

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Penelitian terkait bertujuan untuk mengambil beberapa referensi jurnal terkait yang digunakan dalam mendukung penelitian publikasi ilmiah dalam jurnal lokal.

Penelitian yang dilakukan oleh (Juni, Irna Purba, Muhammad Safiin, Lin Parlina, 2019) dengan judul Penerapan Metode Moora Pada Penilaian Kinerja Tenaga Pegawai Honorer Pada Dinas Ketenaga Kerjaan .Berkesimpulan bahwa dalam menentukan kinerja pegawai honorer pada Dinas Ketenaga Kerjaan membantu dalam menilai psuatu kinerja pegawai

Penelitian yang dilakukan oleh (Tanty Anggeraeni, R.Mahdalena Simanjoran, 2019) dengan judul Sistem Pendukung Keputusan Seleksi Penerimaan Tenaga Kerja Kontrak Dengan Metode MOORA Pada Kantor Kepolisian Sumatera Utara . Dalam metode ini (MOORA)terdapat kriteria – kriteria yang akan dijadikan bahan perhitungan Pada proses penilaian . Hal itu dimaksudkan untuk menentukan tenaga honorer kontrak kerja yang akan di seleksi pada Kepolisian Daerah Sumatera Utaara.

Penelitian yang dilakukan oleh Muhammad (Taufik Hidayat, Fhery Agustin, Frinto Tambunan ,2018) dengan judul sistem pendukung keputusann penentuan kelayakan pegawai untuk mengikuti test kenaikan jabatan. alam proses

pengambilan keputusan bagi manager sehingga memberikan efektifitas keputusan dengan memberikan pertimbangan kepada manager untuk mengambil sebuah keputusan namun tidak mengambil peran manager itu sendiri. Dengan meningkatnya. efektifitas keputusan, manager akan lebih mudah dalam penentuan kelayakan pegawai untuk mengikuti test kenaikan jabatan sehingga hasil yang didapat untuk menentukan kelayakan tersebut berdasarkan perhitungan atas data yang ada. Metode SMART (Simple Multi Attribute Rating Technique) merupakan metode pengambilan keputusan multi kriteria yang didasarkan pada teori bahwa setiap alternatif terdiri dari sejumlah kriteria yang memiliki nilai-nilai dan setiap kriteria memiliki bobot yang menggambarkan seberapa penting dibandingkan dengan kriteria lain.

Penelitian yang dilakukan oleh (Santri W Pasaribu , Ester Rajagukguk , Maret Sitanggang, Robbi Rahim , Leon Andretti Abdillah, 2018)dengan judul Implementasi Multi-Objective Optimization On The Basis Of Ratio Analysis (MOORA) Untuk Menentukan Kualitas Buah Mangga Terbaik. Kesimpulan yang diperoleh dari penelitian yang dilakukan, bahwa metode MOORA(Multi-Objective Optimization on the Basis of Ratio Analysis dapat menyeleksi alternatif dan melakukan perangkingan dalam menentukan kualitas buah mangga terbaik berdasarkan kriteria-kriteria yang sudah ditentukan.

Penelitian yang dilakukan oleh (Dahri Yani Hakim Tanjung, Robiatul Adawiyah ,2018) dengan judul Perancangan Sistem Pendukung Keputusan Dengan Metode Simple Multi Attribute Rating Technique (SMART). Dalam Penilaian Kinerja Dosen (Studi Kasus: Universitas Potensi Utama). Penelitian ini

bertujuan untuk memberikan penilaian terhadap kinerja dosen dengan mengukur kinerja dan menganalisa pencapaian target untuk meningkatkan kualitas layanan internal maupun eksternal perguruan tinggi. Dimana dengan Sistem pendukung keputusan menggunakan metode SMART menawarkan solusi untuk rujukan dalam memberikan penilaian terhadap kinerja dosen. Metode ini meliputi proses penilaian kinerja yang dimulai dari pembobotan kriteria untuk mengetahui bobot kepentingan masing-masing indikator kemudian penjabaran tujuan strategis ke dalam indikator kinerja. Dari pembobotan indikator tersebut dapat menghasilkan bobot alternatif untuk mengetahui nilai tertinggi dari alternatif yang ada. Penelitian dilakukan dengan mencari nilai bobot untuk setiap kriteria, kemudian dilakukan proses perankingan yang akan menentukan alternatif yang optimal, yaitu untuk melihat dosen yang melakukan kinerjanya sesuai dengan kriteria-kriteria yang ditentukan dari Universitas Potensi Utama. Dimana nilai tertinggi dengan hasil akhir 2,36 menjadi peringkat pertama didalam penilaian kinerja dosen.

Penelitian yang dilakukan oleh (Ria Eka Sari, Alfa Saleh,2017) dengan judul Penilaian kinerja Dosen Menggunakan Metode AHP (Studi Kasus Potensi Utama). Dalam mengevaluasi penilaian kinerja dosen diimplementasikan pada SPK dengan menggunakan AHP untuk mendapatkan peringkat tertinggi dari keempat dosen. Penggunaan perangkat lunak sistem pendukung keputusan (SPK) dengan metode Analytical Hierarchy Process (AHP) menggunakan Super Decisions ini terdiri dari 4 kriteria, yaitu kriteria kehadiran dosen, pengumpulan nilai, keterlambatan masuk pbm dan kecepatan masuk pbm pada suatu penilaian

kinerja dosen dan yang menjadi alternatif 4 dosen yang setelah diakumulasikan mendapatkan nilai tertinggi diantara dosen lainnya. Setelah mengetahui parameter, langkah selanjutnya adalah menganalisis kebutuhan sistem, merekayasa pengetahuan, menerapkan metode dan pengujian sistem dengan menggunakan Software Super Decisions. Diharapkan metode ini dapat memudahkan untuk mendukung sesuatu keputusan dengan masalah yang kompleks.

Penelitian yang dilakukan oleh (Ratna Sri Hayati, Siti Aliyah, 2020) dengan judul Sistem Pendukung Keputusan Promosi Jabatan Menggunakan Metode Multi Attribute Utility Theory. Dengan menerapkan Metode MAUT pada Sistem Pendukung Keputusan Promosi Jabatan, dapat memberi saran atau rekomendasi Karyawan terbaik yang dinilai secara objektif.

II.2. Uraian Teoritis

Penelitian yang dilakukan dengan cara langsung pada Penerapan Metode MOORA Dalam Pemilihan Honorer Terbaik Pada Dinas Pertanian dan Perikanan Kota Medan menggunakan beberapa teoritis serta aplikasi atau bahasa pemrograman, yaitu :

II.2.1. Sistem Informasi

Menurut (Rahmat Alfian Syahputra, 2020) sistem informasi merupakan sistem yang menyediakan informasi dengan cara sedemikian rupa sehingga bermanfaat bagi penerima informasi. Seiring dengan perkembangan teknologi yang begitu pesat, sistem informasi yang berbasis *web* dimanfaatkan sebagai sarana peningkatan informasi. Pemanfaatan tersebut akan mempermudah suatu pekerjaan seperti halnya pengolahan data lebih cepat, keputusan yang akan diambil lebih tepat, menghemat waktu dan biaya. Selain itu, sistem informasi yang berbasis *web* juga dapat menjadi sarana promosi yang efisien dan sumber informasi yang dapat diakses oleh pengguna internet yang semakin lama semakin luas.

II.2.2. Sistem Pendukung Keputusan

Sistem yaitu kelompok yang menekankan aturan dari awal dan akhir untuk mencapai tujuan. pendukung keputusan yaitu pola pikir membantu menyelesaikan masalah. Sistem pendukung keputusan adalah aturan dari awal dan akhir dalam menyelesaikan masalah keputusan yang rinci dan jelas. Tahapan dasar tersebut adalah Informasi input disampaikan dalam matriks alternatif dan atribut, yang didasarkan pada matriks keputusan (Pristiwati Fitriani, 2020).

II.3. Metode MOORA (*Multi-Objective Optimization on the basis of Ratio Analysis*)

Metode MOORA (*Multi-Objective Optimization on the basis of Ratio Analysis*) adalah multi objektif sistem mengoptimalkan dua atau lebih atribut

yang saling bertentangan secara bersamaan. Metode ini diterapkan untuk memecahkan masalah dengan perhitungan matematika yang kompleks. Moora diperkenalkan oleh *Brauers* dan *Zavadskas* pada tahun 2006. Pada awalnya metode ini diperkenalkan oleh *Brauers* pada tahun 2004 sebagai “*Multi-Objective Optimization*” yang dapat digunakan untuk memecahkan berbagai masalah pengambilan keputusan yang rumit pada lingkungan pabrik. *Objective Optimization on the Basis of Ratio Analysis* (MOORA) termasuk dalam bagian MCDM yang mampu melakukan proses sekaligus mengoptimalkan dua atau lebih Atribut (sasaran) yang bertentangan dengan batasan tertentu dari rating kinerja pada setiap alternative pada semua atribut. Metode MOORA dapat membantu dalam pengambilan keputusan suatu kasus, akan tetapi perhitungan dengan menggunakan metode MOORA ini hanya yang menghasilkan nilai terbesar yang akan terpilih sebagai alternatif yang terbaik.

Langkah langkah penyelesaian dari Metode MOORA secara lebih terinci dapat dijabarkan sebagai berikut:

1. Penentuan nilai *matriks* keputusan menentukan tujuan untuk mengidentifikasi atribut evaluasi yang bersangkutan.

$$X = \begin{pmatrix} X_{11} & X_{12} & X_{1n} \\ X_{21} & X_{22} & X_{2n} \\ X_{31} & X_{32} & X_{3n} \end{pmatrix}$$

2. Normalisasi *Matriks Braures* (2008) menyimpulkan bahwa untuk penyebut, pilihan terbaik adalah akar kuadrat dari jumlah kuadrat dari setiap alternatif peratribut.

$$X^*_{ij} = X_{ij} / \sqrt{\sum_{i=1}^m X_{ij}^2}$$

3. Mengoptimalkan Atribut Untuk optimasi multiobjektif, ukuran yang dinormalisasi ditambahkan dalam kasus maksimasi (untuk atribut yang menguntungkan) dan dikurangi dalam kasus minimasi (untuk atribut yang tidak menguntungkan).

$$Y_i = \sum_{j=1}^g X_{ij} - \sum_{j=g+1}^n X_{ij}$$

Diman G adalah jumlah atribut yang akan dimaksimalkan (n - g) adalah jumlah atribut yang akan diminimalkan, yi adalah nilai penilaian yang telah dinormalisasikan dari alternatif 1 terhadap semua atribut.

Saat atribut bobot dipertimbangkan, persamaan 3 menjadi sebagai berikut:

$$Y_i = \sum_{j=1}^g W_j X_{ij} - \sum_{j=g+1}^n W_j X_{ij}$$

4. Perangkingan Nilai Yi Nilai Yi bisa positif atau negatif tergantung dari total maksimal dan minimal dalam matriks keputusan. Sebuah urutan peringkat dan Yi menunjukkan pilihan terakhir. Dengan demikian alternatif terbaik memiliki nilai Yi tertinggi, sedangkan alternatif terburuk memiliki nilai yang rendah.

II.4. UML (*Unified Modeling Language*)

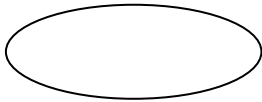
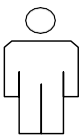
Unified Modelling Language (UML) adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma berorientasi objek. Abstraksi konsep dasar *UML* terdiri dari *structural classification*, *dynamic behavior*, dan model *management* dapat kita pahami main *concepts* sebagai term yang akan muncul pada saat membuat *diagram* dan *view* adalah kategori dari *diagram*

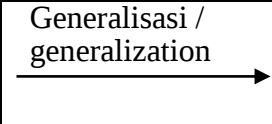
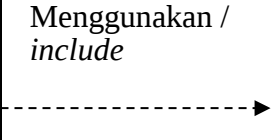
tersebut. *UML* mendefinisikan *diagram-diagram* sebagai *Use case diagram*, *Class diagram*, *Statechart diagram*, *Activity diagram*, *Sequence diagram*, *Collaboration diagram*, *Component diagram*, dan *Deployment diagram*.
(Suendri, 2018 : 2)

A. Use Case Diagram

Use Case Diagram merupakan pemodelan untuk kelakuan sistem informasi yang akan dibangun. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibangun. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada pada sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut.
(Julianto Simatupang, 2019).

Tabel. II. 1. Simbol *Use Case Diagram*

Simbol	Diskripsi
<i>Use Case</i> 	<i>Fungsionalitas</i> yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor: biasanya dinyatakan dengan menggunakan kata kerja diawal di awal <i>frase</i> nama <i>use case</i> .
Aktor/actor 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri.
Asosiasi /association 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
Ekstensi / extend <<extends>>	<i>Relasi use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada

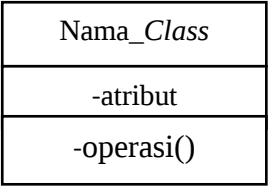
	pemrograman berorientasi objek biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan misal Arah panah mengarah pada <i>use case</i> yang ditambahkan.
Generalisasi / generalization 	Hubungan <i>generalisasi</i> dan spesialisasi (umum – khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
Menggunakan / <i>include</i> 	<i>Include</i> berarti <i>use case</i> yang tambahan akan selalu melakukan pengecekan apakah <i>use case</i> yang ditambahkan telah dijalankan sebelum <i>use case</i> tambahan dijalankan

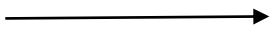
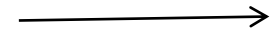
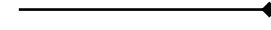
Sumber : (Julianto Simatupang, 2019)

B. Class Diagram

Class Diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. (Julianto Simatupang, 2019)

Tabel II. 1. Simbol *Class Diagram*

Simbol	Diskripsi
Kelas 	Kelas pada struktur sistem.
Antar muka/interface  <i>nama_interface</i>	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
Asosiasi / association 	<i>Relasi</i> antar kelas dengan makna umum, <i>asosiasi</i> biasanya juga disertai dengan <i>multiplicity</i> .

Asosiasi berarah / <i>directed association</i> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
Generalisasi 	Relasi antar kelas dengan makna generalisasi- generalisasi- spesialisasi (umum khusus).
Kebergantungan / <i>dependency</i> 	Relasi antar kelas dengan makna kebergantungan antar kelas
Agregasi / <i>aggregation</i> 	Relasi antar kelas dengan makna semua-bagian (<i>whole- part</i>).

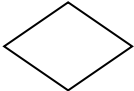
Sumber : (Julianto Simatupang, 2019)

C. Activity Diagram

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Penekanan pada diagram aktivitas adalah menggambarkan aktivitas sistem atau aktivitas yang dapat dilakukan oleh sistem, bukan apa yang dilakukan aktor (Julianto Simatupang, 2019).

Tabel II. 2. Simbol *Activity Diagram*

Simbol	Diskripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.

Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan / <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan <i>aktivitas</i> lebih dari Satu
Penggabungan / Join	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu..
Status akhir 	Status akhir yang dilakukan sistem, sebuah <i>diagram aktivitas</i> memiliki sebuah status akhir.

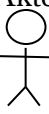
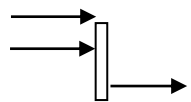
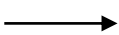
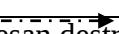
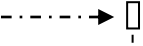
Sumber : (Julianto Simatupang, 2019)

D. Sequence Diagram

Sequence Diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar *diagram sequence* maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang *diinstansiasi* menjadi objek itu. (Julianto Simatupang, 2019).

Tabel II. 3. *Sequence Diagram*

Simbol	Diskripsi
--------	-----------

Aktor 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri.
Garis hidup / lifeline 	Menyatakan kehidupan suatu objek.
Objek 	Menyatakan objek yang berinteraksi pesan.
Waktu aktif 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya
Pesan tipe create << create >> 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat. arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil operasi/metode maka operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi.
Pesan tipe send 1 : masukan 	Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
Pesan tipe return 1 : keluaran 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.
Pesan tipe destroy 	Menyatakan suatu objek mengakhiri hidup objek lain, arah panah yang mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada destroy.

Sumber : (Julianto Simatupang, 2019)

II.6. Website

Menurut (Raniya dkk, 2020) *Website* merupakan sekumpulan halaman yang berhubungan yang dapat diakses public melalui *world wide web*. Suatu *website* biasanya mengandung teks, gambar, suara bahkan video interaktif dan mempunyai kelebihan untuk menghubungkan (*link*) satu dokumen dengan dokumen lainnya (*hypertext*) yang dapat diakses melalui sebuah *browser*. *website* adalah kumpulan halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara dan/atau gabungan dari semuanya, baik

yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan – jaringan halaman. Maka dapat disimpulkan bahwa *website* adalah sekumpulan informasi yang terdiri dari teks, gambar diam atau gerak, animasi, suara yang membentuk satu rangkaian bangunan yang terhubung ke internet.

II.6.1. HTML (*Hypertext Markup Language*)

HTML (Hyper Text Mark Up Language) bahasa pemrograman standar yang digunakan untuk membuat sebuah halaman web, yang kemudian dapat diakses untuk menampilkan berbagai informasi di dalam sebuah penjelajah web Internet HTML dapat juga digunakan sebagai link link antara file-file dalam situs atau dalam komputer dengan menggunakan *localhost*, atau *link* yang menghubungkan antar situs dalam dunia internet. (Sumber : *Pengertian HTML dan Fungsinya, Sejarah HTML, Cara Kerja HTML (temukanpengertian.com)*).

II.6.2. CSS (*Cascading Style Sheet*)

Menurut (Syahril Hasan dkk, 2020). *CSS (Cascading Style Sheet)* adalah sebuah fitur yang diperkenalkan sejak *HTML* versi 4.0 dan berfungsi untuk menangani masalah tampilan pada *HTML* seperti jenis, ukuran dan warna *font*, posisi teks, batas tulisan atau *margin*, warna *background*, dan sebagainya. penting yang perlu diperhatikan adalah cara meletakkan *CSS* dan juga bahasa berbasis *web* lain untuk memudahkan manajemen *file*, *editing*, dan *maintenance*

II.6.3. PHP (*Hypertext Preprocessor*)

Menurut (Khairunnisa Fauzia, 2020), *PHP Hypertext Preprocessor* adalah suatu bahasa *scripting* khususnya digunakan untuk *web server*. *PHP* memiliki sifat *server side scripting* sehingga untuk menjalankan *PHP* harus menggunakan *web server*.

II.6.4. JavaScript

JavaScript adalah bahasa pemrograman untuk *web*. Ini didukung oleh sebagian besar *Web Browser* termasuk *Chrome*, *Firefox*, *Safari*, *internet Explorer*, *Edge*, *Opera*, dan lainnya. Sebagian besar *browser ponsel* untuk ponsel pintar juga mendukung *JavaScript*. Ini terutama digunakan untuk meningkatkan halaman *web* untuk memberikan pengalaman yang lebih ramah pengguna. Ini termasuk memperbarui halaman *web* secara dinamis, peningkatan antarmuka pengguna seperti *menu* dan kotak dialog, *animasi*, grafik 2D dan 3D, peta *interaktif*, pemutar video, dan banyak lagi. Mode penggunaan *JavaScript* ini di *browser web* juga disebut sebagai *javascript sisi klien* atau *Client-Side Javascript*.
(Sumber : *Pengertian JavaScript: Sejarah, Cara Kerja JavaScript dan Manfaatnya!* (rifqimulyawan.com)).

II.6.5. Bootstrap

Menurut (Fahrul Khoirul Khadziq dkk, 2020). Pengertian *bootstrap* adalah sebuah *framework CSS* dari *Twitter* yang menyediakan komponen-komponen antarmuka siap pakai dan telah dirancang sedemikian rupa untuk keperluan desain halaman *website* artistik. Selain komponen-komponen dasar untuk membangun

antarmuka, *bootstrap* juga menyediakan *grid* 12 kolom yang dapat menunjang kinerja anda dalam mendesain *layout* yang rapi. Sebagai *CSS framework*, *Bootstrap* tergolong paket lengkap. Teknologi *HTML*, *CSS*, dan *Javascript* yang ada pada *Bootstrap*, anda tidak hanya dapat *styling* dengan *CSS* saja, akan tetapi dapat juga menggunakan komponen-komponen seperti ikon, tombol dan *navigasi* dengan desain unik khas *Bootstrap*.¹² Selain itu, tersedia banyak *plugin JQuery* yang dapat membantu untuk membuat tampilan *website* yang *atraktif*. *Framework Bootstrap* dibangun menggunakan *Less*, yaitu sebuah *preprocessor* dengan fungsi tambahan seperti *variable*, *mixin* dan fungsi untuk *mengompilasi CSS*. Fitur-fitur yang ditawarkan pada *Bootstrap* dapat membuat sang *developer Bootstrap* kaya jika berorientasi dengan uang. *Framework* ini seharusnya layak untuk menjadi *Framework* berbayar, mengingat fitur-fitur yang ditawarkan sangat lengkap. Kenyataannya, *framework* ini justru tersedia gratis, yaitu dengan menggunakan *lisensi open source*, dimana semua kalangan bebas menggunakan *framework* ini secara gratis tanpa takut terkena masalah legalitas.

II.7. *JQuery*

Menurut (Omar Pahlevi dkk, 2018). *JQuery* adalah *open source add-on* pustaka yang menekankan pada interaksi antara *JavaScript* dan *HTML*. *JQuery JavaScript* merupakan kode *JavaScript* yang telah ditulis dan tinggal menambahkan satu atau dua baris kode untuk memanggil *JQuery*.

II.7.1. Sublime Text

Menurut (Vivi Afifah dkk, 2021), Sublime Text adalah aplikasi editor untuk kode dan teks yang dapat berjalan diberbagai platform sistem operasi menggunakan teknologi Phyton API.

II.7.2. Database

Menurut (Novri Ultariani dkk, 2020), *Database* adalah suatu aplikasi yang menyimpan sekumpulan data. Setiap *database* mempunyai *API* tertentu untuk membuat, mengakses, mengatur, mencari, menyalin data yang ada di dalamnya. Atau juga *Database* yaitu kumpulan file-file yang berhubungan satu dengan yang lainnya, diatur sedemikian rupa sehingga dapat digunakan oleh beberapa program aplikasi *database*.

II.7.3. MySQL

Menurut (Aulia Ikhsan dkk, 2020). *MySQL* adalah *Relational Database Management System (RDMS)* yang cepat dan akurat. Sebuah basis data dapat membuat pengguna untuk menyimpan, mencari, mengurutkan, dan mendapatkan data dengan sangat efisien. *Server MySQL* mengendalikan akses dalam data untuk memastikan bahwa para pengguna dapat bekerja dalam waktu yang bersamaan.