

BAB II

TINJAUAN PUSTAKA

2.1 Teori Dasar

2.2.1 Aplikasi

Program yang dirancang untuk memenuhi persyaratan berbagai tugas dan aktivitas disebut aplikasi. Contohnya adalah permainan, bisnis, komunitas, dan aktivitas periklanan (Pramana, 2021). Aplikasi pada dasarnya adalah versi terkomputerisasi dari operasi manual yang dimaksudkan untuk memaksimalkan pemrosesan data. Aplikasi, seperti Microsoft Word dan Excel, adalah perangkat lunak perusahaan komputer yang dirancang untuk menyelesaikan tugas-tugas tertentu (Dhanta, 2009:32). Menurut Anisyah (2000:30), aplikasi adalah penggunaan, pemakaian, atau penambahan data.

2.2.2 Aplikasi Beasiswa

Aplikasi beasiswa adalah sistem atau platform digital yang dirancang untuk memudahkan proses pengelolaan dan penyaluran beasiswa kepada calon penerima. Aplikasi ini umumnya menyediakan berbagai fitur, seperti pengumpulan dan verifikasi data calon penerima, pencocokan kriteria beasiswa dengan profil pelamar, serta pelaporan dan pemantauan kemajuan penerima beasiswa. Dengan menggunakan aplikasi beasiswa, lembaga atau instansi yang menyelenggarakan program beasiswa dapat lebih efisien dalam mengelola proses seleksi dan pendistribusian dana beasiswa, sementara para pelamar dapat dengan mudah mengakses informasi tentang berbagai program beasiswa yang tersedia dan mengajukan permohonan secara online. Aplikasi ini juga membantu dalam meningkatkan transparansi dan akuntabilitas dalam penyaluran dana beasiswa serta memungkinkan penerima beasiswa untuk memanfaatkan kesempatan pendidikan dengan lebih baik (Hasanudin, 2020).

2.2.3 Beasiswa

Menurut buku Dicky Reynaldo yang berjudul "Beasiswa: Panduan Meraih Beasiswa Pendidikan Dalam dan Luar Negeri," beasiswa adalah jenis bantuan keuangan yang diberikan oleh pihak tertentu kepada orang atau organisasi untuk membiayai kelanjutan studi mereka jika memenuhi standar tertentu. Dalam bukunya "Kiat Meraih Beasiswa Pendidikan Dalam dan Luar

Negeri," ia juga mendefinisikan beasiswa sebagai bantuan keuangan yang diberikan oleh organisasi tertentu kepada individu yang memenuhi syarat untuk membantu kegiatan akademis atau pendidikan. Beasiswa, menurut Magdalena H. Lim dan Windya Sari (2018), adalah bentuk bantuan keuangan yang diberikan kepada orang atau organisasi yang memenuhi persyaratan tertentu. Tujuannya adalah untuk meningkatkan akses ke pendidikan, meningkatkan pengalaman akademis, dan mendorong pengejaran nilai-nilai atau aspirasi profesional.

2.2.4 Tracker

Ketika mengacu pada sistem atau komponen yang digunakan untuk melacak atau memantau perilaku atau data tertentu, kata "pelacak" biasanya mengacu pada aplikasi. Tergantung pada permintaan dan jenis aplikasi, "pelacak" dapat digunakan dengan berbagai cara. Istilah "pelacak" memiliki beberapa definisi umum dalam aplikasi.

1. Tracker Pengguna (*User Tracker*)

Ini adalah jenis tracker yang digunakan untuk melacak perilaku pengguna dalam aplikasi. Ini dapat mencakup informasi seperti aktivitas browsing, tindakan yang diambil oleh pengguna, waktu yang dihabiskan di aplikasi, dan preferensi pengguna.

2. Tracker Kinerja (*Performance Tracker*)

Ini adalah jenis tracker yang digunakan untuk memantau kinerja aplikasi. Ini dapat melacak berbagai metrik kinerja, seperti waktu respons server, waktu pemuatan halaman, penggunaan memori, dan lain-lain. Data ini membantu pengembang dalam mengidentifikasi dan memperbaiki masalah kinerja.

3. Tracker Analitik (*Analytics Tracker*)

Ini adalah jenis tracker yang digunakan untuk mengumpulkan data analitik tentang penggunaan aplikasi. Ini dapat mencakup informasi seperti jumlah pengguna unik, frekuensi penggunaan, tingkat retensi pengguna, konversi, dan lain-lain. Data ini membantu pengembang dan pemilik aplikasi dalam memahami perilaku pengguna dan membuat keputusan berdasarkan data.

4. Tracker Lokasi (*Location Tracker*)

Ini adalah jenis tracker yang digunakan untuk melacak lokasi pengguna dalam aplikasi. Ini dapat digunakan dalam aplikasi yang memerlukan informasi lokasi, seperti aplikasi navigasi, layanan pengiriman, dan lain-lain.

Penggunaan "*tracker*" dalam aplikasi sangat bervariasi tergantung pada kebutuhan dan tujuan aplikasi tersebut. Dalam banyak kasus, "*tracker*" membantu dalam meningkatkan pengalaman pengguna, mengoptimalkan kinerja, meningkatkan keamanan, dan membuat keputusan berdasarkan data (Pane, 2020).

2.2.5 Dashboard

Dasbor adalah antarmuka visual yang menyediakan data penting dengan cara yang jelas dan konsisten menggunakan grafik, tabel, dan metrik lainnya. Dasbor digunakan untuk mengelola, memantau, dan menganalisis kinerja sistem, proses, atau aktivitas secara tepat waktu dan efisien. Data yang ditampilkan di dasbor biasanya dikumpulkan dari berbagai sumber, diproses, dan disajikan dengan cara yang jelas dan mudah dimengerti. Fungsi utama dashboard adalah untuk memberikan gambaran menyeluruh mengenai kinerja atau posisi suatu entitas secara real time atau dalam jangka waktu yang telah ditentukan, sehingga pengguna dapat mengambil keputusan yang tepat dan penuh perhitungan. Dasbor sering digunakan dalam dunia bisnis untuk melacak kinerja operasional, pemasaran, keuangan, dan sumber daya manusia suatu organisasi.

Komponen yang umumnya ada di dalam sebuah dashboard dapat bervariasi tergantung pada tujuan dan konteks penggunaannya, namun beberapa komponen Berikut ini yang sering terlihat di dasbor:

1. Data disajikan secara visual menggunakan grafik dan diagram, termasuk diagram dalam bentuk lingkaran, diagram dalam bentuk batang, diagram dalam bentuk garis, dan dalam bentuk diagram lainnya. Hal ini memudahkan pengguna untuk memahami tren dan pola data dengan cepat.

2. Tabel dan Metrik memberikan Tampilan yang rinci tentang data, termasuk angka-angka spesifik dan nilai-nilai kunci. Metrik ini seringkali disajikan dalam bentuk angka atau persentase untuk menggambarkan kinerja atau pencapaian tertentu.

3. KPI (*Key Performance Indicators*) adalah indikator kinerja kunci yang digunakan untuk mengukur kesuksesan atau pencapaian tujuan tertentu. KPI ini dapat disajikan dalam bentuk grafik, angka, atau indikator visual lainnya untuk memudahkan pemantauan.

4. Filter dan Pemfilteran Data Filter memungkinkan pengguna untuk memilih dan menyesuaikan data yang ingin mereka lihat di dalam dashboard. Ini memungkinkan pengguna untuk fokus pada informasi yang relevan dan penting bagi mereka.

2.2.6 Javascript

Bahasa pemrograman seperti JavaScript digunakan untuk membuat web menjadi dinamis. Bersama HTML dan CSS, JavaScript merupakan salah satu dari tiga teknologi web yang mendasar. Untuk membuat halaman web yang responsif dan berinteraksi dengan pengguna, JavaScript dapat dieksekusi di sisi klien, yaitu browser. Selain itu, JavaScript dapat dieksekusi di sisi server untuk mengontrol logika server dengan memanfaatkan Node.js (Devie Rosa Anamisa dkk, 2022).

JavaScript adalah bahasa pemrograman web yang melakukan pemrosesan di sisi client. Hal ini berarti JavaScript dapat dieksekusi langsung di browser tanpa perlu melakukan request ke server seperti PHP. JavaScript akan aktif ketika terjadi event tertentu di halaman web, baik dari interaksi pengguna maupun perubahan pada halaman itu sendiri. Keberadaan jQuery sebagai library telah meningkatkan penggunaan JavaScript, meskipun sekarang popularitasnya mulai tergeser dengan munculnya framework dan library baru yang menawarkan berbagai fitur lebih canggih.

2.2.7 PHP

Bahasa pemrograman web sisi server yang gratis dan bersumber terbuka. PHP adalah bahasa pemrograman tertanam HTML sisi server yang bekerja dengan HTML. Tujuannya adalah untuk menghasilkan halaman web dinamis, yang diperbarui dengan informasi terbaru yang tersedia bagi pelanggan dengan membuat halaman berdasarkan permintaan. Di server tempat mereka dijalankan, semua skrip PHP dieksekusi (Anhar., 2020).

Bahasa pemrograman seperti PHP (*Hypertext Preprocessor*) sering digunakan dalam pengembangan web. PHP bersifat server-side, artinya HTML

dihasilkan oleh kode PHP dan dikirimkan ke browser pengguna di server web. PHP sering digunakan untuk mengelola formulir, membuat situs web yang dinamis dan interaktif, dan mengakses basis data, di antara aktivitas pengembangan web lainnya. Mungkin PHP OOP, atau pemrograman berorientasi objek, digunakan dalam konteks "Aplikasi Absensi Dosen menggunakan PHP OOP" untuk mengatur, memodulasi, dan memelihara kode PHP. Paradigma OOP memungkinkan untuk menangani kompleksitas aplikasi secara lebih efektif dan meningkatkan keterbacaan dan skalabilitas kode. (Habib, Roni, 2020).

2.2.8 Database

Basis data adalah sebuah informasi yang belum diolah atau sekumpulan data terstruktur yang terorganisir yang diintegrasikan ke dalam sebuah sistem dan disediakan untuk manajemen dan akses oleh pengguna atau aplikasi. Karena data dalam basis data biasanya disusun sebagai tabel atau relasi, tindakan seperti penyimpanan, pengambilan, dan pembaruan data dapat dilakukan dengan menggunakan bahasa kueri seperti SQL (Drs. Afrizal Zein et al., 2023).

Selain data, struktur data juga merupakan komponen penting dalam database. Struktur data mengacu pada cara data disimpan dan diorganisir di dalam database, termasuk penggunaan tabel, indeks, relasi, dan elemen-elemen lainnya yang mendefinisikan cara data terkait dan diatur dalam basis data.

Metadata juga merupakan bagian integral dari sebuah database. Metadata berisi informasi tentang struktur dan isi dari database itu sendiri, seperti definisi tabel, kolom, indeks, dan konfigurasi lainnya yang membantu dalam pengelolaan dan pemahaman data yang disimpan.

Terakhir, perangkat lunak yang mengontrol dan memberikan akses ke database. Sistem manajemen basis data (DBMS) bertanggung jawab untuk menegakkan keamanan data, menjamin integritas data, mengontrol akses pengguna, dan menjalankan prosedur pencadangan data jika terjadi kegagalan sistem. Penanganan dan akses data yang efisien dalam database dimungkinkan oleh sistem manajemen database.

2.2.9 MySql

Sistem manajemen basis data sumber terbuka (DBMS) MySQL hadir dengan dua pilihan lisensi yang berbeda: shareware, yang merupakan perangkat

lunak berlisensi dengan penggunaan terbatas, dan perangkat lunak bebas, yang merupakan perangkat lunak gratis. MySQL dapat digunakan tanpa biaya baik untuk keperluan pribadi maupun komersial di bawah ketentuan GNU General Public License (GPL), sehingga tidak perlu membeli lisensi. Seperti yang ditunjukkan sebelumnya, MySQL dikategorikan sebagai RDBMS (Sistem Manajemen Basis Data Relasional), dan dengan demikian, ide-ide seperti baris, kolom, dan tabel digunakan ketika menggunakan aplikasi basis data ini. Satu atau lebih tabel membentuk sebuah database, misalnya, dalam MySQL (Respaty Namruddin et al., 2023).

2.2.10 Laravel

Framework Laravel dibuat pada tahun 2011 oleh programmer Amerika, Taylor Otwell. Sejak dirilis untuk umum, Laravel secara progresif memenangkan hati para pengembang perangkat lunak di mana-mana. Daya tariknya telah berkembang dari waktu ke waktu, memosisikannya sebagai salah satu kerangka kerja teratas di antara para pengembang perangkat lunak di seluruh dunia. Pada saat buku ini diterbitkan, Laravel telah melampaui banyak kerangka kerja PHP yang telah ada sebelumnya untuk mengukuhkan posisinya sebagai standar industri (Rohi Abdulloh, 2022).

2.2.11 Software Development Life Cycle (SDLC)

Siklus Hidup Pengembangan Perangkat Lunak dan pengembangan perangkat lunak (SDLC) adalah untuk mengatasi masalah secara efisien dan menciptakan sistem berkualitas tinggi yang memenuhi persyaratan pengguna atau tujuan sistem. Dengan menggunakan berbagai model atau metodologi, perangkat lunak dibuat dan dimodifikasi sebagai bagian dari Siklus Hidup Pengembangan Perangkat Lunak (SDLC), sebuah proses rekayasa perangkat lunak. Paradigma SDLC berfungsi sebagai fondasi untuk berbagai metodologi pengembangan perangkat lunak yang populer, seperti SDLC tradisional, SDLC prototipe, dan SDLC berorientasi objek (Nazaruddin et al., 2022).

Menurut buku Syafitri, Siregar, dan Muharni Sistem Informasi Manajemen (2022), Sistem informasi disusun, dirancang, dikembangkan, diuji, dan diimplementasikan dengan menggunakan Siklus Hidup Pengembangan

Perangkat Lunak, sebuah prosedur yang teratur. SDLC sangat penting dalam memastikan bahwa sistem yang dibuat:

- Memenuhi kebutuhan dan tujuan organisasi serta penggunanya.
- Sesuai dengan anggaran dan tenggat waktu yang telah ditetapkan.
- Berkualitas tinggi dan terbebas dari bug.
- Mudah digunakan dan dipelihara.

Penerapan SDLC umumnya terbagi menjadi beberapa tahapan:

1. Perencanaan: Pada tahap ini, dilakukan identifikasi kebutuhan dan tujuan sistem, analisis kelayakan, dan pengembangan rencana proyek yang komprehensif.
2. Analisis: Tahap ini melibatkan studi mendalam terkait kebutuhan sistem, proses bisnis, dan data yang ada.
3. Desain: Spesifikasi sistem yang rinci, termasuk arsitektur sistem, desain antarmuka pengguna, dan desain database, dibuat pada tahap ini.
4. Pengembangan: Tahap ini meliputi pemrograman dan pengujian unit sistem.
5. Pengujian: Sistem secara menyeluruh diuji untuk memastikan bahwa semua persyaratan terpenuhi dan sistem berfungsi dengan baik.
6. Penerapan: Sistem dipasang di lingkungan produksi dan pelatihan pengguna dilakukan.
7. Pemeliharaan: Perbaikan bug, penambahan fitur baru, dan pemberian dukungan kepada pengguna merupakan fokus utama pada tahap ini.

Manfaat utama penerapan SDLC:

- Meningkatkan peluang keberhasilan proyek.

- Meminimalkan risiko kegagalan proyek.
- Memastikan sistem yang dikembangkan memenuhi kebutuhan dan tujuan organisasi.

Syafitri, Siregar, & Muharni (2022) menegaskan pentingnya penerapan SDLC dalam pengembangan sistem informasi. Mereka menyatakan bahwa SDLC membantu organisasi mengembangkan sistem informasi yang efektif dan efisien, sehingga mendukung pencapaian tujuan organisasi.

2.2.12 Rapid Application Development (RAD)

Pada pertengahan tahun 1970-an, Pusat Pengembangan Sistem Dan Gielan di New York Telephone Co. pertama kali mengidentifikasi Pengembangan Aplikasi Cepat (RAD) sebagai metodologi pengembangan perangkat lunak yang sukses. RAD mengintegrasikan konsep dari pendekatan air terjun dengan pengembangan cepat, dan telah terbukti efektif dalam sejumlah penerapan proyek (Hussain et al., 2022). Pendekatan RAD memprioritaskan pekerjaan dalam jangka waktu yang terbatas dan menerapkan fitur-fitur model air terjun:

1) Perencanaan Kebutuhan (*Requirement Planning*)

Pada tahap awal ini, pengembang bersama dengan pemangku kepentingan melakukan pertemuan untuk mengumpulkan dan memahami kebutuhan bisnis serta teknis proyek. Diskusi ini bertujuan untuk mengidentifikasi kebutuhan utama dan menetapkan tujuan proyek secara jelas, jernih. Langkah ini berujung pada dokumen persyaratan menyeluruh yang akan mengarahkan proses pengembangan.

2) Desain Pengguna (*User Design*)

Tahap ini melibatkan kolaborasi yang intens antara pengguna dan pengembang untuk menciptakan prototipe, model, dan arsitektur awal aplikasi. Prototipe ini dibuat dengan cepat untuk mendapatkan umpan balik langsung dari pengguna. Memastikan aplikasi yang dihasilkan benar-benar memenuhi kebutuhan dan harapan konsumen adalah tujuan utama. Prototipe yang mudah dimodifikasi dan disempurnakan sebagai tanggapan terhadap umpan balik pelanggan adalah produk akhir dari tahap ini.

3) Pembangunan dan Konstruksi (*Construction*)

Pada titik ini, pengembang mulai menggunakan alat dan teknik RAD untuk mendesain sistem yang sudah jadi. Aplikasi dengan fungsionalitas lengkap dibuat dari prototipe yang telah dibuat sebelumnya. Untuk mempercepat proses pembangunan, pengembang menggunakan alat pengembangan visual dan komponen yang dapat digunakan kembali. Fase ini bertujuan untuk mengembangkan sistem dengan cepat sambil mempertahankan kualitasnya yang tinggi. Produk akhirnya adalah aplikasi yang hampir selesai, namun masih membutuhkan beberapa penyesuaian kecil sebelum dapat digunakan.

4) Implementasi (*Cutover*)

Aplikasi yang dikembangkan diuji dengan cermat selama tahap implementasi, yang merupakan tahap terakhir sebelum diterapkan ke lingkungan produksi. Untuk memastikan program bebas dari kesalahan dan beroperasi sebagaimana mestinya, pengujian dilakukan. Selain itu, pengguna menerima instruksi yang mereka butuhkan untuk mengoperasikan aplikasi secara efisien. Memastikan aplikasi siap untuk digunakan dalam lingkungan perusahaan yang sebenarnya adalah tujuan dari fase ini. Hasil akhirnya adalah aplikasi yang sudah siap digunakan dan diimplementasikan dalam lingkungan bisnis yang sebenarnya. Model RAD mengikuti tahapan pengembangan sistem secara umum, namun mampu menggunakan kembali komponen yang sudah ada sehingga mengurangi kebutuhan untuk membuat dari awal, menghasilkan pengembangan yang lebih cepat dan efisien.

Metode Pengembangan Iteratif:

Pendekatan pengembangan perangkat lunak yang memecah proses pembangunan aplikasi besar menjadi bagian-bagian kecil disebut model pengembangan iteratif. Setiap bagian, yang disebut sebagai "iterasi", mencakup seluruh proses perencanaan, desain, pengembangan, dan pengujian.

Kelebihan Model Iteratif:

- Karena prosesnya dilakukan secara bertahap, model ini memungkinkan identifikasi dan perbaikan kesalahan di tahap awal untuk mencegahnya terlanjur mempengaruhi tahapan selanjutnya.

- Dengan pendekatan iteratif, manajemen versi menjadi lebih mudah dengan memastikan setiap iterasi baru merupakan perbaikan dari iterasi sebelumnya.
- Jika iterasi baru mengalami kegagalan, iterasi sebelumnya dapat diimplementasikan kembali dengan kerugian minimal.
- Mendapatkan masukan dari pengguna lebih mudah dilakukan, termasuk tanggapan mereka terhadap produk saat ini dan harapan mereka terhadap produk di masa depan. Ini memungkinkan untuk melakukan perbaikan dan perubahan yang diperlukan.
- Menghemat waktu yang biasanya digunakan untuk proses dokumentasi dalam model waterfall.

2.2.13 *Unified Modelling Language (UML).*

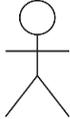
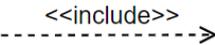
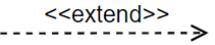
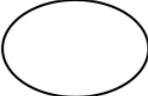
Untuk mendefinisikan komponen sistem yang berbeda, salah satu bahasa yang populer adalah Unified Modeling Language (UML). Pengembang perangkat lunak menggunakan UML untuk memodelkan berbagai bagian sistem seperti diagram aktivitas, diagram kasus penggunaan, dan hierarki kelas. Model-View-Model Interaction (UML) menyediakan dua metode utama untuk mendukung analisis, desain, dan dokumentasi sistem. Ini adalah alat yang sangat membantu untuk menjelaskan sistem yang rumit, terutama sistem bisnis dan perangkat lunak.

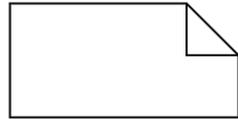
1. Use Case Diagram

Salah satu jenis diagram UML adalah diagram use case, yang menunjukkan bagaimana sistem berkomunikasi dengan pengguna atau sistem lain. Contoh penggunaan mencakup penjelasan rinci tentang operasi sistem serta interaksi pengguna atau aktor lainnya. Diagram ini sering digunakan untuk membantu para pemangku kepentingan berkomunikasi dan memahami kebutuhan fungsional sistem selama fase analisis dan desain. Untuk memastikan sistem memenuhi kebutuhan bisnis, grafik ini juga membantu dalam identifikasi mereka. Simbol-simbol diagram ini juga dapat dijelaskan.

Table 2. 1 Simbol Dari Use Case Diagram

Symbol	Name	Description
--------	------	-------------

	<i>Actor</i>	Memperjelas peran yang dimainkan pengguna saat berinteraksi dengan kasus penggunaan.
	<i>Generalization</i>	Hubungan di mana sebuah entitas turunan (descendant) mengadopsi perilaku dan struktur kekuatan dari entitas yang lebih tinggi atau induk (ancestor).
	<i>Include</i>	Menjelaskan dengan jelas penggunaan kasus dari sumber yang spesifik.
	<i>Extend</i>	menjelaskan bagaimana kapabilitas use case asal diperluas oleh use case tujuan pada saat tertentu.
	<i>Association</i>	Ini adalah usaha untuk menyambungkan satu objek dengan objek lainnya.
	<i>Use Case</i>	Penjelasan mengenai rangkaian langkah yang dieksekusi oleh suatu sistem yang menghasilkan hasil yang dapat diukur untuk seorang pelaku.
	<i>Collaboration</i>	Kolaborasi antara aturan-aturan dan elemen-elemen lain menghasilkan perilaku yang lebih kompleks daripada yang dapat dicapai oleh aturan dan elemen secara terpisah, yang dikenal sebagai sinergi.



Note

Komponen fisik yang ada ketika program berjalan dan merefleksikan sebuah sumber daya komputasi.

2. Activity Diagram

Jenis diagram UML tertentu yang disebut diagram aktivitas sering digunakan untuk mengilustrasikan urutan langkah dalam proses atau alur kerja. Diagram ini sangat jelas menunjukkan bagaimana sistem bekerja dan bagaimana objek atau item yang berbeda berinteraksi satu sama lain.

Activity diagram ini bermanfaat untuk mengilustrasikan proses-proses kompleks serta membantu dalam mengidentifikasi masalah yang mungkin terjadi dalam alur kerja, sehingga memungkinkan hal itu terjaditerciptanya solusi yang lebih berhasil .untuk menciptakan solusiyang lebih sukses . Hal ini juga berfungsi sebagai alat komunikasi bagi pengembang, pemangku kepentingan , dan klien, untuk memastikan bahwa semua persyaratan sistem terpenuhi .

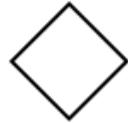
Table 2. 2 Simbol Dari Activity Diagram

Symbol	Name	Description
	<i>Start</i>	Kondisi awal dari aktivitas sistem adalah Diagram aktivitas memiliki kondisi awal.
	<i>Activity</i>	Sistem yang menjalankan suatu aktivitas, biasanya didahului oleh kata kerja.
	<i>End</i>	Sebuah sistem harus memiliki kondisi akhir, yang merupakan kondisi akhir yang dicapai.



Flow final

Simbol Flow Final menandakan akhir dari aliran proses atau aktivitas, menunjukkan bahwa proses telah selesai dan tidak ada langkah lagi yang perlu dilakukan.



Decision

Ketika ada beberapa beberapa pilihan untuk suatu untuk, situasi tersebut disebut asosiasi atau percabangan. suatu aktivitas, situasi tersebut disebut asosiasi atau percabangan .

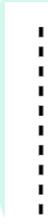
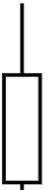
3. Sequence Diagram

Diagram urutan adalah diagram UML yang menunjukkan hubungan antara berbagai hal. Dalam gambar ini, pesan atau pemanggilan metode yang berjalan di antara objek-objek diwakili oleh panah horisontal dan objek-objek itu sendiri sebagai kotak vertikal. Garis waktu yang disejajarkan secara vertikal menampilkan waktu eksekusi, sementara serangkaian panah menunjukkan urutan pesan atau pemanggilan metode.

Diagram urutan adalah alat yang hebat untuk menggambarkan hubungan yang rumit antara komponen sistem. Dengan memanfaatkannya, pengembang dapat mendeteksi masalah apa pun dengan desain sistem sebelum implementasi dan memberikan penjelasan rinci tentang urutan pemanggilan pesan atau metode. Selain sebagai alat referensi yang berguna, diagram ini juga akan membantu dalam perluasan sistem dan pemeliharaan berkelanjutan. Berikut ini adalah simbol-simbolnya:

Table 2. 3 Simbol Dari Sequence Diagram

Symbol	Name	Description
	<i>Objek</i>	Deklarasikan bahwa objek yang berinteraksi dengan atau menyampaikan pesan juga dapat

	<p>berupa instance kelas, dengan nama objek diawali dengan titik koma.</p>
 <p><i>Aktor</i></p>	<p>Aktor dalam diagram urutan adalah pihak luar yang berkomunikasi dengan sistem atau bagian tertentu. Tergantung pada sistem yang dianalisis, aktor dapat mencakup pengguna (orang), sistem lain, atau perangkat eksternal.</p>
 <p><i>Lifeline</i></p>	<p>Menggambarkan kehidupan atau keberadaan suatu objek. Garis hidup biasanya berupa garis vertikal bertitik yang memanjang dari suatu objek.</p>
 <p><i>Activation</i></p>	<p>Bentuk persegi panjang yang diletakkan pada garis hidup menunjukkan bahwa objek akan melakukan suatu tindakan dan digambarkan dalam keadaan aktif ketika berinteraksi dengan pesan.</p>
 <p><i>Message</i></p>	<p>Pesan, yang dinyatakan sebagai tanda panah, menggambarkan komunikasi objek.</p>

2.2.14 Black Box Testing

Perangkat lunak dievaluasi menggunakan teknik yang disebut pengujian "kotak hitam", yang mengabaikan struktur internal dan kode demi menguji fungsionalitas sistem. Pengujian dilakukan dengan menggunakan strategi ini bahkan ketika pemrosesan data internal perangkat lunak tidak diketahui. Penguji tidak melakukan apa pun selain memasukkan data ke dalam antarmuka perangkat

lunak dan melihat output yang dihasilkan. Tujuan utama dari pengujian ini adalah untuk memverifikasi bahwa program sesuai dengan permintaan pengguna dan beroperasi sesuai dengan spesifikasi yang telah ditetapkan; pengujian ini tidak melihat secara spesifik bagaimana program tersebut diimplementasikan. Pengujian kotak hitam memungkinkan untuk menguji dari sudut pandang pengguna akhir, menjamin pengalaman pengguna yang memuaskan.

Pengujian *Black Box Testing* dan klasifikasinya bertujuan untuk meningkatkan kepercayaan pengembang perangkat lunak terhadap fungsi-fungsi aplikasi, serta memastikan bahwa aplikasi dapat beroperasi dengan baik dalam berbagai kondisi dan lingkungan (Manuaba, 2023).

2.1 Tinjauan Studi

Beberapa publikasi ilmiah yang berkaitan dengan subjek penelitian telah dikonsultasikan dalam pengembangan laporan tugas akhir ini. Publikasi-publikasi tersebut antara lain:

- 1) Penelitian pertama yaitu pengembangan dashboard yang dilakukan oleh Dede Bagus Setiawan, dengan judul **“PENGEMBANGAN DASHBOARD INTERAKTIF COVID-19 SEBAGAI ALAT INFORMASI MASYARAKAT DALAM MELIHAT PANDEMI DI INDONESIA”** yang dimuat dalam jurnal *Technologia: Jurnal Ilmiah* Vol.2 No.1 Tahun 2023. Publikasi ini menjelaskan tentang pengembangan Dasbor Interaktif Covid-19 oleh Dede Bagus Setiawan dari Universitas Islam 45 Bekasi sebagai alat informasi publik untuk melacak epidemi di Indonesia. Penciptaan alat yang memungkinkan masyarakat umum untuk mendapatkan informasi yang lebih tepat mengenai kondisi terkini dan lintasan pandemi Covid-19 adalah tujuan dari proyek ini. Informasi tentang tingkat vaksinasi, jumlah kasus, dan kebijakan lokal, nasional, dan internasional semuanya ditampilkan di dasbor ini, yang mengumpulkan data dari berbagai sumber tepercaya dan menyajikannya dalam bentuk visualisasi yang mendidik. Selain itu, dengan menggunakan visualisasi data yang jelas, dasbor ini menawarkan wawasan tentang dampak pandemi terhadap sektor sosial, pendidikan, kesehatan, dan ekonomi.

- 2) Pembahasan kedua yaitu membahas tentang pengembangan dashboard di Universitas Ibnu Sina Batam yang di buat Lastris Yanti. (2019). **“PENGEMBANGAN DASHBOARD UNIVERSITAS IBNU SINA BATAM BERBASIS WEB MOBILE”** (UIS), Artikel ini muncul di Technologia volume 3, edisi 2 tahun 2019: Jurnal Ilmiah. Menciptakan alat yang memungkinkan masyarakat untuk mendapatkan informasi yang lebih akurat mengenai keadaan dan arah masa depan epidemi Covid-19 adalah tujuan utama proyek ini. Biro Administrasi dan Akademik dan Kemahasiswaan (BAAK) kampus menjadi subjek penelitian. Proses ini disebut OOAD UML untuk pemodelan dalam menangani persyaratan, analisis, desain, implementasi, pengujian, dan penyebaran. Seluruh civitas akademika di Universitas Ibnu Sina akan memiliki akses informasi yang mudah dan real-time melalui sistem informasi ini, dengan harapan data yang diberikan akurat dan dapat dipercaya.
- 3) Penelitian ketiga yaitu membahas tentang Perancangan Aplikasi Tracking Sales di PT. Infinity yang di lakukan oleh Yusup Nur Abdurrahman, Sari Susanti. (2021). **” PERENCANGAN APLIKASI TRACKING SALES BERBASIS ANDROID PADA PT. INFINITY”**. E-PROSSIDING SISSTEM INNFORMASI Vol. 2, No. 2, Desember 2021. Pembuatan aplikasi pelacakan penjualan berbasis Android di PT Infinity, sebuah perusahaan pengembangan sumber daya manusia, adalah subjek dari jurnal ini. Dalam konteks bekerja sama dengan perusahaan besar seperti Akulaku, sistem pemantauan dan pengawasan terhadap salesman sangatlah penting, dan metode yang ada saat ini masih mengandalkan WhatsApp GroupChat, yang dapat menyebabkan pemalsuan informasi dan menyulitkan pemantauan. Untuk itu, aplikasi ini dibuat dengan menggunakan metode Waterfall dan Java, bahasa pemrograman berbasis Android, serta dilengkapi dengan teknologi Google Maps API, yang memungkinkan penyediaan informasi posisi salesman secara langsung. Diharapkan dengan adanya aplikasi ini, proses monitoring dan pengawasan terhadap salesman menjadi lebih mudah.
- 4) Penelitian berikutnya yaitu penelitian oleh Anastasia Meyliana, Lina Ayu Safitri, Anik Andriani dengan judul **"PENERAPAN METODE RAPID APPLICATION DEVELOPMENT (RAD) DALAM PERANCANGAN WEBSITE PT SOVVA KREASI INDONESIA"**. Indonesian Journal on

Networking and Security - Volume 11 No 3 – 2022. Pembuatan situs web untuk PT Sovva Kreasi Indonesia, sebuah bisnis yang menawarkan produk perawatan wajah dan tubuh dengan merek Sovva Beauty, adalah subjek dari jurnal ini. Untuk memasarkan dan meningkatkan penjualan produk perusahaan dengan lebih baik, desain situs web diselesaikan dalam penelitian ini dengan menggunakan metodologi Pengembangan aplikasi secara cepat (RAD). Metode black box digunakan untuk pengujian, dan pendekatan analisis PIECES digunakan untuk memeriksa masalah. Melalui wawancara, studi literatur, dan observasi, situs web yang dibuat oleh penelitian ini memiliki potensi untuk meningkatkan penjualan, mengiklankan produk, memperluas audiens divisi pemasaran, dan membuat administrasi data produk serta penyajian laporan menjadi lebih mudah.

- 5) Jurnal oleh Mohammad Reza Fahlevi, Dwi Rizky Rahmawati, Binti Mamluatul Karomah. (2020). ” **RENCANG BANGUN SISTEM INFORMASI PEMBAYARAN SPP BERBASIS WEB MENGGUNAKAN FRAMEWORK LARAVEL 9**”. Judul artikel jurnal ini adalah Jurnal Ilmu Komputer dan Sistem Informasi (JIKOMSI) Volume 6, Nomor 3 Oktober-Desember 2023. Tiga orang peneliti dari Program Studi Teknik Informatika Universitas Nahdlatul Ulama Indonesia, Jakarta, Indonesia melakukan penelitian yang merinci bagaimana menggunakan framework Laravel 9 untuk membuat Sistem Informasi Pembayaran Sumbangan Pembinaan Pendidikan (SPP) secara online. Penelitian tersebut dijelaskan dalam artikel ini. Melalui penggunaan teknologi ini, mereka berharap dapat membantu sekolah dalam mengimplementasikan perubahan terbaru pada administrasi dan sintesis informasi pembayaran SPP.