



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Kata komputer berasal dari bahasa Yunani yaitu *computare* yang berarti menghitung. Fuori mendefinisikan komputer sebagai suatu pemroses data yang dapat melakukan perhitungan besar secara cepat, termasuk perhitungan aritmatika dan operasi logika tanpa campur tangan manusia (Pardede, 2020).

Menurut Robert H. Blissmer, pengertian komputer adalah suatu alat elektronik yang mampu melakukan beberapa tugas, yaitu menerima *input*, memproses *input* sesuai dengan instruksi yang diberikan, menyimpan perintah-perintah dan hasil pengolahannya, serta menyediakan *output* dalam bentuk informasi (Harmayani et al., 2021).

Definisi komputer adalah alat elektronik, dapat menerima *input* data, dapat mengolah data, dapat memberikan informasi, menggunakan suatu program yang tersimpan di memori komputer (*stored program*), dapat menyimpan program dan hasil pengolahan, bekerja secara otomatis (Akhmad, 2019).

Dari beberapa pengertian diatas disimpulkan bahwa Komputer adalah suatu perangkat elektronik yang dapat menerima dan mengelola data mentah hingga menjadi informasi bermanfaat.

2.1.2 Pengertian Perangkat Lunak (*Software*)

Software adalah bagian sistem komputer yang tidak memiliki wujud. *Software* juga bisa memiliki pengertian sebagai data yang berformat digital dan disimpan secara digital yang hanya bisa dibaca oleh komputer (Endraratha, 2022).

Sedangkan menurut Lubis (2020) *Software* merupakan kumpulan beberapa perintah yang dieksekusi oleh mesin komputer untuk menjalankan tugasnya. *Software* merupakan data elektronik yang disimpan sedemikian rupa oleh komputer, data yang disimpan dapat berupa program atau intruksi yang akan dijalankan oleh perintah, atau catatan-catatan yang diperlukan oleh komputer untuk mencapai tujuannya.



Software merupakan data elektronik yang disimpan sedemikian rupa oleh komputer, data yang disimpan dapat berupa program atau intruksi yang akan dijalankan oleh perintah, atau catatan- catatan yang diperlukan oleh komputer untuk mencapai tujuannya (Rahma, 2020).

Dari pengertian diatas disimpulkan bahwa Perangkat Lunak (*Software*) adalah kumpulan instruksi, data atau program yang digunakan untuk mengoperasikan komputer dan menjalankan tugas-tugas tertentu.

2.1.3 Pengertian Perangkat Keras (*Hardware*)

Pengertian *hardware* adalah suatu komponen suatu perangkat elektronik yang fungsinya sebagai penunjang kinerja perangkat tersebut. Kemudian *hardware* dapat disentuh oleh pengguna perangkat tersebut dan digunakan sesuai fungsi dari perangkat tersebut (Widianingsih et al., 2021).

Hardware merupakan perangkat keras yang kasat mata dan memiliki wujud fisik yang dapat diraba (Harmayani et al., 2021).

Menurut James O'Brien dalam (Shiddiq, 2023) *hardware* adalah semua peralatan fisik yang digunakan dalam pemrosesan informasi, termasuk diantaranya CPU, RAM, monitor, *mouse*, *keybord*, *printer*, *scanner*, dan lain-lain. Perangkat keras merupakan media komunikasi yang menghubungkan beberapa jaringan dan memproses paket data sehingga trasmisi data lebih efektif.

Dari pengertian di atas disimpulkan bahwa Perangkat Keras (*Hardware*) merupakan komponen fisik komputer yang bisa dilihat oleh mata dan dipegang yang digunakan dalam pemrosesan informasi.

2.1.4 Pengertian Basis Data (*Database*)

Basis Data (*Database*) terdiri dari kata basis dan data. Basis disebut juga sebagai markas, gudang atau tempat pengumpulan. Sedangkan data merupakan catatan atas kumpulan fakta dunia nyata yang mewakili objek seperti manusia, barang, hewan, konsep, peristiwa dan lain sebagainya yang diwujudkan dalam bentuk huruf, angka, simbol, gambar, teks, bunyi atau kombinasi lainnya. (Rachmadi, 2020).



Menurut Endraratha (2022) “*Database* merupakan suatu kumpulan beberapa tabel yang saling berhubungan satu dengan lainnya, hubungan atau relasi tersebut dapat ditunjukkan sebagai kunci pada setiap tabel tersebut. Satu *database* dapat menunjukkan satu populasi data yang digunakan dalam satu unit kerja suatu lembaga, perusahaan atau organisasi”.

Menurut (Pane, 2020) secara garis besar, basis data merupakan kumpulan informasi yang disimpan pada komputer secara sistematis sehingga dapat diperiksa dikemudian hari menggunakan suatu program komputer untuk mendapatkan informasi dari basis data tersebut.

Dari pengertian di atas disimpulkan bahwa definisi Basis Data (*Database*) merupakan sekumpulan data yang dikelola dengan sedemikian rupa berdasarkan ketentuan tertentu yang saling berkaitan sehingga memudahkan dalam pengelolaannya.

2.1.5 Pengertian Internet

Sudarsono (2021) menjelaskan *Interconnected network* atau yang lebih populer dengan sebutan internet adalah sebuah sistem komunikasi global yang menghubungkan komputer-komputer dan jaringan-jaringan komputer di seluruh dunia.

Menurut Rusman dalam (Rahman et al., 2019) mengemukakan bahwa internet merupakan bentuk penerapan teknologi informasi yang ditunjukkan untuk mempermudah proses pembelajaran yang dikemas dalam bentuk digital konten dan pelaksanaannya membutuhkan sarana komputer yang dikoneksikan ke internet.

Internet adalah suatu jaringan komunikasi yang memiliki fungsi untuk menghubungkan antara satu media elektronik dengan media elektronik yang lain dengan cepat dan tepat. Jaringan komunikasi tersebut, akan menyampaikan beberapa informasi yang dikirim melalui transmisi sinyal dengan frekuensi yang telah disesuaikan. Untuk standar global dalam penggunaan jaringan internet sendiri menggunakan TCP / IP (*Transmission Control Protocol / Internet Protocol*). Istilah TCP / IP merupakan bentuk protokol pertukaran paket yang digunakan oleh



berbagai pengguna global / dunia. Kemudian, proses untuk menghubungkan antara rangkaian internet disebut dengan “*internetworking*” (Noviantoroa et al., 2022).

Dari pengertian di atas disimpulkan bahwa internet adalah sebuah jaringan yang berfungsi untuk menghubungkan antara satu media elektronik dengan media lainnya.

2.2 Teori Judul

2.2.1 Pengertian Sistem

Sistem merupakan suatu tatanan yang terdiri dari sejumlah komponen fungsional dengan tugas atau fungsi khusus yang berkaitan dan kemudian secara bersama-sama memiliki tujuan untuk memenuhi suatu proses atau pekerjaan tertentu (Gede Endraratha, 2022).

Sistem adalah suatu himpunan benda nyata atau abstrak (*a set of thing*) yang terdiri dari bagian-bagian atau komponen - komponen yang saling berkaitan, berhubungan, ketergantungan, saling mendukung, yang secara keseluruhan bersatu dalam satu kesatuan (*unity*) untuk mencapai tujuan tertentu secara efisien dan efektif (Putra et al., 2023).

Pengertian sistem adalah suatu kesatuan yang terdiri komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi atau energi untuk mencapai suatu tujuan (Effendy et al., 2023).

Dari beberapa pengertian di atas disimpulkan bahwa sistem merupakan kumpulan elemen yang saling berhubungan dan berinteraksi dalam satu kesatuan untuk menjalankan suatu proses pencapaian suatu tujuan utama.

2.2.2 Pengertian Informasi

Menurut Agung et al., (2022) menuliskan bahwa Informasi adalah sekumpulan data atau fakta yang telah diproses dan dikelola sedemikian rupa sehingga menjadi sesuatu yang mudah dimengerti dan bermanfaat bagi penerimanya.

Jogianto mengemukakan bahwa informasi merupakan data yang telah diolah kedalam bentuk yang lebih berarti, berguna atau bermanfaat bagi orang yang



menerimanya. Hasil pengolahan data tersebut menggambarkan kejadian (event) yang nyata (fact) yang dapat digunakan untuk membuat atau mengambil keputusan (Effendy et al., 2023).

Sedangkan menurut Susanto dalam (Frisdayanti, 2019) Informasi merupakan hasil dari pengolahan data yang memberikan arti dan manfaat.

Dari beberapa pengertian di atas disimpulkan bahwa pengertian informasi adalah data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat dalam pengambilan keputusan saat ini atau saat mendatang.

2.2.3 Pengertian Sistem Informasi

Definisi dari sistem informasi (SI) secara umum adalah suatu sistem dimana terdiri dari gabungan teknologi atau alat dan aktivitas manusia yang membentuk suatu sistem yang terorganisir. Dimana kombinasi antara hubungan manusia dan teknologi sehingga menciptakan sebuah interaksi manusia, data, informasi, teknologi, dan algoritma (Firmansyah & Herman, 2023).

Menurut Wilkinson dalam (Fatimah & Samsudin, 2019) menuliskan bahwa Sistem informasi adalah kerangka kerja yang mengoordinasikan sumber daya (manusia, komputer) untuk mengubah masukan (*input*) menjadi keluaran (informasi), guna mencapai sasaran – sasaran perusahaan.

Sekumpulan prosedur organisasi yang pada saat dilaksanakan akan memberikan informasi bagi pengambil keputusan dan atau untuk mengendalikan organisasi (Asmara, 2019).

Dari beberapa pengertian di atas disimpulkan bahwa pengertian Sistem Informasi adalah seperangkat komponen yang saling terkait yang mengumpulkan, memproses, menyimpan, dan mendistribusikan informasi untuk mendukung pengambilan keputusan, koordinasi, dan kontrol dalam suatu organisasi.

2.2.4 Pengertian Eksekutif

Menurut kamus besar bahasa Indonesia (KBBI), Eksekutif dapat didefinisikan sebagai sesuatu yang berkenaan dengan pengurusan (pengelolaan, pemerintahan) atau penyelenggaraan sesuatu (Gesi et al., 2019).



Eksekutif digunakan untuk mengidentifikasi manajer tingkat puncak yang mempunyai pengaruh yang kuat terhadap perusahaan yaitu semua manajer yang berada pada tingkat perencanaan strategi. Jika perusahaan besar maka manajer dalam berapa tingkatan yang menyatu dalam struktur organisasi dianggap sebagai eksekutif. Eksekutif adalah semua pimpinan melakukan fungsi-fungsi pimpinan yang sama, merencanakan, mengorganisasikan, menyusun staf, mengarahkan dan mengendalikan. Perencanaan sangat ditentukan pada tingkat eksekutif, sedangkan fungsi-fungsi lain oleh tingkat yang lebih rendah.

Sistem Informasi Eksekutif (*Executive Information System – EIS*) merupakan suatu sistem yang menyediakan informasi bagi eksekutif mengenai kinerja keseluruhan perusahaan. Informasi dapat diambil dengan mudah dan dalam berbagai tingkat rincian (Wijoyo et al., 2019).

Dari beberapa pengertian di atas disimpulkan bahwa pengertian eksekutif merupakan tingkat kekuasaan yang tinggi seperti manajer yang berperan dalam dasar pengambilan keputusan.

2.2.5 Pengertian Manajemen

Istilah manajemen berasal dari kata *management* (bahasa Inggris), berasal dari kata “*to manage*” yang artinya mengurus atau tata laksana. Sehingga manajemen dapat diartikan bagaimana cara mengatur, membimbing dan memimpin semua orang yang menjadi bawahannya agar usaha yang sedang dikerjakan dapat mencapai tujuan yang telah ditetapkan sebelumnya (Hadion Wijoyo, 2021).

Menurut Stoner dalam (Juarni et al., 2023) manajemen adalah proses perencanaan, pengorganisasian, pengarahan, dan pengawasan usaha-usaha anggota organisasi dan penggunaan sumber daya organisasi lainnya agar mencapai tujuan organisasi yang telah ditetapkan.

Sedangkan menurut George R. Terry dalam buku dengan judul “*Principles of Management*” memberikan definisi: “Manajemen adalah suatu proses yang membedakan atas perencanaan, pengorganisasian, penggerakkan pelaksanaan dan pengawasan, dengan memanfaatkan baik ilmu maupun seni agar dapat menyelesaikan tujuan yang telah ditetapkan sebelumnya” (Wijoyo et al., 2021).



Dari beberapa pengertian di atas disimpulkan bahwa pengertian manajemen adalah proses pengorganisasian, pengaturan, pengelolaan SDM, sampai dengan pengendalian agar bisa mencapai tujuan dari suatu kegiatan.

2.2.6 Laporan Masuk Pada Ombudsman RI Perwakilan Sumatera Selatan

Ombudsman adalah Lembaga negara yang mengawasi penyelenggaraan Pelayanan Publik yang diselenggarakan oleh Penyelenggara Negara dan Pemerintah yang diselenggarakan oleh Badan Usaha Milik Negara serta badan Swasta atau perseorangan yang diberi tugas menyelenggarakan pelayanan publik tertentu. Ombudsman bertugas Menerima laporan masuk atas dugaan Maladministrasi dalam penyelenggaraan Pelayanan Publik yang di laporan kan oleh masyarakat yang kemudian dilakukan pemeriksaan dan akan ditindak lanjuti.

2.3 Teori Khusus

2.3.1 Metode Pengembangan Sistem

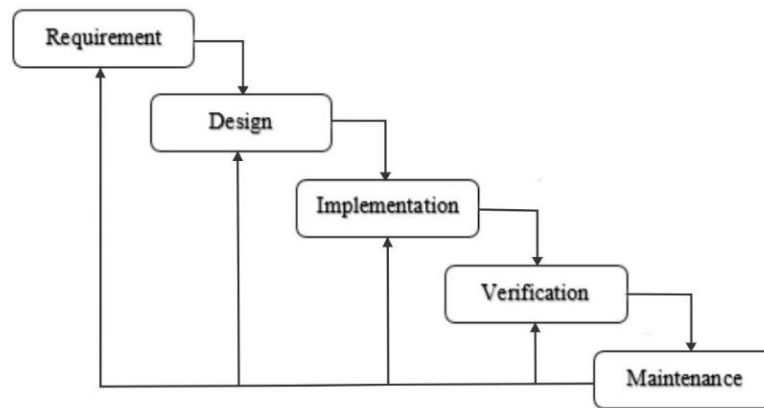
Pada Laporan Akhir ini penulis menggunakan metode pengembang sistem *Waterfall*. Pengertian *Waterfall* menurut Fadillah dan Diah Isti Mona (2023) *Waterfall* adalah rangkaian pada saat proses pengembangan *software* secara berurutan, dimana kemajuan akan tetap terus mengalir kebawah melalui proses perencanaan, pemodelan, implementasi dan pengujian.

Menurut Fachri dan Rizal (2024) Metode *Waterfall* adalah metode pengembangan sistem yang terstruktur dimana setiap tahapan dilakukan secara bertahap dan tidak boleh dilanjutkan sampai tahapan sebelumnya selesai. Metode ini memiliki beberapa keunggulan, termasuk membuat proses perancangan sistem lebih mudah karena tahapan-tahap ini harus dilakukan secara bertahap sampai dengan selesai sehingga proses penelitian tidak terganggu.

Sedangkan menurut Fadillah dan Diah Isti Mona (2023) Pengertian dari metode *waterfall* adalah rangkaian pada saat proses pengembangan *software* secara berurutan, dimana kemajuan akan tetap terus mengalir kebawah melalui proses perencanaan, pemodelan, implementasi dan pengujian.



Dari beberapa pengertian di atas dapat disimpulkan bahwa *waterfall* adalah metode pengembangan sistem yang dimana tiap tahapannya dikerjakan secara berurutan.



Gambar 2.1 Tahapan dalam *waterfall*

Tahapan-tahapan dalam metode *Waterfall* menurut Wahid (2020) adalah sebagai berikut :

a. Requirement

Tahap ini pengembang sistem diperlukan komunikasi yang bertujuan untuk memahami perangkat lunak yang diharapkan oleh pengguna dan batasan perangkat lunak tersebut. Informasi dapat diperoleh melalui wawancara, diskusi atau survei langsung. Informasi dianalisis untuk mendapatkan data yang dibutuhkan oleh pengguna.

b. Design

Pada tahap ini, pengembang membuat desain sistem yang dapat membantu menentukan perangkat keras (*hardware*) dan sistem persyaratan dan juga membantu dalam mendefinisikan arsitektur sistem secara keseluruhan.

c. Implementation

Pada tahap ini, sistem pertama kali dikembangkan di program kecil yang disebut unit, yang terintegrasi dalam tahap selanjutnya. Setiap unit dikembangkan dan diuji untuk fungsionalitas yang disebut sebagai *unit testing*.



d. Verification

Pada tahap ini, sistem dilakukan verifikasi dan pengujian apakah sistem sepenuhnya atau sebagian memenuhi persyaratan sistem, pengujian dapat dikategorikan ke dalam unit testing (dilakukan pada modul tertentu kode), sistem pengujian (untuk melihat bagaimana sistem bereaksi ketika semua modul yang terintegrasi) dan penerimaan pengujian (dilakukan dengan atau nama pelanggan untuk melihat apakah semua kebutuhan pelanggan puas).

e. Maintenance

Ini adalah tahap akhir dari metode *waterfall*. Perangkat lunak yang sudah jadi dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya.

2.3.2 Pengertian Unified Modelling Language (UML)

Menurut (Destriana, 2021) dalam (Irfan et al., 2023) “UML merupakan bahasa buat memvisualisasi, membangun, menspesifikasi serta mendokumentasikan artifacts (bagian dari data yang dipakai buat ditampilkan oleh mekanisme pembuatan piranti lunak, artifact tersebut bisa berbentuk model, uraian ataupun piranti lunak) dari sistem piranti lunak, seperti untuk pemodelan bisnis serta pola non piranti lunak yang lain. Tidak hanya itu UML merupakan bahasa pemodelan yang memakai konsep orientasi object. Dibawah bendera Racional *Software Corps* UML di dirikan oleh James Rumbaugh, Ivar Jacobson, serta Grady Booch. UML sediakan notasi- notasi yang menolong memodelkan sistem dari bermacam prespektif. UML bukan cuma dipakai dalam pemodelan piranti lunak, tetapi nyaris dalam seluruh bidang yang memerlukan pemodelan.

Menurut Abdillah (2021), *Unified Modeling Language* (UML) merupakan suatu bahasa pemodelan piranti lunak yang sudah distandardisasi selaku media penyusunan cetak biru (*blueprints*) piranti lunak (Pressman). UML mampu saja dipakai buat spesifikasi, visualisasi, dokumentasi serta kontruksi sekian banyak bagian-bagian dari sistem yang terdapat dalam piranti lunak. Dengan kata lain, semacam halnya seseorang arsitek dalam membuat dokumen cetak biru yang dipakai oleh industri konstruksi guna membangun suatu bangunan, arsitek piranti



lunak membuat diagram- diagram UML guna menolong programmer/ pengembang membuat perangkat lunak. Guna berikutnya, semakin kita mengenali sebagian kosakata yang dipakai UML, kita akan semakin awam dalam menguasai spesifik.

Unified Modeling Language merupakan salah satu metode pemodelan visual yang digunakan dalam perancangan dan pembuatan sebuah software yang berorientasikan pada objek (Widyastuti et al., 2022).

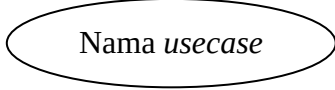
Berdasarkan beberapa definisi diatas, dapat disimpulkan bahwa *Unified Modeling Language* (UML) merupakan salah satu metode pemodelan visual yang digunakan dalam perancangan dan pembuatan sebuah software yang berorientasikan pada objek. UML merupakan sebuah standar penulisan atau semacam *blue print* dimana didalamnya termasuk sebuah bisnis proses, penulisan kelas-kelas dalam sebuah bahasa yang spesifik.

2.3.3 Jenis-Jenis Diagram UML (*Unified Modeling Language*)

2.3.3.1 Pengertian Use Case Diagram

Use case diagram ialah pemodelan guna kegiatan sistem data yang hendak dibentuk. *Use case* menjelaskan suatu hubungan antara satu ataupun lebih aktor menggunakan sistem data yang hendak dibentuk. *Use case* dipakai buat mengenali peranan apa saja yang terdapat untuk suatu sistem data serta siapa saja yang berwenang memakai fungsi- fungsi tersebut (Julianto & Setiawan, 2019). Adapun simbol-simbol diagram *use case* adalah sebagai berikut ini:

Tabel 2. 1 Simbol-simbol *Use Case* Diagram

No.	Simbol	Deskripsi
1.		Fungsionalitas yang ditunjukan oleh system Ketika komponen berpindah pesan antar unit ataupun aktor; sering ditunjukan menggunakan kata kerja di awal frase nama <i>use case</i> .

Lanjutan Tabel 2.1 Simbol-simbol *Use Case Diagram*

2.	Aktor  Nama Aktor	Proses, orang, serta sistem lain yang berhubungan dengan sistem informasi yang sedang dibuat terletak di sistem informasi yang sedang dibuat itu sendiri, sehingga aktor belum tentu orang meskipun simbol aktor ialah gambar seseorang; mayoritas kata benda digunakan didepan frase nama aktor untuk mengekspresikan.
3.	Asosiasi 	Interaksi antar aktor serta <i>use case</i> yang berpartisipasi untuk <i>use case</i> atau <i>use case</i> mempunyai interaksi dengan <i>actor</i> .
4.	Generalisasi / generalitation 	Hubungan generalisasi dan spesialisasi antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
5.	Extend <<extends>> ----->	Hubungan <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang di tambahkan bisa berdiri sendiri bahkan tanpa <i>use case</i> tambahan itu; sama dengan dasar pewarisan pada pemrograman berorientasi objek; Biasanya <i>use case</i> lain memiliki nama yang sama, seperti panah yang menunjuk ke <i>use case</i> yang ditambahkan; biasanya, <i>use case</i> yang diperluas memiliki jenis yang sama dengan <i>use case</i> induknya menunjuk ke <i>use case</i> yang digeneralisasikan (generik).

Lanjutan Tabel 2.1 Simbol-simbol *Use Case Diagram*

6.	<i>Include</i> <include> ----->	Hubungan <i>Use case</i> tambahan ke sebuah <i>use case</i>, dimana <i>use case</i> yang ditambahkan membutuhkan <i>use case</i> untuk menjalankan fungsinya atau berfungsi sebagai syarat untuk eksekusi <i>use case</i>. Ada dua pandangan utama tentang penyertaan dalam kasus penggunaan: 1. <i>Include</i> artinya <i>use case</i> tambahan akan selalu dipanggil saat <i>use case</i> tambahan dijalankan. 2. <i>Include</i> artinya <i>use case</i> tambahan akan selalu dicek apakah <i>use case</i> yang ditambahkan sudah dieksekusi sebelum <i>use case</i> tambahan dijalankan.
----	--	---

Sumber; (Irfan et al., 2023)

2.3.3.2 Pengertian *Activity Diagram*

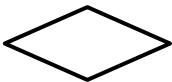
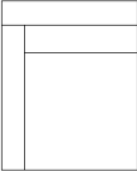
Menurut (Sukanto & Shalahuddin, 2018:161) menjelaskan “Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak”.

Menurut Julianto & Setiawan (2019) *Activity diagram* ataupun diagram aktivitas menggambarkan alur kerja serta aktivitas dari suatu sistem serta mekanisme bisnis ataupun menu dalam piranti lunak. Diagram aktivitas lebih menekankan pada penggambaran kegiatan sistem ataupun aktivitas yang bisa digunakan sistem, dari pada apa yang diperbuat aktor.

Adapun simbol-simbol yang digunakan dalam *activity diagram* adalah sebagai berikut:



Tabel 2.2 Simbol-simbol pada Activity Diagram

No.	Simbol	Deskripsi
1.	Status awal 	Keadaan awal aktivitas sistem, diagram aktivitas mempunyai keadaan awal.
2.	Aktivitas 	Suatu kegiatan yang diperbuat oleh sistem, sering didahului oleh kata kerja.
3.	Percabangan/ <i>decision</i> 	Asosiasi cabang apabila lebih dari satu opsi aktivitas.
4.	Penggabungan/ <i>join</i> 	Untuk menyatukan asosiasi, untuk menyatukan lebih dari satu aktivitas mewujudkan satu aktivitas.
5.	Status akhir 	Keadaan akhir dari eksekusi system, diagram aktivitas mempunyai keadaan akhir.
6.	<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

Sumber : (Irfan et al., 2023)

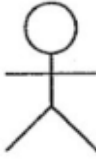
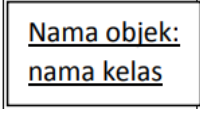


2.3.3.3 Pengertian Sequence Diagram

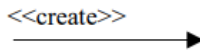
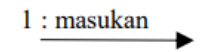
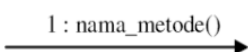
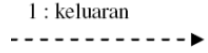
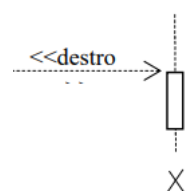
Sequence diagram "menggambarkan perilaku objek dalam kasus penggunaan dengan mengartikan siklus hidup object dan pesan yang dibawa dan masuk di antara objek. Oleh sebab itu, hendak memvisualkan diagram urutan, perlu untuk memahami objek yang berperan dalam kasus penggunaan dan objek yang dipakai ke dalam metode objek kelas". Menghasilkan diagram sekuen juga dibutuhkan guna mengamati skenario yang ada pada use case menurut Rosa dan Shalahuddin dalam (Julianto & Setiawan, 2019).

Berikut ini adalah simbol-simbol *sequence* diagram:

Tabel 2.3 Simbol-simbol *Sequence* Diagram

No.	Simbol	Deskripsi
1.	Aktor 	Proses, orang, ataupun sistem berbeda yang terhubung dengan sistem data yang sedang dibikin terletak di luar sistem informasi yang sedang dibikin itu sendiri, sehingga aktor belum tentu orang meskipun simbol aktor adalah gambar seseorang; biasanya kata benda digunakan di depan frase nama aktor untuk mengekspresikan.
2.	Garis hidup / <i>Lifeline</i> 	Mewakili siklus hidup suatu objek.
3.	Objek 	Mendeklarasikan object yang berinteraksi dengan pesan.
4.	Waktu aktif 	Mendeklarasikan bahwa objek itu aktif dan interaktif, dan segala sesuatu yang berhubungan dengan waktu aktif ini ialah langkah yang dibuat di dalamnya.

**Lanjutan Tabel 2.3** Simbol-simbol *Sequence Diagram*

5.	Pesan tipe Create 	Mendeklarasikan objek menciptakan objek lain, dan panah menunjuk ke objek yang dibikin. Arah panah menunjuk ke object yang memiliki operasi atau metode , sebab ini memanggil operasi atau metode , operasi atau metode yang dipanggil harus ada dalam diagram kelas sesuai dengan kelas objek yang berinteraksi. Kelas objek yang berinteraksi.
6.	Pesan tipe Send 	Merupakan object yang mengirimkan input atau data atau informasi ke object lain, dengan panah yang menunjuk ke objek pengirim.
7.	Pesan tipe call 	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri, Arah panah mengarah pada objek yang memiliki operasi/metode.
8.	Pesan tipe return 	Menunjukkan bahwa suatu objek kembali ke objek yang ditentukan setelah menjalankan operasi atau metode tertentu, dan panah menunjuk ke objek yang menerima pengembalian.
9.	Pesan tipe destroy() 	Deklarasikan objek untuk mengakhiri hidup objek lain, arah panah menunjuk ke objek yang diakhiri, yang terbaik adalah jika ada create maka ada destroy.

Sumber; (Irfan et al., 2023)

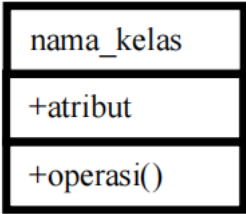


2.3.3.4 Pengertian Class Diagram

Menurut Rosa dan Shalahuddin (2019:141), Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

Adapun simbol-simbol dari *Class Diagram* adalah sebagai berikut:

Tabel 2.4 Simbol-simbol *Class Diagram*

No.	Simbol	Deskripsi
1.	Kelas 	Kelas pada struktur sistem.
2.	Antarmuka (<i>interface</i>)  nama_interface	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
3.	Asosiasi (<i>association</i>) 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
4.	Asosiasi berarah (<i>directed association</i>) 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
5.	Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus).

**Lanjutan Tabel 2.4** Simbol-simbol *Class Diagram*

6.	Ketergantungan (<i>dependency</i>) 	Relasi antar kelas dengan makna kebergantungan antar kelas.
7.	Agregasi (<i>aggregation</i>) 	Relasi antar kelas dengan makna semua bagian (<i>whole-part</i>).

2.3.4 Pengertian Kamus Data

“Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar cara penulisan)” (Sukanto dan Shalahuddin, 2018:72)

Menurut Abdul Kadir dalam (Antares, 2020) Kamus data adalah suatu daftar data elemen yang terorganisir dengan definisi yang tetap dan sesuai dengan sistem, sehingga user dan analisis sistem mempunyai pengertian yang sama tentang input, output dan komponen data store.

Menurut Rosa dan Shalahudin dalam (Sartika Muhammad Hadi, 2019) Kamus Data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar cara penulisan).

Dari beberapa pengertian diatas dapat disimpulkan bahwa Kamus data merupakan penjelasan arus data atau aliran data yang ada.

Kamus data memiliki beberapa simbol untuk menjelaskan informasi tambahan, sebagai berikut :

**Tabel 2.5** Simbol-simbol pada kamus data

Simbol	Keterangan
=	Disusun atau terdiri dari
+	Dan
[]	Baik...atau...
{ }n	n kali diulang atau bernilai banyak
()	Data opsional
...	Batas komentar

Sumber: Sukamto dan Shalahuddin (2018:72)

2.4 Teori Program

2.4.1 Pengertian Hypertext Markup Language (HTML)

HTML atau singkatan dari *HyperText Markup Language* merupakan salah satu bahasa *markup* standar yang digunakan untuk membuat sebuah halaman website. Dengan HTML kita dapat mendirikan sebuah website dengan *coding* tag – tag HTML atau dengan kata lain HTML merupakan tubuh dari sebuah website. Untuk memperindah sebuah website yang dibuat, HTML selalu dibantu dengan CSS yang merupakan *Cascading Style Sheet* yang biasanya berguna untuk mempercantik desain website. (Firmansyah & Herman, 2023)

Menurut Abdulloh dalam (Prasetyo, 2021) “HTML merupakan singkatan dari *Hypertext Markup Language*, yaitu bahasa standar web yang dikelola penggunaannya oleh W3C (*World Wide Web Consortium*) berupa tag-tag yang menyusun setiap elemen dan website. HTML berperan sebagai penyusun struktur halaman website yang menempatkan setiap elemen website sesuai *layout* yang diinginkan”.

Sedangkan menurut Agung et al., (2022) HTML adalah bahasa *markup* yang digunakan untuk membuat sebuah halaman web. Isinya terdiri dari berbagai kode yang dapat menyusun struktur suatu website. HTML terdiri dari kombinasi teks dan simbol yang disimpan dalam sebuah file. Dalam membuat file HTML, terdapat standar atau format khusus yang harus diikuti. Format tersebut telah tertuang dalam



standar kode internasional atau ASCII (*American Standard Code for Information Interchange*).

Berdasarkan beberapa defnisi diatas dapat disimpulkan bahwa HTML adalah bahasa *markup* yang digunakan untuk membuat struktur dan konten halaman web yang terdiri dari tag-tag yang digunakan untuk menandai elemen-elemen halaman web.

HTML memiliki syntax penulisan dengan diawali simbol `<html>` dan diakhiri dengan `</html>`. Pada HTML setiap pembuka `<` tentunya diakhiri dengan penutup berupa `</>` contohnya pada tag pembuka berikut `<head>` maka diakhiri dengan `</head>` atau biasa disebut dengan istilah backslash. Berikut contoh format penulisan HTML :

```
<html>
<head>
    <title> Belajar Web </title>
</head>
    <body>
        //content
    </body>
</html>
```

(Tinambunan et al., 2023).

2.4.2 Pengertian *Cascading Style Sheet* (CSS)

CSS adalah singkatan dari *Cascading Style Sheet* yaitu dokumen website yang berfungsi mengatur elemen HTML dengan berbagai *property-property* yang tersedia sehingga dapat tampil dengan berbagai gaya yang diinginkan.

Cara kerja CSS dalam memodifikasi HTML dengan memilih elemen HTML yang akan diatur kemudian memberikan *property* yang sesuai dengan tampilan yang diinginkan. Dalam memberikan aturan pada elemen HTML, *script* CSS terdiri dari atas tiga bagian yaitu *selector* untuk memilih elemen yang akan diberi aturan, *property* yang merupakan aturan yang diberikan dan *value* sebagai nilai dari aturan yang diberikan.



Cascading Style Sheet (CSS) merupakan suatu teknologi yang digunakan untuk memperindah atau mempercantik halaman website atau konsep sederhana yang berfungsi untuk membuat *style* atau gaya yang lebih diprioritaskan kepada bagaimana data yang ada pada html di tampilkan pada browser (Jantri et al., 2020).

Dari beberapa pengertian diatas dapat disimpulkan bahwa *Cascading Style Sheet* (CSS) adalah merupakan bahasa pemrograman yang digunakan untuk menentukan bagaimana dokumen dan website akan disajikan.

2.4.3 Pengertian *Hypertext Preprocessor* (PHP)

PHP atau singkatan dari *Hypertext Preprocessor* merupakan salah satu bahasa pemrograman *open source* yang digunakan untuk komunikasi ke sisi server yang kemudian transfer hasilnya ke *client* yang melakukan permintaan. (Firmansyah & Herman, 2023)

Menurut Abdulloh dalam (Budi Prasetyo, 2021) “PHP merupakan kependekan dari PHI (*Hypertext Preprocessor*) yaitu bahasa pemrograman web yang dapat disisipkan dalam skrip HTML dan bekerja di sisi server. Tujuan dari bahasa ini adalah membantu para pengembang web untuk membuat web dinamis dengan cepat”.

Sedangkan menurut Anggow (2023) PHP Adalah bahasa *Server-side-scripting* yang menghubungkan HTML untuk membuat halaman web yang dinamis. Dengan *Server-side-scripting* maka *sintaks* dan perintah-perintah PHP akan dieksekusi diserver kemudian hasilnya akan dikirim ke *browser* dengan format HTML.

Berdasarkan pengertian diatas disimpulkan bahwa PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman *server-side* yang digunakan untuk pengembangan aplikasi web.

PHP memiliki ciri khas blok penulisan dalam pemrogramannya sebagai berikut:

- `<? dan ?>`
- `<?php dan ?>`
- `<script language="php">` dan `</script>`



- `<% dan %>`
- Penulisan bahasa pemrograman pada penulisan biasanya diakhiri dengan titik-koma (;). Untuk mengakhiri dari satu perintah sebelum ke perintah lainnya.
- Memiliki ciri khas *CASE SENSITIVE* untuk nama *identifier* yang dibuat oleh *user* dengan simbol \$ dan pemanggilan ulang sebagai contoh Bahwa \$Nama tidak sama dengan \$NAMA dan \$nama karena memiliki *Case sensitive* sendiri.

Berikut contoh format penulisan php :

```
<html>
<head>
    <title> Contoh Sederhana </title>
</head>
<body>
    <?php
        echo("Hallo apakabar? Nama saya PHP script");
    ?>
</body>
</html>
```

2.4.4 Pengertian *My Structured Query Language* (MySQL)

Firmansyah & Herman (2023) mengemukakan bahwa MySQL merupakan sebuah *tool* yang digunakan untuk mengolah sebuah bahasa SQL yang dimana singkatannya merupakan *Structured Query Language*. Dengan kata lain MySQL merupakan sebuah *tool* yang *open source* yang digunakan untuk sistem implementasi dari sistem manajemen basis data relasional (RDBMS).

MySQL merupakan salah satu sistem *database* yang menjadi solusi di aplikasi *database*. Banyak pengguna yang mendukung serta sudah tersedia *mailinglist* dan *homepage* khusus yang berisi tutorial lengkap (Indrianto et al., 2019).



MySQL ini merupakan aplikasi untuk mengelola basis data yang banyak digunakan untuk membangun aplikasi yang menggunakan sebuah database (Rifqi Yusril Maulana, Aryuanto Soetedjo, n.d.).

Berdasarkan pengertian diatas disimpulkan bahwa MySQL merupakan MySQL adalah sistem manajemen basis data relasional (RDBMS) yang bersifat *open source*. MySQL digunakan untuk menyimpan dan mengelola data dalam bentuk table.

2.4.5 Pengertian XAMPP

Menurut Firmansyah & Herman (2023) “XAMPP ialah sebuah web server yang mudah dalam pengoperasiannya dan dapat melayani tampilan halaman web yang dinamis dan dapat diakses secara lokal dengan web server *localhost*”.

XAMPP merupakan website server *open source* yang berjalan pada sistem operasi cross- platform (Windows, Linux, MacOS). Semua yang dibutuhkan untuk mengelola web ada di XAMPP seperti Apache, MySQL/ MariaDB, PHP, serta Perl. Walaupun program di dalamnya lengkap, XAMPP merupakan website server yang simpel serta ringan. XAMPP dipakai untuk membuat website server lokal di pc. Hal ini akan mempermudah kamu dalam meningkatkan, mendesain, serta keperluan testing web (Abadi et al., 2021).

Menurut Fatimah, Samsudin (2019:36) mengemukakan bahwa “XAMPP adalah perangkat lunak bebas, yang mendukung banyak system operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (*localhost*), yang terdiri atas program Apache HTTP Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan *Perl*”.

Dari pengertian diatas dapat disimpulkan bahwa XAMPP merupakan aplikasi yang berfungsi sebagai server yang berdiri (*localhost*).



2.4.6 Pengertian Visual Studio Code

Menurut (Arumsari, 2019) *Microsoft Visual Studio Code* adalah *one-stop shop* yang memungkinkan kita fokus pada proses pengembangan dan melupakan *tools* baru.

Visual Studio Code adalah aplikasi yang dibuat Microsoft dengan tujuan utama bagi desainer web untuk pengembangan aplikasi web dengan ASP.NET dan Node.js. Selain itu, kode studio visual juga mendukung inovasi komparatif seperti HTML, CSS, Less, Backtalk, dan JSON. Dialek pemrograman yang membantu aplikasi kode visual studio adalah Cluster, C++, PHP, SQL, Ruby, Razor, Visual Essential, Java, XML, dan lain-lain. Dalam aplikasi ini terdapat fitur Intellisense Highlight, yaitu sebuah elemen yang membantu kita dalam menyusun kode program melalui popup yang muncul secara konsekuen ketika kita sedang menulis dan menampilkan ide-ide tata bahasa yang dapat kita pilih (Rifqi Yusril Maulana, Aryuanto Soetedjo, n.d.).

Visual Studio Code adalah *editor* perangkat lunak yang sangat ringan namun kuat. Kode sumbernya berjalan dari *desktop*. Dukungan bawaan Untuk JavaScript, Scripts, Node.js, ada berbagai ekstensi. Tersedia dalam bahasa lain seperti C++, C#, Python dan PHP (Noviantoroa et al., 2022).

Berdasarkan pengertian diatas disimpulkan bahwa *Visual Studio Code* adalah salah satu editor teks yang populer digunakan para pengembang perangkat lunak untuk menulis dan mengedit kode.

2.4.7 Pengertian Framework

Menurut Setyawan dan Munari (2020) dalam (Prayogo et al., 2023) , “*Framework* adalah kerangka kerja. *Framework* merupakan *scrip class* dan *function* yang dapat memudahkan *programmer* dalam menangani suatu permasalahan dalam pembuatan aplikasi seperti pemangilan *variabel*, koneksi ke *database* dan lain-lain”.

Framework adalah kumpulan instruksi-instruksi yang dikumpulkan dalam *class* dan *function-function* dengan fungsi masing-masing untuk memudahkan



developer dalam memanggilnya tanpa harus menuliskan *syntax* program yang sama berulang-ulang serta dapat menghemat waktu (Sallaby, 202:50).

framework adalah kerangka yang menggunakan struktur MVC (Model, View, Controller) yang merupakan media dalam pengelolaan suatu data sehingga menghasilkan sesuatu yang diinginkan (Firmansyah & Herman, 2023).

Berdasarkan pengertian diatas dapat disimpulkan bahwa *framework* merupakan kerangka kerja untuk mengembangkan aplikasi berbasis website maupun *desktop*.

2.4.8 Pengertian Bootstrap

Bootstrap adalah salah satu kerangka kerja dari CSS yang digunakan untuk membantu *developer* dalam mempermudah, mempercepat dan memperindah tampilan website tersebut. Bootstrap ini sangat berguna dalam sisi membantu *front – end* atau disebut juga *client – side* (Firmansyah & Herman, 2023).

Sedangkan menurut Abdulloh dalam (Budi Prasetyo, 2021) mengatakan bahwa “Bootstrap merupakan salah satu *framework* CSS paling populer dari sekian banyak *framework* CSS yang ada. Bootstrap memungkinkan desain sebuah web menjadi *responsive* sehingga dapat dilihat dari berbagai macam ukuran *device* dengan tampilan tetap menarik. Bootstrap juga membuat proses pengaturan desain menjadi lebih cepat karena tidak perlu lagi menulis CSS, bahkan hamper tidak perlu kecuali jika memerlukan pengaturan desain yang berbeda dengan *style* Bootstrap. Bootstrap telah didukung oleh hampir semua *browser* baik pada *desktop* maupun *mobile*”.

Bootstrap adalah sebuah *platform* CSS (*Cascading Style Sheet*) yang digunakan untuk merancangan website. Bootstrap merupakan *tool* yang sangat baik untuk digunakan *programmer* saat membuat tampil sebuah website (Noviantoroa et al., 2022).

Berdasarkan pengertian diatas maka disimpulkan bahwa *Bootstrap* merupakan *framework* untuk mendesain sebuah website agar tampilannya lebih menarik.



2.4.9 Pengertian Composer

Composer adalah suatu *tool* yang berfungsi sebagai *manajemen dependencies* PHP. Pada dasarnya Laravel memiliki banyak *library* (*packages*, *modules*, *plug-ins* dan *component*) salah satu yang umum digunakan seperti contoh *User Authentication*, *No Captha* dan masih banyak lainnya. Untuk kita memanggil *library* Laravel secara manual tentu akan sangat sulit dikarenakan tiap *library* diduga juga mempunyai ketergantungan (*Depedencies*) akan *library* lainnya, pada ujungnya akan menyebabkan kekacauan pada *project* yang dikembangkan. Composer berkerja untuk mengatur tiap ketergantungan/*dependencies* pada PHP, kita tinggal memanggil saja *library* yang kita inginkan tanpa perlu memikirkan *dependencies* akan *library* yang kita panggil (Anggow, 2023).

Composer adalah *tools dependency manager* pada PHP, *Dependency* (ketergantungan) sendiri diartikan ketika *project* PHP yang kamu kerjakan masih membutuhkan atau memerlukan *library* dari luar. Composer berfungsi sebagai penghubung antara *project* PHP kamu dengan *library* dari luar (Suharya & Rohman, 2022).

Composer adalah aplikasi *package manager* untuk bahasa pemrograman PHP yang menyediakan format standar untuk mengelola dependensi PHP dan pustaka-pustaka yang diperlukan (Gunanda, 2020).

Berdasarkan pengertian diatas maka disimpulkan bahwa Composer adalah alat *dependency manager* untuk bahasa pemrograman PHP. Dengan kata lain, Composer adalah aplikasi yang diinstal ke perangkat untuk memfasilitasi *developer* menggunakan *library open source* milik orang lain ke dalam *project* yang sedang dibangun.

2.4.10 Pengertian Laravel

Laravel merupakan *framework* aplikasi web berbasis PHP, *open source*, menggunakan konsep *Model-View-Controller* (MVC). Laravel adalah web *development framework* dengan sintaks yang ekspresif dan elegan. *Framework* ini di desain untuk membuat proses pengembangan menjadi lebih menyenangkan bagi pengembang tanpa mengorbankan fungsi aplikasi, meningkatkan kualitas aplikasi



dengan mengurangi beban biaya pengembangan, memudahkan proses pemeliharaan serta meningkatkan produktifitas pekerjaan dengan kode program yang rapi dan terstruktur (Agung et al., 2022).

Laravel merupakan *framework* yang bersifat *highly extensible* yang artinya para pengembang dapat memanfaatkan *ready-to-use first-party-packages* seperti *Envoy*, *Passport*, *Horizon*, *Cashier*, *Jetsream* dan *fortify*. Salah satu fitur yang memudahkan pengembangan adalah proses manajemen *database*, dengan *Eloquent ORM* penulisan tidak perlu menulis *query* secara rumit (Anggow, 2023).

Dari pengertian diatas maka dapat disimpulkan bahwa pengertian Laravel merupakan *framework* yang dapat membantu untuk memaksimalkan penggunaan PHP di dalam proses pengembangan website. *Framework* ini dibuat untuk pengembangan aplikasi website dimana mengikuti arsitektur MVC (*model view controller*).

Laravel merupakan sebuah framework pengembangan aplikasi dengan menggunakan bahasa pemrograman PHP yang paling banyak digunakan oleh programmer dunia. Kelebihan menggunakan framework ini adalah sintaks pada Laravel menggunakan bahasa yang mudah dimengerti, sehingga programmer pemula sekalipun akan mudah paham kegunaan suatu sintaks. Selain itu Laravel memiliki dokumentasi yang cukup lengkap, bahkan setiap versinya memiliki dokumentasi tersendiri mulai dari cara instalasi hingga penggunaan fitur-fiturnya (Gunanda, 2020).

2.4.11 Pengertian Black Box Testing

Menurut Dhaifullah dkk (2022) *Black Box Testing* adalah pengujian tanpa sepengetahuan kerja internal aplikasi yang sedang diuji (AUT). Juga dikenal sebagai pengujian fungsional atau pengujian *input driven*. Teknik pengujian perangkat lunak di mana cara kerja internal dari item yang diuji tidak diketahui oleh *tester*. Pengujian secara *black box* dapat dilakukan dengan 6 cara, yaitu *equivalence partitioning*, *boundary value analysis*, *cause effect graphing*, *fuzzy testing*, dan *model based testing*.



Sedangkan menurut Jantri et al., (2020) *Black box testing* adalah pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak. Jadi dianalogikan seperti kita melihat suatu kotak hitam, kita hanya bisa melihat penampilan luarnya saja, tanpa tau ada apa dibalik bungkus hitam nya. Pengujian perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program untuk mengetahui apakah fungsi, masukan dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.

Pengertian *Black Box Testing* adalah hanya menguji fungsionalitas dan *interface* (antarmuka) tanpa mengetahui proses yang detail dan hanya dapat mengetahui *input* dan *ouput* nya saja. (Ambarsari et al., 2021).

Dari pengertian diatas disimpulkan bahwa *Black box testing* adalah pengujian yang dilakukan untuk mengamati hasil input dan output dari perangkat lunak tanpa mengetahui struktur kode dari perangkat lunak untuk mengetahui apakah perangkat lunak dapat berfungsi dengan baik.