



## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Teori Umum

##### 2.1.1 Pengertian Komputer

Zaharah (2023:37) mengemukakan bahwa komputer adalah mesin yang dirancang khusus untuk memanipulasi informasi yang diberi kode, mesin elektronik yang otomatis melakukan pekerjaan dan perhitungan sederhana dan rumit.

Menurut Baginda, *et al.*, (2022:2) bahwa, komputer bisa diartikan sebagai sebuah alat elektronika yang memiliki kemampuan untuk melakukan pengolahan data baik berupa teks, gambar maupun suara untuk menghasilkan informasi yang kita kehendaki.

Kemudian Syahroni (2019:2) menyatakan bahwa, komputer adalah mesin elektronik yang dapat diprogram untuk memproses data dan menghasilkan informasi.

Dari pengertian diatas dapat disimpulkan bahwa, komputer adalah alat atau perangkat elektronik yang dapat digunakan untuk memproses data, mengolah informasi, dan menjalankan program. Mesin ini dapat diprogram untuk melakukan berbagai tugas dan pengolahan informasi yang kompleks.

##### 2.1.2 Pengertian Perangkat Lunak

Menurut Indah (2021:2), “perangkat lunak adalah sekumpulan data elektronik yang disimpan dan diatur oleh komputer”.

Adapun menurut Allans, *et al.*, (2023:93), “perangkat lunak merupakan kumpulan perintah atau bahasa pemrograman yang berfungsi untuk menerjemahkan perintah manusia ke perintah komputer”.

Sedangkan menurut Hery dan Wahyu (2020:5), “perangkat lunak adalah program yang digunakan untuk mengendalikan perangkat keras komputer dan memproses data”.

Jadi, dapat disimpulkan bahwa perangkat lunak adalah program atau aplikasi yang diinstal pada komputer atau perangkat digital lainnya, yang dapat



digunakan untuk melakukan berbagai tugas dan aktivitas, serta dapat dijalankan pada sistem komputer untuk melakukan tugas tertentu atau mengendalikan perangkat keras komputer dan memproses data.

### 2.1.3 Pengertian Website

*Website* merupakan sebuah media yang memiliki banyak halaman yang saling terhubung (*hyperlink*), dimana *website* memiliki fungsi dalam memberikan informasi berupa teks, gambar, video, suara, dan animasi atau penggabungan semuanya (Elgamar, 2020:3).

Yuni, *et al.*, (2022:795) mengemukakan bahwa *website* adalah kumpulan halaman dalam suatu domain yang memuat tentang berbagai informasi agar dapat dibaca dan dilihat oleh pengguna atau pemakai internet melalui sebuah mesin pencari atau *search engine*.

Sedangkan Yudi (2020:3) menyatakan bahwa *website* adalah suatu aplikasi yang dirancang untuk diakses melalui jaringan internet dengan menggunakan protokol HTTP atau HTTPS.

Kesimpulan yang didapat ialah, *website* adalah suatu halaman atau kumpulan halaman yang dibuat dengan menggunakan bahasa pemrograman tertentu dan diakses melalui internet, wadah atau media digital yang menampung informasi berupa teks, gambar, audio, video, dan animasi yang dapat diakses melalui jaringan internet, serta sebuah aplikasi yang dirancang untuk diakses melalui jaringan internet dengan menggunakan HTTP atau HTTPS.

## 2.2 Teori Judul

### 2.2.1 Pengertian Implementasi

Menurut Betty (2022:8), “implementasi merupakan tahap yang sangat menentukan dalam proses kebijakan karena tanpa implementasi yang efektif maka keputusan pembuat kebijakan tidak akan berhasil dilaksanakan”.

Fitri, *et al.*, (2022:1801) menyatakan bahwa implementasi berarti menyediakan sarana untuk melaksanakan suatu kebijakan dan dapat menimbulkan dampak/akibat terhadap suatu tertentu.



Sedangkan Siti (2020:2) mengemukakan bahwa, implementasi adalah kegiatan atau proses yang bertujuan untuk menerapkan teknologi informasi dalam suatu organisasi perusahaan.

Berdasarkan pengertian diatas dapat disimpulkan bahwa, implementasi adalah suatu proses mengubah kebijakan menjadi tindakan konkret dalam rangka mencapai tujuan tertentu, tahap pelaksanaan dari perencanaan suatu program atau kegiatan, serta kegiatan atau proses yang bertujuan untuk menerapkan teknologi informasi dalam suatu organisasi perusahaan.

### **2.2.2 Pengertian Metode**

Metode adalah suatu cara sistematis yang digunakan demi tercapainya sebuah tujuan (Pasaribu, 2023:1).

Metode adalah perencanaan secara menyeluruh untuk menyajikan materi pembelajaran bahasa secara teratur, tidak ada satu bagian yang bertentangan dan semuanya berdasarkan pada suatu pendekatan tertentu (Sudjana 2023:2).

Sedangkan menurut Ahmad (2020:12), “metode adalah suatu cara yang teratur, sistematis, dan berdasarkan prinsip-prinsip tertentu untuk mencapai tujuan tertentu.”

Dapat disimpulkan bahwa metode adalah serangkaian prosedur atau langkah-langkah yang dilakukan untuk mencapai tujuan tertentu dalam penelitian, suatu cara atau teknik yang digunakan untuk mengumpulkan data atau informasi, serta suatu cara yang teratur, sistematis, dan berdasarkan prinsip-prinsip tertentu untuk mencapai tujuan tertentu.

### **2.2.3 Pengertian *Multi Objective Optimization on The Basic of Ratio Analysis* (MOORA)**

Danang, *et al.*, (2022:68) mengemukakan bahwa MOORA adalah multiobjektif sistem mengoptimalkan dua atau lebih atribut yang saling bertentangan secara bersamaan.



Menurut Hadi (2022:2), “MOORA adalah metode pengambilan keputusan yang menggunakan pemeringkatan alternatif berdasarkan nilai rasio antara kriteria positif dan kriteria negatif,”

Adapun Bambang, *et al.*, (2020:25) mengemukakan bahwa MOORA adalah metode pengambilan keputusan multi-kriteria yang digunakan untuk memilih alternatif terbaik dari beberapa alternatif dengan kriteria yang berbeda dengan cara membandingkan alternatif satu dengan yang lainnya.

Dapat ditarik kesimpulan bahwa MOORA adalah metode pengambilan keputusan multi-kriteria yang digunakan untuk memilih alternatif terbaik dari beberapa alternatif dengan kriteria yang berbeda, menggunakan pemeringkatan alternatif berdasarkan nilai rasio antara kriteria positif dan kriteria negatif, serta membandingkan alternatif satu dengan yang lainnya.

Metode MOORA menggunakan perkalian untuk menghubungkan *rating* atribut, dimana *rating* atribut harus dipangkatkan dulu dengan bobot yang bersangkutan. Secara umum prosedur MOORA meliputi langkah-langkah berikut:

1. Menginput Nilai Kriteria. Langkah pertama pada metode MOORA yaitu dengan menentukan dan memasukkan nilai kriteria pada setiap alternatif yang kemudian akan diproses oleh sistem yang akan menghasilkan sebuah sistem keputusan.
2. Membuat Matriks Keputusan. Berikut adalah perubahan nilai kriteria menjadi sebuah matriks keputusan:

$$X = \begin{bmatrix} x_{11} & x_{1j} & x_{1n} \\ x_{j1} & x_{ij} & x_{jn} \\ x_{m1} & x_{mi} & x_{mn} \end{bmatrix} \dots\dots\dots (1)$$

Keterangan:

$X_{ij}$  = Respon alternatif  $j$  pada kriteria  $i$ .

$i$  = 1,2,3, ...,  $n$  adalah nomor urutan atribut atau kriteria.

$j$  = 1,2,3, ...,  $m$  adalah nomor urutan alternatif.

$X$  = Matriks Keputusan.



3. Menghitung Matriks Normalisasi.

$$X^*_{ij} = \frac{x_{ij}}{\sqrt{\sum_{j=1}^m x_{ij}^2}} \dots \dots \dots (2)$$

Keterangan:

$x_{ij}$  = Respon alternatif  $j$  pada kriteria  $i$ .

$i$  = 1,2,3, ...,  $n$  adalah nomor urutan atribut atau kriteria.

$j$  = 1,2,3, ...,  $m$  adalah nomor urutan alternatif.

$X^*_{ij}$  = Matriks normalisasi alternatif  $j$  pada kriteria  $i$ .

4. Menghitung Nilai Optimasi. Terdapat 2 kondisi yakni untuk menghitung nilai optimasi pada metode MOORA, antara lain sebagai berikut:

a) Jika atribut atau kriteria pada masing-masing alternatif tidak diberikan nilai bobot

$$Y_j = \sum_{i=n}^{i=g} X^*_{ij} - \sum_{i=g+1}^{i=n} X^*_{ij} \dots \dots \dots (3)$$

Keterangan:

$i$  = 1,2,3, ...,  $g$  adalah atribut atau kriteria dengan status *maximized*.

$j$  =  $g+1, g+2, g+3, \dots, n$  adalah atribut atau kriteria dengan status *minimized*.

b) Jika atribut atau kriteria pada masing-masing alternatif diberikan nilai bobot kepentingan

$$Y_i = \sum_{j=1}^g w_j X^*_{ij} - \sum_{j=g+1}^n w_j X^*_{ij} \dots \dots \dots (4)$$

Keterangan:

$i$  = 1,2,3, ...,  $g$  adalah atribut atau kriteria dengan status *maximized*.

$j$  =  $g+1, g+2, g+3, \dots, n$  adalah atribut atau kriteria dengan status *minimized*.

$w_j$  = bobot terhadap alternatif  $j$ .

$Y_i$  = nilai penilai yang sudah dinormalisasi dari alternatif  $j$  terhadap semua atribut  $i$ .



5. Melakukan Perangkingan. Langkah terakhir pada metode MOORA yaitu perangkingan, dengan melihat nilai akhir  $Y_i$  tertinggi maka alternatif tersebut merupakan alternatif tertinggi yang ada pada sistem.

#### 2.2.4 Pengertian Aplikasi

Aplikasi adalah suatu model program komputer yang dibuat dengan menggunakan teknik dan prinsip tertentu agar dapat memproses dan melakukan tugas-tugas khusus (Elmo, 2022:20).

Aplikasi adalah perangkat lunak yang dirancang untuk melakukan tugas tertentu pada suatu sistem komputer. Aplikasi dapat berupa perangkat lunak sistem atau perangkat lunak aplikasi (Kristanto, 2018:20).

Aplikasi adalah program komputer yang dirancang untuk melakukan tugas-tugas spesifik pada suatu perangkat keras atau sistem operasi. Aplikasi dapat berupa perangkat lunak umum atau khusus, dan dapat dikembangkan untuk berbagai keperluan, seperti bisnis, pendidikan, atau hiburan (Sukardi, 2020:33).

Jadi, kesimpulannya ialah aplikasi adalah program komputer atau perangkat lunak yang dirancang untuk melakukan tugas-tugas tertentu pada suatu sistem komputer. Aplikasi dapat berupa perangkat lunak umum atau khusus, dan dapat dikembangkan untuk berbagai keperluan, seperti bisnis, pendidikan, atau hiburan. Aplikasi juga dapat berjalan di berbagai *platform*, seperti komputer, tablet, atau ponsel pintar.

#### 2.2.5 Pengertian Pemilihan

Sukidjo dan Faridatul (2020:16), menyatakan bahwa pemilihan adalah proses memilih alternatif atau pilihan terbaik dari beberapa alternatif yang tersedia.

Sementara Sigit dan Ady (2021:108) menjelaskan bahwa, pemilihan adalah proses memilih antara dua atau lebih pilihan.

Dapat disimpulkan bahwa pemilihan adalah proses memilih alternatif atau pilihan terbaik dari beberapa alternatif yang tersedia.



### **2.2.6 Pengertian Atlet Potensial**

Seperti dijelaskan Puslatnas (2020) bahwa, “atlet potensial adalah atlet yang memiliki bakat, kemampuan teknik, dan fisik yang baik serta berpotensi untuk menjadi atlet berprestasi di masa depan”.

Subagyo (2018) menyatakan, “atlet potensial adalah atlet yang memiliki bakat dan kemampuan fisik yang lebih dibandingkan dengan atlet lainnya di usia yang sama. Atlet potensial dapat didefinisikan melalui beberapa faktor seperti tes fisik dan kemampuan teknis dalam cabang olahraga tertentu.”

Dari pengertian diatas, dapat disimpulkan bahwa atlet potensial adalah atlet yang memiliki bakat dan kemampuan fisik yang lebih dibandingkan dengan atlet lainnya di usia yang sama serta memiliki potensi untuk menjadi atlet berprestasi di masa depan.

### **2.2.7 Pengertian Cabang Olahraga**

Jusuf dan Amran (2020:9) menyatakan, “cabang olahraga merupakan variasi olahraga yang mempunyai kekhasan sendiri dalam bentuk teknik dan peraturannya masing-masing.”

Yoyok (2021:33) mengemukakan bahwa cabang olahraga adalah suatu bentuk variasi atau jenis olahraga yang memiliki peraturan dan cara bermain yang berbeda-beda.

Sementara Heru (2019:7) menjelaskan bahwa, “cabang olahraga adalah bentuk variasi olahraga yang memiliki peraturan dan teknik bermain yang berbeda-beda, serta diakui oleh organisasi olahraga internasional.”

Dari pengertian diatas, dapat disimpulkan bahwa cabang olahraga adalah variasi atau jenis olahraga yang memiliki peraturan, teknik bermain, dan kekhasan masing-masing, serta diakui oleh organisasi olahraga internasional.

### **2.2.8 Pengertian Sepak Bola**

Sepak bola adalah suatu permainan kelompok yang melibatkan banyak unsur seperti fisik, teknik, taktik, dan mental (Herwin, 2022:143).



Sepak bola adalah suatu cabang olahraga yang dimainkan oleh dua tim, masing-masing tim memiliki 11 orang pemain. Tujuan permainan ini adalah memasukkan bola ke dalam gawang lawan sebanyak-banyaknya dengan cara mempergunakan kaki, kepala, dan bagian tubuh lainnya kecuali tangan, kecuali bagi kiper (Djajadiredja, 2021:27).

Dapat disimpulkan bahwa, sepak bola adalah suatu permainan olahraga yang dimainkan oleh dua tim yang terdiri dari sebelas orang pemain dan bertujuan untuk memasukkan bola ke gawang lawan serta mencegah bola masuk ke gawang yang dijaga. Dalam permainan sepak bola, hanya kiper yang diizinkan untuk menggunakan tangannya guna menangkap bola. Permainan ini juga mempergunakan bagian tubuh lainnya seperti kaki dan kepala untuk memainkan bola.

### **2.2.9 Pengertian BPOK PT Pusri Palembang**

BPOK (Badan Pembinaan Olahraga dan Kesenian) PT Pusri Palembang adalah sebuah lembaga internal di lingkungan PT Pusri Palembang yang bertugas mengembangkan dan memajukan kegiatan olahraga dan bertanggung jawab dalam memberikan dukungan dan fasilitas bagi karyawan, keluarga karyawan, dan masyarakat umum dalam mengembangkan bakat dan minat di bidang olahraga dan kesenian. Kegiatan yang dilakukan BPOK PT Pusri Palembang meliputi penyelenggaraan kegiatan olahraga dan kesenian, pembinaan atlet dan seniman, serta penyediaan fasilitas dan perlengkapan olahraga dan kesenian.

### **2.2.10 Pengertian Judul Secara Keseluruhan**

Implementasi Metode *Multi Objective Optimization on The Basic of Ratio Analysis* (MOORA) pada Aplikasi Pemilihan Atlet Potensial Cabang Olahraga Sepak Bola di BPOK PT Pusri Palembang Berbasis *Website* adalah suatu penelitian yang membahas tentang penerapan metode MOORA dalam memilih atlet potensial cabang olahraga sepak bola di BPOK PT Pusri Palembang melalui aplikasi berbasis *website*. Metode MOORA digunakan sebagai alat bantu dalam melakukan evaluasi dan pemilihan atlet potensial dengan mempertimbangkan beberapa kriteria atau



faktor yang menjadi pertimbangan penting dalam penyeleksian atlet. Tujuan dari penelitian ini adalah untuk meningkatkan efektivitas dan efisiensi dalam proses pemilihan atlet potensial, sehingga dapat menghasilkan atlet-atlet yang berkualitas dan mampu bersaing di tingkat nasional maupun internasional.

## 2.3 Teori Khusus

### 2.3.1 Context Diagram (Diagram Konteks)

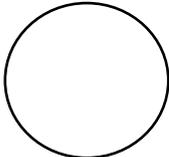
Menurut indyah (2020:51), “diagram konteks adalah diagram yang digunakan untuk menggambarkan sistem secara global”.

Menurut Harun (2018:82), “diagram konteks adalah sebuah diagram sederhana yang menggambarkan hubungan antara *entity* luar, masukan, dan keluaran dari sistem.

Sedangkan, Feri (2019:129) mengemukakan bahwa diagram konteks digunakan untuk mengetahui ruang lingkup dan batasan-batasan yang ada dalam perangkat lunak yang sedang dikerjakan.

Kesimpulannya yaitu diagram konteks adalah alat analisis dan perancangan sistem informasi yang penting dalam menggambarkan batas sistem dan hubungannya dengan entitas eksternal.

**Tabel 2.1** Simbol-Simbol *Diagram Context* (Diagram Konteks)

No.	Simbol	Keterangan
1.	<p><i>Process/Proses</i></p> 	<p>Menggambarkan suatu proses atau sistem yang akan dibangun.</p> <p>Proses dapat digambarkan dengan simbol lingkaran atau persegi panjang dengan sisi-sisi tumpul.</p>
		



**Lanjutan Tabel 2.1** Simbol-Simbol *Diagram Context* (Diagram Konteks)

2.	<i>Entity/Entitas/Terminator</i> 	Menggambarkan entitas atau pengguna dari sistem/aplikasi.
3.	<i>Data Flow/Aliran Data</i> 	Aliran data yang masuk dan keluar dari sistem.

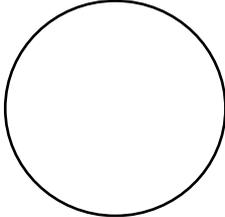
(Sumber: Feri, 2019:129)

### 2.3.2 *Data Flow Diagram (DFD)*

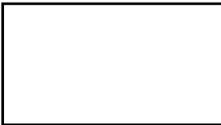
Penulis menggunakan DFD (*Data Flow Diagram*) untuk membangun program aplikasi karena DFD memungkinkan untuk menggambarkan secara grafis alur data dalam sistem. DFD membantu dalam memahami hubungan antara proses, data, dan entitas yang terlibat dalam sistem. DFD juga membantu dalam mengidentifikasi kebutuhan data yang diperlukan oleh sistem, memperjelas proses bisnis, dan mengidentifikasi area yang dapat dioptimalkan. DFD juga membantu dalam mengembangkan desain program dan mengevaluasi efisiensi dan efektivitas sistem.

DFD atau *Data Flow Diagram* adalah suatu diagram yang menggambarkan aliran data pada suatu sistem informasi. DFD digunakan untuk memodelkan sistem informasi agar dapat dipahami dengan mudah oleh pengguna, pengembang, dan pemilik sistem. DFD terdiri dari beberapa simbol yang merepresentasikan entitas, proses dan data yang saling terhubung dalam sistem (Sukanto dan Shalahuddin, 2021:34).

**Tabel 2.2** Simbol-Simbol *Data Flow Diagram (DFD)*

No.	Simbol	Keterangan
1.		Proses atau fungsi atau prosedur; Pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harus menjadi fungsi atau prosedur di dalam kode program.

Lanjutan Tabel 2.2 Simbol-Simbol *Data Flow Diagram* (DFD)

2.		<p>File atau basis data atau penyimpanan (<i>storage</i>);</p> <p>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 [<i>Entity Relationship Diagram</i> (ERD), <i>Conceptual Data Model</i> (CDM), <i>Physical Data Model</i> (PDM)].</p> <p>Catatan:</p> <p>Nama yang diberikan pada sebuah penyimpanan biasanya kata benda.</p>
3.		<p>Entitas luar (<i>External Entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang yang memakai/berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan.</p> <p>Catatan:</p> <p>Nama yang digunakan pada masukan (<i>input</i>) atau keluaran (<i>output</i>) biasanya berupa kata benda.</p>
4.		<p>Aliran data;</p> <p>Merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukan (<i>input</i>) atau keluaran (<i>output</i>).</p> <p>Catatan:</p> <p>Nama yang digunakan pada aliran data biasanya berupa kata benda, dapat diawali dengan kata data misalnya “data atlet” atau tanpa kata data misalnya “atlet”.</p>

(Sumber: Sukanto dan Shalahuddin, 2018:71-72)



Berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD secara singkat:

1. Identifikasi proses: Tahap pertama dalam perancangan DFD adalah mengidentifikasi proses yang akan dimodelkan. Proses ini dapat berupa aktivitas yang dilakukan dalam sebuah organisasi atau sistem informasi.
2. Identifikasi data: Tahap berikutnya adalah mengidentifikasi data yang digunakan dalam proses tersebut. Data yang diidentifikasi bisa berupa data masukan, data keluaran, dan data yang diproses dalam proses tersebut.
3. Penentuan level DFD: Setelah mengidentifikasi proses dan data, tahap selanjutnya adalah menentukan level DFD yang akan dibuat.
4. Pembuatan DFD level 0: Pada level 0, proses utama dari sistem akan diidentifikasi dan digambarkan dalam bentuk lingkaran. Data yang dihasilkan oleh proses tersebut akan dihubungkan dengan proses lain yang akan memproses data tersebut.
5. Pembuatan DFD level 1: Setelah DFD level 0 dibuat, tahap selanjutnya adalah membuat DFD level 1. Pada level ini, proses yang terdapat pada level 0 akan dibagi menjadi beberapa proses yang lebih rinci.
6. Pembuatan DFD level 2: Tahap selanjutnya adalah membuat DFD level 2. Pada level ini, proses yang terdapat pada level 1 akan dibagi menjadi beberapa proses yang lebih rinci lagi.
7. Pembuatan DFD level n: Tahap terakhir adalah membuat DFD level n, yaitu level terakhir dari DFD. Pada level ini, semua proses yang ada telah diuraikan dan dijelaskan secara rinci.

Dalam praktiknya, tahapan-tahapan ini dapat diulangi berkali-kali sampai didapatkan DFD yang optimal dan sesuai dengan kebutuhan pengguna. Sangat penting untuk melakukan komunikasi yang baik dengan pihak-pihak terkait, serta melakukan validasi dan verifikasi secara berkelanjutan guna memastikan kesesuaian antara model DFD dengan sistem atau proses bisnis yang sebenarnya.



### 2.3.3 *Blockchart*

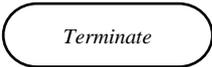
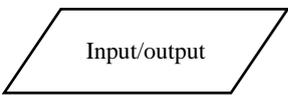
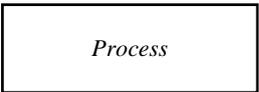
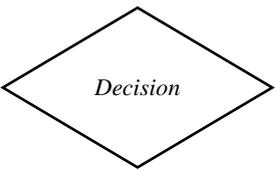
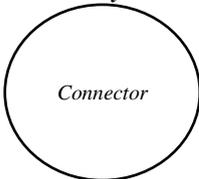
Penulis menggunakan *blockchart* dalam merancang dan mengembangkan program aplikasi karena *blockchart* memudahkan dalam memvisualisasikan struktur logis dari program aplikasi, yaitu dengan membagi program aplikasi menjadi modul-modul atau blok-blok yang saling terkait. Dengan menggunakan *blockchart*, penulis dapat memperjelas hubungan antara modul-modul atau blok-blok tersebut, sehingga memudahkan dalam merancang struktur program aplikasi yang lebih terstruktur dan efisien. Selain itu, *blockchart* juga memudahkan dalam mengkomunikasikan desain program aplikasi kepada tim pengembang lainnya dan *stakeholders*, serta memudahkan dalam pemeliharaan dan pengembangan program aplikasi di masa depan.

### 2.3.4 *Flowchart (Diagram Alir)*

Penulis menggunakan *flowchart* dalam merancang dan mengembangkan program aplikasi karena *flowchart* membantu dalam memvisualisasikan alur kerja dari suatu program aplikasi secara sistematis. Dengan *flowchart*, penulis dapat merencanakan dan mengidentifikasi langkah-langkah yang diperlukan dalam suatu proses secara jelas dan terstruktur. Hal ini memudahkan dalam mengembangkan program aplikasi dengan lebih efisien dan efektif, serta meminimalkan kesalahan atau kecacatan pada program yang dihasilkan. *Flowchart* juga memudahkan dalam melakukan *debugging* atau pemecahan masalah pada program aplikasi yang sudah ada.

Menurut Harwikarya (2021:47), “*flowchart* adalah sebuah diagram yang digunakan untuk mempresentasikan urutan langkah-langkah dalam suatu proses bisnis atau teknis. *Flowchart* terdiri dari simbol-simbol yang merepresentasikan berbagai jenis aktivitas atau keputusan yang terkait dengan suatu proses.”

**Tabel 2.3** Simbol-Simbol *Flowchart* (Diagram Alir)

No.	Simbol	Keterangan
1.	<p><i>Terminal point symbol</i></p> 	<p>Simbol terminal merupakan lambang untuk mengawali dan menutup suatu proses. Ketika akan membuat langkah logis suatu penyelesaian masalah menggunakan diagram alir maka terminal akan mengawali dan menutup langkah-langkah logis tersebut.</p>
2.	<p><i>Input/Output symbol</i></p> 	<p>Simbol <i>input/output</i> berfungsi untuk membaca <i>input</i> dan menampilkan <i>output</i>. Contoh <i>input</i> ketika membaca tinggi dan alas segitiga, <i>output</i> ketika menampilkan luas segitiga tersebut.</p>
3.	<p><i>Process symbol</i></p> 	<p>Simbol proses digunakan untuk menunjukkan pengeluaran yang dilakukan oleh komputer. Contoh pada perhitungan luas segitiga, maka proses akan menghitung luas segitiga, yaitu:</p> $\text{Luas} = \frac{1}{2} \text{ alas} \times \text{tinggi}.$
4.	<p><i>Decision symbol</i></p> 	<p>Simbol keputusan yang digunakan untuk suatu penyeleksian kondisi di dalam program. Contohnya ketika memeriksa apakah nilai mahasiswa &gt; 60, jika iya, maka lulus, jika tidak, maka gagal.</p>
5.	<p><i>Connector symbol</i></p> 	<p>Simbol penghubung digunakan untuk menunjukkan sambungan dari bagan air yang terputus di halaman yang masih sama atau di halaman yang berbeda.</p>

(Sumber: Harwikarya, et al., 2017:21)

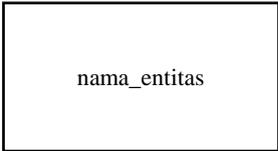
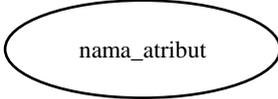
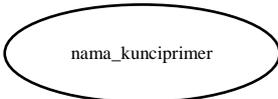


### 2.3.5 Entity Relationship Diagram (ERD)

Penulis menggunakan ERD (*Entity Relationship Diagram*) dalam merancang dan mengembangkan program aplikasi karena ERD dapat membantu dalam memvisualisasikan hubungan antara entitas dalam suatu sistem atau aplikasi. Dengan menggunakan ERD, penulis dapat dengan mudah mengidentifikasi dan merancang hubungan antara entitas seperti tabel pada *database*, sehingga memudahkan dalam mengembangkan program aplikasi yang terintegrasi dan efisien. Selain itu, ERD juga membantu dalam menghindari kesalahan dalam desain dan memastikan konsistensi data dalam sistem atau aplikasi yang dibangun.

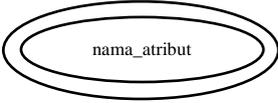
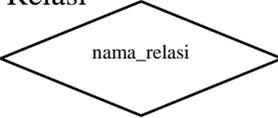
Seperti yang dikemukakan oleh Sukamto dan Shalahuddin (2018:34) bahwa, “*Entity Relationship Diagram* (ERD) merupakan suatu diagram yang digunakan untuk menggambarkan hubungan antara entitas dalam suatu sistem basis data. ERD merupakan model yang sangat penting dalam perancangan basis data karena dapat memudahkan dalam memvisualisasikan hubungan antara entitas, sehingga memungkinkan dalam merancang basis data yang efisien dan terstruktur.”

**Tabel 2.4** Simbol-Simbol *Entity Relationship Diagram* (ERD)

No.	Simbol	Keterangan
1.	Entitas/ <i>Entity</i> 	Entitas merupakan data inti yang akan tersimpan; Bakal tabel pada basis data; Benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer; Penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel.
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 id;



**Lanjutan Tabel 2.4** Simbol-Simbol *Entity Relationship Diagram* (ERD)

		Kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut bersifat unik (berbeda tanpa ada yang sama).
4.	<p>Atribut multinilai/<i>multivalued</i></p> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat dimiliki nilai lebih dari satu.
5.	<p>Relasi</p> 	Relasi yang menghubungkan antar entitas; Biasanya diawali dengan kata kerja.
6.	<p>Asosiasi/<i>Association</i></p> 	Penghubung antara relasi dan entitas dimana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut dengan <i>one to many</i> menghubungkan entitas A dan entitas B.

(Sumber: Sukamto dan Shalahuddin, 2018:50-51)

### 2.3.6 Data Dictionary (Kamus Data)

Menurut Tim Garuda Eduka (2021:92), “kamus data (*data dictionary*), merupakan pusat penyimpanan informasi data-data dari basis data yang memuat skema basis data”.

Sukamto dan Shalahuddin (2019:63) mendefinisikan, “kamus data sebagai dokumen atau *database* yang berisi deskripsi lengkap tentang data yang ada dalam



basis data, termasuk struktur, tipe data, batasan-batasan, relasi antar data, dan deskripsi lain yang relevan dengan data”.

Kesimpulannya yaitu kamus data adalah dokumen atau *database* yang berisi deksripsi lengkap tentang data dalam suatu basis data, seperti nama field, tipe data, dan keterangan lainnya. Fungsi utama kamus data adalah untuk menjaga konsistensi dan integritas data dalam suatu basis data, kamus data juga mencakup deskripsi lengkap tentang struktur, batasan-batasan, relasi antar data, dan deskripsi lain yang relevan dengan data.

Kamus data memiliki beberapa notasi untuk menjelaskan informasi tambahan berupa simbol berikut:

**Tabel 2.5** Simbol-Simbol Kamus Data (*Data Dictionary*)

No.	Simbol	Keterangan
1.	=	Disusun atau terdiri dari
2.	+	Dan
3.	[   ]	Baik... atau...
4.	{ } <sup>n</sup>	n kali diulang/bernilai banyak
5.	( )	Data opsional
6.	*...*	Batas komentar

(Sumber: Sukamto dan Shalahuddin, 2018:74)

## 2.4 Teori Program

### 2.4.1 Sekilas Tentang *Visual Studio Code*

#### 2.4.1.1 Pengertian *Visual Studio Code*

Fitriani dan Maulana (2022:36) berpendapat “*visual studio code* adalah sebuah *text editor* ringan dan handal yang dibuat oleh *Microsoft* untuk sistem operasi *multiplatform*, artinya tersedia juga untuk versi *Linux*, *Mac*, dan *Windows*.

Syafrizal (2020:1) menyatakan “*Visual Studio Code* adalah *text editor* yang sangat populer dan memiliki banyak fitur yang dapat membantu para *developer*. *VS Code* juga mudah digunakan dan dapat diinstal pada berbagai sistem operasi.



Sedangkan menurut Jauhari (2019:9), *visual studio code* adalah *text editor* yang sangat populer di kalangan *developer*. *VS Code* dapat digunakan untuk mengembangkan berbagai macam aplikasi, seperti *web*, *mobile*, dan *desktop*. Selain itu, *VS Code* juga dapat mengintegrasikan dengan berbagai bahasa pemrograman dan layanan *cloud*.

Dari beberapa pengertian di atas, dapat disimpulkan bahwa *Visual Studio Code* adalah *text editor* yang sangat populer di kalangan *developer* di Indonesia. *VS Code* memiliki berbagai fitur yang sangat baik, seperti kemampuan untuk mengintegrasikan dengan berbagai bahasa pemrograman dan layanan *cloud*, memiliki fitur *debugging* yang baik, serta tersedia banyak *extension* dan *plugin* yang dapat memudahkan pekerjaan para *developer*.

#### **2.4.1.2 Keunggulan Visual Studio Code**

*Visual Studio Code (VS Code)* adalah *text editor* yang dibuat oleh Microsoft untuk Windows, Linux, dan MacOS pada tahun 2015 dan telah menjadi salah satu *text editor* yang paling populer digunakan oleh para pengembang di seluruh dunia. Keunggulan *Visual Studio Code* yaitu:

1. *Open Source*: Artinya kode sumbernya terbuka untuk semua orang dan dapat diakses, diubah, dan didistribusikan secara bebas.
2. Sudah ter-*install plugin* EMMET: *Plugin* EMMET merupakan *plugin* yang sangat membantu bagi *web developer* karena dapat menghemat waktu dan mempercepat dalam pembuatan sebuah *web*.
3. Mudah untuk mengelola *Extensions*: Pengelolaan *extensions* yang sangat mudah sehingga mudah dipahami bagi pengguna baru memakai *VS Code*. Pengguna hanya perlu ke menu *extensions* atau CTRL+SHIFT+X dan pilih *extensions* mana yang mau di-*install* atau bisa dicari di kolom *search*.
4. *Extensions* yang banyak: Banyak *extensions* *VS Code* untuk memudahkan pekerjaan contohnya *auto rename tag* sangat berguna bagi *web developer* dan masih banyak lagi *extensions* yang dapat mempermudah pekerjaan.



5. Kostumisasi tampilan: Di *VS Code* juga bisa merubah tampilan seperti *icon*, *font*, dan warna sesuai keinginan pengguna atau pengguna bisa meng-*install* berbagai tema seperti *material theme* atau tema lainnya yang disukai.
6. Terintegrasi dengan GIT: Di *VS Code* sudah terintegrasi dengan *git* jadi pengguna lebih mudah dalam menyelesaikan *conflict* atau mengetahui mana baris yang berubah atau ditambahkan. Jadi akan lebih memudahkan dalam kerja tim.
7. Dukungan bahasa: *VS Code* mendukung banyak bahasa pemrograman populer seperti *JavaScript*, *Python*, *C++*, dan banyak lagi. Ini memungkinkan pengembang untuk bekerja dengan banyak jenis proyek yang berbeda.

#### 2.4.1.3 Kekurangan *Visual Studio Code*

Adapun kekurangan dalam menggunakan *Visual Studio Code* yaitu:

1. Performa: *VS Code* cukup memakan sumber daya, terutama jika pengguna banyak memakai ekstensi. Jika mesin komputer atau laptop pengguna memiliki memori terbatas, ini dapat menghambat performa editor.
2. Belum menghafal *shortcut key*: Bagi sebagian pengguna yang baru sebelumnya memakai *sublime text* dan ingin mencoba *vs code* mungkin akan sedikit kebingungan dengan *shortcut key* di *vs code* karena tidak seperti di *sublime text* untuk melakukan suatu tindakan menggunakan tombol 'CTRL', sedangkan di *vs code* pengguna menggunakan tombol 'ALT'.

#### 2.4.2 Sekilas Tentang *MySQL*

##### 2.4.2.1 Pengertian *MySQL*

Menurut Fitriani dan Maulana (2022:37), kata "*SQL*" dari "*MySQL*" adalah singkatan dari "*Structured Query Language*", adalah bahasa standar yang paling umum digunakan untuk mengakses *database*.

Menurut Hartanto (2020:22), *MySQL* adalah salah satu sistem manajemen basis data yang paling populer dan banyak digunakan di dunia.



Sedangkan menurut Suharyanto (2019:54), *MySQL* adalah sebuah sistem manajemen basis data relasional yang didistribusikan secara gratis dan terbuka untuk umum.

Berdasarkan pengertian dari para ahli di atas, dapat disimpulkan bahwa *MySQL* adalah sebuah sistem manajemen basis data relasional yang populer dan banyak digunakan di seluruh dunia. *MySQL* didistribusikan secara gratis dan terbuka untuk umum, sehingga banyak digunakan oleh para pengembang dan organisasi dalam mengelola basis data mereka.

#### 2.4.2.2 Keunggulan *MySQL*

Berikut ini merupakan beberapa kelebihan *MySQL*, diukur dari segi penggunaan dan fitur yang dimiliki, yaitu:

1. Bersifat *open source*: Untuk penggunaan dari *MySQL* dapat diakses oleh siapa saja dan kapan saja. Karena memiliki dua lisensi, yaitu *Free Software* dan *Shareware*. Selain itu juga termasuk dalam sistem manajemen *database* yang disediakan secara gratis. Serta, memiliki banyak komunitas dan dokumentasi untuk membantu pengguna dalam mengembangkan aplikasi menggunakan *database server*.
2. Mendukung penggunaan *multi user*: *MySQL* juga dapat digunakan secara bersama-sama dalam satu waktu karena bersifat *multi user*. Sehingga, pengguna tidak perlu khawatir ketika mengakses basis data secara bersamaan. Keuntungannya disini adalah, tim *developer* dapat mengerjakan *database* secara bersamaan (*team work*).
3. Terintegrasi dengan bahasa pemrograman yang lain: Kelebihan lain adalah dapat terintegrasi dengan bahasa pemrograman yang lain seperti pada *PHP*. Dengan menggunakan *PHPMyAdmin* sebagai perantara, pengguna sudah bisa membuat basis data pada *web server* dengan lebih efektif. Kemudian, dapat digunakan juga untuk kebutuhan pengembangan perangkat lunak dengan menggunakan *framework* seperti *Laravel* dan *Codeigniter*.
4. Sajian tipe data yang bervariasi: Contohnya adalah tipe data *varchar*, *integer*, *float*, *double*, *date*, *timestamp*, dll. Dalam hal ini, pengguna dapat



mengelompokkan berbagai data sesuai dengan kriteria informasi secara akurat dan efektif. Pengelolaan *database* dalam *server* juga akan terbantu dengan adanya banyak jenis tipe data untuk mengembangkan perangkat lunak yang kompleks dan terstruktur.

5. Struktur tabel cenderung fleksibel: Jika dibandingkan dengan struktur tabel dari *database* yang lain seperti *Oracle* dan *Postgre*, *MySQL* tergolong lebih mudah. Karena sebagai contoh dalam proses eksekusi pada 'ALTER TABLE' jadi lebih mudah digunakan dan cepat diproses.
6. Sistem keamanan yang telah terjamin: *MySQL* dapat mengatur hak akses pengguna (*user*) dengan enkripsi tinggi. Selain itu, terdapat beberapa fitur keamanan yang lain seperti *level*, nama pada *host*, dan pembuatan *subnet mask*. Jadi, bukan berarti dengan perangkat lunak yang bersifat *open source* membuat keamanan menjadi buruk.
7. Tidak membutuhkan kapasitas RAM yang besar: *MySQL* dapat diinstal pada perangkat dengan kapasitas RAM terbatas. Jika dibandingkan dengan *database* lain, dapat dijalankan dengan kapasitas memori yang kecil. Sangat cocok digunakan oleh setiap orang untuk membantu pengembangan aplikasi dengan konsumsi RAM yang rendah.
8. Perangkat lunak *portable*: *Portable* disini maksudnya adalah sistem basis data dapat diolah dalam berbagai *platform* (*multi platform*). Sehingga dapat dijalankan baik dari sisi *Windows*, *Linux*, maupun *Mac OS*.
9. *Support* untuk spesifikasi *hardware* yang rendah: Untuk pengguna yang memiliki spek PC/Laptop yang masih rendah. Dengan menggunakan DBMS ini, pengguna juga bisa membuat sistem basis data untuk *server* dalam sebuah *website*.
10. Memiliki performa yang baik: Kecepatan dalam penanganan kueri sangat cepat. Kemudian, juga dapat menangani proses *SQL* per satuan waktu.



### 2.4.2.3 Kekurangan MySQL

Selain memiliki banyak keunggulan, pastinya juga memiliki beberapa kekurangan. Berikut merupakan beberapa kekurangan dari sisi fitur dan penggunaan, yaitu:

1. Kurang cocok untuk mengembangkan aplikasi berbasis *game* dan *mobile*: Karena, manajemen *database* yang dimiliki kurang cocok untuk pengelolaan aplikasi tersebut. Lebih mudah untuk menggunakan *database server* dengan metode *cloud* yang lebih *real time*.
2. *Techniqal support* yang kurang mendukung: *Techniqal support* yang dimiliki *MySql* sendiri dirasa kurang mendukung. Karena, belum menyediakan *support* yang baik dari segi teknisnya. Sehingga, pengguna mengalami kesulitan saat menghubungi *techniqal support*.
3. Kurang mendukung untuk pemakaian *database* dengan kapasitas besar: Apabila pengguna menginginkan sebuah *database server* yang dapat menampung data dalam jumlah besar, maka pengguna dapat mencoba menggunakan *Big Data* atau *BigQuery*. Karena, *MySql* dikhususkan supaya ramah dengan perangkat yang memiliki spesifikasi rendah.

## 2.4.3 Sekilas Tentang XAMPP

### 2.4.3.1 Pengertian XAMPP

Menurut Fitriani dan Maulana (2022:37), *Xampp* adalah sebuah *software* yang berfungsi untuk menjalankan *website* berbasis *PHP* dan menggunakan pengolah data *MySQL* di komputer *local*.

Menurut Munir dan Darmawan (2021:136), *Xampp* adalah sebuah paket perangkat lunak yang mengintegrasikan *Apache*, *MySql*, *PHP*, dan *Perl* dalam satu instalasi sehingga memudahkan pengembangan dan pengujian aplikasi web lokal.

Sedangkan menurut Hermanto dan Satriawan (2018:11), *Xampp* adalah sebuah perangkat lunak yang dapat digunakan untuk membuat dan menjalankan server web lokal tanpa memerlukan konfigurasi yang rumit.

Dapat disimpulkan bahwa *Xampp* adalah sebuah paket perangkat lunak yang mengintegrasikan *Apache*, *MySql*, *PHP*, dan *Perl* dalam satu instalasi untuk



membuat dan menjalankan server web lokal dengan mudah dan cepat. *Xampp* sangat berguna bagi pengembang web dalam mengembangkan dan menguji aplikasi web di lingkungan lokal sebelum diunggah ke server produksi.

#### 2.4.3.2 Keunggulan XAMPP

*Xampp* memberikan banyak fitur dan kelebihan yang dapat dinikmati oleh para *web developer*. Berikut adalah beberapa keunggulan *Xampp*, yaitu:

1. Gratis diunduh dan digunakan: Salah satu alasan mengapa *Xampp* begitu banyak digemari oleh para *programmer* apalagi *web developer*, sebab sifatnya yang *free* atau gratis untuk diunduh dan digunakan. Pengguna dapat mengunduhnya secara gratis pada *website* resminya dan mendapatkan versi terbarunya secara rutin.
2. Kapasitas yang cukup: Berdasarkan versi terbaru kemampuannya yang sudah dapat diandalkan, memiliki sebuah kapasitas yang cukup mumpuni yakni sekitar 60.000 tabel dengan suatu jumlah *record* hingga mencapai sekitar 5.000.000.000, jumlah ini biasanya akan bertambah ketika *Xampp* mengeluarkan versi terbarunya.
3. Data yang aman: Keamanan datanya yang cukup aman walaupun tak sehebat dan sebagus keamanannya dari *Postgre* maupun *Oracle*.
4. Dapat digunakan untuk semua sistem operasi: *Engine* atau mesin ini bersifat *multiplatform* sehingga mampu untuk diaplikasikan di berbagai macam sistem operasi komputer mulai dari *Windows* hingga yang paling susah *Mac Os*.
5. Kecepatan proses: Kelebihan paling utama pada *engine* ini ialah ada pada kecepatannya. Semua proses yang ada dalam sistem ini sangat cepat, ringkas, dan mudah.

#### 2.4.3.3 Kekurangan XAMPP

Meskipun *Xampp* memiliki banyak keunggulan, ada juga beberapa kekurangan yang membuat *Xampp* tidak bisa bekerja dengan maksimal, yaitu:



1. Kurang cocok untuk menangani sebuah data dengan jumlah yang cukup besar: Baik itu untuk menyimpan sebuah data maupun untuk coba memproses data.
2. Memiliki sebuah keterbatasan kemampuan suatu kinerja: Keterbatasannya ada pada *server*, bagi yang berencana untuk mengelola data dalam jumlah yang besar, maka *Xampp* tidak akan bisa digunakan. Bahkan, proses mengelola data akan terhambat sehingga hasilnya tidak akan keluar sesuai dengan yang diinginkan. Oleh karena itu, pastikan diawal jumlah kapasitas yang dibutuhkan agar tidak sampai terjadi *error*.

#### **2.4.4 Sekilas Tentang HTML**

##### **2.4.4.1 Pengertian HTML**

Menurut Devie dan Fifin (2020:21), *HTML* adalah singkatan dari *Hypertext Markup Language*, yang memungkinkan seorang *user* untuk membuat dan menyusun bagian paragraf, *heading*, *link* atau tautan, dan *blockquote* untuk halaman *web* dan aplikasi.

Menurut Susanto (2020:1), *HTML* adalah bahasa *markup* yang digunakan untuk membuat struktur dasar halaman *web* dan menentukan konten dan tata letak halaman.

Sedangkan menurut Wibowo (2019:136), *HTML* adalah bahasa *markup* yang digunakan untuk membuat halaman *web* dengan menambahkan tag atau kode yang menjelaskan bagaimana halaman *web* akan ditampilkan di *browser*.

Dapat disimpulkan bahwa, *HTML* adalah bahasa *markup* yang digunakan untuk membuat struktur dasar halaman *web* dan menentukan konten serta tata letak halaman. *HTML* digunakan untuk menambahkan tag atau kode yang menjelaskan bagaimana halaman *web* akan ditampilkan di *browser*. *HTML* menjadi dasar dalam pembuatan halaman *web* dan sangat penting bagi pengembangan *web*.



#### 2.4.4.2 Keunggulan *HTML*

*HTML (HyperText Markup Language)* adalah suatu bahasa *standard* yang digunakan untuk menampilkan halaman *web*. Kelebihan dari *HTML* adalah sebagai berikut, yaitu:

1. Memiliki banyak sumber dengan komunitas yang sangat besar dan penggunaan yang sangat luas.
2. Bahasa pemrograman dijalankan di semua *web browser*.
3. Dapat dipelajari dengan mudah oleh pengembang web pemula.
4. Bahasa pemrograman memiliki struktur yang rapi dan konsisten sehingga mudah untuk dipelajari.
5. Bahasa pemrograman *open source* (gratis).
6. Mudah diintegrasikan dengan Bahasa *back-end* seperti *Node.js* dan *PHP*.
7. *Maintan* dilakukan langsung W3C (*World Wide Web Consortium*).
8. Digunakan untuk pembuatan struktur konten pada *website* yang dapat dijalankan seperti *javascript*.

#### 2.4.4.3 Kekurangan *HTML*

Ada juga beberapa kekurangan yang membuat *HTML* tidak bisa bekerja dengan maksimal, yaitu:

1. Penggunaan *HTML* murni hanya dapat diimplementasikan untuk halaman *web* statis.
2. Bahasa pemrograman ini tidak mendukung *user* untuk menjalankan *logic* sehingga semua halaman yang dibuat harus dibuat secara terpisah walaupun menggunakan elemen yang sama.
3. Terdapat beberapa fitur baru yang terkadang tidak dapat digunakan pada *browser* dengan cepat.
4. Perilaku *browser* yang tidak dapat diprediksi membuat proses *tag* baru terkendala.



## 2.4.5 Sekilas Tentang CSS

### 2.4.5.1 Pengertian CSS

Menurut Sarjono dan Suryanto (2021:155), CSS adalah bahasa yang digunakan untuk mengontrol dan memodifikasi tampilan halaman *web* dengan menentukan properti seperti warna, ukuran, dan tata letak elemen.

Menurut Mulyanto (2020:17), CSS adalah bahasa yang digunakan untuk memisahkan tampilan dan struktur halaman *web* dengan mengatur gaya dan penampilan elemen-elemen *HTML*.

Sedangkan menurut Arianto dan Widodo (2019:16), CSS adalah Bahasa yang digunakan untuk mengubah tampilan dan gaya elemen-elemen dalam halaman *web* dengan mengontrol properti seperti warna, *font*, dan *margin*.

Dapat disimpulkan bahwa CSS adalah bahasa yang digunakan untuk mengontrol dan memodifikasi tampilan halaman *web* dengan mengatur properti seperti warna, ukuran, dan tata letak elemen. CSS memungkinkan pemisahan antara tampilan dan struktur halaman *web*, sehingga memungkinkan pengembang *web* untuk dengan mudah mengubah gaya dan penampilan elemen-elemen dalam halaman. CSS menjadi elemen penting dalam desain *web modern*.

### 2.4.5.2 Keunggulan CSS

CSS (*Cascading Style Sheets*) adalah bahasa yang digunakan untuk mengatur tampilan dan format halaman *web*. Berikut adalah beberapa keunggulan CSS, ialah:

1. Mudah untuk para pemula: *Framework CSS* dapat menjadi pilihan yang tepat untuk pengguna yang masih pemula, karena pengguna hanya cukup menguasai *basic* dari *HTML* dan *CSS*. Dengan begitu pengguna sudah dapat menciptakan sebuah *front-end* yang baik dari bantuan *framework*. Selain itu telah tersedia juga berbagai fitur pendukung pada setiap *framework* yang telah memiliki panduan serta dokumentasi yang *detail*. Jadi dengan begitu pengguna dapat tinggal mengikuti lalu membuat *desain* yang sesuai dengan keinginan.



2. Elemen yang siap digunakan: Salah satu tantangan bagi para pengembang *front-end* ialah menyatukan dari setiap elemen UI, yang biasanya bermulai dari navigasi, tombol, tipografi, dan masih banyak lagi. Semua ini dilakukan agar elemen UI dapat berfungsi secara maksimal sehingga tidak dapat menyulitkan para pengguna, tetapi pastinya hal ini akan sangat menguras banyak waktu bila cara membuat semuanya dari nol. Tetapi dengan menggunakan *CSS framework*, semua proses dapat selesai lebih cepat. Dalam setiap *framework* ini telah tersedia berbagai *style* elemen yang siap digunakan, dan untuk sisanya dapat tinggal focus pada sisi kreatif pada proyek pengguna, contohnya memikirkan *grid*, *layout*, dan lainnya.
3. Menghasilkan performa yang stabil: Hampir semua jenis *CSS framework* ini bersifat *open source* sehingga siapa pun akan terlibat dalam pengembangan *framework* sendiri, walaupun begitu setiap *CSS framework* memiliki komunitas yang sangat kuat. Hal ini menjadi kabar baik bagi para pengguna, karena selain dapat menggunakan *framework CSS* secara gratis, pengguna dapat juga memastikan setiap kode yang dirancang akan minim dari isu seperti *error* atau kode yang *typo*. Sehingga kualitas pada performa *website* dan aplikasi dapat terjamun stabil serta dapat bekerja secara responsif.
4. Memiliki sistem grid: Grid ini merupakan struktur yang berbentuk sebuah garis vertikal, garis horizontal, dan atau garis diagonal. Garis ini akan membentuk kolom dengan jarak serta lebar yang sama. Grid berfungsi untuk dapat memudahkan dalam mengatur setiap elemen UI yang ada pada *layout* agar dapat rapi dan terstruktur.

#### 2.4.5.3 Kekurangan CSS

Berikut beberapa kekurangan *CSS framework* yang perlu diketahui, yaitu:

1. Sulit mengembangkan desain: Pada setiap jenis *framework CSS* memiliki standar kode atau sistem grid tertentu. Hal inilah yang mengakibatkan beberapa *framework* akan membatasi pengguna dalam merancang desain yang sesuai dengan keinginan, contohnya dalam mengubah ukuran *layout*, model tombol, lebar grid, dan masih banyak lagi. Selain itu juga, biasanya



pengguna akan *stuck* dengan model *framework* yang itu-itu saja, hal ini terjadi karena pengguna tidak dapat melakukan perubahan secara cepat pada beberapa kode. Dan jika pengguna ingin melakukan banyak modifikasi maka waktu yang akan dihabiskan pun cukup banyak. Terlebih jika proyek yang dibuat memerlukan parameter atau sebuah standar desain yang tidak konvensional dan harus unik.

2. Banyak kode yang tidak penting: Untuk dapat memenuhi kebutuhan dari para pengguna, maka masing-masing *framework* telah menyiapkan banyak pilihan elemen, fitur, dan variabel. Tetapi biasanya akan ada banyak *script* atau struktur kode yang tidak diperlukan. Bila pengguna termasuk yang cukup ahli maka mungkin dapat dengan mudah dalam menghapus baris kode yang tidak perlu digunakan. Namun bila pengguna masih pemula hal ini akan sangat merepotkan, karena bisa saja pengguna menghapus kode yang tidak tepat. Sehingga bukan menghasilkan performa yang optimal namun membuat *front end* menjadi berantakan.
3. Struktur desain yang *mainstream*: Bila ingin melakukan banyak perubahan kode maka waktu yang diperlukan cukup banyak. Sehingga bila pengguna menggunakan *framework* ini maka kemungkinan besar juga akan ditemukan pada sebuah *website* atau aplikasi dari pengguna lain. Terkecuali bila pengguna sudah canggih maka akan lebih mudah dalam memodifikasi kode.

## 2.4.6 Sekilas Tentang JavaScript

### 2.4.6.1 Pengertian JavaScript

Menurut Hidayatullah dan Wibisono (2022:16), *JavaScript* adalah bahasa pemrograman yang digunakan untuk mengontrol dan memanipulasi elemen halaman *web*, melakukan validasi data, serta menyediakan interaksi dengan pengguna.

Menurut Wardani, *et al.*, (2021:153), *JavaScript* adalah bahasa pemrograman yang digunakan untuk mengembangkan fitur interaktif pada halaman *web*, seperti validasi form, animasi, dan pengolahan data.



Sedangkan menurut Nurhayati dan Fauzi (2019:1099), *JavaScript* adalah bahasa pemrograman yang digunakan untuk memanipulasi elemen halaman *web* secara dinamis, melakukan validasi input, dan mengatur interaksi dengan pengguna.

Dapat disimpulkan bahwa *JavaScript* adalah bahasa pemrograman yang digunakan untuk mengontrol dan memanipulasi elemen halaman *web* untuk membuat fitur interaktif, melakukan validasi data, dan menyediakan interaksi dengan pengguna. Dengan *JavaScript*, halaman *web* dapat menjadi lebih responsif dan menawarkan pengalaman yang lebih kaya bagi pengguna.

#### **2.4.6.2 Keunggulan *JavaScript***

*JavaScript* adalah bahasa pemrograman yang digunakan untuk membuat interaksi dan pengalaman dinamis di halaman *web*. Berikut adalah beberapa keunggulan *JavaScript*, yaitu:

1. Kekayaan fungsionalitas: *JavaScript* memungkinkan pengembang untuk membuat aplikasi *web* yang interaktif dan dinamis. Bahasa ini mendukung banyak fitur, termasuk manipulasi DOM (*Document Object Model*), animasi, validasi formulir, pengolahan data, dan masih banyak lagi.
2. Kompatibilitas: *JavaScript* didukung oleh semua *browser* utama seperti Google Chrome, Mozilla Firefox, Safari, dan Microsoft Edge. Hal ini memungkinkan pengembang untuk membuat aplikasi *web* yang dapat diakses oleh pengguna dari berbagai *platform*.
3. Integrasi mudah: *JavaScript* dapat dengan mudah diintegrasikan dengan bahasa pemrograman lain dan kerangka kerja seperti Node.js, yang memungkinkan pengembang untuk menggunakan *JavaScript* di sisi *server* juga. Ini memungkinkan pengembang aplikasi *full-stack* dengan menggunakan satu bahasa pemrograman.
4. Pengembangan cepat: *JavaScript* memiliki sintaks yang mudah dipahami dan dipelajari, yang mempercepat proses pengembangan. Selain itu, ada juga banyak perpustakaan (*library*) dan kerangka kerja (*framework*) yang tersedia,



seperti React.js, Angular.js, dan Vue.js, yang dapat mempercepat pengembangan aplikasi.

5. Komunitas yang besar: *JavaScript* memiliki komunitas pengembang yang besar dan aktif. Komunitas ini menyediakan dukungan, dokumentasi, dan berbagai sumber daya untuk membantu pengembang dalam mempelajari dan mengembangkan aplikasi *JavaScript* dengan lebih baik.

### 2.4.6.3 Kekurangan *JavaScript*

*JavaScript* adalah bahasa pemrograman yang populer dan kuat yang digunakan secara luas dalam pengembangan *web*. Meskipun memiliki banyak keunggulan, *JavaScript* juga memiliki beberapa kekurangan, yaitu:

1. Keamanan: *JavaScript* dapat rentan terhadap serangan keamanan seperti injeksi skrip lintas situs (XSS) dan serangan *man-in-the-middle*. Para pengembang harus berhati-hati dalam mengimplementasikan tindakan keamanan yang tepat untuk melindungi aplikasi *web* dari serangan yang mungkin terjadi.
2. Keterbatasan kontrol akses: *JavaScript* memiliki keterbatasan dalam mengendalikan akses ke sumber daya lintas domain, seperti mengakses elemen dari *frame* atau *iframe* yang berasal dari domain lain. Ini disebabkan oleh kebijakan keamanan perambatan yang dikenal sebagai “sama dengan asal”.
3. Kinerja terbatas pada beban berat: Meskipun *JavaScript* sangat cepat dalam menjalankan tugas-tugas kecil, ia mungkin menghadapi tantangan kinerja jika digunakan untuk memproses beban berat atau aplikasi yang membutuhkan perhitungan yang rumit.

## 2.4.7 Sekilas Tentang *PHP*

### 2.4.7.1 Pengertian *PHP*

Menurut Rizal dan Kurniawan (2022:46), *PHP* adalah bahasa pemrograman yang digunakan untuk membangun aplikasi *web* dinamis dengan



kemampuan untuk berinteraksi dengan basis data dan menghasilkan konten dinamis.

Menurut Effendi dan Wicaksana (2021:85), *PHP* adalah bahasa pemrograman yang digunakan untuk mengembangkan aplikasi *web* dengan fitur seperti pengolahan form, interaksi dengan basis data, dan pembuatan konten dinamis.

Sedangkan menurut Devie dan Fifin (2020:4), *PHP* disebut sebagai bahasa *script server-side* karena *PHP* diproses pada komputer *server*. Bahasa ini didesain untuk pengembangan sebuah *website* dan saat ini menjadi bahasa pemrograman yang banyak digunakan.

Dapat disimpulkan bahwa *PHP* adalah bahasa pemrograman yang digunakan untuk mengembangkan aplikasi *web* dengan fitur-fitur seperti pengolahan *form*, interaksi dengan basis data, dan pembuatan konten dinamis. *PHP* memungkinkan pengembang *web* untuk menciptakan aplikasi *web* yang interaktif dan dinamis dengan kemampuan untuk mengelola data dari basis data serta menghasilkan halaman yang dapat berubah-ubah sesuai dengan kebutuhan pengguna.

#### **2.4.7.2 Keunggulan *PHP***

*PHP* adalah bahasa pemrograman yang banyak digunakan dalam pengembangan *web*. Berikut ini adalah beberapa keunggulan *PHP*, antara lain:

1. Kemudahan pengguna: *PHP* didesain untuk mudah dipahami dan dipelajari oleh pengembang pemula. Sintaksisnya mirip dengan bahasa pemrograman C, sehingga banyak pengembang yang sudah familiar dengan struktur kode *PHP*.
2. Kompatibilitas: *PHP* kompatibel dengan berbagai sistem operasi, seperti Windows, Linux, macOS, dan juga kompatibel dengan sebagian besar *server web* seperti Apache dan Nginx. Hal ini memudahkan pengembang untuk menggunakan *PHP* di lingkungan pengembangan pengguna.
3. Fleksibilitas: *PHP* dapat digunakan dalam berbagai jenis aplikasi *web*, mulai dari *website* sederhana hingga proyek yang kompleks. *PHP* juga mendukung



berbagai jenis basis data, termasuk MySQL, PostgreSQL, dan Oracle, sehingga memungkinkan pengembang untuk mengintegrasikan aplikasi dengan mudah.

4. Kaya akan *framework*: *PHP* memiliki berbagai kerangka kerja (*framework*) yang populer seperti Laravel, Symfony, dan CodeIgniter. Kerangka kerja ini menyediakan berbagai fitur dan alat bantu untuk mempercepat pengembangan aplikasi *web*, seperti sistem rute (*routing*), manajemen basis data, dan keamanan.
5. Komunitas yang aktif: *PHP* memiliki komunitas pengembang yang besar dan aktif di Indonesia maupun secara global. Komunitas ini menyediakan dukungan, sumber daya, dan tutorial untuk membantu pengembang *PHP* meningkatkan kemampuan mereka.

#### 2.4.7.3 Kekurangan *PHP*

Seperti halnya dengan bahasa pemrograman lainnya, *PHP* juga memiliki beberapa kekurangan, antara lain:

1. Performa yang relatif lambat: *PHP* cenderung memiliki kinerja yang lebih lambat dibandingkan dengan beberapa bahasa pemrograman lainnya, terutama dalam menghadapi beban kerja yang besar. Hal ini disebabkan oleh beberapa faktor, seperti penggunaan interpreter dan manajemen memori yang kurang efisien.
2. Pengelolaan tipe data yang lemah: *PHP* memiliki tipe data dinamis yang dapat mempermudah pengembangan, tetapi juga dapat menyebabkan kesalahan pemrograman yang sulit dideteksi. Beberapa kesalahan pemrograman yang terkait dengan tipe data dapat menyebabkan masalah keamanan atau *bug* pada aplikasi.
3. Kurangnya standar kualitas kode: *PHP* memiliki tingkat standar kualitas kode yang bervariasi. Beberapa pengembang *PHP* mungkin tidak menerapkan praktik pengkodean yang baik, yang dapat menghasilkan kode yang sulit dibaca, dipelihara, dan diperluas.



4. Kurangnya dukungan untuk pemrograman asinkron: *PHP* tidak secara alami mendukung pemrograman asinkron, yang merupakan fitur yang semakin penting dalam pengembangan *web modern*. Meskipun ada beberapa solusi tambahan, seperti *ReactPHP*, tetapi dukungan asinkronitas masih terbatas dibandingkan dengan bahasa pemrograman lain seperti *JavaScript*.
5. Kurangnya kontrol pada bagian basis data: meskipun *PHP* dapat terintegrasi dengan berbagai sistem basis data, kontrol terhadap penggunaan basis data tidak seketat pada beberapa bahasa pemrograman lain. Hal ini dapat mempengaruhi keamanan aplikasi jika tidak dikelola dengan hati-hati.

## **2.4.8 Sekilas Tentang *Bootstrap***

### **2.4.8.1 Pengertian *Bootstrap***

Menurut Fitriani dan Maulana (2022:38), *Bootstrap* adalah sebuah *framework* yang dibuat dengan menggunakan bahasa dari *HTML* dan *CSS*, namun juga menyediakan efek *javascript* yang dibangun menggunakan *jquery*.

Menurut Santoso dan Eka (2021:49), *Bootstrap* adalah *framework front-end* yang memudahkan pengembang *web* dalam membangun tampilan *website* yang responsif dan memiliki komponen UI yang siap pakai.

Sedangkan menurut Rahmawati dan Purnomo (2020:56), *Bootstrap* adalah *framework CSS* yang membantu pengembang *web* dalam membangun tampilan *website* dengan cepat dan mudah, serta menyediakan komponen UI yang responsif dan siap digunakan.

Dapat disimpulkan bahwa *Bootstrap* adalah sebuah *framework CSS* yang digunakan dalam pengembangan *web* untuk membangun tampilan *website* yang responsif dan *mobile-friendly*. *Bootstrap* menyediakan komponen UI yang siap pakai dan mempermudah pengembang *web* dalam membangun tampilan dengan cepat dan konsisten di berbagai perangkat. Dengan menggunakan *Bootstrap*, pengembang *web* dapat meningkatkan efisiensi dalam pembangunan tampilan *website* yang menarik dan responsif.



#### 2.4.8.2 Keunggulan *Bootstrap*

*Bootstrap* adalah *framework front-end* yang digunakan untuk membangun tampilan responsif dan *mobile-friendly* dalam pengembangan *web*. Berikut adalah beberapa keunggulan *Bootstrap*, yaitu:

1. Responsif: *Bootstrap* menyediakan sistem grid yang responsif, yang memungkinkan pengembang untuk dengan mudah membangun tata letak yang responsif dan dapat menyesuaikan tampilan dengan berbagai perangkat dan ukuran layar. Hal ini membantu meningkatkan pengalaman pengguna pada berbagai perangkat, mulai dari *desktop* hingga perangkat *mobile*.
2. Komponen siap pakai: *Bootstrap* menyediakan berbagai komponen UI siap pakai seperti tombol, formulir, navigasi, jumbotron, dan lainnya. Dengan menggunakan komponen-komponen ini, pengembang dapat dengan cepat membangun antarmuka yang menarik dan konsisten tanpa perlu membuat semuanya dari awal.
3. Kustomisasi mudah: *Bootstrap* memiliki struktur *CSS* yang terorganisir dengan baik dan fleksibel. Ini memudahkan pengembang untuk mengubah tampilan dan gaya elemen-elemen *Bootstrap* sesuai dengan kebutuhan proyek. Selain itu, *Bootstrap* juga menyediakan *Sass variables* dan *mixins* yang memudahkan kustomisasi lebih lanjut.
4. Dukungan *browser* yang luas: *Bootstrap* didukung oleh sebagian besar *browser* modern, termasuk Google Chrome, Mozilla Firefox, Safari, dan Microsoft Edge. Hal ini memastikan kompatibilitas yang baik dan pengalaman yang seragam bagi pengguna di berbagai *platform*.
5. Komunitas yang aktif: *Bootstrap* memiliki komunitas pengembang yang besar dan aktif, baik di Indonesia maupun secara global. Komunitas ini menyediakan dokumentasi yang kaya, sumber daya yang berguna, dan berbagai tema dan *template* yang dapat digunakan untuk memperluas fungsionalitas dan gaya *Bootstrap*.



### 2.4.8.3 Kekurangan *Bootstrap*

*Bootstrap* adalah kerangka kerja *CSS* yang populer yang digunakan untuk membangun tata letak dan desain responsif dalam pengembangan *web*. Namun, seperti halnya dengan kerangka kerja lainnya, *Bootstrap* juga memiliki beberapa kekurangan, antara lain:

1. Ukuran file yang besar: Karena *Bootstrap* dilengkapi dengan banyak fitur dan komponen, ukuran file *CSS* dan *JavaScript*-nya cenderung lebih besar dibandingkan dengan menggunakan kode *CSS* dan *JavaScript* khusus yang disesuaikan. Hal ini dapat mengakibatkan waktu unduhan yang lebih lama untuk pengguna dan mempengaruhi kinerja aplikasi *web*.
2. Ketergantungan terhadap kerangka kerja: Penggunaan *Bootstrap* membatasi fleksibilitas dan kreativitas desain. Karena *Bootstrap* memiliki struktur dan gaya yang sudah ditentukan, sulit untuk mengubah tampilan secara signifikan tanpa mengorbankan struktur dasar *Bootstrap*. Hal ini dapat menyebabkan aplikasi *web* menggunakan *Bootstrap* terlihat serupa satu sama lain.
3. Kurangnya kustomisasi: Meskipun *Bootstrap* menawarkan banyak komponen yang siap pakai, ada keterbatasan dalam kustomisasi. Ketika ingin mencapai tampilan yang unik dan berbeda, seringkali diperlukan penyesuaian *CSS* tambahan. Hal ini dapat memerlukan pengetahuan yang lebih mendalam tentang *CSS* dan memakan waktu lebih lama dalam proses pengembangan.
4. Kompatibilitas browser yang terbatas: Beberapa fitur dan komponen *Bootstrap* mungkin tidak sepenuhnya kompatibel dengan semua *browser* lama. Ini dapat menyebabkan tampilan yang tidak konsisten atau masalah fungsionalitas pada *browser* yang tidak didukung sepenuhnya.
5. Peningkatan kecepatan pemuatan yang diperlukan: *Bootstrap* menggunakan file *JavaScript* untuk mengaktifkan beberapa fitur interaktif dan responsif. Penggunaan file *JavaScript* tambahan dapat memperlambat waktu pemuatan halaman, terutama pada perangkat dengan koneksi internet yang lambat.



## 2.4.9 Sekilas Tentang *Google Chrome*

### 2.4.9.1 Pengertian *Google Chrome*

Menurut Roni, *et al.*, (2020:11), *Google Chrome* adalah sebuah penjelajah *web* sumber terbuka yang dikembangkan oleh *Google* dengan menggunakan mesin *rendering webkit*.

Menurut Rasyid (2020:43), *Google Chrome* adalah peramban *web* yang menyediakan antarmuka pengguna yang intuitif, kecepatan akses yang tinggi, dan dukungan fitur-fitur modern seperti sinkronisasi data dan ekstensi.

Sedangkan menurut Yudhantara, *et al.*, (2019:1), *Google Chrome* adalah peramban *web* yang memiliki kecepatan akses yang tinggi, stabilitas yang baik, serta kompatibilitas yang luas dengan berbagai situs *web* dan aplikasi.

Dapat disimpulkan bahwa *Google Chrome* adalah peramban *web* yang dikembangkan oleh *Google* dengan fokus pada kecepatan, keamanan, kinerja yang baik, serta antarmuka pengguna yang intuitif. *Google Chrome* memiliki fitur-fitur modern, seperti sinkronisasi data dan dukungan ekstensi. Dengan performa yang tinggi dan kompatibilitas yang luas, *Google Chrome* menjadi salah satu peramban *web* yang populer dan banyak digunakan oleh pengguna di seluruh dunia.

### 2.4.9.2 Keunggulan *Google Chrome*

*Google Chrome* sebuah peramban *web* (*web browser*) yang paling populer dan banyak digunakan di seluruh dunia. Berikut adalah beberapa keunggulan *Google Chrome*, yaitu:

1. Kecepatan dan kinerja: *Google Chrome* terkenal karena kecepatan dan kinerjanya yang tinggi. *Browser* ini menggunakan mesin *rendering* yang efisien, V8, yang memungkinkan penggunaan *web* yang lancar dan responsif.
2. Interoperabilitas dan kompatibilitas: *Chrome* mendukung standar *web* terkini dengan baik, memastikan kompatibilitas yang baik dengan banyak situs *web* dan aplikasi. *Browser* ini mendukung teknologi *web* seperti *HTML5*, *CSS3*, dan *JavaScript* yang membantu meningkatkan pengalaman pengguna.
3. Antarmuka pengguna yang *user-friendly*: *Google Chrome* menawarkan antarmuka pengguna yang bersih dan sederhana. Ikon dan menu yang mudah



diakses, serta kemampuan untuk mengelompokkan tab, membuat pengalaman menjelajah *web* lebih mudah dan nyaman.

4. Sinkronisasi antar perangkat: *Chrome* memiliki fitur sinkronisasi yang kuat, memungkinkan pengguna untuk mengakses *bookmark*, riwayat penelusuran, dan preferensi pengguna dari berbagai perangkat yang terhubung ke akun Google yang sama.

#### 2.4.9.3 Kekurangan *Google Chrome*

Meskipun memiliki banyak keunggulan, *Chrome* juga memiliki beberapa kekurangan. Berikut ini adalah beberapa kekurangan *Google Chrome*, antara lain:

1. Penggunaan sumber daya yang tinggi: *Chrome* dikenal memiliki konsumsi memori yang tinggi, terutama ketika menghadapi banyak *tab* yang terbuka secara bersamaan. Hal ini dapat menyebabkan penurunan kinerja pada sistem dengan sumber daya terbatas atau menyebabkan komputer menjadi lambat.
2. Privasi dan keamanan: Meskipun *Chrome* memiliki fitur keamanan yang baik, ada kekhawatiran tentang privasi pengguna. Google, sebagai perusahaan yang mengembangkan *Chrome*, diketahui mengumpulkan dan menganalisis data pengguna untuk berbagai tujuan. Ini bisa menjadi perhatian bagi pengguna yang mengutamakan privasi mereka.
3. Tergantung pada layanan Google: *Chrome* terintegrasi dengan berbagai layanan Google. Seperti pencarian, Gmail, dan Google Drive. Hal ini bisa menjadi kekurangan bagi pengguna yang tidak ingin terlalu bergantung pada ekosistem Google atau ingin menggunakan alternatif dari layanan tersebut.
4. Kurangnya pembaruan yang konsisten: *Chrome* sering melakukan pembaruan dengan cepat, tetapi kecepatan pembaruan ini dapat menjadi kendala bagi pengembang dan pengguna yang mengandalkan stabilitas versi sebelumnya. Terkadang, pembaruan *Chrome* juga dapat menyebabkan inkompatibilitas dengan beberapa situs *web* atau ekstensi.

## 2.5 Referensi Penelitian Terdahulu

Dalam Tugas Akhir ini penulis menggunakan beberapa jurnal sebagai referensi. Dibawah ini akan diuraikan dari referensi tersebut.

**Tabel 2.6** Referensi Penelitian Terdahulu

No	Judul/Penulis/Tahun	Masalah	Metode	Hasil
1	Suryanto, A.A. <i>et al.</i> , (2023:1721-1731), dalam jurnal <i>Building of Informatics, Technology and Science</i> (BITS), Volume 4, No 4, dengan judul “Penerapan Metode MOOSRA dan MOORA dalam Keputusan Pemilihan Produk Asuransi Terbaik”.	Pendapat masyarakat terhadap pemilihan produk asuransi sangat rendah, dikarenakan masyarakat masih kurang mengerti untuk mengasuransikan diri pada suatu produk yang akan menjamin kebutuhan hidup mereka.	MOOSRA; MOORA	Menghasilkan sebuah sistem yang dapat membantu dalam menentukan produk asuransi terbaik. Dari hasil penelitian yang telah dilakukan dengan metode MOOSRA dan MOORA untuk Pemilihan Produk Asuransi lebih mudah dan tepat dibandingkan dengan cara kerja manualnya.
2	Sihombing, S. <i>et al.</i> , (2022:151-158), dalam jurnal KOMIK (Konferensi Nasional Teknologi Informasi dan Komputer), Volume 6, Nomor 1, dengan judul “Sistem Pendukung Keputusan Pemilihan Kepala Desa Terbaik di Kecamatan Batang Kuis Menggunakan Metode MOORA”.	Dalam memilih kepala desa terbaik hanya melihat dari kemampuan yang dimiliki oleh kepala desa saja, padahal ada beberapa kriteria yang dapat dipertimbangkan oleh pihak kecamatan dalam memilih kepala desa terbaik.	MOORA	Menghasilkan sebuah sistem dalam melakukan pemilihan kepala desa terbaik dengan menggunakan beberapa kriteria yang telah ditentukan sebelumnya. Kriteria yang digunakan dalam pemilihan ini ialah Kinerja (C1), Bermasyarakat (C2), Pengetahuan (C3), Kedisiplinan (C4), dan Sikap (C5).

Lanjutan Tabel 2.6 Referensi Penelitian Terdahulu

3	Nizar, M.M. <i>et al.</i> , (2021:34-42), dalam Jurnal Informatika dan Sistem Informasi (JIFoSI), Volume 2, Nomor 1, dengan judul “Implementasi Metode MOORA pada Sistem Pendukung Keputusan Pemilihan <i>Smartwatch</i> Terbaik”.	Masih banyak pembeli kurang mengerti jenis, merek, kualitas, dan harga <i>smartwatch</i> yang beredar di pasaran. Padahal ada <i>smartwatch</i> murah yang sesuai dengan kebutuhan terlepas dari <i>brand</i> .	MOORA	Menghasilkan sebuah sistem yang dapat membantu untuk memberikan rekomendasi dalam memilih <i>smartwatch</i> agar sesuai dengan kebutuhan pembeli. Kriteria yang digunakan dalam pemilihan <i>smartwatch</i> terbaik adalah RAM (GB), CPU, Penyimpanan (GB), Baterai (Mah), Fitur, Konektivitas, Harga (Rp), dan Berat (g).
4	Pasaribu, I.P. <i>et al.</i> , (2021:497-507), dalam jurnal Seminar Nasional Informatika (SENATIKA), dengan judul “SPK Untuk Pemilihan Asisten Pelatih Sepak Bola dengan Metode <i>Profile Matching</i> pada PSSI Kabupaten Asahan”.	Pihak PSSI Kabupaten Asahan dalam memilih dan menyeleksi pemilihan asisten pelatih sepak bola masih menggunakan cara manual sehingga dalam pelaksanaannya dinilai kurang optimal.	<i>Profile Matching</i>	Menghasilkan sebuah sistem yang dapat mencocokkan antara <i>profile</i> individu dengan kriteria yang diinginkan sehingga mempermudah pemilih dalam menentukan keputusannya untuk memilih asisten pelatih sepak bola terbaik.

Lanjutan Tabel 2.6 Referensi Penelitian Terdahulu

5	Kristanto, P.B. dan Sulistyowati (2021:306-312), dalam jurnal Seminar Nasional Sains dan Teknologi Terapan IX, dengan judul “Sistem Pendukung Keputusan Pemilihan Pemain Sepak Bola Berdasarkan <i>Skill</i> Dan <i>Intelegency</i> Pada SSB PSAD Brawijaya Surabaya Menggunakan Metode MOORA”.	SSB PSAD Brawijaya Surabaya Kesulitan untuk memperhitungkan dan menentukan pemain agar menghindari kesalahan dalam pemilihan pemain terbaik berdasarkan <i>skill</i> dan <i>intelegency</i> .	MOORA	Menghasilkan sebuah sistem untuk mempermudah pelatih dan kepala sekolah dalam memilih pemain terbaik yang sesuai dengan kriteria yang diinginkan oleh pelatih SSB PSAD Brawijaya Surabaya.
6	Ulandari, N.W.A. <i>et al.</i> , (2020:53-58), dalam Jurnal Eksplora Informatika, Volume 10, No 1, dengan judul “Implementasi Metode MOORA pada Proses Seleksi Beasiswa Bidikmisi di Institut Teknologi dan Bisnis STIKOM Bali”	Pemberian beasiswa Bidikmisi dilakukan melalui proses seleksi yang dilaksanakan setiap tahun ajaran baru, oleh karena itu membutuhkan suatu metode yang dapat mempertimbangkan berbagai kriteria dalam proses seleksi.	MOORA	Menghasilkan sebuah sistem dalam mengambil suatu keputusan yang cepat dan tepat guna pemberian beasiswa Bidikmisi di Institut Teknologi dan Bisnis STIKOM Bali.