



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Perangkat Lunak

Perangkat lunak atau disebut juga dengan *software* merupakan salah satu komponen dalam komputer. Seperti namanya yaitu perangkat lunak, sifatnya berbeda dengan *hardware* atau perangkat keras, jika perangkat keras adalah komponen yang nyata yang dapat dilihat dan disentuh oleh manusia, maka *software* atau perangkat lunak tidak dapat disentuh dan dilihat secara fisik. *Software* memang tidak tampak secara fisik dan tidak berwujud benda tapi kita bisa mengoperasikannya, melalui *software* atau perangkat lunak inilah suatu komputer dapat menjalankan suatu perintah.

Menurut Rianto (2021:5), mendefinisikan bahwa "*Software* (Perangkat Lunak) merupakan suatu data yang diprogram sedemikian rupa dan disimpan dalam bentuk digital yang tidak terlihat secara fisik tetapi tersimpan dalam media penyimpanan komputer."

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

Berdasarkan beberapa definisi perangkat lunak diatas, dapat disimpulkan bahwa perangkat lunak atau *software* adalah instruksi-instruksi atau data yang diformat secara digital, yang bisa dibaca dan ditulis oleh komputer sesuai kehendak pemakai.

2.1.2 Pengertian Komputer

Komputer semula dipergunakan sebagai alat yang dipakai untuk mengolah data serta untuk menggambarkan orang yang perkerjaannya melakukan



perhitungan aritmatika dengan atau tanpa alat bantu, tetapi arti kata ini kemudian dipindahkan kepada mesin itu sendiri.

Pengertian komputer menurut Rianto (2021:3), "Komputer merupakan suatu perangkat elektronika yang memiliki kemampuan untuk menerima dan mengolah data menjadi informasi, menjalankan program yang tersimpan dalam memori, serta dapat bekerja secara otomatis berdasarkan perangkat aturan tertentu."

Menurut Prawiro (dalam Harmayani, dkk, 2021:1), "Komputer adalah perangkat elektronik yang dapat digunakan untuk mengolah data sesuai dengan prosedur yang telah dirumuskan sebelumnya sehingga menghasilkan informasi bermanfaat bagi penggunaannya."

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa komputer adalah alat elektronik yang dapat mengolah data digital, dengan mengikuti serangkaian program yang digunakan untuk membantu pelaksanaan pekerjaan.

2.1.3 Pengertian *Internet*

Internet merupakan suatu perpustakaan besar yang didalamnya terdapat jutaan bahkan miliaran informasi yang berupa teks dalam bentuk media elektronik selain itu internet dikenal sebagai dunia maya, karena hampir seluruh aspek kehidupan di dunia nyata ada di *internet*. *Internet* juga banyak memberikan manfaat di berbagai bidang seperti bidang pendidikan, bidang kesehatan, bidang informasi, bidang bisnis, bidang sosial dan hiburan, dan lain sebagainya.

Pengertian *internet* menurut Ladiqi dan Wekke (2018:16-17), "*Internet* merupakan singkatan dari *International Networking* (jaringan internasional), secara teknis merupakan dua komputer atau lebih saling berhubungan membentuk jaringan komputer hingga meliputi jutaan komputer di dunia yang saling berinteraksi dan bertukar informasi."

Menurut Sobri, dkk (2017:4), "*Internet* merupakan sebuah dunia maya jaringan komputer yang berbentuk dari miliaran komputer di dunia."



Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa *Internet* adalah jaringan global yang menghubungkan seluruh jaringan di dunia agar saling berinteraksi dan melakukan pertukaran informasi.

2.1.4 Pengertian Data

Data dapat digambarkan sebagai fakta dan angka yang belum diolah, data yang belum diolah atau data mentah tidak dapat membantu dalam pengambilan keputusan, namun data adalah bahan mentah yang terorganisir, terstruktur dan diinterpretasikan untuk menciptakan sistem informasi yang berguna.

Menurut Pamungkas (2017:1), "Data merupakan nilai yang merepresentasikan deskripsi dari suatu objek atau kejadian."

Menurut Pane, dkk, (2020:7), "Data adalah kumpulan keterangan-keterangan atau deskripsi dasar dari suatu hal (objek atau kejadian) yang diperoleh dari hasil pengamatan (observasi), dan dapat diolah menjadi bentuk yang lebih kompleks, seperti: informasi, *database*, atau solusi untuk masalah tertentu."

Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa data adalah kumpulan atas angka, karakter, dan fakta yang akan diklasifikasi, dimodifikasi, dan diolah oleh program agar dapat menjadi informasi yang dapat berguna bagi penggunaannya.

2.1.5 Pengertian Basis Data (*Database*)

Menurut Pamungkas (2017:2), "Basis data merupakan kumpulan data terhubung yang disimpan secara bersama-sama pada suatu media, yang diorganisasikan berdasarkan sebuah skema atau struktur tertentu, dan dengan *software* untuk melakukan manipulasi untuk kegunaan tertentu."

Menurut A.S dan Shalahudin (2018:28) "sistem basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi tersedia saat dibutuhkan. Pada intinya basis data adalah media untuk penyimpanan data agar dapat diakses dengan mudah dan cepat".




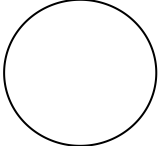
Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa basis data (*database*) adalah kumpulan data yang saling berhubungan, diorganisasi secara sistematis, menggunakan program komputer agar dapat dimanipulasi dan diperoleh dengan cepat dan mudah.

2.2 Teori Khusus

2.2. *Data Flow Diagram (DFD)*

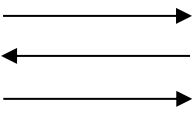
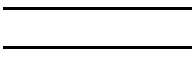
Menurut Kristanto (2018:61), “DFD adalah suatu model logika data atau proses yang dibuat untuk menggambarkan darimana asal data dan kemana tujuan data yang keluar dari *system*, dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut.”

Tabel 2.1 Simbol-Simbol Data *Flow Diagram*

No	Simbol	Nama Simbol	Keterangan
1.		Entitas Luar (<i>External Entity</i>)	Entitas Luar atau masukan atau keluaran atau orang yang memakai/berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang di modelkan.
2.		Proses	Proses pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur.



Lanjutan Tabel 2.1 Simbol-Simbol *Data Flow Diagram*


3.		Aliran Data	Aliran data merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukan (<i>input</i>) atau keluaran (<i>output</i>).
4.		<i>File</i> atau basis data	Pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya dibuat menjadi tabel-tabel basis data yang dibutuhkan, tabel-tabel ini juga harus sesuai dengan perancangan tabel-tabel pada basis data

Sumber : Kristanto (2018:64-65)

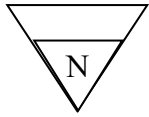
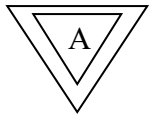
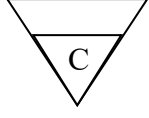

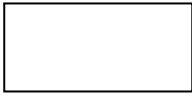
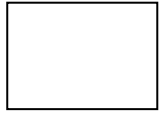

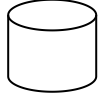
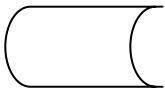

2.2.3 Flowchart

Menurut Rusmawan (2019:48), "Flowchart (bagan alir) merupakan sebuah gambaran dalam bentuk diagram alir dari algoritma-algoritma dalam suatu program yang menyatakan arah alur program tersebut."


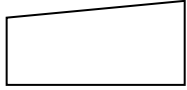
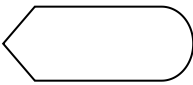



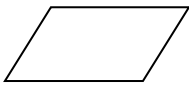
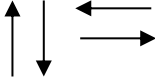
Tabel 2.2 Simbol-Simbol *Flowchart*

No	Nama Simbol	Simbol	Fungsi
1.	Simbol dokumen		Menunjukkan dokumen input untuk proses manual, mekanik atau komputer.

Lanjutan Tabel 2.2 Simbol-Simbol *Flowchart*

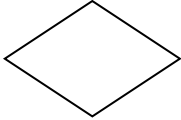


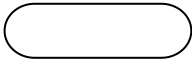
2.	Simbol simpanan offline		File non-komputer yang diarsip urut angka (<i>numerical</i>).
			File non-komputer yang diarsip urut huruf (<i>alphabetical</i>).
			File non-komputer yang diarsip urut tanggal (<i>cronological</i>).
3.	Simbol kartu plong		Menunjukkan <i>input/output</i> yang menggunakan kartu plong (<i>punched card</i>).
4.	Simbol proses		Menunjukkan kegiatan proses dari operasi program komputer.
5.	Simbol operasi luar		Menunjukkan operasi yang dilakukan di luar proses operasi komputer.
6.	Simbol pita magnetic		Menunjukkan <i>input/output</i> memakai pita magnetik.
7.	Simbol <i>hard disk</i>		Menunjukkan <i>input/output</i> menggunakan <i>hard disk</i> .
8.	Simbol <i>diskette</i>		Menunjukkan <i>input/output</i> menggunakan <i>diskette</i> .
9.	Simbol drum magnetic		Menunjukkan <i>input/output</i> menggunakan drum magnetik.

Lanjutan Tabel 2.2 Simbol-Simbol *Flowchart*

10.	Simbol pita kertas berlubang		Menunjukkan <i>input/output</i> menggunakan pita kertas berlubang.
11.	Simbol <i>keyboard</i>		Menunjukkan <i>input</i> menggunakan <i>on-line keyboard</i> .
12.	Simbol <i>display</i>		Menunjukkan <i>output</i> yang ditampilkan di <i>monitor</i> .
13.	Simbol pita <i>control</i>		Menunjukkan penggunaan pita kontrol (<i>control tape</i>) dalam <i>batch control</i> total untuk pencocokan di proses <i>batch processing</i> .
14.	Simbol hubungan komunikasi		Menunjukkan proses transmisi data melalui <i>channel</i> komunikasi.
15.	Simbol penghubung		Menunjukkan penghubung ke halaman yang masih sama atau ke halaman lain.
16.	Simbol <i>input/output</i>		Simbol <i>input/output</i> (<i>input/output symbol</i>) digunakan untuk mewakili data <i>input/output</i> .
17.	Simbol garis alir		Simbol garis alir (<i>flow lines symbol</i>) digunakan untuk menunjukkan arus proses.



Lanjutan Tabel 2.2 Simbol-Simbol *Flowchart*

18.	Simbol keputusan		Simbol keputusan (<i>decision symbol</i>) digunakan untuk suatu penyeleksian kondisi didalam program.
19.	Simbol proses terdefinisi		Simbol proses terdefinisi (<i>predifined process symbol</i>) digunakan untuk menunjukkan suatu operasi yang rinciannya ditunjukkan di tempat lain.
20.	Simbol persiapan		Simbol persiapan (<i>preparation symbol</i>) digunakan untuk memberi nilai awal suatu besaran.
21.	Simbol titik terminal		Simbol titik terminal (<i>terminal point symbol</i>) digunakan untuk awal dan akhir dari suatu proses.

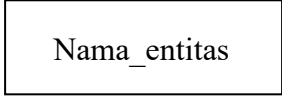
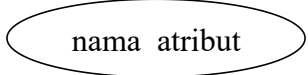
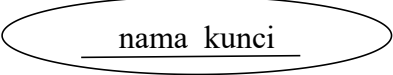
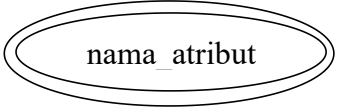
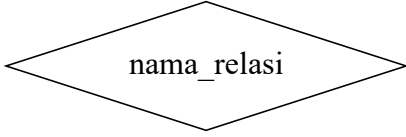
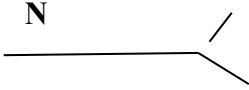
Sumber : Rusmawan (2019:49)

2.2.4 Entity Relationship Diagram (ERD)

Rosa, Salahuddin (2018:50), “ERD adalah pemodelan awal basis data yang dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional.”

Rosa, Salahuddin (2018:50) menjelaskan simbol – simbol yang digunakan dalam ERD, yaitu :


Tabel 2.3 Simbol-Simbol *Entity Relationship Diagram (ERD)*

No.	Simbol	Deskripsi
1.	Entitas / <i>Entity</i> 	Entitas merupakan data inti yang akan disimpan
2.	Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas
3.	Atribut kunci primer 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan; biasanya berupa <i>id</i> ; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama)
4.	Atribut multivalui / <i>multivalue</i> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu
5.	Relasi 	Relasi yang menghubungkan antar entitas biasanya diawali dengan kata kerja
6.	Asosiasi / <i>association</i> 	Penghubung antara relasi dan entitas dimana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian.

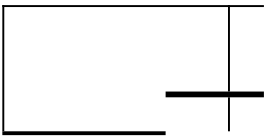

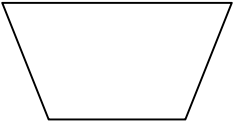

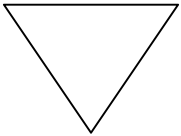
Sumber : Rosa, Salahuddin (2018:50)



2.2.5 Blockchart Diagram

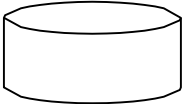

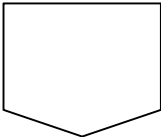
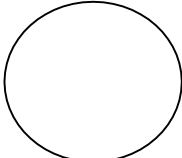

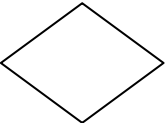

Menurut Kristanto dalam Noor dkk (2018:22), “Blockchart berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan block chart harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi.” Sedangkan simbol-simbol yang sering digunakan dalam blockchart dapat dilihat pada tabel berikut ini:

Tabel 2.4 Simbol-Simbol *Blockchart Diagram*

No.	Simbol	Keterangan
1.		Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bendel/berkas atau cetakan.
2.		Multi dokumen.
3.		Proses manual.
4.		Proses yang dilakukan oleh komputer.
5.		Menandakan dokumen yang diarsipkan (arsip manual).



Lanjutan Tabel 2.4 Simbol-Simbol *Blockchart Diagram*

6.		Data penyimpanan (data storage).
7.		Proses apa saja yang tidak terdefinisi termasuk aktifitas fisik.
8.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain.
9.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliranlain pada halaman yang sama.
10.		Terminasi yang menandakan awal dan akhir dari suatu aliran.
11.		Pengambilan keputusan(decision).
12.		Layar peraga(monitor).



2.3 Teori Judul

2.3.1 Pengertian Penerapan

Menurut Ali (2017:51), “Implementasi (penerapan) merupakan penyediaan sarana untuk melaksanakan sesuatu yang menimbulkan dampak atau akibat terhadap sesuatu”.

Berdasarkan beberapa definisi diatas maka dapat disimpulkan bahwa penerapan adalah mempraktekkan atau cara melaksanakan sesuatu berdasarkan sebuah teori.

2.3.2 Pengertian Metode *Customer Relationship Management (CRM)*

Menurut Kotler dan Keller (2018:189) “CRM merupakan proses mengelola informasi secara rinci tentang masing-masing pelanggan dan secara cermat mengelola semua kesempatan pelanggan menghadapi merek dan produk demi memaksimalkan kesetiaan pelanggan. CRM memungkinkan perusahaan memberikan layanan pelanggan yang unggul melalui penggunaan yang efektif atas informasi laporan individual”.

Dari pengertian tersebut maka dapat disimpulkan CRM merupakan suatu strategi bisnis dalam mengelola hubungan antara organisasi dengan pelanggan yang bertujuan untuk memahami, mengantisipasi dan mengelola kebutuhan pelanggan sehingga menghasilkan kepuasan pelanggan.

2.3.3 Pengertian *Framework of Dynamic CRM*

Menurut Alim, dkk (2021:11) “Salah satu fitur yang terdapat pada CRM adalah Framework of Dynamic CRM, framework ini menjelaskan serangkaian tahapan pada pembangunan dan penerapan CRM. Tahapan pada Framework of Dynamic CRM yaitu pertama akuisisi; ditujukan untuk memperluas customer/mendapatkan customer baru untuk lebih menambah pada sistem yang ingin dibuat, kedua tahap retensi; ditujukan guna mendukung peningkatan kinerja SDM agar terjadi perubahan tata cara pelayanan pada sistem yang ingin dibuat, dan tahap ke tiga extensi; ditujukan untuk menjaga loyalitas pelanggan”.



Berdasarkan pengertian diatas, pada dasarnya Framework of Dynamic CRM ialah suatu cara bagaimana customer dapat merasakan sebuah layanan yang diberikan oleh pelaku bisnis.

2.3.4 Pengertian Jadwal

Menurut Kamus Besar Bahasa Indonesia “Jadwal adalah pembagian waktu berdasarkan rencana pengaturan urutan kerja, daftar atau tabel kegiatan atau rencana kegiatan dengan pembagian waktu pelaksanaan yang terperinci.

Dapat disimpulkan bahwa jadwal adalah daftar atau tabel kegiatan atau rencana kegiatan dengan pelaksanaan yang terperinci.

2.3.5 Pengertian Maintenance

Menurut kamus besar bahasa Indonesia (KBBI) arti dari kata perawatan sendiri ialah proses, cara, perbuatan memelihara. Perawatan atau pemeliharaan (maintenance) adalah aktivitas penjadwalan secara berkala terhadap fasilitas/mesin untuk tetap menjaga performa dari agar tetap berfungsi dengan baik sesuai dengan kondisi awal mula mesin tersebut ada.

Dapat disimpulkan bahwa *maintenance* adalah kombinasi dari berbagai tindakan yang dilakukan untuk menjaga suatu aset dan memperbaikinya agar selalu dalam keadaan siap pakai untuk melaksanakan produktivitas secara efektif dan efisien sesuai dengan standar (fungsional dan kualitas).

2.3.6 Pengertian Pelanggan

Menurut Triwibowo (2019:19) “Konsumen atau pelanggan merupakan stake holder bisnis. Bisnis tidak akan berjalan kalau tidak ada konsumen yang menggunakan produk atau jasa yang ditawarkan.

Dapat disimpulkan bahwa pelanggan atau pemakai suatu produk adalah orang-orang yang berhubungan secara langsung maupun tidak langsung dengan perusahaanperusahaan bisnis.



2.3.7 Pengertian Website

Mnurut Marisa (2017:1), “*Website* dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk mempublikasikan informasi berupa teks, gambar, dan program multimedia lainnya berupa animasi (gambar gerak, tulisan gerak), suara, dan atau gabungan dari semuanya itu baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait antara satu halaman dengan halaman yang lain yang sering disebut sebagai *hyperlink*”.

Maka, dapat disimpulkan bahwa *Website* adalah halaman yang berisi informasi berupa teks, gambar, suara, dan video yang dapat disimpan pada komputer lokal maupun *server web*, yang terhubung melalui link *hypertext* dan dapat dijadikan atau ditampilkan pada browser.

2.4 Teori Program

2.4.1 Pengertian HTML



Sumber : wikipedia.org (2021)

Gambar 2.1 Logo HTML

Menurut Devi (2020:1), "*Hypertext Markup Language* (HTML) adalah sebuah bahasa markah yang digunakan untuk membuat sebuah halaman web, menampilkan berbagai informasi dalam sebuah penjelajah web internet dan memformat hiperteks sederhana yang ditulis dalam berkas format ASCII agar dapat menghasilkan tampilan wujud yang terintegrasi."

Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa HTML (*Hypertext Markup Language*) adalah skrip yang berupa tag-tag untuk membuat dan mengatur struktur serta menandai bagian-bagian dari sebuah halaman website untuk menampilkan konten di web.



2.4.2 Pengertian JavaScript



Sumber : pngwing.com (2021)

Gambar 2.2 Logo JavaScript

Menurut Sulistiono (2018:3), "Javascript adalah bahasa pemrograman tingkat tinggi untuk membuat website dinamis."

Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa JavaScript adalah bahasa pemrograman web yang pemrosesanya dilakukan di sisi client untuk memberikan efek dinamis, interaktif dan bersifat independen, yang dieksekusi di browser.

2.4.3 Pengertian CSS



Sumber : pnggg.com (2021)

Gambar 2.3 Logo CSS

Menurut Abdulloh (2018:45), “ CSS adalah singkatan dari Cascading Style Sheet yaitu dokumen web yang berfungsi mengatur elemen HTML dengan berbagai property yang tersedia sehingga dapat tampil dengan berbagai gaya yang diinginkan.”

Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa CSS adalah bahasa pengkodean atau dokumen yang berfungsi mengatur elemen HTML dengan berbagai *property* untuk menata gaya tampilan halaman web agar lebih cantik dan indah saat di tampilkan di web browser.



2.4.4 Pengertian PHP



Sumber : php.net (2021)

Gambar 2.4 Logo PHP

Menurut Supono dan Putratama (2018:3), "PHP (PHP: *Hypertext Preprocessor*) adalah suatu bahasa pemrograman yang digunakan untuk menerjemahkan baris kode program menjadi kode mesin yang dapat dimengerti oleh komputer yang bersifat *server-side* yang dapat ditambahkan ke dalam HTML."

Berdasarkan definisi di atas, maka dapat disimpulkan bahwa PHP adalah bahasa pemrograman web berbasis *server-side programming* (bahasa pemrograman yang diproses disisi server) yang memarsing kode PHP dari kode web dengan ekstensi (.php), yang kemudian akan dikirim ke browser web.

2.4.5 Pengertian XAMPP



Sumber : mrenesia.web.id (2021)

Gambar 2.5 Logo XAMPP

Menurut Nurhidayah, dkk (2020:40), menyatakan bahwa "XAMPP adalah aplikasi yang bersifat *open source* yang bisa digunakan di banyak sistem operasi seperti Windows, Linux, dan Mac OS."

Berdasarkan definisi di atas, maka dapat disimpulkan bahwa XAMPP merupakan aplikasi pendukung bersifat *open source* untuk membaca file yang memiliki format PHP.



2.4.6. Pengertian MySQL



Sumber : stickpng.com (2021)

Gambar 2.6 Logo MySQL

Menurut Fathoroni, dkk (2020:55), "MySQL adalah *database management system* yang menggunakan bahasa *SQL* sebagai bahasa penghubung antara perangkat lunak aplikasi dengan *database server*."

Menurut Fitri (2020:7), "MySQL merupakan *database engine* atau *server database* yang mendukung bahasa *database SQL* sebagai bahasa interaktif dalam mengelola data."

Berdasarkan definisi di atas, dapat disimpulkan bahwa MySQL adalah *database management system* sebagai bahasa interaktif penghubung perangkat aplikasi dengan *database server* yang digunakan dalam melakukan pengelolaan data. Berikut adalah fungsi-fungsi dari MySQLi.

Tabel 2.5 Fungsi-Fungsi MySQLi

No.	Fungsi MySQLi	Keterangan
1.	<code>mysqli_affected_rows()</code>	Mengembalikan jumlah baris yang berpengaruh dalam operasi MySQL sebelumnya.
2.	<code>mysqli_autocommit()</code>	Menghidupkan atau mematikan modifikasi basis data yang dilakukan secara otomatis.
3.	<code>mysqli_change_user()</code>	Mengubah pengguna koneksi <i>database</i> yang ditentukan.



Lanjutan Tabel 2.5 Fungsi-Fungsi MySQLi

No.	Fungsi MySQLi	Keterangan
4.	<code>mysqli_character_set_name()</code>	Mengembalikan <i>set</i> karakter default untuk koneksi <i>database</i> .
5.	<code>mysqli_close()</code>	Menutup koneksi <i>database</i> yang sebelumnya dibuka.
6.	<code>mysqli_commit()</code>	Melakukan transaksi saat ini.
7.	<code>mysqli_connect_errno()</code>	Mengembalikan kode kesalahan dari kesalahan koneksi terakhir.
8.	<code>mysqli_connect_error()</code>	Mengembalikan deskripsi kesalahan dari kesalahan koneksi terakhir.
9.	<code>mysqli_connect()</code>	Membuka koneksi baru ke server MySQL.
10.	<code>mysqli_data_seek()</code>	Menyesuaikan pointer hasil ke baris arbitrer di set hasil.
11.	<code>mysqli_debug()</code>	Melakukan operasi <i>debugging</i> .
12.	<code>mysqli_dump_debug_info()</code>	Tuang info debug ke dalam log.
13.	<code>mysqli_free_result()</code>	Membebaskan memori yang terkait dengan hasil.
14.	<code>mysqli_errno()</code>	Mengembalikan kode kesalahan terakhir untuk panggilan fungsi terbaru.
15.	<code>mysqli_error_list()</code>	Mengembalikan daftar kesalahan untuk panggilan fungsi terbaru.
16.	<code>mysqli_error()</code>	Mengembalikan deskripsi kesalahan terakhir untuk panggilan fungsi terbaru.
17.	<code>mysqli_fetch_all()</code>	Mengambil semua baris hasil sebagai <i>array asosiatif</i> , <i>array numerik</i> , atau keduanya.



Lanjutan Tabel 2.5 Fungsi-Fungsi MySQLi

No.	Fungsi MySQLi	Keterangan
18.	<code>mysqli_fetch_array()</code>	Mengambil baris hasil sebagai <i>asosiatif</i> , <i>array numerik</i> , atau keduanya.
19.	<code>mysqli_fetch_assoc()</code>	Mengambil baris hasil sebagai <i>array asosiatif</i> .
20.	<code>mysqli_fetch_field_direct()</code>	Mengembalikan meta-data untuk bidang tunggal di <i>set</i> hasil, sebagai objek.
21.	<code>mysqli_fetch_field()</code>	Mengembalikan bidang berikutnya dalam <i>set</i> hasil, sebagai objek.
22.	<code>mysqli_fetch_fields()</code>	Mengembalikan <i>array</i> objek yang mewakili bidang dalam <i>set</i> hasil.
23.	<code>mysqli_init()</code>	Menginisialisasi MySQLi dan mengembalikan sumber daya untuk digunakan dengan <code>mysqli_real_connect()</code> .
24.	<code>mysqli_fetch_lengths()</code>	Mengembalikan panjang kolom dari baris saat ini di <i>set</i> hasil.
25.	<code>mysqli_fetch_object()</code>	Mengembalikan baris saat ini dari hasil yang ditetapkan, sebagai objek.
26.	<code>mysqli_field_count()</code>	Mengembalikan jumlah kolom untuk kueri terbaru.
27.	<code>mysqli_field_tell()</code>	Mengembalikan posisi cursor bidang.
28.	<code>mysqli_fetch_row()</code>	Mengambil satu baris dari hasil- <i>set</i> dan mengembalikannya sebagai <i>array</i> .
29.	<code>mysqli_insert_id()</code>	Mengembalikan id yang dibuat secara otomatis yang digunakan dalam permintaan terakhir.
30.	<code>mysqli_info()</code>	Mengembalikan informasi tentang kueri yang baru saja dieksekusi.



Lanjutan Tabel 2.5 Fungsi-Fungsi MySQLi

No.	Fungsi MySQLi	Keterangan
31.	<code>mysqli_insert_id()</code>	Mengembalikan id yang dibuat secara otomatis yang digunakan dalam permintaan terakhir.
32.	<code>mysqli_more_results()</code>	Cek apakah ada lebih banyak hasil dari kueri multi.
33.	<code>mysqli_multi_query()</code>	Melakukan satu / lebih <i>query</i> pada <i>database</i> .
34.	<code>mysqli_next_result()</code>	Mempersiapkan set hasil selanjutnya dari <code>mysqli_multi_query()</code> .
35.	<code>mysqli_get_charset()</code>	Mengembalikan objek <i>set</i> karakter.
36.	<code>mysqli_get_client_info()</code>	Mengembalikan versi pustaka MySQL.
37.	<code>mysqli_fetch_array()</code>	Mengambil baris hasil sebagai asosiatif, array numerik, atau keduanya.
38.	<code>mysqli_fetch_assoc()</code>	Mengambil baris hasil sebagai <i>array asosiatif</i> .
39.	<code>mysqli_fetch_field_direct()</code>	Mengembalikan meta-data untuk bidang tunggal di <i>set</i> hasil, sebagai objek.
40.	<code>mysqli_fetch_field()</code>	Mengembalikan bidang berikutnya dalam <i>set</i> hasil, sebagai objek.
41.	<code>mysqli_fetch_fields()</code>	Mengembalikan <i>array</i> objek yang mewakili bidang dalam <i>set</i> hasil.
42.	<code>mysqli_init()</code>	Menginisialisasi MySQLi dan mengembalikan sumber daya untuk digunakan dengan <code>mysqli_real_connect()</code> .
43.	<code>mysqli_fetch_lengths()</code>	Mengembalikan panjang kolom dari baris saat ini di <i>set</i> hasil.
44.	<code>mysqli_fetch_object()</code>	Mengembalikan baris saat ini dari hasil yang ditetapkan, sebagai objek.



Lanjutan Tabel 2.5 Fungsi-Fungsi MySQLi

No.	Fungsi MySQLi	Keterangan
45.	<code>mysqli_field_count()</code>	Mengembalikan jumlah kolom untuk kueri terbaru.
46.	<code>mysqli_field_tell()</code>	Mengembalikan posisi cursor bidang.
47.	<code>mysqli_fetch_row()</code>	Mengambil satu baris dari hasil-set dan mengembalikannya sebagai array.
48.	<code>mysqli_insert_id()</code>	Mengembalikan id yang dibuat secara otomatis yang digunakan dalam permintaan terakhir.
49.	<code>mysqli_info()</code>	Mengembalikan informasi tentang kueri yang baru saja dieksekusi.
50.	<code>mysqli_insert_id()</code>	Mengembalikan id yang dibuat secara otomatis yang digunakan dalam permintaan terakhir.
51.	<code>mysqli_more_results()</code>	Cek apakah ada lebih banyak hasil dari kueri multi.
52.	<code>mysqli_multi_query()</code>	Melakukan satu atau lebih query pada <i>database</i> .
53.	<code>mysqli_next_result()</code>	Mempersiapkan <i>set</i> hasil selanjutnya dari <code>mysqli_multi_query()</code> .
54.	<code>mysqli_get_charset()</code>	Mengembalikan objek <i>set</i> karakter.
55.	<code>mysqli_get_client_info()</code>	Mengembalikan versi pustaka MySQL.
56.	<code>mysqli_get_client_stats()</code>	Mengembalikan statistik tentang klien per-proses.
57.	<code>mysqli_get_client_version()</code>	Mengembalikan versi pustaka klien MySQL sebagai <i>integer</i> .
58.	<code>mysqli_get_connection_stats()</code>	Mengembalikan statistik tentang koneksi klien.



Lanjutan Tabel 2.5 Fungsi-Fungsi MySQLi

No.	Fungsi MySQLi	Keterangan
59.	<code>mysqli_get_host_info()</code>	Mengembalikan nama <i>host server</i> MySQL dan jenis koneksi.
60.	<code>mysqli_get_proto_info()</code>	Mengembalikan versi protokol MySQL.
61.	<code>mysqli_get_server_info()</code>	Mengembalikan versi server MySQL.
62.	<code>mysqli_get_server_version()</code>	Mengembalikan versi <i>server</i> MySQL sebagai bilangan bulat.
63.	<code>mysqli_kill()</code>	Meminta <i>server</i> untuk membunuh utas MySQL.
64.	<code>mysqli_num_fields()</code>	Mengembalikan jumlah bidang dalam <i>set</i> hasil.
65.	<code>mysqli_refresh()</code>	Me- <i>refresh</i> tabel atau <i>cache</i> , atau me- <i>reset</i> informasi server replikasi.
66.	<code>mysqli_rollback()</code>	Putar kembali transaksi saat ini untuk basis data.
67.	<code>mysqli_select_db()</code>	Mengubah <i>database</i> default untuk koneksi.
68.	<code>mysqli_set_charset()</code>	Menetapkan <i>set</i> karakter klien default.
69.	<code>mysqli_set_local_infile_default()</code>	<i>Unsets handler</i> yang ditentukan pengguna untuk memuat perintah <i>infile</i> lokal.
70.	<code>mysqli_set_local_infile_handler()</code>	Atur fungsi panggilan balik untuk perintah <i>LOAD DATA LOCAL INFILE</i> .
71.	<code>mysqli_sqlstate()</code>	Mengembalikan kode kesalahan SQLSTATE untuk operasi MySQL terakhir.
72.	<code>mysqli_num_rows()</code>	Mengembalikan jumlah baris dalam hasil yang ditetapkan.



Lanjutan Tabel 2.5 Fungsi-Fungsi MySQLi

No.	Fungsi MySQLi	Keterangan
73.	<code>mysqli_options()</code>	Menetapkan opsi koneksi tambahan dan memengaruhi perilaku koneksi.
74.	<code>mysqli_ping()</code>	Ping koneksi <i>server</i> , atau mencoba menyambung kembali jika koneksi sudah turun.
75.	<code>mysqli_prepare()</code>	Mempersiapkan pernyataan SQL untuk dieksekusi.
76.	<code>mysqli_query()</code>	Melakukan kueri terhadap basis data.
77.	<code>mysqli_real_connect()</code>	Membuka koneksi baru ke <i>server</i> MySQL.
78.	<code>mysqli_real_escape_string()</code>	Melepaskan karakter khusus dalam <i>string</i> untuk digunakan dalam pernyataan SQL.
79.	<code>mysqli_real_query()</code>	Menjalankan <i>query</i> SQL.
80.	<code>mysqli_reap_async_query()</code>	Mengembalikan hasil dari kueri <i>async</i> .
81.	<code>mysqli_thread_id()</code>	Mengembalikan ID utas untuk koneksi saat ini.
82.	<code>mysqli_ssl_set()</code>	Digunakan untuk membuat koneksi yang aman menggunakan SSL.
83.	<code>mysqli_stat()</code>	Mengembalikan status sistem saat ini.
84.	<code>mysqli_stmt_init()</code>	Menginisialisasi pernyataan dan mengembalikan objek untuk digunakan dengan <code>mysqli_stmt_prepare()</code> .
85.	<code>mysqli_store_result()</code>	Mentransfer hasil yang ditetapkan kueri terakhir.
86.	<code>mysqli_thread_safe()</code>	Mengembalikan apakah pustaka klien dikompilasi sebagai aman-utas.



Lanjutan Tabel 2.5 Fungsi-Fungsi MySQLi

No.	Fungsi MySQLi	Keterangan
87.	<code>mysqli_use_result()</code>	Pengambilan hasil yang disetel dari kueri terakhir yang dieksekusi <code>mysqli_real_query()</code> .
88.	<code>mysqli_warning_count()</code>	Mengembalikan jumlah peringatan dari kueri terakhir dalam koneksi.

Sumber : W3school (2019).

2.4.7 Pengertian *Sublime Text*



Sumber : en.m.wikipedia.org (2021)

Gambar 2.7 Logo *Sublime Text*

Menurut Harani dan Hasanah (2020:84), menyatakan bahwa "Aplikasi *Sublime Text* merupakan sebuah aplikasi editor digunakan pada kode dan teks yang dapat berfungsi di berbagai platform *operating system* (OS) dengan menggunakan teknologi *Phyton API*".

Sublime Text menjunjung berbagai jenis bahasa pemrograman dan dapat menyuguhkan fitur *syntax highlight* hampir pada semua bahasa pemrograman yang didukung maupun dikembangkan oleh komunitas seperti; C, C++, SQL, TCL, Textile, CSS, D, LaTeX, Lisp, Lua, PHP, Python, XML, Dylan, Erlang, Java, JavaScript, OCaml, Perl, C#, HTML, Markdown, MATLAB, Groovy, Ruby, R and Haskell.

Berikut karakteristik yang diunggulkan dari aplikasi *Sublime Text* :



1. *Go to Anything*

Go to anything digunakan untuk membuka file yang menarik dengan satu project file yang sedang kita kerjakan/lakukan pada sublime lalu dengan cara menekan CTRL+P maka kita akan dapat mencari file apa yang akan kita buka dengan menuliskan nama file nya, lalu kita dapat mencari baris kata dengan keyword # atau @ untuk melompat ke simbol setelah menekan CTRL+P.

2. *Multiple Selections*

Multiple selections digunakan untuk membuat perubahan *code* pada saat yang sama dalam beberapa baris yang berbeda. Dapat dicoba dengan cara menekan CTRL+L, atau dapat juga dengan mengganti kata yang sama dalam baris berbeda dengan memblok kata yang akan diganti kemudian tekan CTRL+D.

3. *Command Pallete*

Command pallete digunakan untuk fungsi yang satu ini jarang digunakan sebenarnya namun fungsi ini banyak sekali hal yang bisa dilakukan seperti menutup semua file, *convert file: lower case*, *remove tag*, dan masih banyak lagi. Dengan menekan CTRL+SHIFT+P.

4. *Distraction Free Mode*

Distraction Free Mode digunakan untuk merubah tampilan menjadi layar penuh, dibutuhkan ketika *user* akan fokus pada kerjaan yang dikerjakan maka dengan fungsi ini akan sangat membantu dengan menekan SHIFT+F11.

5. *Split Editing*

Split editing untuk mendapatkan hasil yang maksimal dari monitor layar lebar dengan dukungan *editing* perpecahan. Melakukan *editing* di sisi file dengan sisi, atau mengedit dua lokasi di satu file.

6. *Instant Project Switch*

Mengambil seluruh file yang dimasukkan kedalam project pada aplikasi ini terhubung dengan fitur *Go to anything* untuk menjelajahi semua file yang ada ataupun untuk beralih ke file dalam project lainnya dengan cepat.



7. *Plugin API*

Aplikasi ini memiliki plugin API berbasis *Python* iyang membuat aplikasi ini sangat tangguh.

8. *Customize Anything*

Aplikasi sublime ini memiliki dan memberikan *user/pengguna* fleksibilitas dalam hal pengaturan/pengarahan fungsional dalam aplikasi ini.

9. *Cross Platform*

Aplikasi ini bisa berjalan hampir pada semua *operating system modern* seperti *Windows, Linux based operating system*, dan *OS X*.

Sumber : Harani dan Hasanah (2020:84-87).