

## BAB II TINJAUAN PUSTAKA

### 2.1 Teori Dasar

#### 2.1.1 Sistem

Sistem berasal dari bahasa Latin (*Systema*) dan bahasa Yunani (*Sustema*) yang berarti suatu kesatuan yang terdiri dari komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi atau energi untuk mencapai suatu tujuan. Istilah ini sering dipergunakan untuk menggambarkan suatu entitas yang berinteraksi. Menurut Hutahaean (2015) Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan kegiatan atau untuk melakukan sasaran yang tertentu. Ciri-ciri sistem menurut Mulyani (2017) yaitu sebagai berikut:

- Memiliki komponen-komponen: Sistem terdiri dari beberapa komponen yang saling berhubungan dan saling tergantung satu sama lain.
- Komponen-komponen tersebut terintegrasi: Komponen-komponen sistem saling terhubung dan bekerja sama untuk mencapai tujuan bersama.
- Memiliki batasan sistem: Batasan sistem menentukan apa yang termasuk dalam sistem dan apa yang tidak.
- Memiliki tujuan yang jelas: Sistem memiliki tujuan yang ingin dicapai.
- Memiliki lingkungan: Sistem memiliki lingkungan yang mempengaruhinya dan dipengaruhinya.
- Memiliki *input*, proses, dan *output*: Sistem menerima *input*, memprosesnya, dan menghasilkan *output*.

Pengertian sistem secara umum mengartikan bahwa sistem adalah suatu kesatuan komponen atau elemen yang saling berhubungan untuk menjalankan sebuah proses untuk mencapai suatu tujuan tertentu.

### **2.1.2 Informasi**

Informasi adalah sekumpulan data atau fakta yang diberikan dan dijelaskan kepada khalayak banyak dan memberikan arti dan kegunaan bagi orang yang menerimanya. Informasi memiliki banyak bentuk seperti tulisan, gambar, suara atau gabungan dari ketiganya yang bertujuan untuk memberikan pengetahuan mengenai suatu subjek atau topik tertentu. Informasi juga dapat dianggap sebagai langkah-langkah yang berisi pengumpulan, pengolahan, penyimpanan dan distribusi data atau fakta yang berguna. Informasi merupakan hal penting yang dapat mempengaruhi suatu keputusan dan bahkan untuk penyelesaian masalah. Menurut Kenneth Laudon dan Jane Laudon (2018), Informasi adalah data yang telah diubah menjadi bentuk yang bermakna dan berguna untuk tujuan pengambilan keputusan atau tindakan tertentu. Mereka menekankan pentingnya informasi dalam konteks sistem informasi dan teknologi informasi.

### **2.1.3 Sistem Informasi**

Sistem informasi adalah suatu Kumpulan dari manusia, *software*, *hardware*, prosedur dan data yang bekerja bersama untuk mengolah, menyimpan, mengumpulkan dan menyebarkan informasi yang diperlukan oleh kebutuhan operasional dan keputusan suatu organisasi atau bahkan untuk kebutuhan informasi yang harus diketahui masyarakat luas. Menurut Ken Laudon dan Jane Laudon (2021), sistem informasi adalah kumpulan elemen yang terintegrasi, yang terdiri dari orang, perangkat keras, perangkat lunak, jaringan, dan data, yang bekerja bersama-sama untuk mengumpulkan, memproses, menyimpan, dan menyebarkan informasi guna mendukung pengambilan keputusan, koordinasi, pengendalian, analisis, dan visualisasi di organisasi dan masyarakat. Definisi ini mencerminkan kompleksitas dan pentingnya sistem informasi dalam konteks saat ini, di mana teknologi informasi dan pengelolaan data memainkan peran yang semakin vital dalam operasi organisasi dan pengambilan keputusan. Sedangkan menurut R. Kelly Rainer Jr. dan Hugh J. Watson (2020), sistem informasi adalah struktur terorganisir yang mencakup teknologi, orang-orang, data, dan prosedur yang bekerja bersama untuk menghasilkan

informasi yang berguna bagi organisasi. Penjelasan mengenai sistem informasi yang diuraikan memberikan kita suatu pengetahuan bagaimana pentingnya peran sistem informasi dalam membantu kebutuhan masyarakat atau suatu organisasi dikarenakan sistem informasi dapat mengintegrasikan antara teknologi, prosedur bisnis, data dan orang-orang dalam mengelola suatu informasi secara efektif dan efisien. Suatu sistem informasi memiliki ciri-ciri, menurut Carol Yacht (2022), ciri-ciri sistem informasi seperti kecepatan (kemampuan sistem untuk memproses informasi dengan cepat), ketepatan (akurasi dan kebenaran informasi yang dihasilkan), dan penggunaan (kemudahan penggunaan sistem oleh pengguna dalam mengakses dan menggunakan informasi). Sedangkan menurut James A. O'Brien (2022), ciri sistem informasi ialah termasuk integrasi (kemampuan untuk mengintegrasikan berbagai komponen sistem seperti perangkat keras, perangkat lunak, data, dan prosedur), keterpaduan (konsistensi antara berbagai elemen dalam sistem), dan ketersediaan (kemampuan untuk memberikan akses yang tepat dan waktu yang tepat terhadap informasi).

#### **2.1.4 Data**

Menurut Peter Bruce dan Andrew Bruce (2020), data didefinisikan sebagai fakta, angka, atau informasi yang dihasilkan dari pengamatan, eksperimen, atau proses lainnya, yang dapat dianalisis untuk mendapatkan wawasan atau informasi yang bermanfaat. Dilampirkan juga ciri-ciri data yaitu menurut Kenneth C. Laudon dan Jane P. Laudon (2021), Ciri-ciri data menurut Kenneth C. Laudon dan Jane P. Laudon termasuk keberagaman (data dapat berasal dari berbagai sumber dan bentuk), relevansi (data memiliki hubungan dengan subjek atau topik tertentu), akurasi (data harus tepat dan benar), dan waktu yang tepat (data harus tersedia pada saat dibutuhkan). Sedangkan menurut David M. Kroenke dan Randall J. Boyle (2022), ciri-ciri data ialah konsistensi (data harus konsisten dan tidak bertentangan), validitas (data harus sesuai dengan standar atau kriteria tertentu), kebenaran (data harus benar dan akurat), dan relevansi (data harus relevan dengan tujuan atau kebutuhan pengguna). Dengan menggabungkan ciri-ciri data menurut Laudon dan Laudon (2021) serta Kroenke dan Boyle

(2022), dapat disimpulkan bahwa data yang baik adalah data yang relevan, akurat, lengkap, tepat waktu, tersedia, dapat diandalkan, dan bersifat valid. Ciri-ciri ini merupakan pedoman penting dalam pengelolaan dan penggunaan data untuk mendukung pengambilan keputusan, analisis, dan pemahaman yang lebih baik dalam berbagai konteks.

Secara umum, data adalah bagian informasi yang berbeda yang dikumpulkan dan diterjemahkan untuk beberapa tujuan. Data dapat tersedia dalam bentuk yang berbeda, seperti bit dan byte yang disimpan dalam memori elektronik, angka atau teks pada selembar kertas, atau fakta yang disimpan dalam pikiran seseorang.

### **2.1.5 Unified Modeling Language (UML)**

Menurut Evans, Eric (2023) menyatakan bahwa UML merupakan alat yang kuat untuk mendefinisikan dan memvisualisasikan domain model. Menurutnya, UML membantu dalam komunikasi antara pengembang dan pemangku kepentingan bisnis dengan cara yang lebih terstruktur dan visual. Pada dasarnya *Unified Modeling Language* (UML) adalah sebuah bahasa pemodelan yang digunakan untuk menggambarkan, merencanakan, dan mendokumentasikan desain perangkat lunak berbasis objek. UML memberikan notasi yang kaya dan sistematis untuk menggambarkan struktur dan perilaku sistem perangkat lunak, serta hubungan antara komponen-komponennya.

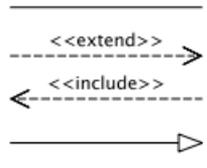
Teori dasar mengenai UML meliputi konsep-konsep utama yang digunakan dalam pemodelan perangkat lunak berbasis objek. Tujuan utama UML adalah menyediakan metode yang jelas dan terstruktur untuk menggambarkan struktur dan perilaku sistem perangkat lunak. Dengan UML, tim pengembang dapat berkomunikasi secara efektif dan memahami desain sistem dengan lebih baik. UML menggunakan notasi grafis yang terdiri dari berbagai jenis diagram, seperti diagram kelas, diagram use case, diagram aktivitas, diagram urutan, dan lain-lain. Setiap jenis diagram memiliki tujuan dan fokus tertentu dalam menggambarkan aspek tertentu dari sistem. Elemen-elemen dasar dalam UML meliputi kelas, objek, atribut, metode, hubungan antar kelas (seperti asosiasi, komposisi, dan pewarisan),

*use case*, aktor, aktivitas, pesan, dan lain-lain. Setiap elemen memiliki peran dan hubungan yang ditentukan dalam model UML. UML digunakan untuk memodelkan struktur sistem perangkat lunak, termasuk entitas-entitas utama seperti kelas, objek, antarmuka, paket, dan hubungan antara mereka.

Diagram kelas / *Class Diagram* adalah salah satu jenis diagram UML yang sering digunakan untuk menggambarkan struktur kelas dan hubungan antar kelas. Selain struktur, UML juga digunakan untuk memodelkan perilaku sistem perangkat lunak. Diagram aktivitas, diagram urutan, dan *diagram state* adalah beberapa jenis diagram UML yang digunakan untuk menggambarkan perilaku sistem dan interaksi antara objek-objeknya. UML dapat digunakan dalam berbagai tahap pengembangan perangkat lunak, mulai dari analisis kebutuhan, desain, implementasi, hingga pengujian dan dokumentasi. Setiap tahap pengembangan dapat menggunakan jenis diagram UML yang sesuai untuk memahami dan mengkomunikasikan aspek yang relevan dari sistem. Lalu, UML dapat digunakan bersama dengan berbagai metode pengembangan perangkat lunak, seperti metode *waterfall*, metode *agile*, dan lain-lain. Hal ini memungkinkan UML untuk diintegrasikan dengan proses pengembangan yang ada dan digunakan secara fleksibel sesuai dengan kebutuhan proyek. Dikarenakan di era sekarang penuh dengan kemudahan, Terdapat berbagai alat bantu (*tools*) yang mendukung penggunaan UML, seperti Rational Rose, Visual Paradigm, dan Enterprise Architect. Alat-alat ini menyediakan lingkungan kerja yang mempermudah pembuatan, analisis, dan dokumentasi model UML. Salah satu aspek kunci dari penggunaan UML adalah kemampuannya untuk meningkatkan komunikasi dan pemahaman antara anggota tim pengembangan perangkat lunak, pemangku kepentingan, dan pemakai sistem. Dengan menggunakan notasi yang terstandarisasi dan terstruktur, UML membantu mengurangi kesalahpahaman dan memfasilitasi diskusi yang efektif dalam tim pengembangan. Macam-macam UML dapat dilihat sebagai berikut :

### a. Use Case Diagram

Menurut Grady Booch, Ivar Jacobson, dan James Rumbaugh (2021), use case diagram adalah representasi visual dari fungsionalitas sistem dan interaksinya dengan aktor eksternal. Mereka menekankan pentingnya diagram ini dalam komunikasi antara tim pengembang dan pemangku kepentingan, memastikan bahwa semua pihak memiliki pemahaman yang sama mengenai kebutuhan sistem. Sedangkan menurut Craig Larman (2023), use case diagram tidak hanya penting untuk mengidentifikasi fungsionalitas sistem tetapi juga untuk merencanakan iterasi pengembangan. Menurut Larman, use case diagram membantu dalam memprioritaskan fitur-fitur yang akan dikembangkan berdasarkan nilai bisnis dan kebutuhan pengguna.

Symbol	Reference Name
	Actor
	Use case
	Relationship

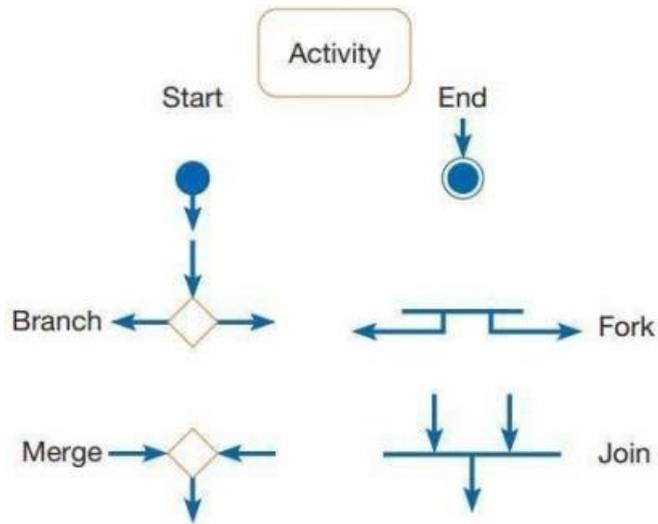
Gambar 2.1 Notasi Use Case Diagram

Sumber (Xuan Truong Vu, 2015)

### b. Activity Diagram

Menurut Booch, Grady (2021), *Activity diagram* adalah representasi visual dari alur kerja dari langkah-langkah aktivitas dan aksi secara bertahap dengan dukungan untuk pilihan, iterasi, dan concurrent. Ini adalah alat penting dalam memahami dan mendokumentasikan aspek dinamis dari sistem. Sedangkan menurut Arlow, Jim, dan Ila Neustadt (2020), *Activity diagram* adalah diagram alur yang mewakili alur kontrol dari satu aktivitas ke aktivitas lainnya dalam suatu sistem. Diagram ini memberikan

pandangan tingkat tinggi tentang fungsionalitas sistem dan cara kerjanya dalam situasi yang berbeda.



Gambar 2. 2 Notasi Activity Diagram

Sumber (Dennis, Wixom dan Tegarden, 2015)

**c. Sequence Diagram**

Menurut Dennis, Alan, Barbara Haley Wixom, and David Tegarden (2015), *sequence diagram* adalah alat yang digunakan untuk menggambarkan bagaimana objek dalam sistem berinteraksi melalui pesan yang dikirimkan dalam urutan waktu. Diagram ini memberikan wawasan yang berharga mengenai jalur komunikasi yang tepat antara komponen-komponen dalam sistem.

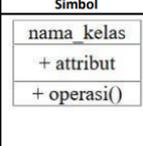
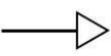
Nama	Simbol	Fungsi
Object		Menggambarkan sebuah <i>class</i> atau <i>object</i> .
Activation boxes		Menggambarkan panjang waktu yang dibutuhkan sebuah <i>object</i> dalam mengerjakan tugasnya
Actors		Menggambarkan pengguna yang berinteraksi dengan sistem
Lifeline		Menggambarkan "garis hidup" sebuah <i>object</i>
Message		Menggambarkan pesan atau interaksi antar <i>object</i>
Message to Self		Menggambarkan pesan balikan atau reaksi dari <i>object</i> sebelumnya

Gambar 2. 3 Notasi Sequence Diagram

Sumber (Ani Yoraeni, 2015)

#### d. Class Diagram

Menurut Pressman, Roger S. (2020), *class diagram* adalah representasi grafis dari struktur dan hubungan objek dalam suatu sistem, termasuk atribut dan metode dari masing-masing kelas. Class diagram sangat penting dalam tahap perancangan untuk memastikan integritas struktural dari sistem yang dikembangkan.

Simbol	Nama	Keterangan
	Kelas	Kelas pada struktur sistem.
	Interface	Sama dengan konsep interface dalam pemrograman berorientasi objek.
	Association	Relasi antarclass dengan arti umum, asosiasi biasanya juga disertai dengan Multiplicity.
	Directed Association	Relasi antarkelas dengan makna kelas yang atau digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan multiplicity.
	Generalisasi	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus).
	Dependency	Relasi antarkelas dengan makna ketergantungan antarkelas.
	Aggregation	Relasi antarkelas dengan makna semua-bagian (whole-part).

Gambar 2. 4 Notasi Class Diagram

Sumber (Ani Yoraeni, 2015)

#### 2.1.6 Pemodelan Perangkat Lunak

Dalam proses pembuatan suatu perangkat lunak, dibutuhkan beberapa perencanaan yang matang untuk hasil yang lebih baik dan sesuai dengan kebutuhan pengguna. Maka dari itu dibutuhkan suatu pemodelan yang dinamakan pemodelan perangkat lunak. Pemodelan perangkat lunak adalah suatu proses yang menggambarkan garis besar aplikasi, desain, dan dokumentasi dari aspek-aspek perangkat lunak yang akan direncanakan serta dibangun sesuai dengan kebutuhan pengguna. Menurut Grady Booch (2020) dalam buku terbarunya, Booch menjelaskan bahwa pemodelan perangkat lunak ialah suatu proses untuk merepresentasikan garis besar ataupun

abstrak dari suatu sistem perangkat lunak yang akan dibuat dan meliputi pemahaman struktur, perilaku dan aspek dari sistem yang akan dibuat. Representasi dari sebuah perangkat lunak yang akan direncanakan memiliki bermacam-macam model yang ditawarkan untuk melakukan pemodelan seperti *Waterfall*, *V-Model*, *Incremental Model*, *Spiral Model*, *Agile Model*, *Scrum*, *Prototyping Model*, *RAD* dan lainnya. Tiap model memiliki karakteristik yang berbeda-beda tergantung dengan Keputusan pengembang dan sifat serta lingkungan pengguna. Hal tersebut sangat mempengaruhi pemilihan model perangkat lunak dikarenakan banyak hal yang harus dipertimbangkan dalam perencanaan.

### **2.1.7 Bahasa Program**

Menurut Pressman, Roger S. (2020), Alat utama yang digunakan untuk menulis perangkat lunak, yang terdiri dari sintaks dan semantik yang memungkinkan pengembang untuk menulis kode yang dapat diterjemahkan ke dalam tindakan oleh komputer. Bahasa pemrograman juga mencakup konsep-konsep yang memungkinkan modularitas, abstraksi, dan pengelolaan kompleksitas dalam perangkat lunak. Secara umum, Bahasa pemrograman, dalam konteks teknologi informasi, merujuk pada sistem aturan dan sintaks yang digunakan oleh *programmer* untuk menulis kode yang dapat dieksekusi oleh komputer. Sebagai alat komunikasi antara manusia dan mesin, bahasa program memungkinkan *programmer* untuk menginstruksikan komputer dalam melakukan tugas-tugas tertentu, seperti pemrosesan data, pengelolaan informasi, pengembangan aplikasi, dan banyak lagi. Bahasa program dapat berupa bahasa tingkat tinggi seperti Python, Java, C++, atau bahasa tingkat rendah seperti bahasa rakitan, tergantung pada kompleksitas dan tujuan pengembangan perangkat lunak. Setiap bahasa pemrograman memiliki aturan sintaks dan struktur yang berbeda, yang ditentukan oleh desain bahasa tersebut. Aturan ini mengatur cara penulisan kode program, termasuk tata letak, penggunaan kata kunci, operasi aritmatika, logika, dan lainnya. Bahasa program juga memiliki berbagai jenis pernyataan dan konstruksi, seperti perulangan, percabangan,

fungsi, kelas, dan objek, yang memungkinkan *programmer* untuk membangun aplikasi dengan logika yang kompleks.

Pengembangan bahasa program merupakan bidang yang terus berkembang seiring dengan kemajuan teknologi. Bahasa pemrograman modern sering kali didukung oleh berbagai platform dan *framework* untuk memfasilitasi pengembangan aplikasi yang lebih efisien. Selain itu, bahasa program juga dapat digunakan untuk berbagai tujuan, mulai dari pengembangan perangkat lunak desktop, aplikasi web, hingga pengembangan kecerdasan buatan dan analisis data. Kemampuan bahasa program dalam mengubah instruksi manusia menjadi tindakan yang dapat dipahami oleh komputer membuatnya menjadi komponen penting dalam dunia teknologi informasi. Seiring dengan kompleksitas tugas yang dihadapi oleh perangkat lunak modern, bahasa program terus berkembang untuk memenuhi tuntutan pengembangan perangkat lunak yang semakin kompleks dan canggih.

#### **2.1.8 Basis Data**

Basis data adalah suatu kumpulan data yang terorganisir dengan jumlah yang banyak untuk suatu kebutuhan tertentu yang memungkinkan pengguna untuk menyimpan, mengambil dan mengelola data secara cepat dan efisien. Menurut Silberschatz, Korth, & Sudarshan (2019), Basis data merupakan kumpulan data yang luas dan kompleks yang dapat diakses, dimodifikasi, serta diperbarui oleh berbagai aplikasi. Silberschatz, Korth, & Sudarshan (2019) menekankan pentingnya menjaga integritas, keamanan, dan kinerja dalam pengelolaan data tersebut. Basis data juga tidak luput dari pengelolaan manusia yang mengoperasikan serta mengelola penyimpanan data untuk suatu kepentingan. Peran manusia dalam pengelolaan basis data menjadi sangat penting dikarenakan data merupakan kumpulan fakta yang akan dijadikan informasi untuk suatu kepentingan tertentu sehingga menjadi titik kritis di sebuah perusahaan ataupun segala pekerjaan yang membutuhkan basis data. Menurut Elmasri & Navathe (2020), Basis data merupakan kumpulan data yang terstruktur dan disimpan secara elektronik dalam sistem komputer. Basis data ini dirancang untuk memungkinkan

akses, manipulasi, dan pengelolaan data secara efisien, biasanya melalui penggunaan sistem manajemen basis data (DBMS).

### **2.1.9 Framework**

Menurut Munawar (2018), *framework* adalah kerangka kerja yang digunakan sebagai alat bantu untuk mempermudah pengembangan aplikasi dengan menggunakan kode-kode program yang telah ada atau mengikuti pola tertentu. Munawar (2018) juga menjelaskan bahwa *framework* dapat dikategorikan menjadi *Framework Front-end* dan juga *back-end*.

### **2.1.10 Software Development Life Cycle (SDLC)**

*Software Development Life Cycle* (SDLC) adalah proses yang berurutan pada suatu awal perencanaan pembuatan sistem ataupun aplikasi dan melibatkan beberapa tahapan seperti analisis kebutuhan, perancangan sistem, implementasi, pengujian dan pemeliharaan. Penekanan pada setiap tahapan bertujuan untuk menghasilkan sistem yang matang sehingga setiap tahapan memiliki *output* yang maksimal dengan memperoleh hasil akhir yang juga tepat sasaran dan sesuai kebutuhan. SDLC memiliki orientasi dalam menekankan kualitas pada tiap tahapannya dengan menggabungkan aspek teknis dan manajerial dalam suatu pengembangan sistem informasi (Kendall & Kendall, 2020). Teori ini ditekankan menurut ahli lain yang menggambarkan SDLC sebagai suatu model dalam perancangan perangkat lunak dengan proses dan hasil yang komprehensif. SDLC dalam prosesnya juga melibatkan pengembangan dalam aspek manajemen resiko dan kesesuaian sumber daya alam. Siklus hidup dalam suatu SDLC adalah suatu siklus yang harus diintegrasikan kepada bisnis dan teknologi sehingga menghasilkan suatu rancangan sistem yang baik dan sesuai kebutuhan pada tiap tahapannya (Pressman & Maxim, 2020). Beberapa tahapan dalam SDLC yaitu :

#### **1. *Planning* (Perencanaan)**

Fase perencanaan adalah langkah pertama yang penting dalam menetapkan tujuan proyek, menentukan ruang lingkup, dan mengidentifikasi sumber daya yang dibutuhkan. Resiko suatu

proyek tanpa perencanaan yang tepat selalu berpotensi menjadi mahal ataupun gagal (Kendall & Kendall, 2020)

2. *System Analysis* (Analisis Sistem)

Analisis sistem adalah fase di mana kebutuhan pengguna ditangkap dan dianalisis secara mendalam. Perlu ditekankan bahwa analisis persyaratan harus mencakup persyaratan fungsional dan non-fungsional serta mempertimbangkan faktor-faktor seperti kinerja, keamanan, dan skalabilitas (Sommerville, 2019)

3. *System Design* (Desain Sistem)

Perancangan sistem adalah pembuatan cetak biru teknis untuk sistem yang ingin dibangun. Hal ini termasuk desain arsitektur sistem, desain antarmuka, desain *database*, dan desain proses. Penekanan diterapkan bahwa desain harus mempertimbangkan aspek modularitas untuk memfasilitasi pengujian dan pemeliharaan di masa depan. (Pressman & Maxim, 2020)

4. *Implementation* (Implementasi)

Implementasi adalah fase yang mengubah sistem yang dirancang menjadi kode perangkat lunak yang dapat dieksekusi. Tidak lupa juga pengembang harus mengikuti standar pengkodean yang ditetapkan penting untuk memastikan pengembangan yang efisien dan pengujian yang mudah. (Kendall & Kendall, 2020)

5. *Testing* (Pengujian)

Tahap pengujian seringkali dianggap sebagai tahap penting untuk memastikan bahwa sistem beroperasi sesuai spesifikasi. Pengujian meliputi pengujian unit, pengujian integrasi, pengujian sistem, dan pengujian penerimaan pengguna (UAT). Setiap jenis pengujian dirancang untuk mengidentifikasi dan memperbaiki kesalahan sebelum sistem dimulai. (Pressman & Maxim, 2020)

6. *Deployment* (Penerapan)

Penerapan atau *Deployment* adalah fase di mana sistem yang diuji diperkenalkan ke lingkungan produksi. Ahli mencatat bahwa implementasi sering kali memerlukan pelatihan pengguna dan

migrasi data dari sistem lama ke sistem baru. Penting untuk memastikan bahwa semua pengguna akhir menerima pelatihan yang sesuai sebelum sistem diimplementasikan sepenuhnya (Schwalbe, 2021)

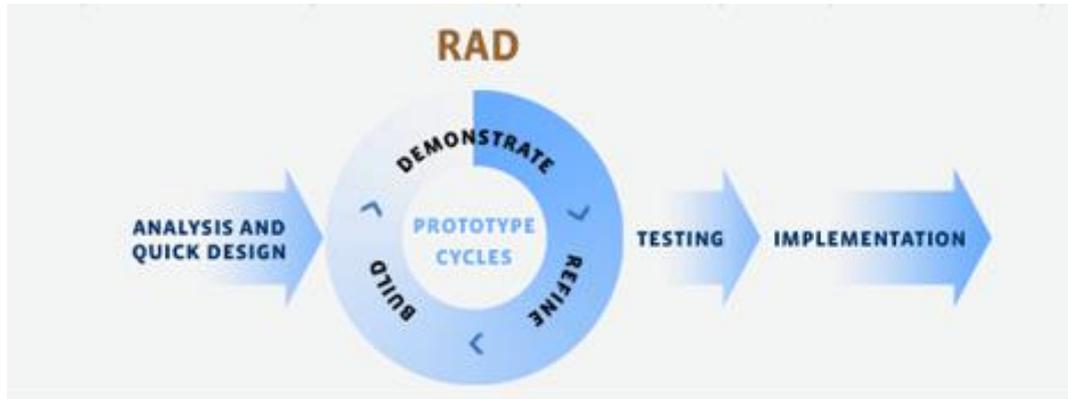
#### 7. *Maintenance* (Pemeliharaan)

Pemeliharaan adalah fase pemantauan dan pemutakhiran sistem yang diterapkan secara terus menerus. Perlu ditekankan bahwa pemeliharaan bukan hanya tentang memperbaiki kesalahan, namun juga tentang mengadaptasi sistem terhadap perubahan kebutuhan bisnis dan teknologi baru. (Boehm, 2020)

#### 1.1.11 *Rapid Application Development (RAD)*

*Rapid Application Development (RAD)* adalah metode pengembangan perangkat lunak yang menitikberatkan pada kecepatan, iterasi, dan kolaborasi tim untuk menghasilkan produk dalam waktu singkat. Pendekatan ini diperkenalkan sebagai solusi atas keterbatasan model waterfall yang cenderung memakan waktu lebih lama. Dalam RAD, proses pengembangan perangkat lunak melibatkan pembuatan prototipe dan pengumpulan umpan balik dari pengguna secara berulang, sehingga memastikan produk yang dihasilkan sesuai dengan kebutuhan bisnis. Metode ini membagi tahapan pengembangan menjadi beberapa fase utama, seperti perencanaan, pembuatan prototipe, iterasi, dan implementasi. Dengan pembagian ini, pengembang dapat lebih fleksibel dalam menyesuaikan perubahan kebutuhan di tengah proses pengembangan (Boehm, 2022). Di era modern, terutama dalam konteks agile development, pendekatan RAD semakin banyak diterapkan karena kemampuannya untuk mempercepat waktu peluncuran tanpa mengorbankan kualitas. Berdasarkan penelitian terbaru, RAD memiliki beberapa keunggulan, seperti peningkatan kerja sama antara pemangku kepentingan dan pengembang, fleksibilitas dalam menghadapi perubahan, serta penghematan biaya pengembangan dibandingkan dengan metode tradisional (Smith &

Turner, 2023). Namun, RAD juga memiliki tantangan, seperti kebutuhan akan sumber daya yang lebih besar dan risiko kegagalan jika tim pengembang kurang berpengalaman. Oleh karena itu, penerapan RAD membutuhkan perencanaan yang matang dan tim yang terampil agar dapat berjalan dengan baik.



Gambar 2. 5 RAD (Rapid Application Development)

Sumber (Dede Kurniadi dan Asri Mulyani, 2016)

#### 1.1.12 *Black Box Testing*

Black Box Testing adalah metode pengujian perangkat lunak yang berfokus pada bagaimana aplikasi berfungsi dari perspektif pengguna akhir, tanpa mempertimbangkan bagaimana kode atau struktur internalnya bekerja. Pengujian dilakukan dengan menguji input dan output dari sistem untuk memastikan bahwa aplikasi berjalan sesuai dengan spesifikasi yang ditetapkan. Penguji berperan layaknya pengguna yang ingin memastikan perangkat lunak dapat memenuhi kebutuhan mereka. Pendekatan ini umumnya digunakan untuk mengidentifikasi berbagai masalah, seperti kesalahan fungsional, masalah antarmuka, atau masalah kompatibilitas antar platform (Beizer, 2021). Biasanya, Black Box Testing diterapkan pada tahap akhir pengembangan perangkat lunak untuk memastikan produk siap digunakan. Menurut riset terbaru, metode ini memiliki keuntungan besar, seperti kemudahan implementasi dan kemampuannya untuk digunakan oleh penguji yang tidak perlu memiliki pemahaman mendalam tentang kode sumber (Kumar & Sharma, 2023). Namun, Black Box Testing juga memiliki beberapa

keterbatasan, salah satunya adalah kesulitan dalam mendeteksi kesalahan yang ada pada logika internal sistem. Oleh karena itu, metode ini sering dipadukan dengan teknik lain, seperti White Box Testing, untuk memberikan gambaran yang lebih menyeluruh dalam pengujian perangkat lunak.

SDLC (*Software Development Life Cycle*) adalah proses yang berurutan pada suatu awal perencanaan pembuatan sistem ataupun aplikasi dan melibatkan beberapa tahapan seperti analisis kebutuhan, perancangan sistem, implementasi, pengujian dan pemeliharaan. Penekanan pada setiap tahapan bertujuan untuk menghasilkan sistem yang matang sehingga setiap tahapan memiliki *output* yang maksimal dengan memperoleh hasil akhir yang juga tepat sasaran dan sesuai kebutuhan.

## **2.2 Tinjauan Studi**

Beberapa hasil penelitian sebelumnya yang relevan dengan pembuatan dan pengembangan sistem ataupun aplikasi yang menghubungkan minat bakat dengan mahasiswa ataupun siswa dengan tujuan mengembangkan keterampilan minat bakat tersebut dengan disalurkan kepada kegiatan yang terkait. Berikut beberapa penelitian yang dapat dijadikan referensi serta acuan pada penelitian ini.

Penelitian yang berjudul “Sistem Monitoring Penelurusan Minat dan Bakat Mahasiswa” oleh Iin Nurkarima, Masna Wati dan Novianti Puspitasari (2020) pada jurnal Explore p-ISSN 2087-894 dan e-ISSN 2656-615 Volume 10 No 2 Tahun 2020 Universitas Mulawarman melakukan penelitian serta pembuatan sistem monitoring penelurusan minat dan bakat mahasiswa. Secara umum, konsep dari sistem monitoring penelurusan minat dan bakat mahasiswa ini ialah suatu sistem untuk membantu mengumpulkan data serta memantau kegiatan mahasiswa. Sistem monitor ini merupakan salah satu dari proses pengumpulan data dari berbagai sumber untuk menilai suatu pelaksanaan program ataupun kegiatan. Sistem monitoring ini juga dikembangkan untuk mengaktualisasi minat bakat mahasiswa serta membantu mahasiswa mengidentifikasi dan mengembangkan minat bakat mereka melalui keikutsertaan dalam berbagai *event*, lomba maupun kegiatan lainnya yang relevan dengan minat bakat mahasiswa. Tujuan dari pembuatan sistem monitoring ini ialah untuk membantu para mahasiswa dalam meningkatkan pencapaian mahasiswa tidak hanya didalam lingkup akademik namun juga pada

bidang non akademik seperti hobi dan keterampilan lainnya. Hasil dari implementasi sistem ini menunjukkan bahwa dari 880 mahasiswa yang diamati, 90 di antaranya memiliki pencapaian yang tercatat dalam sistem dengan total 220 data pencapaian. Sistem ini menggunakan metode *black box* sebagai alat *testing* untuk memastikan sistem berjalan dengan baik. Kesimpulan dari penelitian ini ialah sistem monitoring minat bakat sangat penting dan memunculkan hasil yang positif dalam memfasilitasi keterampilan mahasiswa yang relevan dengan jurusan maupun keterampilan lain yang dimiliki oleh tiap mahasiswa. Pengelolaan data mengenai minat bakat mahasiswa juga memberikan kemudahan untuk mencari delegasi ke tiap-tiap kegiatan maupun *event* sehingga mahasiswa dapat secara *real* mendapatkan pengalaman ataupun prestasi di bidang yang diminati baik di bidang akademik maupun non akademik.

Selanjutnya, studi yang berjudul “Perancangan Sistem Informasi Manajemen Kegiatan Mahasiswa Berbasis Web” oleh Ana Juita Oktasari dan Denny Kurniadi (2019) pada Jurnal Vokasional Teknik Elektronika dan Informatika E-ISSN 2302-3295 Vol 7, No. 4, Desember 2019 Universitas Negeri Padang. Secara garis besar, penelitian ini memberikan hasil dari sebuah sistem informasi manajemen pada unit kegiatan mahasiswa. Sistem informasi manajemen UKM ini memiliki fungsi dalam mengelola wadah dari suatu UKM agar UKM yang ada di Universitas Negeri Padang dapat lebih mudah dalam hal pengajuan, pelaksanaan, pelaporan sampai penyaluran minat bakat yang dimiliki mahasiswa. Suatu UKM dinilai sangat penting untuk mewadahi mahasiswa yang memiliki keterampilan tertentu. Maka dari itu, dibutuhkan suatu pengelolaan yaitu sistem informasi manajemen UKM untuk menunjang serta mempermudah dalam hal pengembangan minat bakat mahasiswa. Selain itu, tujuan dari dibuatnya sistem informasi manajemen UKM ini ialah untuk membuat proses kegiatan UKM menjadi terstruktur, mempermudah pengelolaan data minat bakat mahasiswa dan pengajuan kegiatan UKM. Dari penelitian ini dapat diambil kesimpulan bahwa suatu sistem yang digunakan untuk mengelola suatu hal pada penelitian ini yaitu UKM, dapat meningkatkan produktifitas dan memberikan semangat lebih dari kemudahan yang diberikan sehingga mahasiswa dapat menyalurkan minat bakatnya dengan mudah dan terkelola dengan baik. Penggunaan sistem pada zaman serba digital memang

sangat penting dan dapat terlihat dari manfaat pembuatan sistem informasi manajemen UKM yaitu membantu pengelolaan, penyaluran dan strukturalisasi dengan menggunakan *tools-tools* pemrograman seperti PHP, *framework* dan *database* yang dirancang khusus untuk mengelola kegiatan-kegiatan UKM di Universitas Negeri Padang yang tentunya mempunyai mahasiswa dengan keterampilan dan minat bakat yang berbeda baik di bidang akademik maupun non akademik.

Lalu, studi yang berkaitan dengan penelitian yang sedang dilakukan juga didapatkan yang berjudul “PERANCANGAN DAN IMPLEMENTASI APLIKASI SISTEM INFORMASI PENGELOLAAN DATA UKM (UNIT KEGIATAN MAHASISWA)BERBASIS WEB DI POLITEKNIK TEDC BANDUNG” oleh D. S Ramdan dan Syam Ardy Bangun Putra (2020) pada Jurnal TEDC (e) ISSN 2776 – 723X (p) ISSN 1978-0060 Vol 14 No 1 (2020) Politeknik TEDC Bandung. Secara garis besar, studi ini membahas tentang perancangan suatu sistem informasi pengelolaan UKM berbasis web pada Politeknik TEDC Bandung. Seperti yang diketahui bahwa minat dan bakat tiap mahasiswa berbeda-beda dan terkadang sulit di kelola secara manual karena antusias yang besar dan jumlah mahasiswa yang banyak dalam satu UKM. Objek penelitian pada studi ini ialah Unit Kegiatan Mahasiswa pada Politeknik TEDC Bandung yang meliputi proses di dalam UKM tersebut seperti keanggotaan, program kegiatan serta profil UKM. Masalah utama dari studi ini ialah belum ada suatu sistem yang terintegrasi dan dapat mengelola data UKM yang menyebabkan sulitnya pengelolaan suatu organisasi UKM dalam hal administratif dan data. Sebelumnya proses pengelolaan dilakukan secara manual namun hal tersebut memiliki banyak resiko seperti data hilang dan rusak. Disisi lain, mahasiswa juga kesulitan untuk mendapatkan informasi mengenai kegiatan UKM yang diselenggarakan dan menghasilkan kebingungan mahasiswa dalam menyalurkan minat bakatnya. Solusi dari masalah dalam studi ini ialah membangun suatu sistem untuk mengelola data UKM secara terintegrasi. Sistem ini dirancang untuk mempermudah pengelolaan para anggota UKM dalam menjalankan tugas harian sebagai wadah minat bakat mahasiswa. Selain itu, penyaluran informasi serta minat bakat mahasiswa dapat dilakukan lebih efektif dengan adanya informasi kegiatan dan data mahasiswa yang terintegrasi dengan UKM serta minat bakat yang

dimiliki masing-masing mahasiswa. Tujuan dari studi ini adalah untuk melakukan pengembangan terhadap suatu UKM agar lebih efisien dan efektif serta memudahkan penyaluran informasi bagi mahasiswa. Sistem ini diharapkan dapat memecahkan masalah pengelolaan data yang terstruktur dan terorganisir sehingga penyaluran minat mahasiswa melalui UKM dapat berjalan dengan lancar.

Selanjutnya ditemukan juga studi yang berjudul “UJI BLACK BOX PADA SISTEM INFORMASI MINAT BAKAT PENERIMAAN MAHASISWA BARU” oleh Taufik, Ruki Rizal Nul Fikri dan Isnandar Agus (2023) pada Jurnal Teknika ISSN 0854-3143 e-ISSN 2622-3481 Vol 17 No 1 (2023) Institut Bakti Nusantara. Secara umum, penelitian berfokus pada suatu pengujian sistem informasi minat bakat pada Institut Bakti Nusantara dan memastikan bahwa sistem berjalan dengan baik tanpa adanya hambatan. Pada pengujian ini, peneliti menggunakan pengujian *black box* dengan memperhatikan aspek pengujian fungsi sistem berdasarkan input dan *output*. Masalah yang dihadapi pada penelitian ini adalah peneliti ingin memastikan bahwa sistem informasi minat bakat mahasiswa berfungsi dengan baik setelah berjalan dan di implementasikan pada objek penelitian. Peneliti melakukan pengujian untuk memastikan bahwa keputusan yang diberikan oleh sistem merupakan keputusan akurat dan objektif serta peneliti juga memastikan sistem dapat mengelola data dengan benar dalam menangani informasi input yang tidak lengkap ataupun tidak valid. Maka dari itu diberikan suatu solusi berupa penerapan uji *black box* untuk mendeteksi potensi kelemahan atau kesalahan yang terdapat pada sistem informasi minat bakat. Pengujian ini dilakukan dengan menerapkan *scenario* nyata yaitu proses penerimaan mahasiswa dan pengelolaan mahasiswa dengan minat bakat. Tujuan dari pengujian ini ialah menguji fungsionalitas sistem dan keandalan sistem minat bakat untuk mengidentifikasi minat bakat dan memperbaiki potensi kelemahan serta kerusakan sistem. Kesimpulan dari penelitian ini ialah peneliti berusaha mengembangkan sistem informasi minat bakat mahasiswa yang telah berjalan dalam membantu proses penerimaan siswa, pengelolaan sampai penentuan minat bakat mahasiswa. Hal ini sangat penting karena kita dapat membuktikan bahwa sistem benar-benar digunakan dalam pekerjaan sehari-hari sesuai kebutuhan dan dapat mempermudah operator dalam menentukan minat bakat mahasiswa dan pengelolaan data mahasiswa yang

memiliki minat bakat. Dari setiap pengujian diharapkan hal yang diperbaiki dapat meningkatkan kualitas sistem, efisiensi dan juga efektifitas sistem dalam membantu pekerjaan operator sehari-hari.

Terakhir, studi yang juga berkaitan dan berguna sebagai referensi pada penelitian ini yaitu studi berjudul “RANCANG BANGUN APLIKASI MONITORING UNIT KEGIATAN MAHASISWA (UKM) UNIVERSITAS TEKNOLOGI SUMBAWA BERBASIS WEB” oleh Sri Lis Apriliani, Shinta Esabella dan M. Julkarnain (2020) pada jurnal Hexagon Teknik dan Sains Fakultas Teknik Universitas Teknologi Sumbawa e-ISSN 2721-3188 P-ISSN 2721-3714 Volume 1 Nomor 2, Juli 2020. Secara umum, studi ini menawarkan sebuah solusi dari suatu masalah yang ada pada Unit Kegiatan Mahasiswa (UKM) yang berada di Universitas Teknologi Sumbawa. Masalah yang dihadapi oleh peneliti yaitu bagian kemahasiswaan mendapatkan kesulitan dalam memantau keaktifan UKM beserta anggotanya yang berjumlah cukup banyak. Belum ada sebuah fasilitas khusus yang disediakan kampus untuk menyajikan informasi mengenai UKM, keaktifan anggota UKM dan pengelolaan UKM lainnya. Hal ini mempersulit bagian kemahasiswaan dalam mengetahui aktivitas mahasiswa dalam UKM serta prestasi yang telah dicapai oleh setiap UKM. Peneliti menawarkan solusi berupa Pembangunan aplikasi monitoring berbasis web yang memungkinkan bagian kemahasiswaan untuk memantau keaktifan UKM, program, prestasi dan anggota secara rinci dan tentunya efektif dan efisien. Tujuan dari penelitian ini ialah untuk memberikan kemudahan dan efektivitas waktu bagian kemahasiswaan dalam mengelola dan memonitor aktivitas UKM di Universitas Teknologi Sumbawa. Diharapkan dengan adanya aplikasi ini maka pengelolaan data UKM akan lebih terstruktur dan mudah diakses. Hasil dari penelitian ini ialah setelah selesai maka akan memungkinkan pihak bagian kemahasiswaan untuk langsung melakukan monitoring dan evaluasi terhadap setiap UKM dan memantau prestasi yang telah dicapai demi mengembangkan lebih lanjut minat bakat dari tiap mahasiswa yang menjadi anggota UKM pada Universitas Teknologi Sumbawa.

Dari berbagai penelitian yang telah membahas pentingnya sistem berbasis teknologi untuk berperan pada suatu pengembangan bidang di universitas menunjukkan seberapa pentingnya studi yang sedang berjalan pada penelitian ini yang berkaitan dengan bidang minat bakat pada suatu universitas. Misalnya, penelitian "Sistem Monitoring Penelurusan Minat dan Bakat Mahasiswa" pada tahun 2020 menunjukkan bahwa sistem *monitoring* dapat membantu mahasiswa mengenali dan mengembangkan bakat mereka melalui berbagai kegiatan yang relevan. Penelitian lain, seperti "Perancangan Sistem Informasi Manajemen Kegiatan Mahasiswa Berbasis Web" pada tahun 2019 dan "Rancang Bangun Aplikasi Monitoring UKM Universitas Teknologi Sumbawa" pada tahun 2020, menekankan pentingnya sistem berbasis web untuk mempermudah pengelolaan data Unit Kegiatan Mahasiswa (UKM), memantau aktivitas, dan melaporkan hasil kegiatan. Kesimpulan dari berbagai studi ini menunjukkan bahwa pemanfaatan teknologi dapat membuat pengelolaan minat bakat lebih efisien, terstruktur, dan mendukung mahasiswa dalam mencapai prestasi baik di bidang akademik maupun non-akademik. Selain itu, pengujian sistem, seperti uji black box dapat sangat membantu dalam memastikan bahwa sistem berjalan dengan baik dan dapat diandalkan.

Dari beberapa penelitian terdahulu, dapat disimpulkan suatu perbedaan yang membuat penelitian yang sedang berjalan mengenai minat bakat mahasiswa berbeda dari yang terdahulu. Penelitian terdahulu lebih fokus kepada proses analisa dan penggunaan untuk bidang yang terkait. Walaupun tidak semuanya, tetapi kebanyakan penelitian terdahulu hanya menguntungkan biro atau bidang sebagai pengguna. Pada penelitian yang berjalan di studi ini, mahasiswa juga mendapatkan keuntungan yang sangat penting sebagai aktor yang dapat mengajukan minat bakatnya. Hal ini sangat menguntungkan bagi situasi akademik maupun non akademik tiap individu mahasiswa.