kerja secara lebih efisien karena alur data dapat dikelola dengan baik dan diakses secara mudah. Sistem informasi juga dirancang untuk meminimalkan kesalahan, mempercepat proses administrasi, serta meningkatkan akurasi data yang digunakan dalam aktivitas organisasi. Dengan dukungan teknologi komputer, sistem informasi mampu melakukan otomatisasi berbagai proses yang sebelumnya dilakukan secara manual, sehingga kualitas dan ketepatan informasi meningkat.

Sistem informasi merupakan kumpulan komponen yang membentuk kesatuan untuk mengolah data transaksi harian untuk mendukung kegiatan operasional suatu organisasi dengan menghasilkan laporan tertentu (Revita, 2023). Penggunaan sistem informasi menjadi sangat penting bagi organisasi yang membutuhkan akurasi dan ketelitian dalam pengelolaan data, termasuk pengelolaan stok barang yang jumlah dan pergerakannya dinamis seperti pada apotek. Sistem yang mampu mencatat keluar-masuk obat, memantau ketersediaan stok secara real-time, dan menyediakan laporan yang cepat akan sangat membantu dalam memastikan operasional berjalan lancar. Pengelolaan manual yang rentan terhadap kesalahan pencatatan dapat diminimalkan melalui penerapan sistem informasi berbasis komputer. Melalui pemanfaatan teknologi seperti PHP dan

MySQL, sistem informasi dapat dibangun dengan fungsi yang lebih terstruktur, terintegrasi, dan mudah digunakan sehingga mampu mendukung peningkatan efisiensi serta kualitas pelayanan.

2.2. Sistem Informasi Stock Obat

Sistem informasi stock obat merupakan sebuah sistem yang dirancang untuk mengelola seluruh data persediaan obat secara terkomputerisasi dan terintegrasi, mulai dari pencatatan obat masuk, obat keluar, hingga pemantauan jumlah stok dan masa kedaluwarsa. Sistem ini membantu meminimalkan kesalahan pencatatan yang sering terjadi pada proses manual, serta mempercepat penyajian informasi yang dibutuhkan oleh pihak pengelola. Dengan adanya sistem informasi, proses pengendalian persediaan dapat dilakukan secara lebih akurat karena data tersimpan dalam basis data yang terstruktur dan mudah diakses. Selain itu, sistem ini mampu menghasilkan laporan stok secara berkala yang berguna untuk perencanaan pengadaan obat, pengawasan ketersediaan, serta pengambilan keputusan yang lebih tepat.

Penerapan sistem informasi stock obat memberikan kemudahan dalam mengelola persediaan secara efisien melalui pemanfaatan teknologi berbasis web dan database terintegrasi. Seluruh aktivitas pengelolaan obat dapat dipantau secara real-time, sehingga memudahkan pihak apotek dalam mengetahui kondisi stok, mengantisipasi kekurangan obat, serta mengurangi risiko kelebihan persediaan. Penggunaan sistem ini juga mendukung proses administrasi yang lebih tertib karena data tersimpan dengan aman dan dapat diakses sesuai kebutuhan. Dengan pengelolaan yang lebih baik, kualitas pelayanan kepada pelanggan dapat

ditingkatkan melalui ketersediaan obat yang terjamin dan proses kerja yang lebih cepat serta terorganisir.

2.2.1. Pengertian Sistem Informasi Stock Obat

Sistem informasi stok obat merupakan suatu sistem yang dirancang untuk mengelola persediaan obat secara terstruktur, terintegrasi, dan otomatis dengan memanfaatkan teknologi informasi. Sistem ini berfungsi untuk mencatat, memantau, dan mengendalikan jumlah obat yang masuk dan keluar, termasuk informasi terkait kategori obat, jumlah persediaan, harga, serta masa kedaluwarsa. Penggunaan sistem informasi membantu memastikan ketersediaan obat selalu terpantau, meminimalkan human error, dan mempercepat proses administrasi. Selain itu, sistem ini juga menyediakan laporan stok yang akurat dan real-time sehingga memudahkan proses pengambilan keputusan terkait pemesanan, pengadaan, dan pengendalian obat. Dengan adanya sistem yang terkomputerisasi, pengelolaan persediaan dapat dilakukan secara lebih efisien dan mampu mendukung peningkatan kualitas layanan di sektor farmasi.

Pengembangan sistem informasi stok obat menjadi landasan penting untuk meningkatkan efektivitas pengelolaan persediaan di Apotek Jaya Raya Sigambal, mengingat kebutuhan akan data yang akurat dan proses administrasi yang cepat. Sistem ini memungkinkan seluruh proses pencatatan stok dilakukan secara digital, sehingga masalah seperti selisih persediaan, keterlambatan pengecekan masa kedaluwarsa, dan kesulitan dalam menelusuri riwayat keluar-masuk obat dapat diminimalkan. Implementasi teknologi berbasis PHP dan *MySQL* memberikan fleksibilitas dalam pengelolaan *database* serta kemudahan akses bagi pengguna, sehingga pegawai apotek dapat menjalankan tugasnya dengan lebih terorganisir.

Melalui penerapan sistem informasi ini, operasional apotek menjadi lebih efisien, proses pengawasan stok lebih terkontrol, dan kualitas layanan terhadap pelanggan dapat meningkat secara signifikan.

2.2.2. Konsep Dasar Sistem Informasi Stock Obat

Sistem informasi stok obat merupakan sebuah mekanisme terkomputerisasi yang dirancang untuk membantu proses pengelolaan persediaan obat secara lebih efisien, akurat, dan terstruktur. Sistem ini mencakup beberapa komponen penting seperti pencatatan stok masuk dan keluar, pemantauan ketersediaan obat, pengingat masa kedaluwarsa, hingga penyusunan laporan persediaan secara berkala. Dengan memanfaatkan teknologi basis data, informasi mengenai persediaan dapat diperbarui secara real-time sehingga meminimalkan risiko ketidaksesuaian data antara jumlah fisik dan jumlah yang tercatat. Selain itu, sistem juga berfungsi sebagai alat pengendalian yang membantu pihak pengelola dalam memastikan ketersediaan obat tetap stabil dan sesuai kebutuhan konsumen.

Penerapan konsep sistem informasi stok obat sangat relevan untuk meningkatkan efektivitas pengelolaan obat di Apotek Jaya Raya Sigambal, terutama karena proses yang sebelumnya masih dilakukan secara manual rentan menimbulkan kesalahan pencatatan dan keterlambatan dalam pemantauan ketersediaan obat. Pemanfaatan teknologi berbasis PHP dan *MySQL* mampu memberikan solusi yang lebih terstruktur dengan menyediakan fitur-fitur pengelolaan persediaan yang mudah diakses dan diperbarui oleh petugas apotek. Kehadiran sistem ini memungkinkan informasi stok ditampilkan secara lebih akurat, mendukung percepatan proses pelayanan, serta mengurangi potensi kerugian akibat kelalaian dalam pengecekan masa kedaluwarsa maupun

kekosongan obat. Dengan adanya sistem yang terintegrasi, aktivitas operasional dapat berjalan lebih optimal dan mendukung peningkatan kualitas layanan apotek secara keseluruhan.

2.3. Alat Bantu dalam Pengembangan Sistem

Alat bantu dalam pengembangan sistem merupakan elemen penting yang digunakan untuk memvisualisasikan kebutuhan, alur kerja, serta struktur logika dari sebuah perangkat lunak agar lebih mudah dipahami dan dianalisis. Berbagai jenis diagram dalam *Unified Modeling Language (UML)* seperti *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, dan *Class Diagram* berfungsi untuk menggambarkan interaksi pengguna dengan sistem, alur aktivitas, urutan proses, hingga struktur kelas dalam sistem secara detail. Penggunaan UML membantu pengembang menyusun sistem secara lebih terorganisasi, mengurangi risiko miskomunikasi, serta mempercepat proses analisis dan desain. Selain itu, UML juga mempermudah pembuatan dokumentasi yang dibutuhkan untuk pengembangan lanjutan maupun pemeliharaan sistem di masa depan.

Penerapan UML memberikan gambaran menyeluruh mengenai bagaimana sistem obat dirancang mulai dari identifikasi aktor seperti admin atau petugas apotek melalui *Use Case Diagram*, penyusunan langkah-langkah proses pencatatan dan pemantauan stok melalui *Activity Diagram*, hingga penjelasan interaksi antar objek sistem menggunakan *Sequence Diagram*. Selanjutnya, struktur basis data dan komponen sistem diperjelas melalui *Class Diagram* sehingga hubungan antar data obat, transaksi, dan pengguna dapat terlihat dengan jelas. Pemodelan ini memungkinkan penyusunan desain sistem yang lebih presisi, memastikan bahwa seluruh kebutuhan pengelolaan stok obat dapat diakomodasi,

serta mendukung proses pengembangan sistem berbasis PHP dan *MySQL* agar berjalan lebih efektif dan tepat sasaran.

2.3.1. *Unified Modeling Language (UML)*

Unified Modeling Language (UML) adalah suatu metode dalam pemodelan secara visual yang digunakan sebagai sarana perancangan sistem berorientasi objek (Wahyuni et al., n.d.). UML menyediakan berbagai jenis diagram, seperti *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, dan *Class Diagram*, yang masing-masing memiliki fungsi untuk menggambarkan alur kerja, interaksi antarkomponen, hingga struktur objek dalam sistem. Pemodelan menggunakan UML membantu pengembang memahami kebutuhan sistem, memetakan proses bisnis, serta merancang arsitektur perangkat lunak secara lebih jelas dan konsisten. Dengan adanya standar ini, pengembang dapat mengkomunikasikan rancangan sistem secara lebih efektif, baik kepada tim teknis maupun pemangku kepentingan non-teknis.

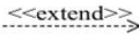
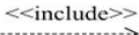
UML mempunyai kelebihan, diantaranya dapat memberikan bahasa pemodelan visual atau gambar bagi developer dari berbagai macam pemrograman maupun proses umum rekayasa, menyatukan informasi-informasi terbaik yang ada dalam pemodelan, memberikan suatu gambaran model atau sebagai bahasa pemodelan visual yang ekspresif dalam pengembangan sistem, dapat memodelkan sistem berorientasi objek, mempermudah developer untuk membaca suatu sistem, serta berguna sebagai blueprint yang nantinya dapat menjelaskan informasi yang lebih detail dalam perancangan berupa coding suatu program (Hidayati et al., 2023). Penggunaan UML memberikan gambaran yang sistematis terkait alur pengelolaan stok, hubungan antar-entitas seperti obat, pemasok, transaksi keluar-

masuk barang, serta interaksi pengguna dengan fitur-fitur sistem. Diagram use case dapat memperlihatkan kebutuhan utama seperti pencatatan stok, pembaruan data obat, dan pembuatan laporan, sementara *Activity Diagram* menggambarkan langkah-langkah operasional yang semula dilakukan secara manual menjadi terstruktur dalam sistem digital. *Sequence Diagram* dapat menunjukkan urutan proses ketika pengguna berinteraksi dengan sistem, sedangkan *Class Diagram* membantu merancang struktur *database* yang lebih terorganisir menggunakan tabel-tabel yang saling terhubung. Pemodelan ini mempermudah proses perancangan sistem berbasis PHP dan *MySQL* karena setiap komponen telah digambarkan secara visual, sehingga sistem yang dihasilkan lebih terukur, mudah dikembangkan, dan sesuai dengan kebutuhan pengelolaan persediaan obat.

2.3.2. Use Case Diagram

Use Case Diagram merupakan salah satu alat pemodelan dalam *Unified Modeling Language (UML)* yang digunakan untuk menggambarkan interaksi antara pengguna (*aktor*) dengan sistem. *Use Case Diagram* menggambarkan interaksi antara sistem internal, sistem eksternal, dan user. Dengan kata lain, secara grafik menjelaskan siapa yang menggunakan sistem, dan dengan cara apa user berinteraksi dengan sistem (Gunawan et al., 2022). *Use Case Diagram* membantu memvisualisasikan kebutuhan fungsional secara sederhana namun jelas, sehingga pengembang dapat memahami alur kerja sistem dari sudut pandang pengguna. Selain itu, diagram ini berperan penting dalam mengidentifikasi batasan sistem, peran aktor, serta hubungan antarproses yang akan dibangun.

Tabel 2. 1. Simbol-Simbol Use Case

Nama Komponen	Keterangan	Simbol
<i>Usecase</i>	<i>Usecase</i> adalah menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, yang dinyatakan dengan menggunakan kata kerja	
<i>Actor</i>	<i>Actor</i> adalah pengguna sistem. <i>Actor</i> tidak terbatas hanya manusia saja, jika sebuah sistem berkomunikasi dengan aplikasi lain dan membutuhkan <i>input</i> atau memberikan <i>output</i> , maka aplikasi tersebut juga bisa dianggap sebagai <i>actor</i> .	
Asosiasi	Asosiasi digunakan untuk menghubungkan <i>actor</i> dengan <i>usecase</i> / komunikasi antara <i>actor</i> dan <i>usecase</i> .	
Ektensi	Relasi <i>usecase</i> tambahan ke sebuah <i>usecase</i> dimana <i>usecase</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>usecase</i> tambahan itu.	
<i>Include</i>	Relasi <i>usecase</i> tambahan ke sebuah <i>usecase</i> dimana <i>usecase</i> yang ditambahkan memerlukan <i>usecase</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalkannya syarat ini.	

Sumber: (Ariansyah & Wijaya, 2021)

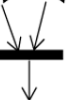
Use Case Diagram merupakan pemodelan pengoperasian sistem informasi yang dibuat (Samsudin & Januar, 2024). Penerapan *Use Case Diagram* pada pengembangan sistem informasi stok obat di Apotek Jaya Raya Sigambal menjadi langkah penting untuk memastikan bahwa seluruh kebutuhan operasional apotek terakomodasi dengan baik. Diagram ini membantu menggambarkan proses utama seperti pengelolaan data obat, pencatatan stok masuk dan keluar, pemantauan masa kedaluwarsa, hingga pembuatan laporan persediaan yang dibutuhkan oleh pegawai apotek. Setiap aktor seperti admin maupun petugas apotek memperoleh gambaran jelas mengenai perannya dalam sistem melalui pemodelan tersebut, sehingga membantu memastikan bahwa alur kerja yang dibangun sesuai dengan kebutuhan nyata di lapangan. Selain itu, *Use Case Diagram* memberikan dasar yang kuat bagi pengembang untuk menghindari kesalahan dalam mendesain fitur, memastikan sistem lebih efisien, mudah

digunakan, dan benar-benar mampu meningkatkan akurasi pengelolaan persediaan obat.

2.3.3. Activity Diagram

Activity Diagram merupakan representasi grafik yang menggambarkan alur kerja suatu sistem secara bertahap (Nugraha et al., 2024). Diagram ini menampilkan urutan proses, keputusan, serta interaksi antar aktivitas dari awal hingga akhir sehingga memudahkan pemahaman terhadap bagaimana suatu sistem bekerja. Dengan simbol-simbol yang sederhana seperti initial *node*, *action*, *decision node*, *merge*, *fork*, *join*, dan *final node*, *Activity Diagram* membantu pengembang melihat gambaran alur proses secara menyeluruh. Diagram ini sangat berguna untuk memetakan proses bisnis, menggambarkan skenario pemakaian sistem, ataupun mengidentifikasi alur aktivitas yang membutuhkan perbaikan. Melalui visualisasi tersebut, analisis kebutuhan dapat dilakukan lebih tepat, sehingga pengembangan sistem menjadi lebih terarah dan minim kesalahan.

Tabel 2. 2. Simbol-Simbol Activity Diagram

Nama Komponen	Keterangan	Simbol
<i>Start</i> (Mulai)	Menyatakan awal dari suatu proses	
<i>Stop</i> (Berhenti)	Menyatakan akhir dari proses	
<i>Decision</i>	Menyatakan kondisi dari suatu proses (pilihan untuk mengambil keputusan).	
<i>Action</i>	Menyatakan aksi yang dilakukan dalam suatu arsitektur sistem.	
<i>Fork</i> (Percabangan)	menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu	
<i>Join</i> (Penggabungan)	Digunakan untuk menunjukkan adanya dekomposisi	

Sumber: (Ariansyah & Wijaya, 2021)

Activity Diagram menggambarkan work flow (aliran kerja) proses bisnis dan urutan dalam sebuah proses. Diagram ini mirip dengan flowchart karena memodelkan workflow dari satu aktivitas ke aktivitas lainnya (Ariansyah & Wijaya, 2021). Alur aktivitas yang sebelumnya berjalan secara manual dapat divisualisasikan dengan jelas sehingga terlihat bagian mana yang membutuhkan otomatisasi. Diagram ini membantu menguraikan langkah-langkah kerja pegawai apotek, interaksi dengan sistem, serta keputusan yang muncul selama proses pengelolaan stok obat. Dengan adanya pemetaan aktivitas tersebut, sistem informasi yang dibangun dapat menyesuaikan kebutuhan nyata di lapangan, memastikan setiap fungsi saling terhubung, serta menghadirkan alur kerja yang lebih efisien dan terstruktur bagi proses operasional apotek.

2.3.4. Class Diagram

Class Diagram adalah menggambarkan keadaan (atribut / properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode/fungsi) (Ariansyah & Wijaya, 2021). Diagram ini menampilkan atribut, operasi, serta relasi seperti asosiasi, agregasi, komposisi, serta pewarisan sehingga mempermudah pengembang memahami bagaimana data dan fungsi saling terhubung. *Class Diagram* berperan sebagai blueprint yang membantu menyusun alur kerja sistem secara logis sebelum diimplementasikan ke dalam bahasa pemrograman. Dengan adanya pemetaan struktur sistem melalui *Class Diagram*, proses pengembangan dapat dilakukan lebih terarah, terstruktur, dan mampu meminimalkan kesalahan desain sejak tahap awal.

Tabel 2. 3. Simbol-Simbol Class Diagram

Nama Komponen	Keterangan	Simbol
Kelas	Kelas pada struktur sistem	Nama kelas
		+Atribut
		Operasi()
Asosiasi	Kelas antar kelas dengan makna umum, asosiasi biasanya juga disertai <i>multiplicity</i>	—————
Asosiasi berarah	Kelas antar kelas dengan makna kelas yang satu digunakan kelas yang lain asosiasi biasanya juga disertai <i>multiplicity</i>	—————>
Generalisasi	Kelas antar kelas dengan makna generalisasi-spesialisasi (umum-khusus).	—————▷
Kebergantungan/ <i>dependency</i>	Kelas antar kelas dengan makna kebergantungan antar kelas.	----->
Agregasi/ <i>Aggregation</i>	Kelas antar kelas dengan makna semua bagian.	—————◊

Sumber: (Ariansyah & Wijaya, 2021)

Class Diagram menjelaskan hubungan antar kelas serta penjelasan secara detail dalam setiap kelas system (Ramadani, 2025). Perancangan ini membantu memastikan bahwa setiap proses mulai dari pencatatan stok, transaksi keluar masuk obat, sampai pengelolaan data pengguna dapat terimplementasi secara konsisten dan efisien pada sistem berbasis PHP dan *MySQL* yang dikembangkan. Selain itu *Class Diagram* mendukung pengembang dalam mengidentifikasi kebutuhan *database* dengan lebih tepat sehingga struktur tabel dan relasinya dapat disusun sesuai dengan kebutuhan operasional apotek. Dengan demikian *Class Diagram* berperan sebagai landasan yang memperkuat kualitas desain sistem serta meningkatkan keandalan hasil implementasinya.

2.3.5. *Sequence Diagram*

Sequence Diagram merupakan gambaran interaksi antar objek dan mengindikasikan komunikasi diantara obyek-obyek tersebut (Fernandes et al., 2022). Diagram ini menunjukkan bagaimana pesan dikirimkan dari satu objek ke

objek lainnya serta bagaimana proses tersebut berlangsung secara berurutan dari awal hingga akhir. Dengan menampilkan aktor, objek, lifeline, dan message, *Sequence Diagram* membantu memberikan pemahaman yang jelas mengenai logika proses, aliran aktivitas, dan hubungan antar komponen sistem secara dinamis. Fungsi utamanya adalah memvisualisasikan proses yang terjadi dalam skenario tertentu sehingga pengembang dapat melihat struktur interaksi yang dibutuhkan untuk menjalankan suatu fitur atau fungsi sistem secara efisien dan konsisten.

Tabel 2. 4. Simbol-Simbol Sequence Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menggambar orang yang sedang berinteraksi dengan sistem.
2		<i>Entity Class</i>	Menggambarkan hubungan yang akan dilakukan
3		<i>Boundary Class</i>	Menggambarkan sebuah gambaran dari form
4		<i>Control Class</i>	Menggambarkan penghubung antara boundary dengan tabel
5		<i>A focus of Control & A Life Line</i>	Menggambarkan tempat mulai dan berakhirnya message
6		<i>A message</i>	Menggambarkan Pengiriman Pesan

Sumber: <https://badoystudio.com/uml/simbol-sequence-diagram/>

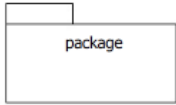
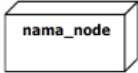
Penerapan *Sequence Diagram* sangat penting untuk menggambarkan alur kerja pengelolaan obat, mulai dari proses login pengguna, pencatatan stok obat, pembaruan data masuk dan keluar, hingga pembuatan laporan persediaan. *Sequence Diagram* ini bertujuan untuk menampilkan interaksi antar objek dalam

dua dimensi (Jurnal et al., 2025). Setiap interaksi antara pengguna, sistem, dan *database* dapat divisualisasikan secara rinci sehingga mempermudah proses perancangan dan pengembangan sistem berbasis PHP dan *MySQL*. Dengan visualisasi tersebut, alur proses yang sebelumnya dilakukan secara manual menjadi lebih terstruktur, mudah dianalisis, serta dapat dipastikan berjalan sesuai kebutuhan operasional apotek. *Sequence Diagram* juga membantu dalam mendeteksi potensi kesalahan alur, memastikan setiap kebutuhan fungsional terakomodasi, serta menjadi acuan penting dalam tahap implementasi dan pengujian sistem .

2.3.6. *Component Diagram*

Component Diagram merupakan salah satu diagram dalam Unified Modeling Language (UML) yang digunakan untuk menggambarkan struktur fisik dari sebuah sistem perangkat lunak. Diagram ini menampilkan komponen-komponen utama yang membentuk sistem, seperti modul, library, file, atau subsistem, beserta hubungan dan ketergantungan di antara komponen tersebut. Dengan menggunakan Component Diagram, pengembang dapat memahami bagaimana bagian-bagian sistem saling berinteraksi, bagaimana fungsi-fungsi dikelompokkan, serta bagaimana alur komunikasi antar komponen berjalan. Diagram ini juga membantu dalam proses perencanaan arsitektur sistem karena memberikan gambaran yang jelas mengenai pembagian tanggung jawab setiap komponen, sehingga memudahkan pengembangan, pemeliharaan, dan pengujian sistem secara terstruktur.

Tabel 2. 5. Simbol-Simbol Component Diagram

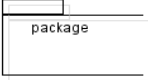
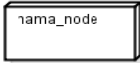
Simbol	Deskripsi
Package 	package merupakan sebuah bungkus dari satu atau lebih <i>node</i>
Node 	biasanya mengacu pada perangkat keras (<i>hardware</i>), perangkat lunak yang tidak dibuat sendiri (<i>software</i>), jika di dalam <i>node</i> disertakan komponen untuk mengkonsistenkan rancangan maka komponen yang diikutsertakan harus sesuai dengan komponen yang telah didefinisikan sebelumnya pada diagram komponen
Kebergantungan / dependency 	Kebergantungan antar <i>node</i> , arah panah mengarah pada <i>node</i> yang dipakai
Link 	relasi antar <i>node</i>

Penggunaan Component Diagram sangat membantu dalam menggambarkan rancangan system informasi yang dibangun karena mampu menunjukkan keterkaitan antara komponen antarmuka pengguna, proses bisnis, logika aplikasi, dan basis data MySQL. Melalui diagram ini dapat terlihat bagaimana modul input data, pengolahan data, laporan, serta pengelolaan database saling terhubung dan bekerja secara terintegrasi. Kejelasan pembagian komponen membuat proses pengembangan system berbasis PHP dan MySQL menjadi lebih terarah, meminimalkan kesalahan dalam implementasi, serta memudahkan pengembang dalam melakukan pengembangan lanjutan atau perbaikan system. Selain itu, Component Diagram juga berperan penting dalam memastikan bahwa setiap komponen memiliki fungsi yang jelas dan saling mendukung untuk menghasilkan system informasi yang efisien dan andal.

2.3.7. Deployment Diagram

Deployment Diagram merupakan salah satu diagram dalam Unified Modeling Language (UML) yang digunakan untuk menggambarkan arsitektur fisik dari sebuah system, termasuk hubungan antara perangkat keras, perangkat lunak, dan komponen system yang dijalankan di dalamnya. Diagram ini menunjukkan bagaimana komponen aplikasi ditempatkan pada node atau server tertentu, seperti web server, application server, dan database server, serta bagaimana komunikasi antar node tersebut berlangsung. Dengan adanya Deployment Diagram, pengembang dapat memahami konfigurasi system secara menyeluruh, mulai dari alur distribusi aplikasi, penggunaan jaringan, hingga ketergantungan antar komponen. Diagram ini sangat membantu dalam proses perencanaan implementasi system karena memberikan system nyata mengenai lingkungan operasional yang akan digunakan.

Gambar 2. 1. Simbol-Simbol Deployment Diagram

Simbol	Deskripsi
<i>Package</i> 	Package merupakan sebuah bungkusan dari satu atau lebih
<i>Node</i> 	Biasanya mencakup pada perangkat keras (hardware), perangkat lunak yang tidak dibuat sendiri (software), jika didalam node disertakan komponen untuk mengkonsistenkan rancangan maka komponen yang telah didefinisikan sebelumnya pada diagram komponen.
Kebergantungan/ <i>dependency</i> 	Ketergantungan antar node, arah panah mengarah pada node yang dipakai.
<i>Link</i> 	Relasi antar node

Penerapan Deployment Diagram pada pengembangan system informasi berbasis web memberikan system yang jelas mengenai penempatan aplikasi PHP pada web server, pengelolaan database MySQL pada server database, serta akses pengguna melalui perangkat client seperti system atau perangkat mobile. Hubungan antar node dapat divisualisasikan untuk memastikan alur komunikasi data berjalan system dan efisien, termasuk proses pengiriman dan penerimaan data dari system. Kejelasan struktur ini memudahkan pengelolaan system, proses pemeliharaan, serta pengembangan lanjutan karena setiap komponen telah didefinisikan sesuai fungsinya. Dengan rancangan deployment yang baik, system dapat berjalan lebih stabil, terstruktur, dan mampu mendukung kebutuhan operasional secara optimal.

2.4. Database

Basis data atau disebut juga sebagai *database* dalam system informasi system informasi yang disimpan dalam media elektronik atau system secara sistematis (Setiawan et al., 2022). Dalam pengembangan system informasi modern, *database* berfungsi sebagai pusat penyimpanan informasi yang memungkinkan proses pengolahan data dilakukan secara efisien, konsisten, dan terstruktur. Penggunaan *database* memungkinkan pengguna melakukan berbagai aktivitas seperti pencarian, penyisipan, pembaruan, hingga penghapusan data secara cepat tanpa mengorbankan integritas data. Dengan system yang terpusat, *database* membantu mengurangi duplikasi data, meminimalkan kesalahan pencatatan, serta mempermudah proses pemeliharaan data dalam jangka system.

Database atau basis data, system informasi yang disimpan dalam computer secara sistematis sehingga dapat diperiksa menggunakan suatu program

system untuk memperoleh informasi (Damanik et al., 2021). Pemanfaatan *database* menjadi elemen penting dalam system informasi karena mampu mengakomodasi kebutuhan penyimpanan data obat yang beragam, seperti nama obat, jumlah stok, harga, kategori, hingga tanggal kedaluwarsa. Sistem yang dirancang membutuhkan basis data yang stabil dan terstruktur agar proses pencatatan keluar-masuk obat dapat dilakukan secara real-time serta menghasilkan informasi stok yang akurat. Penggunaan *MySQL* sebagai manajemen *database* mendukung pengolahan data dalam jumlah besar sekaligus menjaga performa system tetap optimal. Dengan adanya *database* yang tersusun rapi, proses pemantauan stok menjadi lebih mudah, pengambilan system dapat dilakukan lebih cepat, dan risiko kesalahan akibat pencatatan manual dapat berkurang secara signifikan.

2.4.1. *MySQL*

MySQL system *database* yang menghubungkan script PHP menggunakan perintah query sehingga pengguna *MySQL* mudah dalam menggunakan perintah SQL (Helmina et al., 2023). Struktur system yang fleksibel serta dukungan terhadap berbagai tipe data memungkinkan pengembang untuk mengelola, menyimpan, dan memproses informasi secara terorganisir. Selain itu, *MySQL* memiliki kemampuan untuk melakukan query kompleks, integrasi dengan berbagai system pemrograman, serta system keamanan yang dapat disesuaikan, sehingga menjadi pilihan ideal untuk pengembangan aplikasi yang membutuhkan pengolahan data terpusat dan akurat.

Penggunaan *MySQL* sangat relevan untuk mendukung pengelolaan data inventaris karena mampu menyimpan informasi stok obat secara terstruktur, mulai

dari jumlah persediaan, detail obat, system keluar-masuk barang, hingga masa kedaluwarsa. Selain itu, *MySQL* juga merupakan program pengakses *database* yang bersifat jaringan sehingga dapat digunakan untuk aplikasi multiuser (Janitra et al., 2022). Keunggulan *MySQL* dalam memproses data secara cepat memungkinkan aktivitas pengawasan stok menjadi lebih efisien, terutama system data harus diperbarui secara berkala sesuai aktivitas operasional apotek. Integrasinya yang baik dengan system pemrograman PHP juga mempermudah pembentukan antarmuka system yang interaktif dan system. Dengan memanfaatkan *MySQL*, proses pencatatan dapat diminimalkan dari potensi kesalahan manusia, sementara laporan stok dapat dihasilkan secara otomatis dan real-time, sehingga mendukung kebutuhan pengambilan system yang lebih akurat dan tepat waktu.

2.4.2. Structure Query Language (SQL)

Structure Query Language (SQL) merupakan system standar yang digunakan untuk mengakses, mengelola, dan memanipulasi data dalam system manajemen basis data relasional. SQL menyediakan serangkaian perintah yang memungkinkan pengguna melakukan berbagai operasi seperti membuat system, menambah data, memperbarui informasi, melakukan pencarian, hingga menghapus data secara efisien. Kemampuannya dalam menangani data dalam jumlah besar, struktur sintaks yang jelas, serta fleksibilitasnya dalam digunakan di berbagai platform menjadikan SQL sebagai fondasi utama dalam pengembangan aplikasi berbasis *database*. Selain itu, SQL juga mendukung integritas dan keamanan data melalui penerapan aturan seperti primary key, foreign key, serta mekanisme pengaturan hak akses pengguna. Hal ini membuat SQL menjadi

teknologi yang sangat andal untuk memastikan data tersimpan rapi, konsisten, dan mudah dikelola.

Penggunaan SQL berperan penting sebagai inti dari pengelolaan *database* yang menyimpan data obat, data stok, serta informasi keluar-masuk barang yang diperlukan untuk membangun system informasi . Perintah-perintah SQL memungkinkan proses pengolahan data dilakukan secara otomatis dan terstruktur, sehingga mempermudah penyimpanan, pemanggilan, dan pembaruan informasi setiap kali terjadi transaksi di apotek. SQL juga membantu memastikan keakuratan dan konsistensi data karena setiap perubahan pada stok obat dapat terekam secara langsung melalui query yang dijalankan oleh system berbasis PHP. Dengan dukungan struktur *database* yang dibangun menggunakan SQL, pengelolaan persediaan menjadi lebih cepat, terpantau, dan minim kesalahan dibandingkan dengan metode manual.

2.4.3. Konsep Dasar *Database MySQL*

MySQL merupakan sebuah *database* server yang *free*, artinya bebas menggunakan *database* untuk keperluan pribadi atau usaha tanpa harus membeli atau membayar lisensinya (Fauzan et al., 2024). *MySQL* mendukung penggunaan query SQL (Structured Query Language) untuk melakukan operasi seperti penyimpanan, pengubahan, penghapusan, serta pengambilan data dari *system*-tabel yang saling berhubungan. Struktur *database* dalam *MySQL* dirancang dengan prinsip normalisasi agar data tersimpan dengan efisien, menghindari redundansi, dan memastikan integritas data tetap terjaga. Keunggulan *MySQL* lainnya meliputi kemampuan menangani banyak pengguna secara bersamaan, keamanan yang dapat dikonfigurasi, kompatibilitas dengan berbagai platform, serta kemudahan

integrasi dengan system pemrograman seperti PHP, *Python*, atau *Java*. Kombinasi fitur tersebut menjadikan *MySQL* sebagai pilihan utama dalam pengembangan system informasi berbasis web.

Pemanfaatan *MySQL* memberikan kemampuan untuk menyimpan dan mengelola data obat secara lebih terstruktur, mulai dari informasi stok, transaksi keluar-masuk barang, kategori obat, hingga masa kedaluwarsa yang membutuhkan pemantauan berkala. *Database* yang dirancang dengan baik memungkinkan system menampilkan informasi secara cepat dan akurat sehingga mempermudah proses analisis dan pengambilan system. Hubungan antar system membantu menjaga konsistensi data, misalnya antara data obat dengan system transaksi, sehingga system dapat mendeteksi perubahan stok secara otomatis tanpa perlu pencatatan manual. Keandalan *MySQL* dalam menangani volume data yang terus bertambah juga mendukung kebutuhan pengelolaan yang dinamis, membuat proses administrasi menjadi lebih efektif, teratur, dan minim kesalahan.

2.4.4. Tipe-Tipe Data pada *MySQL*

MySQL menyediakan berbagai tipe data yang dirancang untuk menyimpan informasi sesuai kebutuhan struktur *database*, mulai dari tipe numerik, string, hingga tipe khusus seperti tanggal dan waktu. Tipe numerik meliputi INT, BIGINT, FLOAT, dan DOUBLE yang digunakan untuk menyimpan nilai bilangan dengan system presisi berbeda. Untuk penyimpanan teks, *MySQL* menyediakan tipe CHAR, VARCHAR, TEXT, hingga LONGTEXT yang mampu menampung berbagai ukuran data. Sementara itu, tipe DATE, TIME, DATETIME, dan TIMESTAMP digunakan untuk menyimpan informasi terkait waktu, yang sangat penting dalam pencatatan berbasis kronologis. Selain itu, *MySQL* juga

mendukung tipe data biner seperti BLOB, serta tipe logika BOOLEAN yang digunakan untuk nilai true/false. Keragaman tipe data ini memberikan fleksibilitas bagi pengembang dalam merancang struktur *database* yang efisien dan sesuai kebutuhan system.

Pemilihan tipe data yang tepat sangat berpengaruh terhadap kecepatan akses, efisiensi penyimpanan, serta akurasi informasi dalam system informasi yang dibuat untuk mengelola persediaan obat. Setiap entitas seperti data obat, kategori, stok, transaksi masuk dan keluar, hingga masa kedaluwarsa membutuhkan tipe data yang spesifik agar informasi tersimpan dengan benar dan mudah dikelola. Misalnya, jumlah stok lebih efisien disimpan dengan tipe INT, nama obat menggunakan VARCHAR, sedangkan tanggal kadaluarsa perlu menggunakan tipe DATE agar system dapat melakukan pengecekan otomatis terkait keberadaan obat yang mendekati batas pemakaian.

2.5. Sekilas Tentang PHP

PHP merupakan system pemrograman berbasis server-side yang banyak digunakan untuk membangun aplikasi web dinamis. Bahasa ini memiliki sintaks yang sederhana, fleksibel, dan mudah dipelajari sehingga cocok digunakan oleh pengembang pemula maupun system. PHP juga didukung oleh komunitas besar, dokumentasi yang lengkap, serta berbagai framework dan library yang membantu mempercepat proses pengembangan aplikasi. Selain itu, PHP mampu berintegrasi dengan berbagai jenis *database*, termasuk *MySQL*, sehingga memungkinkan pengelolaan data dalam jumlah besar secara efektif. Keunggulan lain yang dimiliki PHP system kemampuannya berjalan pada system semua jenis server dan

system operasi, menjadikannya salah satu system pemrograman yang paling system dalam pengembangan system informasi berbasis web.

Pemilihan PHP sebagai teknologi utama dalam system informasi memberikan kemudahan dalam merancang fitur-fitur seperti pencatatan stok obat, pemantauan keluar-masuk persediaan, laporan otomatis, hingga pengingat masa kedaluwarsa obat. Integrasinya yang kuat dengan *MySQL* memungkinkan penyimpanan data yang terstruktur, cepat, dan aman sehingga proses manajemen persediaan dapat berjalan lebih efisien. PHP juga mendukung pengembangan antarmuka yang interaktif dan system sehingga memudahkan pengguna apotek dalam mengakses informasi secara real-time. Dengan memanfaatkan fleksibilitas dan kemampuan PHP, system yang dihasilkan dapat dioptimalkan agar mampu memenuhi kebutuhan operasional apotek secara lebih akurat, stabil, dan mudah dikelola.

2.6. Aplikasi Visual Studio Code

Visual Studio Code system sebuah *source-code editor* yang dikembangkan oleh Microsoft dan banyak digunakan oleh programmer karena bersifat ringan, cepat, serta mendukung berbagai system pemrograman. Aplikasi ini menyediakan fitur penting seperti *syntax highlighting*, *intellisense*, *debugging*, *version control* melalui Git, serta dukungan ekstensi yang sangat luas sehingga pengguna dapat menambahkan fungsi sesuai kebutuhan pengembangan perangkat lunak. Tampilan antarmuka yang sederhana namun fleksibel membuat pengguna mudah menyesuaikan lingkungan kerja, sementara kemampuan lintas platform memungkinkan VS Code digunakan pada Windows, macOS, maupun Linux.

Kombinasi fitur-fitur tersebut menjadikan Visual Studio Code salah satu editor yang sangat system di kalangan pengembang aplikasi modern.



Gambar 2. 2. Aplikasi Visual Studio Code

Visual Studio Code digunakan sebagai alat bantu untuk mempercepat proses penulisan dan pengelolaan kode program, terutama system membangun system berbasis PHP dan *MySQL*. Fitur *intellisense* membantu meminimalkan kesalahan penulisan kode, sementara integrasi Git memudahkan pengembang dalam melakukan penyimpanan versi, pembaruan, serta pelacakan perubahan selama proses pembuatan system informasi. Lingkungan kerja yang system juga mendukung pengelolaan file proyek yang kompleks, memungkinkan system antarmuka, pengaturan struktur folder, hingga pengujian fungsi dilakukan dengan lebih efisien. Dengan memanfaatkan berbagai ekstensi seperti PHP Debug, *MySQL*, ataupun Live Server, proses pengembangan menjadi lebih cepat, terorganisir, dan menghasilkan aplikasi yang lebih stabil.

2.7. Aplikasi XAMPP

XAMPP merupakan sebuah paket aplikasi yang berisi Apache, *MySQL*/MariaDB, PHP, dan Perl yang dirancang untuk mempermudah proses pengembangan aplikasi berbasis web. Platform ini bersifat open-source dan dapat dijalankan pada berbagai system operasi, sehingga menjadi pilihan system bagi

pengembang untuk membangun, menguji, dan menjalankan aplikasi web secara *system* sebelum dipublikasikan ke server sebenarnya. XAMPP menyediakan lingkungan server yang lengkap dan mudah digunakan, memungkinkan pengguna melakukan konfigurasi dan pengujian aplikasi tanpa memerlukan perangkat server fisik. Selain itu, antarmuka yang sederhana dan fitur seperti phpMyAdmin memberikan kemudahan dalam pengelolaan *database* sehingga semakin mendukung proses pengembangan aplikasi berbasis PHP dan *MySQL*.



Gambar 2. 3. Aplikasi XAMPP

Penggunaan XAMPP sangat mendukung proses perancangan *system* informasi karena menyediakan lingkungan pengembangan yang stabil dan terintegrasi antara web server, *system* pemrograman, dan *system* basis data. Seluruh komponen yang dibutuhkan untuk membangun aplikasi dapat dijalankan secara *system* sehingga proses pembuatan, pengujian, serta debugging dapat dilakukan dengan lebih cepat dan efisien. Kemudahan akses ke phpMyAdmin membantu dalam merancang struktur *database* obat, sementara modul Apache dan PHP memungkinkan pengembang membangun fitur-fitur seperti pencatatan stok, transaksi keluar-masuk obat, hingga pembuatan laporan secara dinamis. Dengan memanfaatkan XAMPP, pengembangan *system* menjadi lebih terstruktur karena seluruh komponen berjalan dalam satu paket terpadu yang mampu mempercepat alur kerja dan meningkatkan kualitas hasil perancangan aplikasi.

2.8. Penelitian Terdahulu

Tabel 2. 6. Penelitian Terdahulu

No	Penulis	Judul	Tahun	Hasil
1	(Indrawan & Mahendra, 2022)	SISTEM INFORMASI PADA PT. DJAYA BUAH BERSINAR DENPASAR BERBASIS WEB	2022	Sistem informasi berbasis web yang dibangun mampu membantu PT. Djaya Buah Bersinar dalam mengelola persediaan barang secara lebih terstruktur dan akurat dengan menerapkan metode First-In-First-Out, sehingga permasalahan selisih stok dan pencatatan manual dapat diminimalkan. Hasil pengujian menunjukkan bahwa system tersebut efektif mendukung proses pemesanan, penerimaan, pengeluaran, hingga pelaporan barang, sehingga meningkatkan efisiensi dan ketepatan pengelolaan system.
2	(Handayani et al., 2023)	PERANCANGAN SISTEM INFORMASI BARANG BERBASIS WEB MENGGUNAKAN METODE AGILE SOFTWARE	2023	Hasil penelitian menunjukkan bahwa system informasi yang dikembangkan mampu mempermudah proses pengelolaan barang masuk dan barang keluar sehingga data persediaan menjadi lebih tertata, akurat, dan mudah diakses oleh karyawan. Penerapan system informasi ini juga

		DEVELOPMENT		berhasil mengurangi terjadinya kesalahan pencatatan dan duplikasi data, serta meningkatkan efisiensi kerja dalam pengelolaan persediaan di Toko Azura Pekanbaru.
3	(Widiarta et al., 2023)	Rancang Bangun Sistem Informasi Menggunakan Metode Agile Software Development (Studi Kasus Toko Nada)	2023	Sistem informasi yang dibangun mampu membantu Toko Nada dalam mengelola data persediaan barang secara lebih terstruktur, akurat, dan efisien dibandingkan dengan pencatatan manual yang sebelumnya digunakan. Penerapan system ini juga mempercepat proses pencatatan transaksi dan pembuatan laporan, sehingga meminimalkan kesalahan stok serta mendukung pengambilan system terkait pengelolaan persediaan barang.
4	(Rianto, 2023)	Rancang Bangun Sistem Informasi Menggunakan Metode Rapid Application Development	2023	Hasil penelitian menunjukkan bahwa penerapan system informasi yang terkomputerisasi mampu meningkatkan akurasi pencatatan data dan efisiensi pengelolaan persediaan dibandingkan system manual yang sebelumnya digunakan. Selain itu, system informasi yang dikembangkan dengan metode RAD membantu mempercepat proses pengolahan data, meminimalkan

				kesalahan, serta mendukung peningkatan kualitas pelayanan dan kepuasan pelanggan.
5	(Handoko et al., 2024)	Analisis Kinerja Sistem Informasi Papoetoys Jombang Menggunakan Metode Pieces Framework	2024	Hasil penelitian menunjukkan bahwa system informasi (Simtory) yang digunakan di CV. Putra Putri Jombang telah mampu mendukung pengelolaan stok dan penjualan secara terintegrasi antara kantor dan system sesuai dengan kebutuhan system. Evaluasi menggunakan metode PIECES membuktikan bahwa system informasi tersebut memberikan system kepuasan pengguna yang baik pada seluruh system penilaian, sehingga dinilai efektif dalam menunjang proses bisnis system.