



BAB II TINJAUAN PUSTAKA

2.1 Teori Umum

2.4.1 Komputer

Menurut Kadir dan Triwahyuni (2013:2) “Komputer adalah mesin serbaguna yang dapat dikontrol oleh program, digunakan untuk mengolah data menjadi informasi”.

Kadir (2017:2), “Komputer merupakan peralatan elektronik yang bermanfaat untuk melaksanakan berbagai pekerjaan yang dilakukan manusia.”

Dari definisi di atas penulis dapat menyimpulkan bahwa komputer adalah alat digital serba guna yang bisa mengecek data yang bersifat gelombang.

2.4.2 Perangkat Lunak

Rosa dan Shalahuddin (2018:2), Perangkat lunak adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (user manual).

Kadir (2017:2), “Perangkat lunak adalah instruksi-instruksi yang ditujukan kepada komputer agar dapat melaksanakan tugas sesuai kehendak pemakai.”

Berdasarkan definisi di atas dapat disimpulkan bahwa perangkat lunak adalah data yang diformat, dan disimpan secara digital, dan berbagai informasi yang bisa dibaca, dan ditulis oleh komputer.



2.4.3 Pengertian Perangkat Keras

Sallaby, Utami, dan Arliando (2015:173), Perangkat keras (hardware) adalah perangkat yang terdiri dari alat yang dapat difungsikan untuk mendukung suatu operasi tertentu yang dipasang dalam sebuah sistem computer.

Kurnianingsih (2020:1), Perangkat keras merupakan perangkat yang dapat dilihat dan disentuh secara langsung oleh manusia, Perangkat keras komputer adalah semua bagian fisik computer.

Berdasarkan definisi di atas dapat di simpulkan bahwa perangkat keras adalah bagian fisik komputer, dan dibedakan dengan data yang berada di dalamnya atau yang beroperasi di dalam komputer

2.4.4 Metode Pengembangan Sistem

Sukanto dan Shalahuddin (2018:28) menjelaskan tentang metode pengembangan sistem yaitu *waterfall*. “Metode air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*)”. Model air terjun menyediakan pendekatan alur hidup terturut mulai dari analisis, desain, pengkodean, pengujian dan pemeliharaan.

a. Analisis kebutuhan perangkat lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk mespesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

b. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke



representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

c. Pembuatan Kode Program

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

d. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

2.2 Teori Judul

2.4.1 Aplikasi

Juansayha (2015:2) menjelaskan “Secara istilah pengertian aplikasi adalah suatu program yang siap untuk digunakan yang dibuat untuk melaksanakan suatu fungsi bagi pengguna jasa aplikasi serta penggunaan aplikasi lain yang dapat digunakan oleh suatu sasaran yang akan dituju”.

Sedangkan pada pengertian lain menurut Suhimarita dan Susianto dalam Asropuddin (2019:24) “ Aplikasi adalah sofwer[sic!] yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu, misalnya Ms, Word, Ms Axcel”.

Berdasarkan definisi diatas penulis menarik kesimpulan bahwa aplikasi merupakan perangkat lunak yang sangat mendukung manusia untuk mengerjakan tugas- tugasnya.



2.4.2 Try Out

Try out adalah wahana pembiasaan siswa terhadap penyelesaian soal-soal ujian. (Harmadya,dkk, 2015:577)

Try out merupakan salah satu prosedur evaluasi yang komprehensif, sistematis, dan objektif yang hasilnya dapat dijadikan sebagai dasar dalam pengambilan keputusan dalam proses pengajaran yang dilakukan oleh guru. (Nugraha, dkk, 2020: 138)

Berdasarkan definisi di atas maka penulis menyimpulkan bahwa *try out* adalah percobaan sebelum menghadapi ujian yang sebenarnya.

2.4.3 TOEFL

Kadarmo dan Kasmini (2015:1), “*TOEFL* adalah singkatan dari *Test Of English as a Foreign Language*. *TOEFL* merupakan sebuah *proficiency test*, yaitu tes untuk menguji kemampuan seseorang dalam penguasaan bahasa Inggris tanpa dikaitkan dengan proses belajar mengajar”.

TOEFL adalah suatu tes yang digunakan untuk mengukur kemampuan berbahasa Inggris bagi orang yang menggunakan bahasa Inggris sebagai bahasa keduanya (nonnative). (Nasser,R & Saldriani, 2019:185)

Berdasarkan definisi di atas maka penulis menyimpulkan bahwa *TOEFL* adalah uji coba untuk mengukur kemampuan Bahasa Inggris

2.4.4 Online

“Dalam bahasa Indonesia istilah *online* dipadankan menjadi dalam jaringan (daring), yaitu perangkat elektronik yang terhubung ke jaringan internet.” (badan pengembangan dan pembinaan bahasa, 2015).

Penulis menyimpulkan *Online* adalah jaringan perangkat elektronik terhubung ke jaringan internet sebagai alat memindai informasi dari orang sat uke orang lain.



2.4.5 Website

Abdullah (2018:1), “Website dapat diartikan sebagai kumpulan halaman yang berisi informasi data digital baik berupa teks, gambar, animasi, suara dan video atau gabungan dari semuanya yang disediakan melalui jalur koneksi internet sehingga dapat diakses dan dilihat oleh semua orang di seluruh dunia.”

Sidik (2019:3), “Website adalah jaringan sajian dan layanan informasi yang mendunia, yang sangat mudah diakses dan digunakan oleh pengguna Internet, karena pada halaman informasi yang ditampilkan setiap kata bisa memiliki tautan (link) atau sambungan informasi yang ada pada halaman lain, yang berasal dari suatu komputer lain yang terhubung dengan internet.”

Dari beberapa definisi di atas penulis menyimpulkan bahwa Website adalah suatu dokumen berupa kumpulan halaman web yang saling terhubung dan isinya terdiri dari berbagai informasi berbentuk teks, suara, gambar, video, dan lainnya, dimana semua data tersebut disimpan pada server hosting.

2.4.6 Aplikasi *Try Out TOEFL Online* pada Lembaga Bahasa dan Pendidikan Profesional (LBPP) LIA Palembang berbasis *Website*

Aplikasi *Try Out TOEFL Online* pada Lembaga Bahasa dan Pendidikan Profesional (LBPP) LIA Palembang berbasis *Website* adalah sebuah aplikasi berupa simulasi yang dapat menunjang pembelajaran sebelum menghadapi tes *TOEFL* yang sebenarnya.

2.3 Teori Khusus

2.4.1 Pengertian UML (*Unified Modeling Language*)

Sugiarti (2018:100) menjelaskan bahwa “Unified Modeling Language (UML) adalah sebuah bahasa yang telah menjadi standar dalam industri untuk visualisasi, merancang, dan mendokumentasikan sistem peranti lunak.”



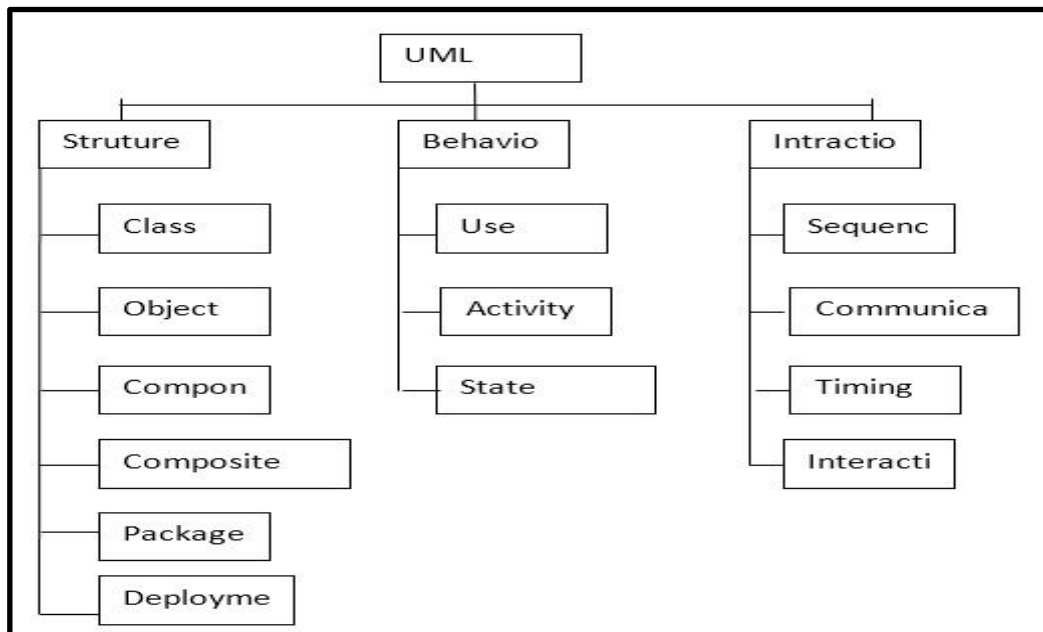
Munawar (2018:49) menjelaskan bahwa “UML (*Unified Modelling Language*) adalah salah satu alat bantu yang sangat handal di dunia pengembangan sistem yang berorientasi obyek.”

Inillah beberapa tujuan atau fungsi dari penggunaan UML:

1. Dapat memberikan bahasa permodelan visual kepada pengguna dari berbagai macam pemrograman maupun proses rekayasa.
2. Dapat menyatukan praktik-praktik terbaik yang ada dalam permodelan.
3. Dapat memberikan model yang siap untuk digunakan, merupakan bahasa pemrograman visual yang ekspresif untuk mengembangkan sistem, dan untuk saling tukar model secara mudah.

2.4.2 Macam – macam diagram *Unified Modeling Language*

Shalahuddin dan Rosa (2016:140), Pada UML terdapat 13 buah diagram yang selanjutnya dikelompokkan menjadi 3 kategori. Pembagian kategori dan macam – macam diagram tersebut dapat dilihat pada tabel dibawah ini :



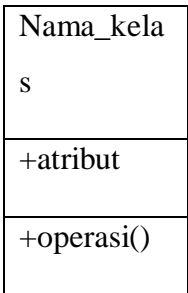



Tabel 2.1 Macam – Macam Diagram UML





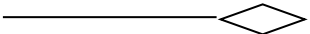
2.4.3 Pengertian *Class Diagram*

Menurut Sugiarti (2018:119) menjelaskan bahwa “Diagram kelas atau class diagram menggambarkan struktur sistem dari segi pendefinisian class-class yang akan dibuat untuk membangun sistem.”

Tabel 2.2 Simbol-simbol dalam *class diagram*

Simbol	Deskripsi
Kelas 	Kelas pada struktur system
Antarmuka/ <i>interface</i>  Nama_interface	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
Asosiasi/ <i>association</i> 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan multiplicity
Asosiasi berarah/ <i>directed association</i> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan multiplicity

**Tabel 2.3** Lanjutan Tabel Simbol-simbol dalam *class diagram*

Generalisasi 	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus)
Keberganungan/ <i>dependency</i> 	Relasi antarkelas dengan makna kebergantungan antarkelas
Agregasi/ <i>aggregation</i> 	Relasi antarkelass dengan makna semua-bagian (whole-part)

(Sumber: Shalahudin rosa, 2016:146-147)

2.4.4 Pengertian *Use Case Diagram*

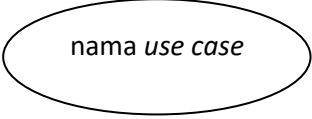
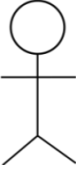

Munawar (2018:89) menjelaskan bahwa “Use case adalah deskripsi fungsi dari sebuah system dari perspektif pengguna. Use case bekerja dengan cara mendeskripsikan tipikal interaksi antara user (pengguna) sebuah system dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah system dipakai.”

Syarat penamaan pada use case adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada use case yaitu pendefinisian apa yang disebut aktor dan use case.

- a. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
- b. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.


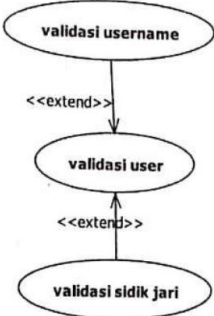


Tabel 2.4 Simbol-simbol dalam *Use case*

Simbol	Deskripsi
<p><i>Use Case</i></p> 	<p>Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal di awal <i>frase</i> nama <i>use case</i>.</p>
<p>Aktor/<i>actor</i></p> 	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal <i>frase</i> nama aktor.</p>
<p>Asosiasi/<i>association</i></p> 	<p>Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.</p>


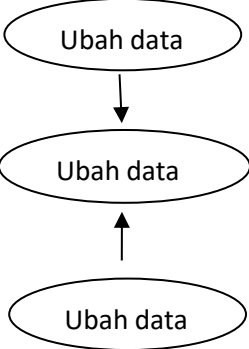
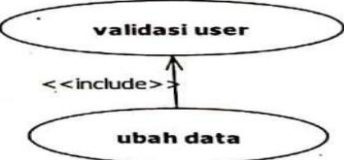


Tabel 2.5 Lanjutan Tabel Simbol-simbol dalam *Use case*

<p>Ekstensi/extend</p> <p style="text-align: center;"><<extend>></p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walaupun tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan, misal;</p>  <pre> graph TD A([validasi user]) -.-> <<extend>> B([validasi username]) C([validasi sidik jari]) -.-> <<extend>> A </pre> <p>Arah panah mengarah pada <i>use case</i> yang ditambahkan; biasanya <i>use case</i> yang menjadi extend-nya merupakan jenis yang sama dengan <i>use case</i> yang menjadi induknya.</p>
--	--



Tabel 2.6 Lanjutan Tabel Simbol-simbol dalam *Use case*

<p>Generalisasi/<i>generalization</i></p> 	<p>Hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya, misalnya:</p>  <p>Arah panah mengarah pada use case yang menjadi generalisasinya (umum</p>
	<p>ditambahkan selalu dijalankan sebelum <i>use case</i> tambahan dijalankan, missal pada kasus berikut:</p>  <p>Kedua interpretasi di atas dapat dianut salah satu keduanya tergantung pada pertimbangan dan interpretasi yang.</p>

(Sumber: Shalahudin rosa, 2016:156-158)






2.4.5 Pengertian Activity Diagram

Menurut Sugiarti (2018:134) menjelaskan bahwa “Activity Diagram menggambarkan berbagai aliran activity dalam sistem yang sedang dirancang, bagaimana masing-masing aliran berawal, decision yang mungkin terjadi, dan bagaimana mereka berakhir. Activity Diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.”

Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut:



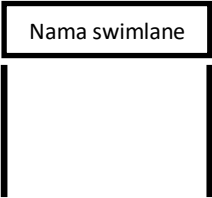
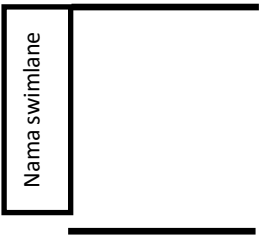
1. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan
2. Urutan atau pengelompokan tampilan dari sistem / user interface dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan
3. Rancangan pengujian dimana seriap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya
4. Rancangan menu yang ditampilkan pada perangkat lunak

Tabel 2.7 Simbol-simbol dalam *Activity Diagram*

Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
Percabangan / <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu



Tabel 2.8 Lanjutan Tabel Simbol-simbol dalam *Activity Diagram*

Penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
Simbol	Deskripsi
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
Swimlane  Atau 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi



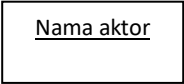


2.4.6 Pengertian Sequence Diagram

Menurut Sugiarti (2018:130) mengatakan bahwa “Diagram sekuens menggambarkan behavior objek pada Usecase dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antarobjek.”


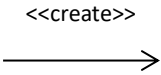
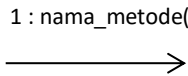
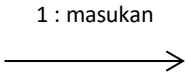
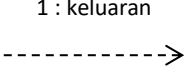
Banyaknya diagram sekuen yang harus digambar adalah sebanyak pendefinisian *Usecase* yang memiliki proses sendiri atau yang penting semua *Usecase* yang telah didefinisikan interaksinya pesan sudah dicakup pada diagram sekuens. Oleh karena itu, semakin banyak *Usecase* yang didefinisikan, maka diagram sekuens yang harus dibuat juga semakin banyak.

Tabel 2.5 Simbol Diagram *Sequence*

No	Simbol	Nama	Keterangan
1.		Garis hidup <i>/ lifeline</i>	Menyatakan kehidupan suatu objek
2.	 Atau 	Aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor



Tabel 2.9 Lanjutan Tabel Simbol Diagram *Sequence*

3.		Waktu aktif	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya
4.		Pesan tipe <i>create</i>	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat
5.		Pesan tipe <i>call</i>	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri, arah panah mengarah pada objek yang memiliki operasi/metode
6.		Pesan tipe <i>send</i>	Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim
7.		Pesan tipe keluaran	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian



2.4 Teori Program

2.4.1 MySQL

Rusmawan (2019:97), “MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (Database Management System) atau DBMS yang multithread, multi-user, dengan sekitar 6 juta instalasi diseluruh dunia.

Haqi dan Setiawan (2019:8) mengatakan bahwa “MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (database management system) atau DBMS yang multithread, multi-user, dengan sekitar 6 juta instalasi di seluruh dunia.”

Penulis menyimpulkan bahwa MySQL adalah sebuah program yang banyak digunakan oleh komputer.

2.4.2 Pengertian PHP (PHP Hypertext Preprocessor)

Menurut Anwar dan Irawan dalam Sibero (2017:114) “PHP (*Personal Home Page*) adalah pemrograman (*interpreter*) adalah proses penerjemahan baris sumber menjadi kode mesin yang dimengerti komputer secara langsung pada saat baris kode dijalankan”.

Hal serupa juga dijelaskan oleh Hakim (2019:114) “PHP (singkatan dari *Personal Home Page*) adalah bahasa pemrograman yang digunakan untuk membuat Aplikasi Web”.

Jadi berdasarkan uraian diatas, maka dapat disimpulkan bahwa PHP adalah perintah – perintah server yang disertai bahasa pemrograman scrip yang diletakkan didalam server *web*.

2.4.3 Pengertian XAMPP

Menurut Madcom dalam Ayu dan Permatasari, 2018:20 “Xampp adalah sebuah paket kumpulan software yang terdiri dari *Apache, MySQL, PhpMyAdmin, PHP, Perl, Filezilla*, dan lain.”



Sedangkan menurut Sunoro (2019:11) “Xampp adalah perangkat lunak yang bersifat *open source*, aplikasi *Apache* (web server) yang mudah diinstal dan berisi mariaDB, PHP, dan Perl”.

Berdasarkan definisi di atas maka penulis menyimpulkan bahwa XAMPP adalah sebuah paket yang didalamnya sudah tersedia database server MySQL dan support php programming.

2.4.4 Pengertian Basis Data (*Database*)

Menurut Kadir dan Triwahyuni (2013:325) “Basis Data adalah suatu pengorganisasian sekumpulan data yang saling terkait sehingga memudahkan aktivitas untuk memperoleh informasi”.

Sedangkan menurut Saputra (2019:14) “Basis Data merupakan kumpulan data-data yang saling berhubungan satu dengan yang lain yang disimpan dalam perangkat keras komputer dan akan diolah menggunakan perangkat lunak. Basis data sendiri merupakan kumpulan file-file yang mempunyai kaitan antara satu file dengan file yang lain sehingga membentuk satu bangunan data”.