

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **1.1 Penelitian Sebelumnya**

Penelitian terdahulu ini merupakan langkah yang penting dalam proses penelitian karena memberikan landasan teoritis yang diperlukan dengan meninjau penelitian-penelitian terdahulu yang relevan dengan topik penelitian, penelitian saat ini dapat memperoleh pemahaman yang lebih baik tentang konteks dan permasalahan yang sedang dihadapi. Tinjauan Pustaka membantu dalam mengidentifikasi kerentanan atau kesenjangan dalam penelitian yang sudah ada, yang dapat dijadikan sebagai landasan untuk penelitian yang sedang dilakukan [9].

Studi-studi sebelumnya yang mengulas penggunaan metode SDLC dalam merancang sistem informasi dan penjualan memberikan pandangan yang berharga bagi penelitian ini. Penelitian dapat diperoleh wawasan mengenai keberhasilan dan tantangan yang dihadapi dalam menerapkan metode SDLC dalam konteks penjualan. Tinjauan pustaka ini memiliki batasan tersendiri. Fokus utama dari penelitian-penelitian terdahulu mungkin lebih terbatas pada perbandingan antara metode yang digunakan dan hasil yang diperoleh [10].

Penelitian ini dapat memperluas cakupan tinjauan pustaka tersebut dengan menggali lebih dalam aspek-aspek yang mungkin belum tercakup secara memadai. Tinjauan pustaka yang komprehensif memungkinkan penelitian ini untuk menemukan kesempatan-kesempatan untuk berinovasi dan menyumbangkan kontribusi baru dalam penelitian yang sudah ada. -hal ini memungkinkan penelitian untuk tidak hanya mereplikasi temuan yang sudah ada, tetapi juga untuk mengeksplorasi area baru dan memberikan wawasan yang lebih dalam atau solusi yang lebih inovatif terhadap tantangan yang ada dalam perancangan sistem informasi dan penjualan [11]. Tabel 2.1 menunjukkan penelitian terkait metode uji pada sistem .

Tabel 2.1 Tinjauan Pustaka

| <i>No</i> | <i>Judul</i>   | <i>Compare</i>  | <i>Contrast</i>  | <i>Criticize</i>   | <i>Synthesis</i>  | <i>Summaries</i>   |
|-----------|--|---|--|--|---|--|
| 1         | Perancangan Sistem Informasi Untuk Mengontrol Sistem Pembelian, Persediaan Dan Penjualan Dengan Metode <i>System Development Life Cycle (SDLC)</i> [12]. | Perbandingan tersebut menjelaskan pentingnya teknologi informasi sebagai sumber daya dan elemen kunci dalam strategi bersaing Toko Bangunan Subur. Hal ini sesuai dengan pandangan umum bahwa teknologi informasi berperan penting dalam mengoptimalkan kinerja bisnis. | Pentingnya teknologi informasi dalam mengintegrasikan kebutuhan informasi toko bangunan Subur, tidak ada perbandingan langsung dengan metode atau pendekatan alternatif lebih lanjut, ringkasan ini tidak menjelaskan keterbatasan atau kelemahan yang mungkin muncul dalam implementasi sistem. | Pada penelitian ini dapat memberikan gambaran tentang pentingnya teknologi informasi dan penggunaan <i>System Development Life Cycle (SDLC)</i> dalam manajemen pembelian, gudang dan distribusi di Toko Subur Building. Namun, ringkasan ini tidak memberikan informasi rinci tentang bagaimana SDLC diimplementasikan atau bagaimana sebenarnya sistem manajemen gudang bekerja. | Analisis ini fokus pada pentingnya teknologi informasi dan SDLC dalam menghadapi tantangan baru di toko bangunan subur. Hal ini memperkuat pemahaman bahwa mengintegrasikan persyaratan informasi dengan pendekatan sistem yang tepat dapat meningkatkan manajemen inventaris dan menghindari tingkat stok. | Kesimpulan ini menjelaskan pentingnya penerapan teknologi informasi dan SDLC dalam mengelola pembelian, inventaris dan penjualan di Toko Subur <i>Building</i> . Sistem manajemen Gudang yang digunakan membantu mengontrol Gudang untuk menghindari kekosongan dan keamanan Gudang. |

| <i>No</i> | <i>Judul</i>   | <i>Compare</i>  | <i>Contrast</i>  | <i>Criticize</i>  | <i>Synthesis</i>   | <i>Summaries</i>   |
|-----------|--|---|--|---|--|--|
| 2         | Perancangan Sistem Informasi Pengarsipan Data Penduduk Pada Kantor Camat Bilah Hulu Kabupaten Labuhan Batu Dengan Metode <i>System Development Life Cycle (SDLC)</i> [13]. | Perbandingan ini menjelaskan sistem informasi sebagai sistem organisasi internal yang memenuhi berbagai kebutuhan pemrosesan acara harian, operasional, administratif, dan strategis. Perbandingan dapat dibuat dengan definisi dan karakteristik sistem informasi di organisasi lain atau dalam konteks industri yang berbeda. | Meskipun menjelaskan penggunaan Siklus Hidup Pengembangan Sistem (SDLC) sebagai pendekatan untuk analisis dan desain sistem, ringkasan ini tidak dapat dibandingkan dengan metodologi lain yang dapat digunakan dalam penelitian atau pengembangan sistem. | Hal ini memberikan gambaran tentang sistem informasi dan SDLC, tetapi tidak memberikan penjelasan rinci tentang bagaimana implementasi SDLC menggunakan model Waterfall dapat memberikan manfaat khusus untuk penelitian tersebut. Selain itu, seseorang dapat mengkritik batasan model air terjun yang dinamis namun berurutan dalam konstruksi perangkat lunak. | Menyajikan definisi sistem informasi sebagai sistem yang memenuhi berbagai kebutuhan organisasi dan menjelaskan penggunaan SDLC dengan pendekatan waterfall dalam penelitian ini. Pendekatan R&D digunakan untuk mengembangkan sistem untuk fase tertentu. Pada dasarnya ringkasan ini merangkum konsep sistem informasi | Kesimpulan ini menjelaskan sistem informasi adalah sistem internal organisasi yang menanggapi berbagai kebutuhan pemrosesan transaksi dan mendukung kegiatan operasional dan strategis. Dalam penelitian ini, pendekatan SDLC digunakan dengan model <i>waterfall</i> , model klasik, dinamis, dan Metode penelitian yang digunakan adalah penelitian pengembangan R&D |
| 3         | Pengembangan Sistem Informasi Arus Kas Dengan Metode SDLC ( <i>System Development Life</i>   | Manajemen arus kas yang baik digambarkan sebagai kunci keberlanjutan organisasi dalam menjaga operasi keuangan.   | Arus kas dianggap sangat penting bagi bisnis, terutama dalam kegiatan operasional yang bergantung pada   | Artikel ini membahas tentang pentingnya pengembangan sistem informasi arus kas untuk  | Pembahasan ini menekankan pentingnya manajemen arus kas yang baik untuk bisnis dan pengembangan sistem informasi arus kas untuk  | Kesimpulan ini membahas tentang Manajemen arus kas yang baik sangat penting bagi suatu organisasi dalam  |

| No | Judul  | Compare  | Contrast  | Criticize  | Synthesis  | Summaries   |
|----|--|--|---|--|--|---|
|    | <i>Cycle</i> ) pada Madin Al-Jannah [14].  | Perbandingan dapat ditarik dengan cara lain di mana manajemen arus kas penting untuk pemeliharaan operasi keuangan organisasi.   | informasi kas. Kontras yang dapat dibuat dengan menjelaskan apa yang terjadi ketika informasi arus kas tidak dikelola dengan baik dan bagaimana kegagalan dalam manajemen arus kas dapat mengakibatkan kerugian pada bisnis.  | menjaga keakuratan informasi arus kas masuk dan keluar. Namun, kritik tersebut tidak ditujukan kepada kelemahan atau tantangan yang mungkin timbul selama proses pengembangan sistem. <i>online</i> .  | mengurangi kesalahan dan kerugian. Ini mencerminkan pemahaman menyeluruh tentang hubungan antara manajemen arus kas, sistem informasi, dan kesuksesan bisnis.  | menjaga operasi keuangan. Mengembangkan sistem informasi arus kas membantu menjaga keakuratan informasi arus kas, mengurangi kesalahan dan mencegah kerugian pada bisnis. Penggunaan SDLC dalam pengembangan  |
| 4  | Perancangan Sistem Informasi Kepegawaian Dengan Metode <i>System Development Life Cycle</i> Pada Kantor Kelurahan Cimuning [15]. | Perbandingan dalam konteks ini, teknologi informasi di pemerintahan dianggap sebagai faktor yang mendukung efisiensi, produktivitas, dan efisiensi kerja pegawai. Perbandingan dapat dilakukan dengan membandingkan dampak teknologi informasi di instansi pemerintah. | Meskipun teknologi informasi dipandang penting untuk meningkatkan efisiensi kerja di instansi pemerintah, namun terdapat perbedaan antara instansi yang sudah mengoptimalkan penggunaan teknologi dengan instansi yang masih bermasalah atau keterbatasannya dalam penggunaannya. | Kritik ini tidak kalah dengan kritik terhadap penggunaan teknologi informasi di otoritas publik. Namun, kritik konstruktif dapat diberikan pada setiap tantangan atau kendala yang mungkin dihadapi dalam pengembangan dan implementasi sistem informasi sumber daya | Hal ini menjelaskan bahwa teknologi informasi dalam pemerintahan dapat meningkatkan efisiensi pegawai dan mendukung semua tugas. Tujuannya adalah untuk membuat sistem informasi kepegawaian menggunakan pemrograman <i>VisualBasic.Net</i> yang memfasilitasi dan meningkatkan pemrosesan | Teknologi informasi digunakan dalam otoritas publik untuk meningkatkan efisiensi karyawan dan mendukung tugas-tugas yang ada. Kesimpulan ini membahas tentang pengembangan sistem informasi kepegawaian dengan menggunakan pemrograman <i>VisualBasic.Net</i> . |

| <i>No</i> | <i>Judul</i>  | <i>Compare</i>  | <i>Contrast</i>  | <i>Criticize</i>   | <i>Synthesis</i>  | <i>Summaries</i>   |
|-----------|---|---|--|--|---|--|
| 5         | Rancangan Sistem Informasi Persediaan Stok Obat Dengan Metode <i>System Development Life Cycle</i> Di Apotek Nusa Farma Nusa Penida [16]. | Rancangan sistem informasi persediaan obat Apotek Nusa Farma Nusa Penida dapat dibandingkan dengan rancangan sistem informasi persediaan obat apotek lain. Perbandingan ini dapat mencakup fitur- fitur yang ditawarkan, pendekatan yang digunakan dalam pengembangan sistem, dan manfaat yang diharapkan dari penerapan sistem tersebut. | Perencanaan sistem informasi persediaan obat Apotek NusaFarma Nusa Penida dapat dibandingkan dengan keadaan sebelum sistem tersebut diimplementasikan. Perbedaannya terletak pada efisiensi pengelolaan persediaan obat, keakuratan informasi persediaan obat, dan aksesibilitas bagi pengguna sistem. | manusia.<br>Tidak ada kritik dalam judulnya. Namun, poin kritik dapat berupa kekurangan pada sistem sebelumnya, kelemahan pada metode SDLC yang digunakan atau kemungkinan kesalahan dalam pemberian obat. | Perancangan sistem informasi persediaan obat di Apotek Nusa Farma Nusa Penidamenggunakan pendekatan <i>System Development Life Cycle</i> (SDLC) untuk mengembangkan sistem yang efektif. Tujuan utamanya adalah untuk meningkatkan pengelolaan persediaan obat, meningkatkan akurasi data persediaan obat dan memberikan manfaat bagi pengguna sistem. Menggabungkan elemen SDLC dan kebutuhan khusus | Kesimpulan ini membahas tentang Metodologi System Development Life Cycle (SDLC) digunakan dalam perancangan sistem informasi persediaan obat Apotek Nusa Farma Nusa Penida untuk meningkatkan pengelolaan persediaan Tujuan utamanya adalah untuk meningkatkan efisiensi dan akurasi data persediaan obat. Rancangan sistem mencerminkan komitmen apotek |

| <i>No</i> | <i>Judul</i>  | <i>Compare</i>   | <i>Contrast</i>  | <i>Criticize</i>   | <i>Synthesis</i>  | <i>Summaries</i>  |
|-----------|---|--|--|--|---|---|
| 6         | Perancangan Sistem Informasi Penjadwalan Mata Kuliah Dengan Metode SDLC ( <i>System Development LifeCycle</i> ) [17]. | Perbandingan dalam konteks ini dapat mengacu pada aspek perkuliahan mahasiswa dengan ketersediaan waktu dan kurikulum terkait dosen. Waspadai konflik apa pun antara jadwal kursus siswa dan kurikulum instruktur untuk menghindari konflik. | \Dalam konteks ini terlihat adanya kebutuhan perkuliahan mahasiswa dengan ketersediaan waktu dan kurikulum dosen. Ketika ada perbedaan yang signifikan antara kedua aspek ini, akan sulit untuk menemukan pasangan | beberapa potensi masalah yang mungkin timbul adalah kesulitan dalam mengatasi konflik penjadwalan antara kuliah mahasiswa dan ketersediaan waktu dan kurikulum fakultas. | Saat membuat jadwal, aspek mahasiswa dan dosen harus diperhatikan. Perspektif mahasiswa juga mencakup kuliah-kuliah lain yang diberikan oleh mahasiswa, sedangkan perspektif dosen mencakup kemungkinan fakultas tidak dapat mengajar dengan baik karena. | Kesimpulan ini membahas tentang membuat rencana perkuliahan harus memperhatikan aspek mahasiswa dan dosen. Metode SDLC digunakan sebagai pendekatan |

| <i>No</i> | <i>Judul</i>  | <i>Compare</i>   | <i>Contrast</i>  | <i>Criticize</i>   | <i>Synthesis</i>  | <i>Summaries</i>   |
|-----------|---|--|--|--|---|--|
| 7         | Perancangan Sistem Informasi Penjualan CV Mitra Tani Menggunakan Metode <i>Prototype</i> [18].                        | Perbandingan dalam konteks ini, dapat dibandingkan penggunaan sistem informasi penjualan dengan proses manual yang dilakukan oleh CV Mitra Tani.   | Dalam konteks ini, sistem informasi penjualan dapat dibandingkan dengan proses manual CV Mitra Tani.   | Bahwa penggunaan sistem informasi penjualan mungkin memerlukan biaya pengembangan dan pemeliharaan   | Dalam hubungannya dengan CV Mitra Tani, penggunaan sistem informasi penjualan berbasis web direncanakan untuk mengatasi permasalahan yang   | Penggunaan sistem informasi penjualan menawarkan keuntungan besar untuk meningkatkan efektivitas dan efisiensi proses bisnis. Sebagai toko serba ada, CV Mitra Tani  |
| 8         | Application of Digital Forensics to Identify Human Voices Using the System Development Life Cycle (SDLC) Method [19]. | Penerapan Forensik Digital untuk Mengidentifikasi Suara Manusia Menggunakan Metode Siklus Hidup Pengembangan Sistem (SDLC)" menunjukkan penggunaan metode SDLC dalam mengembangkan kerangka forensik audio untuk mengidentifikasi suara manusia. Di sisi lain, penelitian lain, seperti "Penerapan Pembelajaran Transfer | Penerapan Forensik Digital untuk Mengidentifikasi Suara Manusia Menggunakan Metode Siklus Hidup Pengembangan Sistem (SDLC)" menekankan penggunaan pendekatan SDLC untuk pengembangan kerangka forensik audio, penelitian lain, seperti "Penerapan Pembelajaran | "Menerapkan Forensik Digital pada Deteksi Suara Manusia Menggunakan Metodologi Siklus Hidup Pengembangan Sistem (SDLC)" tidak mengkritik aspek spesifik dari pendekatan SDLC atau proses pengembangan kerangka forensik audio. | Menerapkan Forensik Digital pada Pengenalan Suara Manusia Menggunakan Metodologi Siklus Hidup Pengembangan Sistem (SDLC)"; mensintesis penggunaan pendekatan SDLC untuk mengembangkan kerangka forensik audio untuk pengenalan suara manusia. | Kesimpulannya, penelitian menunjukkan bahwa SDLC dapat berfungsi sebagai kerangka kerja untuk mengembangkan kerangka forensik audio dengan berkolaborasi pada beberapa kerangka forensik audio. Tahapan SDLC yaitu. Desain, Analisis, Desain, Implementasi dan Pemeliharaan bersifat indikatif dalam proses pengembangan |

| <i>No</i> | <i>Judul</i>  | <i>Compare</i>  | <i>Contrast</i>  | <i>Criticize</i>  | <i>Synthesis</i>  | <i>Summaries</i>  |
|-----------|---|---|--|---|---|---|
|           |   | End-to-End untuk Identifikasi Perangkat Pererekam Sumber untuk Keamanan Audio Berkelanjutan," berfokus pada penerapan model pembelajaran mendalam untuk identifikasi sumber audio digital | Transfer End-to-End untuk Identifikasi Perangkat Pererekam Sumber untuk Keamanan Audio Berkelanjutan," menyoroti penerapan model pembelajaran mendalam untuk identifikasi sumber audio digital |   |   | kerangka kerja. Framework yang dikembangkan menghasilkan 7 langkah utama, 33 sub-langkah dan 3 kondisi, total 40 langkah, yang dapat mendukung proses identifikasi audio forensik. Hasil analisis spektogram dan nada menunjukkan efektivitas kerangka kerja yang dikembangkan menggunakan forensik digital dalam pengenalan suara manusia. |
| 9         | Analysis and Design of Student Guidance Information System through Software Development Life Cycle (SDLC) and Waterfall Model [20]. | Metode pengembangan perangkat lunak Waterfall dan SDLC adalah dua metode yang paling banyak digunakan untuk merancang, mengimplementasikan, dan melaksanakan proyek perangkat             | Metode Waterfall mengikuti pendekatan linier dan sekuensial, sedangkan SDLC merupakan proses terstruktur yang terdiri dari beberapa fase. Waterfall lebih kaku dan kurang fleksibel            | Tidak ada kritik spesifik terhadap metode Waterfall atau SDLC tersebut. | pengembangan sistem informasi manajemen skripsi (SIBIMA) menggunakan metode SDLC dengan model Waterfall. SIBIMA bertujuan untuk mendukung pelayanan pendidikan yang menyesuaikan dengan konteks nilai-nilai | pengembangan sistem informasi kepemimpinan skripsi (SIBIMA) menggunakan metode SDLC dengan model Waterfall. SIBIMA bertujuan untuk mendukung layanan pendidikan yang beradaptasi dengan   |

| <i>No</i> | <i>Judul</i>  | <i>Compare</i>   | <i>Contrast</i>  | <i>Criticize</i>                     | <i>Synthesis</i>   | <i>Summaries</i>  |
|-----------|---|--|--|--------------------------------------|--|---|
|           |   | lunak.   | dibandingkan SDLC.   |                                      | Universitas Katolik Indonesia Atma Jaya yaitu Kristiani, Unggul.   | konteks nilai-nilai Universitas Katolik Atma Jaya, Indonesia.   |
| 10        | Student Absence Information System at Pancur Ido Elementary School Using Rapid Application Development Method [21]. | pengolahan data secara manual dengan pola komputersasi yang memudahkan proses memasukkan dan mencari data yang telah tersimpan dalam database. Selain itu juga membandingkan sistem yang digunakan saat ini di SD Pancur Ido dengan sistem berbasis web yang diperlukan untuk mengatasi permasalahan tersebut. | ketidakefisienan dan ketidakefektifan sistem yang digunakan saat ini di SD Pancur Ido dalam memantau kehadiran siswa dengan efektivitas sistem berbasis web. | Tidak ada kritik pada penelitian ini | pemanfaatan teknologi informasi dalam pengolahan data dan pembuatan database agar proses kerja menjadi lebih optimal, cepat, tepat, dan hemat biaya. | pemanfaatan teknologi informasi dalam pengolahan data, pembuatan database, dan pemanfaatan sistem informasi kehadiran siswa di bidang pendidikan. Hal ini juga menyoroti ketidakefisienan dan ketidakefektifan sistem yang saat ini digunakan |

Kesimpulan dari 10 jurnal diatas mencakup berbagai topik seperti pengembangan sistem informasi, manajemen sistem informasi SDM dan desain sistem informasi. Beberapa publikasi membahas penerapan metode perangkat lunak seperti System Development Life Cycle (SDLC) dan Model Waterfall pada pengembangan sistem informasi. Selain itu, terdapat juga penelitian yang berfokus pada penerapan forensik digital pada pengenalan suara manusia. Beberapa jurnal membandingkan metode pengembangan perangkat lunak dan sistem informasi yang berbeda, sementara jurnal lainnya menekankan pentingnya teknologi informasi dalam konteks organisasi yang

berbeda. Meskipun beberapa jurnal memberikan gambaran umum tentang topik yang dibahas, ada juga jurnal yang memberikan analisis lebih mendalam tentang metode dan konsep yang digunakan.

## 1.2 Landasan Teori

### 1.2.1 Sistem informasi

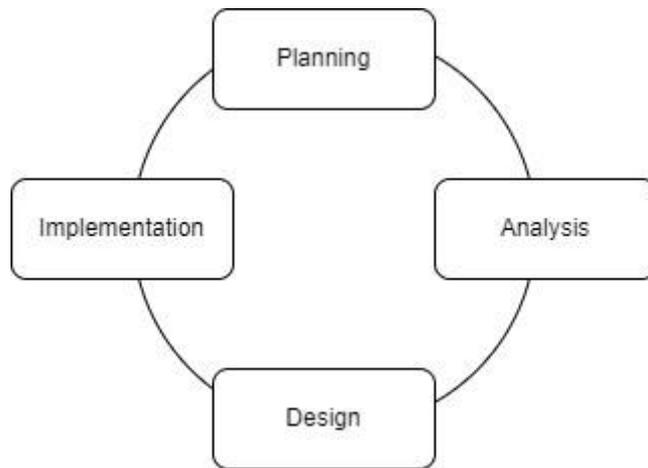
Sistem informasi dapat digambarkan sebagai kerangka kerja yang mencakup komponen teknis, proses, orang, dan informasi yang bersama-sama mengumpulkan, menyimpan, mengatur, memproses, dan mengirimkan informasi yang berguna dan signifikan dalam suatu organisasi. Tujuannya adalah untuk mendukung pengambilan keputusan, mengendalikan proses, dan menerapkan proses bisnis secara efektif [22].

### 1.2.2 System development life cycle

Metode *System Development Life Cycle* (SDLC) adalah metode yang sistematis digunakan dalam pengembangan sistem komputer. SDLC menyediakan kerangka kerja yang sistematis dan terorganisir untuk perencanaan, desain, pengembangan, implementasi, dan pemeliharaan sistem komputer. Pengembangan SDLC telah berkembang dari waktu ke waktu dengan kemajuan teknologi dan perubahan kebutuhan bisnis. Menurut Barry Boehm, seorang peneliti dan profesor di bidang rekayasa perangkat lunak memperkenalkan konsep "*model spiral*" dengan metode SDLC [23].

### 1.2.3 Langkah-langkah system development life cycle

Tahap-tahap yang dilewati untuk pengembangan sistem penggunaan SDLC (*System Development Life Cycle*) adalah sebagai berikut :



Gambar 2.1 Langkah-Langkah Metode SDLC

#### 1.2.3.1 Tahap Awal : Perencanaan (*Planning*)

Langkah pertama yang penting dalam analisis kebutuhan dalam pengembangan sistem informasi. Fase perencanaan ini, persyaratan bisnis ditentukan, tujuan proyek ditetapkan dan anggaran, jadwal, dan sumber daya yang diperlukan. (*User specification*) proses pengumpulan informasi tentang kebutuhan dan kebutuhan pengguna akhir sistem informasi, (*Feasibility Study*) termasuk evaluasi kelayakan proyek dari sudut pandang teknis dan teknologi serta perencanaan atau pengembangan proyek sistem informasi.

#### 1.2.3.2 Tahap kedua : Analisis (*Analysis*)

Tahap di mana pengembang bekerja sama dengan pemangku kepentingan dan pengguna akhir untuk memastikan bahwa semua masalah dan kebutuhan pengguna teridentifikasi dengan jelas. Tujuan utama dari tahap analisis adalah memahami secara menyeluruh sistem yang akan dikembangkan agar dapat merancang solusi yang tepat dan efektif, hal ini dilakukan dengan menguraikan dan menerapkan diagram use case sebagai alat untuk mewakili

interaksi antara pengguna dan sistem dan untuk mengidentifikasi komponen sistem, objek-objek dan penggunaan yang berbeda.

#### 1.2.3.3 Tahap ketiga : Desain (*Design*)

Tahap desain merupakan tahap penting setelah tahap analisis. Tahap desain yaitu merencanakan sistem informasi secara detail, yang akan dikembangkan berdasarkan kebutuhan dan kebutuhan yang telah ditentukan sebelumnya. Tahap desain menghasilkan seperti diagram arsitektur sistem, diagram hubungan entitas (*ERD*), rencana antarmuka pengguna, rencana basis data, dan spesifikasi teknis yang membentuk dasar untuk fase pengembangan berikutnya.

#### 1.2.3.4 Tahap keempat : Implementasi (*Development*)

Tahap dimana sistem informasi yang direncanakan pada langkah sebelumnya akan diimplementasikan pada tahap ini. Tahap ini melibatkan komponen sistem pengkodean dan digabungkan menjadi sistem yang lengkap dan berfungsi. Pada tahap implementasi atau pengembangan, pengembangan sistem, pemangku kepentingan dan pengguna akhir harus memastikan bahwa sistem dapat berhasil diimplementasikan untuk memastikan pengembangan sistem sesuai jadwal dan mencapai tujuan yang diinginkan

### 1.2.4 Kelebihan dan kekurangan metode SDLC

Pengembangan sistem memiliki kelebihan dan kekurangannya, begitu juga dengan SDLC. Menurut beberapa peneliti mengenali kelebihan SDLC, seperti pendekatan terstruktur, pemahaman kebutuhan yang baik, manajemen risiko yang efektif, dan dokumentasi

lengkap. Peneliti juga menemukan kekurangan, seperti fleksibilitas yang terbatas, tambahan waktu dan biaya, kesulitan dalam mengevaluasi persyaratan awal, dan ketidaksesuaian untuk proyek kecil, tetapi sangat berguna bagi pengembang sistem pemula [24].

#### 1.2.5 MySQL (Database)

Sistem basis data atau biasa disebut sebagai *database*, adalah sekumpulan yang terorganisir dari informasi terkait yang disimpan secara elektronik. Tujuan dari sistem basis data adalah untuk menyimpan, mengelola, dan membuat informasi dapat diakses secara efisien, dalam pengembangan sistem informasi ini menggunakan aplikasi sistem basis data yaitu *MySQL*. *MySQL* merupakan sistem manajemen basis data relasional yang populer dan banyak digunakan, dalam *MySQL*, data disimpan, dikelola, dan digunakan secara efisien. *MySQL* memiliki berbagai kegunaan seperti penyimpanan, manajemen data, pemrosesan data, skalabilitas untuk mengelola data dalam jumlah besar. Saat menggunakan *MySQL*, penting untuk memantau kinerja sistem dan melakukan pemeliharaan rutin untuk pengoptimalan query, serta menjaga keamanan dan membuat backup data secara berkala [25].

#### 1.2.6 XAMPP

*XAMPP* adalah perangkat lunak *web server* lokal yang sangat berguna dan populer untuk pengembangan, uji coba, dan pengelolaan situs web atau aplikasi *web*. Dengan menggunakan *XAMPP*, pengembang dapat memanipulasi dan mengelola database, menjalankan *script* PHP, dan menginstal WordPress atau Laravel dengan mudah dan efisien [26].

#### 1.2.7 Visual Studio Code

*Visual Studio Code* adalah editor teks sumber terbuka serbaguna yang dikembangkan oleh *Microsoft*. *Visual Studio code* dirancang untuk mendukung berbagai bahasa pemrograman dan platform termasuk *HTML*, *CSS*, *JavaScript*, *Python*, *C++*, dan banyak lagi. Editor ini menawarkan fitur canggih untuk meningkatkan produktivitas pengembang, seperti pembuatan sintaks, penyelesaian kode otomatis, *debugging*, manajemen proyek, dan integrasi dengan sistem kontrol versi. *Visual Studio Code* memiliki komunitas yang aktif dan luas, sehingga pengguna memiliki banyak sumber daya, tutorial, dan dukungan yang dapat mereka gunakan [27].

#### 1.2.8 Flowchart

*Flowchart* adalah diagram visual yang digunakan untuk menggambarkan logika atau alur proses dari suatu sistem atau proses. *Flowchart* membantu memvisualisasikan langkah-langkah atau keputusan yang dibulat dalam urutan tertentu yang membuat proses lebih mudah dipahami dan dikomunikasikan. Menurut Richard Bellman seorang tokoh *flowchart* adalah metode standar untuk merepresentasikan algoritma dalam bentuk diagram. Algoritma itu sendiri adalah sekumpulan instruksi atau langkah logis untuk memecahkan masalah atau mencapai tujuan tertentu. Diagram alir membantu memvisualisasikan algoritme dengan cara yang lebih mudah dipahami dan dapat ditindak lanjuti. *Flowchart* menggunakan simbol khusus untuk mewakili langkah atau keputusan dalam suatu proses dan menggunakan garis untuk menghubungkan simbol dan menunjukkan jalur yang harus diikuti [28].

#### 1.2.9 Unified Modeling Language (UML)

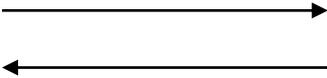
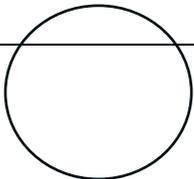
*Unified Modeling Language (UML)* adalah sebuah bahasa standar yang digunakan dalam rekayasa perangkat lunak untuk memodelkan, mendokumentasikan, dan merancang sistem perangkat

lunak. UML menyediakan serangkaian notasi grafis yang memungkinkan pengembang perangkat lunak untuk menggambarkan struktur, perilaku, dan interaksi sistem secara visual. Pengembang dapat membuat berbagai jenis diagram yang mencakup berbagai aspek dari sistem, seperti diagram kelas, diagram kasus penggunaan, diagram aktivitas, dan banyak lagi. Tujuan utama dari UML adalah untuk menyediakan cara yang konsisten dan terstandarisasi untuk menyampaikan informasi tentang sistem perangkat lunak kepada para pemangku kepentingan [29].

#### 1.2.9.1 Data alir diagram (*Data Flow Diagram*)

*Diagram Data Flow* (DFD) adalah suatu alat visual yang digunakan untuk menggambarkan aliran data dan proses dalam suatu sistem, memudahkan analisis dalam desain sistem dan memahami bagaimana data mengalir dan diproses dalam sistem. DFD diagram terdiri dari beberapa bagian, yaitu proses, data store, external entity, dan data flow yang membantu dalam memahami bagaimana data mengalir dan diproses dalam sistem, meningkatkan kemampuan analisis dan desain sistem, serta mengidentifikasi masalah dan memperbaiki sistem [30].

Tabel 2. 2 Simbol Data Flow Diagramm

| Simbol  | Fungsi   |
|---|--|
|  | External entity / Entitas luar. Simbol ini menunjukkan orang, organisasi , atau sistem yang berada di luar sistem tetapi berinteraksi dengan sistem      |
|  | Data Flow diberi simbol panah. Simbol ini menunjukkan satu data tunggal atau kumpulan logis suatu data, selalu diawali atau di akhiri pada suatu proses. |
|  | Proses adalah aktivitas atau fungsi  |

|   |  |
|---|--|
|   | yang dilakukan untuk alasan bisnis yang spesifik , bisa berupa manual maupun terkomputerisasi                    |
|  | Data store adalah Kumpulan data yang disimpan dengan cara tertentu. Data yang mengalir disimpan dalam data store |

### 1.2.9.2 Entity relationship diagram (ERD)

Diagram Relasi Entitas adalah alat visual yang digunakan dalam desain basis data untuk menggambarkan entitas, atribut, dan hubungan antara entitas tersebut. Diagram ini, entitas direpresentasikan oleh kotak, sedangkan atributnya dijelaskan di dalam kotak. Hubungan antara entitas ditunjukkan dengan garis yang menghubungkannya, dan jenis hubungan ini dapat dinyatakan menggunakan tanda panah atau notasi khusus lainnya. Diagram Relasi Entitas membantu dalam memodelkan struktur data dengan jelas, memahami hubungan antara berbagai entitas dalam basis data, dan merencanakan desain yang efisien dan efektif untuk sistem informasi [31].

### 1.2.9.3 Class diagram

Diagram Kelas adalah alat visual yang digunakan dalam pemrograman berorientasi objek untuk menggambarkan struktur kelas dalam sistem perangkat lunak. Dalam diagram ini, kelas direpresentasikan oleh kotak, dengan nama kelas ditulis di dalamnya. Atribut kelas dan metode yang dimilikinya juga dijelaskan di dalam kotak tersebut. Hubungan antara kelas-kelas ditunjukkan dengan garis yang menghubungkannya, yang menggambarkan hubungan seperti pewarisan, asosiasi, atau ketergantungan. Diagram Kelas membantu dalam memodelkan

struktur objek dan hubungan antar objek dalam sistem, sehingga mempermudah pemahaman dan perancangan sistem perangkat lunak [32].

#### 1.2.9.4 Use case diagram

Use case diagram adalah alat visual yang digunakan dalam rekayasa perangkat lunak untuk memodelkan interaksi antara sistem dan pemakainya. Dalam diagram ini, aktor-aktor yang mewakili pemakai sistem direpresentasikan oleh simbol luar, sedangkan kasus penggunaan atau skenario penggunaan sistem direpresentasikan oleh elips atau lingkaran. Hubungan antara aktor dan kasus penggunaan ditunjukkan dengan garis, menggambarkan bagaimana aktor berinteraksi dengan sistem dalam berbagai situasi. Diagram Kasus Penggunaan membantu dalam memahami kebutuhan dan persyaratan pengguna, serta mengidentifikasi fungsi-fungsi utama sistem yang harus diimplementasikan untuk memenuhi kebutuhan tersebut [33].

#### 1.2.9.5 Activity Diagram

Diagram Aktivitas adalah alat visual yang digunakan dalam rekayasa perangkat lunak untuk menggambarkan alur kerja atau aktivitas dalam sistem. Dalam diagram ini, aktivitas atau tindakan direpresentasikan oleh simbol elips, sedangkan alur aliran antara aktivitas tersebut direpresentasikan oleh panah. Diagram Aktivitas membantu dalam memvisualisasikan langkah-langkah atau proses yang harus dilakukan dalam suatu sistem atau algoritma [34].

#### 1.2.10 Bahasa pemrograman

Bahasa pemrograman adalah sistem komunikasi yang digunakan untuk memberikan instruksi ke komputer atau sistem komputer lainnya. Bahasa ini memungkinkan pemrogram untuk mengatur dan

mengatur fungsi yang ingin dilakukan komputer. Bahasa pemrograman menggunakan kata dan aturan khusus yang memungkinkan pemrogram menulis program dan membuat algoritme atau logika pemrosesan data. Bahasa pemrograman memungkinkan orang untuk berinteraksi dengan komputer dan mengembangkan berbagai aplikasi [35]. Pemrograman *web* juga memiliki Bahasa-bahasa yang digunakan seperti :

#### 1.2.10.1 HTML (Hypertext Markup Language)

*Hypertext Markup Language (HTML)* adalah bahasa markup standar yang digunakan untuk membuat dan mendesain halaman *web*. *HTML* menyediakan struktur dan elemen yang digunakan untuk mengatur konten, tata letak, dan tampilan halaman *web*. *HTML* memungkinkan pengembang *web* untuk menentukan bagaimana elemen seperti teks, gambar, tautan, tabel, dan formulir ditampilkan di *browser*. Dalam *HTML*, setiap elemen diidentifikasi oleh tag yang diapit tanda kurung sudut (<>). *Tag* memberi petunjuk pada browser tentang item apa yang akan ditampilkan dan cara menampilkannya [36].

#### 1.2.10.2 CSS (Cascading Style Sheet)

*CSS (Cascading Style Sheet)* adalah bahasa yang digunakan untuk menentukan tata letak atau gaya visual halaman *web* yang ditulis dalam *HTML*. *CSS* memisahkan tata letak dari struktur dokumen, memberi pengembang *web* kontrol yang mudah atas tampilan elemen halaman *web*. *CSS* memungkinkan pengembang *web* untuk menentukan berbagai properti gaya untuk elemen *HTML*, seperti warna, ukuran, tata letak, dan *font*. *CSS* memungkinkan halaman *web* memiliki tampilan yang

konsisten karena gaya yang ditentukan dalam CSS dapat diterapkan secara konsisten ke banyak halaman [37].

#### 1.2.10.3 PHP (Personal Home Page)

*PHP (Personal Home Page)* adalah bahasa pemrograman *open source* yang dirancang khusus untuk pengembangan aplikasi *web*. Dikembangkan oleh Rasmus Lerdorf pada tahun 1994, *PHP* telah menjadi salah satu bahasa pemrograman paling populer di dunia. *PHP* dirancang untuk memfasilitasi pengembangan aplikasi *web* yang dinamis dan interaktif. Bahasa ini dapat disematkan langsung ke dalam kode *HTML* dan dapat berinteraksi dengan basis data, menghasilkan konten dinamis, dan melakukan berbagai tugas pemrosesan data di sisi *server*. Salah satu keunggulan utama *PHP* adalah fleksibilitasnya. Bahasa ini dapat berjalan di berbagai platform seperti *Windows*, *MacOS*, *Linux* dan *server web* populer seperti *Apache* dan *Nginx*. Selain itu, *PHP* juga mendukung berbagai *database* seperti *MySQL*, *PostgreSQL*, dan *Oracle*, yang memungkinkan *developer* menyimpan dan mengelola data dengan mudah [38].

#### 1.2.10.4 Java script

Bahasa pemrograman *JavaScript* adalah bahasa pemrograman tingkat tinggi yang sering digunakan dalam pengembangan aplikasi *web*. *JavaScript* dirancang untuk membuat halaman *web* menjadi interaktif dan dinamis dengan menambahkan fungsi-fungsi seperti animasi,

validasi formulir, manipulasi elemen *HTML*, dan banyak lagi. Bahasa skrip yang dapat dieksekusi di sisi klien (*browser*), yang berarti kode *JavaScript* dieksekusi oleh *browser* pengguna, bukan oleh *server web*. *JavaScript* juga digunakan secara luas dalam pengembangan aplikasi berbasis *server* menggunakan *runtime* seperti *Node.js*. Keunggulan utama *JavaScript* adalah kemampuannya untuk berintegrasi dengan *HTML* dan *CSS* dengan mudah, serta dukungan luas dari berbagai *browser* dan platform [39].

#### 1.2.11 User acceptance test

*User acceptance test* adalah proses pengujian yang dilakukan oleh pengguna atau pemangku kepentingan pada tahap akhir pengembangan perangkat lunak untuk memastikan bahwa sistem memenuhi persyaratan dan ekspektasi mereka. Tujuannya adalah untuk memverifikasi apakah sistem telah siap untuk digunakan secara operasional dalam lingkungan yang nyata. Uji penerimaan pengguna melibatkan serangkaian skenario pengujian yang dirancang untuk mencakup berbagai kasus penggunaan yang mungkin terjadi dalam situasi nyata. Hasil dari uji penerimaan pengguna dapat digunakan sebagai tolak ukur untuk menentukan apakah sistem siap untuk diimplementasikan [40]. Rumus perhitungan UAT :

|  |       |
|--|-------|
| $P = \frac{S}{\text{Skor}} \times 100$ | (2.1) |
|--|-------|

P = merupakan persentase UAT

S = Skor yang diperoleh dalam UAT

Skor adalah skor maksimum yang dapat dicapai dalam UAT

Rumus ini digunakan untuk menghitung persentase keberhasilan UAT dengan membagi skor yang diperoleh oleh skor

maksimum yang dapat dicapai, lalu hasilnya dikalikan dengan 100%. Persentase ini memberikan gambaran seberapa baik system yang telah melewati pengujian oleh pengguna akhir. Semakin tinggi persentasenya, semakin baik hasil UAT-nya

#### 1.2.12 Blackbox testing

Pengujian *blackbox* adalah metode pengujian perangkat lunak di mana sistem diuji tanpa memperhatikan struktur internal atau logika kode programnya. Dalam pengujian kotak hitam, pengujian dilakukan berdasarkan spesifikasi fungsional sistem, tanpa mengetahui bagaimana sistem tersebut diimplementasikan secara internal. Tujuannya adalah untuk mengevaluasi fungsi-fungsi sistem dari sudut pandang pengguna akhir, dengan menguji interaksi antara input dan output sistem. Pengujian ini fokus pada perilaku eksternal sistem, dan tidak memperhatikan detail bagaimana sistem mencapai output tertentu dari input yang diberikan [41].