

BAB II

TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Penelitian terdahulu ini menjadi salah satu acuan penulis dalam melakukan penelitian sehingga dapat memperkaya teori yang digunakan dalam mengkaji penelitian yang dilakukan. Dari penelitian terdahulu, penulis tidak menemukan penelitian dengan judul yang sama seperti judul penelitian penulis. Namun penulis mengangkat beberapa penelitian sebagai referensi dalam memperkaya bahan kajian pada penelitian penulis. Berikut merupakan penelitian terdahulu berupa beberapa jurnal terkait dengan penelitian yang dilakukan penulis.

Pada penelitian sebelumnya (Ilham, I., dan Apriadi, D., 2020) dalam jurnal yang berjudul **“Sistem Pendukung Keputusan Pemilihan Balita Sehat dengan Menggunakan Metode Simple Additive Weighting (SAW)”**, Salah satu yang menjadi persoalan dalam penyelenggaraan pemilihan bayi sehat adalah petugas masih kesulitan dalam menentukan siapa bayi sehat yang terpilih karena banyaknya kriteria yang harus dipenuhi sehingga memerlukan waktu yang cukup lama dan juga rumit dapat dikatakan masih kurang efektif dan efisien. Karena proses pemilihan Balita Sehat masih menggunakan blangko sebagai media penilaian Balita Sehat. Pemilihan atau penilaian balita sehat dengan menggunakan blangko sebagai media penilaian masih membutuhkan proses yang lama saat perekapan nilai dan sering terjadi kesalahan, selain itu juga blangko yang dikumpulkan panitia juga sering terjadi kerusakan dan hilang karena tercampur berkas yang lain. Selain itu panitia kesulitan dalam menentukan balita sehat karena terlalu banyak kriteria sehingga membutuhkan proses yang lama. Dengan mencoba membangun sebuah sistem yang mampu membantu panitia dalam menentukan keputusan untuk pemilihan Balita Sehat. Metode penelitian yang digunakan pada jurnal ini mulai dari mengidentifikasi masalah, mengumpulkan data, analisis data dan penerapan metode *Simple Additive Weighting* (SAW). Aplikasi yang dibuat menggunakan bahasa pemrograman PHP. Pada implementasinya, Sistem Pendukung Keputusan yang telah dibuat dapat melakukan *input* data penilaian gigi, penilaian biodata, penilaian berat badan, penilaian imunisasi dan dapat menampilkan data hasil penilaian, hasil

penilaian balita yang dinyatakan sehat, dan hasil penilaian balita yang dinyatakan gugur. Dalam pengujian yang dilakukan Sistem Pendukung Keputusan pemilihan balita sehat yang dibuat berhasil dijalankan dengan baik.

Penelitian yang lain yaitu (Sari,R.,K. Dan Hamdani, 2016) dalam jurnal yang berjudul **“Sistem Pendukung Keputusan Pemilihan Balita Sehat Menggunakan Penalaran Fuzzy Tsukamoto (Studi Kasus: Puskesmas Wonorejo Samarinda)”**, memiliki permasalahan pada saat penyelenggaraan pemilihan balita sehat adalah petugas masih kesulitan dalam menentukan siapa balita sehat yang terpilih karena banyaknya kriteria yang harus dipenuhi. Hal tersebut menyebabkan penilaian yang dilakukan untuk tiap balita memerlukan waktu yang cukup lama dan juga rumit. Karena itu dalam proses penilaian masih sering terjadi ketidaksengajaan obyektivitas pemilihan balita sehat, sehingga keputusan yang diambil kurang bisa dipertanggungjawabkan hasilnya. Dibutuhkan satu sistem terkomputerisasi yang dapat membantu tim penilai untuk membantunya memperoleh suatu keputusan yang baik sesuai dengan penilaian. Dengan adanya Sistem Pendukung Keputusan (SPK) dengan menggunakan metode penalaran *fuzzy* tsukamoto dapat menjadi salah satu alternatif pemecahan masalah tersebut. Aplikasi yang dibuat menggunakan bahasa pemrograman Visual Basic (VB). Metode Penelitian yang digunakan pada jurnal ini menggunakan metode pengumpulan data seperti observasi, wawancara dan studi pustaka. Pada implementasinya, Sistem Pendukung Keputusan yang telah dibuat dapat melakukan *input* data balita, data penilaian balita dan dapat menampilkan data hasil penilaian balita sehat. Berdasarkan pengujian yang dilakukan, Sistem Pendukung Keputusan pemilihan balita sehat yang dibuat berhasil dijalankan dengan baik.

Penelitian berikutnya (Irma,2020) dalam jurnal yang berjudul **“Sistem Pendukung Keputusan Pemilihan Pemenang Lomba Balita Sehat Dengan Menerapkan Metode Weighted Aggerated Sum Product Assessment (Waspas) Pada Posyandu Desa Karang Anyar”**, memiliki permasalahan dalam proses penilaiannya, para panitia masih menggunakan proses manual dengan cara memberi nilai pada setiap kriteria kemudian mengambil nilai rata-ratanya untuk setiap alternatif. Alternatif yang memiliki rata-rata paling tinggi yang menjadi

pemenang lomba. Cara ini kurang efektif karena hasil akhir dari perhitungan tidak sesuai dan dapat menimbulkan ketidakadilan bagi para peserta lomba. Agar pemilihan pemenang lomba balita sehat mampu menghasilkan hasil akhir yang akurat dan sesuai maka diperlukan sebuah sistem pendukung keputusan pemilihan pemenang lomba balita sehat. Aplikasi yang dibuat menggunakan bahasa pemrograman Visual Basic (VB). Metode penelitian yang digunakan pada jurnal ini yaitu mengidentifikasi masalah dengan melakukan tinjauan langsung ke lapangan, dan mempelajari literatur yang bersumber dari beberapa buku dan jurnal yang berhubungan dengan permasalahan yang ada. Pada implementasinya, Sistem Pendukung Keputusan dengan menggunakan metode *weighted aggerated sum product assessment* (waspas) yang telah dibuat dapat melakukan *input data user*, data peserta lomba balita sehat, data kriteria, data bobot nilai kriteria, data penilaian dan dapat menampilkan data hasil perhitungan penilaian lomba balita sehat. Berdasarkan pengujian yang dilakukan, Sistem Pendukung Keputusan pemilihan pemenang lomba balita sehat yang dibuat berhasil dijalankan dengan baik.

Berdasarkan penelitian terdahulu pada jurnal pertama yang telah dipublikasikan, terdapat persamaan antara penelitian yang telah dilakukan dengan penelitian yang akan dilakukan, yaitu bahasa pemrograman yang digunakan dalam pembuatan Sistem Pendukung Keputusan pemilihan balita sehat menggunakan bahasa pemrograman PHP dan *database management system* MySQL. Akan tetapi pada penelitian yang akan dilakukan, digunakan pula metode *Weighted Product* (WP). Selain itu terdapat perbedaan pada metode yang digunakan, pada penelitian terdahulu pada jurnal kedua dan ketiga yaitu menggunakan bahasa pemrograman visual basic dan menggunakan metode *Simple Additive Weighting* (SAW) dan *Weighted Aggerated Sum Product Assessment* (Waspas).

Dari ke 3 penelitian tersebut dapat dilihat perbandingan penelitian terdahulu dan penelitian sekarang, untuk lebih jelas dapat dilihat ditabel 2.1 sebagai berikut :

Tabel 2.1 Perbandingan penelitian terdahulu dan penelitian sekarang.

No.	Penelitian	Persamaan	Perbedaan
-----	------------	-----------	-----------

1.	Ilham, I., & Apriadi, D.(2020).Sistem Pendukung Keputusan Pemilihan Balita Sehat dengan Menggunakan Metode Simple Additive Weighting (SAW).	1). Menggunakan bahasa pemrograman PHP.	1). Menggunakan Metode <i>Simple Additive Weighting</i> (SAW).
2.	Sari, R. K., & Hamdani(2016).Sistem Pendukung Keputusan Pemilihan Balita Sehat Menggunakan Penalaran Fuzzy Tsukamoto (Studi Kasus: Puskesmas Wonorejo Samarinda).	-	1). Menggunakan Metode Penalaran Fuzzy Tsukamoto . 2). Menggunakan bahasa pemrograman Visual Basic (VB).
3.	Irma, (2020). Sistem Pendukung Keputusan Pemilihan Pemenang Lomba Balita Sehat Dengan Menerapkan Metode Weighted Aggerated Sum Product Assessment (Waspas) Pada Posyandu Desa Karang Anyar	-	1). Menggunakan Metode <i>Weighted Aggerated Sum Product Assessment</i> (Waspas). 2). Menggunakan bahasa pemrograman Visual Basic (VB).

2.2. Sistem Pendukung Keputusan

Menurut Fajarika (2019) Sistem merupakan kumpulan sub-sub sistem (elemen) yang saling berkorelasi satu dengan yang lainnya untuk mencapai tujuan

tertentu. Sebagai contoh yaitu sebuah perusahaan memiliki sistem manajerial yang terdiri dari *bottom management*, *middle management* dan *top management* yang memiliki tujuan untuk mencapai kemajuan masyarakat. Sistem Pendukung Keputusan dapat diartikan sebagai suatu sistem yang dirancang yang digunakan untuk mendukung manajemen dalam pengambilan keputusan.

Menurut Turban dalam jurnal yang ditulis oleh Irma (2020) sistem pendukung keputusan merupakan suatu sistem interaktif yang membantu pengambilan keputusan melalui penggunaan data dan model-model keputusan untuk memecahkan masalah-masalah yang sifatnya semi terstruktur dan tidak terstruktur.

2.3. Website

Menurut Zufria dan Azhari dalam jurnal yang ditulis oleh Zahir (2019:3) *Website* adalah sebuah komponen yang terdiri dari teks, gambar, suara, animasi sehingga menjadi media informasi yang menarik dikunjungi oleh orang lain.

Menurut Prasetyo dalam jurnal yang ditulis oleh Hendrawan,dkk. (2017:55) *Website* adalah sekumpulan dokumen yang berada pada *server* dan dapat dilihat oleh *user* dengan menggunakan *browser*. Dokumen itu bisa terdiri dari beberapa halaman. Tiap-tiap halamannya memberi informasi atau interaksi yang beraneka ragam. Informasi dan interaksi itu bisa berupa tulisan, gambar atau bahkan dapat ditampilkan dalam bentuk video, animasi, suara, dan lain-lain.

Halaman *website* biasanya berupa dokumen yang ditulis dalam format *HyperText Markup Language* (HTML), yang bisa diakses melalui HTTP, HTTP adalah suatu protokol yang menyampaikan berbagai informasi dari *server website* untuk ditampilkan kepada para *user* atau pemakai melalui *web browser* (Nofyat,dkk., 2018:12).

2.4. Database

Basis data (*database*) merupakan kumpulan data yang saling berkaitan dan berhubungan satu dengan yang lain, tersimpan di perangkat keras komputer dan menggunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data didalam basis

data perlu diorganisasikan sedemikian rupa supaya informasi yang dihasilkan berkualitas (Fauzi,dkk., 2021:42).

Siregar,dkk. (2018:115) menjelaskan *database* atau basis data adalah sekumpulan informasi yang disimpan dalam komputer secara sistematis dan merupakan sumber informasi yang dapat diperiksa menggunakan suatu program komputer dan berfungsi untuk menyimpan informasi atau data.

2.5. HTML

Menurut Suryatiningsih dalam jurnal yang ditulis oleh Fauzi, dkk., (2021:43) “*HyperText Markup Language* (HTML) adalah bahasa yang digunakan untuk membuat suatu situs *web* atau *homepage*. Setiap dokumen dalam *web* ditulis dalam format HTML. Semua format dokumen, *hyperlink* yang dapat diklik, gambar, dokumen multimedia *form* yang dapat diisi dan sebagainya didasarkan atas HTML.

Menurut Puspitasari (2016:228) *HyperText Markup Language* merupakan suatu metode untuk mengimplementasikan konsep *hypertext* dalam suatu naskah atau dokumen. HTML sendiri bukan tergolong pada suatu bahasa pemrograman karena sifatnya yang hanya memberikan tanda (*marking up*) pada suatu naskah teks dan bukan sebagai program.

2.6. PHP

Menurut Simanjuntak, dkk. (2017:28) PHP (*Perl Hypertext Preprocessor*) adalah bahasa pemrograman yang digunakan secara luas untuk penanganan pembuatan dan pengembangan sebuah *web* dan biasa digunakan pada HTML.

Menurut Yuhendra dan Yulianto (2015) “PHP adalah singkatan dari *Perl Hypertext Preprocessor* yang digunakan sebagai bahasa *script server-side* dalam pengembangan *web* yang disisipkan pada dokumen HTML.”

Menurut Puspitasari (2016:228) PHP atau yang memiliki kepanjangan *Perl Hypertext Preprocessor* merupakan suatu bahasa pemrograman yang difungsikan untuk membangun suatu *website* dinamis, PHP menyatu dengan kode HTML, HTML digunakan sebagai pembangun atau pondasi dari kerangka *layout web*,

sedangkan PHP difungsikan sebagai prosesnya, sehingga dengan adanya PHP tersebut sebuah *web* akan sangat mudah di *maintenance*.

2.7. MySQL

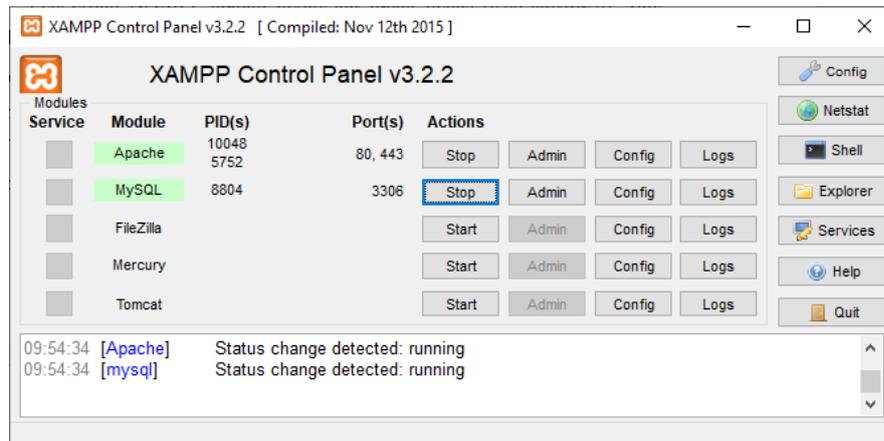
Menurut Puspitasari (2016:229) MySQL merupakan *database* yang pertama kali didukung oleh bahasa pemrograman *script* untuk internet. MySQL dan PHP dianggap sebagai pasangan *software* pembangun aplikasi *web* yang ideal. MySQL lebih sering digunakan untuk membangun aplikasi berbasis *web*, umumnya pengembangan aplikasinya menggunakan bahasa pemrograman *script* PHP.

Menurut Suryadi dan Zulaikhah dalam jurnal yang ditulis oleh Catriwati dkk. (2020:103) menjelaskan bahwa, “MySQL merupakan perangkat lunak sistem manajemen berbasis data SQL (Bahasa Inggris: *database management system*) atau DBMS yang *multi thread* dan *multi user*, dengan sekitar enam juta instalasi di seluruh dunia”.

2.8. XAMPP

Menurut Iqbal dalam jurnal yang ditulis oleh Putra (2019:82) menyatakan XAMPP merupakan *software server apache* dimana dalam XAMPP yang telah tersedia *database server* seperti MySQL dan PHP *programming*. XAMPP memiliki keunggulan yaitu cukup mudah dioperasikan, tidak memerlukan biaya serta mendukung instalasi pada Windows dan Linux. Keuntungan lain yang didapatkan adalah hanya dengan melakukan instalasi cukup satu kali kemudian didalamnya tersedia MySQL, *apache web server*, *database server* PHP *support* (PHP 4 dan PHP 5) dan beberapa modul lainnya.

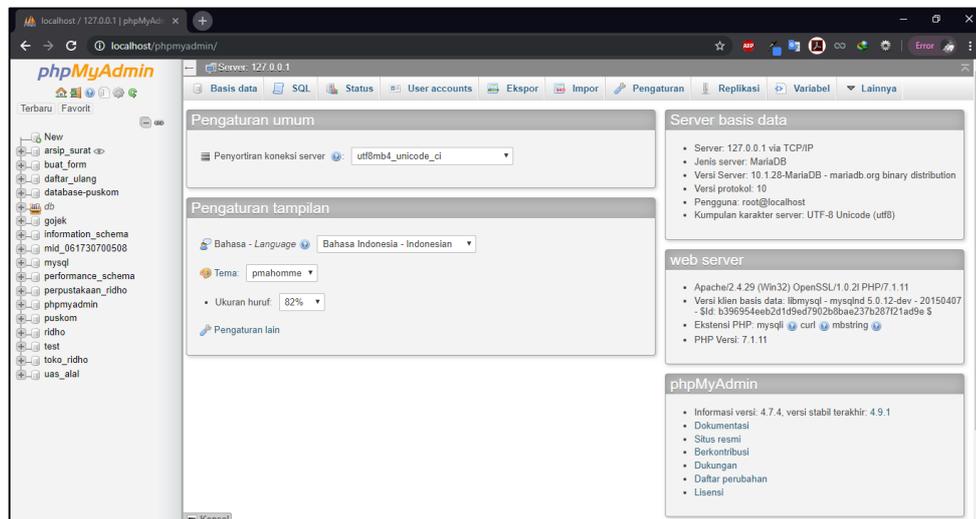
XAMPP ialah *software* yang di dalamnya terdapat *server* MySQL dan didukung oleh PHP sebagai bahasa pemrograman untuk membuat *website* dinamis serta terdapat *web server apache* yang dapat dijalankan di beberapa *platform* seperti OS X, Windows, Linux, Mac, dan Solaris (Putra, 2019:82).



Gambar 2.1 Tampilan *control panel* XAMPP

2.9. PHPMyAdmin

PHPMyAdmin adalah perangkat lunak bebas yang ditulis dalam bahasa pemrograman PHP yang digunakan untuk menangani administrasi MySQL. PHPMyAdmin mendukung berbagai operasi MySQL diantaranya mengelola basis data, tabel-tabel, bidang (*fields*), relasi (*relations*), indeks, pengguna (*users*), perijinan (*permissions*), dan lain-lain (Barri, dkk., 2015:25).



Gambar 2.2 Tampilan *dashboard* PHPMyAdmin

2.10. Apache

Menurut Muslim dan Dayana (2016:40-41) *Apache* adalah sebuah nama *web server* yang bertanggung jawab pada *request-response* HTTP dan *logging* informasi

secara detail. Selain itu, *Apache* juga diartikan sebagai suatu *web server* yang kompak, modukar, mengikuti standar protokol HTTP, dan tentu saja sangat digemari.

2.11. CSS

Cascading Style Sheets (CSS) adalah suatu bahasa *style sheet* yang digunakan untuk mengatur tampilan suatu dokumen yang ditulis dalam bahasa *markup*. Penggunaan yang paling umum dari CSS adalah untuk memformat halaman *web* yang ditulis dengan HTML dan XHTML. Walaupun demikian, bahasanya sendiri dapat dipergunakan untuk semua jenis dokumen XML termasuk SVG dan XUL. Spesifikasi CSS diatur oleh *World Wide Web Consortium* (W3C) (Puspitasari, 2016:228).

Menurut Muslim dan Dayana (2016) CSS singkatan dari *cascading style sheets*, berisi rangkaian instruksi yang menentukan bagaimana suatu *text* akan tertampil di halaman *web*. Perancangan desain *text* dapat dilakukan dengan mendefinisikan *fonts* (huruf), *colors* (warna), *margins* (ukuran), latar belakang (*background*), ukuran font (*font size*) dan lain-lain. Elemen-elemen seperti *colors* (warna), *fonts* (huruf), *sizes* (ukuran) dan *spacing* (jarak) disebut juga *styles*. *Cascading Style Sheets* juga bisa berarti meletakkan *styles* yang berbeda pada lapisan yang berbeda. CSS terdiri dari *style sheet* yang memberitahukan *browser* bagaimana suatu dokumen akan disajikan. Fitur-ditur baru pada halaman *web* lama dapat ditambahkan dengan bantuan *style sheet*. Saat menggunakan CSS, tidak perlu lagi untuk menulis *font*, *color* atau *size* pada setiap paragraf atau pada setiap dokumen.

2.12. JavaScript

Menurut Yani dan Saputra (2018:111) “*JavaScript* adalah bahasa yang digunakan untuk membuat program yang digunakan agar dokumen HTML yang ditampilkan pada sebuah *browser* menjadi lebih interaktif, tidak sekedar indah saja. *Javascript* memberikan beberapa fungsionalitas ke dalam halaman *web*, sehingga

dapat menjadi sebuah program yang disajikan dengan menggunakan antar muka web”.

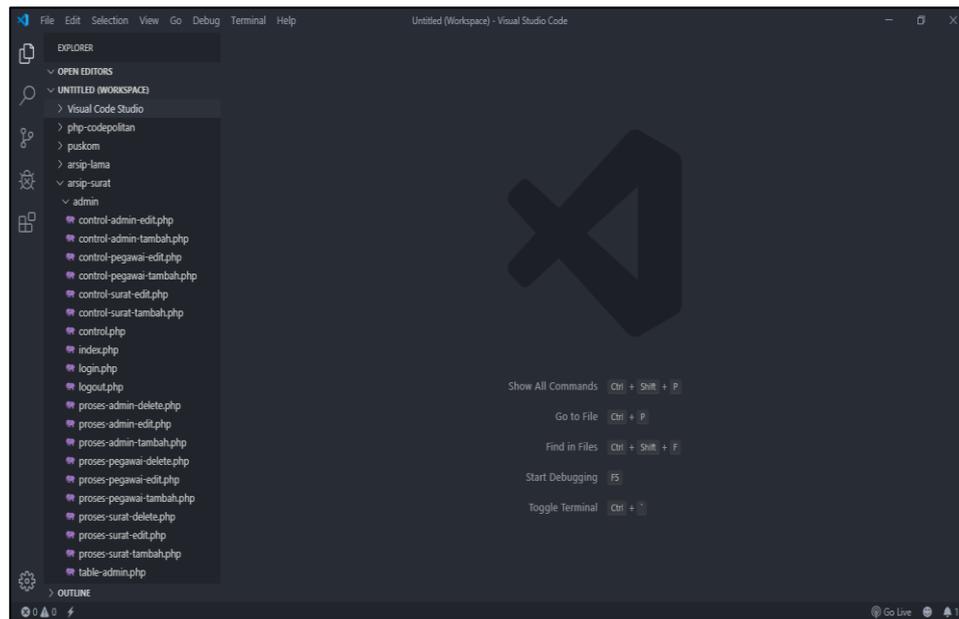
2.13. JQuery

Jquery adalah *open source add-on* pustaka *Javascript* yang menekankan pada interaksi antara *Javascript* dan HTML. *JQuery* merupakan kode *Javascript* yang telah ditulis dan tinggal menambahkan satu atau dua baris kode untuk memanggil *Jquery* (Pahlevi, dkk., 2018:28).

Jquery adalah kumpulan kode *Javascript* siap pakai. Keunggulan menggunakan *Jquery* dibandingkan dengan *Javascript* standar, yaitu menyederhanakan kode *Javascript* dengan cara memanggil fungsi-fungsi yang disediakan oleh *Jquery*. *Javascript* sendiri merupakan bahasa *scripting* yang bekerja disisi *client/browser* sehingga *website* bisa lebih interaktif (Lavarino dan Yustanti, 2016:75).

2.14. Visual Studio Code

Menurut Permana dan Romadlon (2019:155) *Visual Studio Code* (*VS Code*) ini adalah sebuah teks editor ringan dan handal yang dibuat oleh Microsoft untuk sistem operasi *multiplatform*, artinya tersedia juga untuk versi *Linux*, *Mac*, dan *Windows*. Teks editor ini secara langsung mendukung bahasa pemrograman *JavaScript*, *Typescript*, dan *Node.js*, serta bahasa pemrograman lainnya dengan bantuan *plugin* yang dapat dipasang via *marketplace Visual Studio Code* (seperti C++, C#, Python, Go, Java, dan seterusnya).



Gambar 2.3 Tampilan *dashboard* Visual Studio Code

2.15. Bootstrap

Menurut Pahlevi, dkk., (2018:28) *Bootstrap* adalah sebuah *framework* untuk CSS dan berupa produk *open source* yang dibuat oleh Mark Otto dan Jacob Thornton. Pada awalnya *bootstrap* ini dibuat untuk membuat standarisasi *front end* untuk semua *programmer* di perusahaannya. *Bootstrap* telah berubah dari yang sebelumnya CSS-*driven* proyek ke sebuah *host* dari JavaScript *plugins* dan ikon yang dapat dengan mudah digunakan untuk formulir dan tombol. *Bootstrap* merupakan *framework* ataupun *tools* untuk membuat aplikasi *web* ataupun situs *web responsive* secara cepat, mudah dan gratis. *Bootstrap* terdiri dari CSS dan HTML untuk menghasilkan *grid*, *layout*, *typography*, *table*, *form*, *navigation*, dan lain-lain. Didalam *Bootstrap* juga sudah terdapat *Jquery plugins* untuk menghasilkan komponen UI yang cantik seperti *transitions*, modal, *dropdown*, *scrollspy*, *tooltip*, *tap*, *popover*, *alert*, *button*, *carousel* dan lain-lain. Dengan bantuan *bootstrap*, kita bisa membuat *responsive website* dengan cepat dan mudah dan dapat berjalan sempurna pada *browser-browser* populer seperti chrome, firefox, safari, opera dan internet explorer (Masuara,dkk., 2015:44).

2.16. Metode Weighted Product

Menurut Khoir,dkk (2016:61-62) Metode *Weighted Product* (WP) merupakan perkalian untuk menghubungkan *rating* atribut, dimana *rating* setiap atribut harus dipangkatkan dengan bobot atribut yang bersangkutan. Proses tersebut sama halnya dengan proses normalisasi. Dalam penentuan nilai kepentingan atau bobot pada aplikasi Sistem Pendukung Keputusan sebagai alat bantu, pencarian nilai bobot atribut menggunakan penilaian secara subyektif yaitu penskalaannya dari 1 sampai 4 berdasarkan penilaian disesuaikan dengan tingkat sumbangan dari pengguna. Metode *weighted product* memerlukan proses normalisasi karena metode ini mengalikan hasil penilaian setiap atribut. Hasil perkalian tersebut belum bermakna jika belum dibandingkan (dibagi) dengan nilai standar. Ditunjukkan pada persamaan sebagai berikut:

$$S_i = \pi_{j=1}^n X_{ij}^{W_j}$$

Keterangan :

S : Menyatakan preferensi alternatif yang dianalogikan sebagai vektor S

X : Menyatakan nilai kriteria

W : Menyatakan bobot kriteria

i : Menyatakan alternatif

j : Menyatakan kriteria

n : Menyatakan banyaknya kriteria

Relatif dari setiap alternatif, ditunjukkan pada persamaan sebagai berikut:

$$V_i = \frac{\pi_{j=1}^n X_{ij}^{W_j}}{\pi_{j=1}^n (X_{ij}) W_j}$$

Keterangan :

V : Menyatakan preferensi alternatif yang dianalogikan sebagai vektor V

X : Menyatakan nilai kriteria

W : Menyatakan bobot kriteria

i : Menyatakan alternatif

j : Menyatakan kriteria

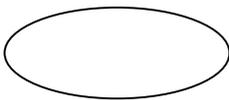
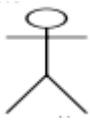
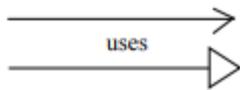
n : Menyatakan banyaknya kriteria

2.17. Use Case Diagram

Use Case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat (Rosa dan Shalahuddin, 2015).

Berikut pada Tabel 2.2 adalah simbol-simbol yang digunakan dalam *use case diagram* disertai dengan keterangan fungsinya.

Tabel 2.2 *Use Case Diagram*

No.	Simbol	Keterangan
1.	<p><i>Use Case</i></p> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit dan aktor.
2.	<p>Aktor</p> 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi.
3.	<p>Asosiasi</p> 	Komunikasi antar aktor dan <i>Use Case</i> yang berpartisipasi.
4.	<p><i><<extend>></i></p> 	Relasi <i>Use Case</i> tambahan ke sebuah <i>Use Case</i> dimana <i>Use Case</i> yang ditambah dapat berdiri sendiri walau tanpa <i>Use Case</i> tambahan.
5.	<p>Generalisasi/generalization</p> 	Hubungan generalisasi dan spesialisasi antara dua buah <i>Use Case</i> yang mana fungsi yang satu lebih umum dari yang lainnya.
6.	<p><i><<include>></i></p> 	Relasi <i>Use Case</i> tambahan ke sebuah <i>Use Case</i> dimana <i>Use Case</i> yang ditambahkan memerlukan <i>Use Case</i> ini untuk menjalankan fungsinya.

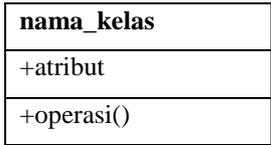
(Sumber: Rosa dan Shalahuddin, 2015)

2.18. Class Diagram

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi definisi kelas-kelas yang akan dibuat untuk membangun sistem. *Class diagram* dibuat agar pembuat program atau *programmer* membuat kelas-kelas sesuai rancangan *class diagram* agar antara dokumentasi perancangan dan perangkat lunak sinkron (Rosa dan Shalahuddin, 2018:142).

Berikut pada Tabel 2.3 adalah simbol-simbol yang digunakan dalam *class diagram* disertai dengan keterangan fungsinya:

Tabel 2.3 Simbol-simbol *Class Diagram*

No.	Simbol	Keterangan
1.	<p>Kelas</p> 	Merupakan kelas pada struktur sistem.
2.	<p>Antarmuka/<i>Interface</i></p>  <p>nama_interface</p>	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
3.	<p>Asosiasi</p> 	Merupakan relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
4.	<p>Asosiasi Berarah</p> 	Merupakan relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
5.	<p>Generalisasi/<i>Generalization</i></p> 	Merupakan relasi antarkelas dengan makna generalisasi-spesialisasi (umum-khusus).
6.	<p>Kebergantungan/<i>Dependency</i></p> 	Merupakan relasi antarkelas dengan makna kebergantungan antarkelas.

7.	Agregasi/ <i>Aggregation</i> 	Merupakan relasi antarkelas dengan makna semua-bagian (<i>whole-part</i>).
----	---	--

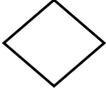
(Sumber: Rosa dan Shalahuddin, 2018:146-147)

2.19. Activity Diagram

Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Diagram aktivitas berfungsi untuk menggambarkan aktivitas sistem bukan yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa dan Shalahuddin, 2018:161).

Berikut pada Tabel 2.4 adalah simbol-simbol yang digunakan dalam *activity diagram* disertai dengan keterangan fungsinya:

Tabel 2.4 Simbol-simbol *Activity Diagram*

No.	Simbol	Keterangan
1.	Status Awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki status awal.
2.	Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3.	Percabangan/ <i>Decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4.	Penggabungan/ <i>Join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
5.	Status Akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

6.	<p style="text-align: center;"><i>Swimlane</i></p> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.
----	--	--

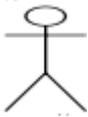
(Sumber : Rosa dan Shalahuddin, 2018:162-163)

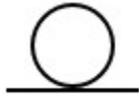
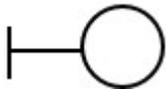
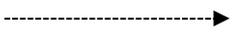
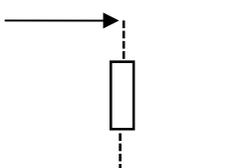
2.20. Sequence Diagram

Diagram sekuen atau *sequence diagram* merupakan diagram yang menggambarkan kelakuan dari objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek (Rosa dan Shalahuddin, 2018:165).

Berikut pada Tabel 2.5 adalah simbol-simbol yang digunakan dalam *sequence diagram* disertai dengan keterangan fungsinya:

Tabel 2.5 Simbol-simbol *Sequence Diagram*

No.	Simbol	Keterangan
1.	<p style="text-align: center;">Garis/<i>Lifeline</i></p> 	Menyatakan kehidupan suatu objek.
2.	<p style="text-align: center;">Objek</p> <div style="border: 1px solid black; padding: 2px; width: fit-content; margin: auto;"> <u>nama objek : nama</u> <u>kelas</u> </div>	Menyatakan objek yang berinteraksi pesan.
3.	<p style="text-align: center;">Aktor</p>  <p style="text-align: center;">nama aktor</p> <p style="text-align: center;">atau</p> <div style="border: 1px solid black; padding: 2px; width: fit-content; margin: auto;"> <u>nama aktor</u> </div> <p style="text-align: center;">Tanpa waktu aktif</p>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
4.	<p style="text-align: center;">Waktu aktif</p> 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu

		aktif ini adalah sebuah tahapan yang dilakukan di dalamnya.
5.	<p><i>Entity Class</i></p> 	Digunakan untuk menggambarkan hubungan kegiatan yang akan dilakukan.
6.	<p><i>Boundary Class</i></p> 	Digunakan untuk menggambarkan sebuah <i>form</i> .
7.	<p><i>Control Class</i></p> 	Digunakan untuk menghubungkan <i>boundary</i> dengan tabel.
8.	<p>Pesan tipe <i>create</i></p> <p><<create>></p> 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.
9.	<p>Pesan tipe <i>call</i></p> <p>1 : nama_metode()</p> 	Menyatakan suatu objek memanggil operasi atau metode yang ada pada objek lain atau dirinya sendiri. Arah panah mengarah pada objek yang memiliki operasi atau metode, karena ini memanggil operasi atau metode maka operasi atau metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi.
10.	<p>Pesan tipe <i>send</i></p> <p>1 : masukan</p> 	Menyatakan bahwa suatu objek mengirimkan data atau masukan atau informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
11.	<p>Pesan tipe <i>return</i></p> <p>1 : keluaran</p> 	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.
12.	<p>Pesan tipe <i>destroy</i></p> 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i> .

(Sumber : Rosa dan Shalahuddin, 2018:165-167)