

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Sistem Pakar merupakan suatu kecerdasan buatan yang berisi ilmu pengetahuan dari satu atau lebih pakar mengenai suatu bidang spesifik. Penelitian yang dilakukan oleh peneliti terdahulu menjadi referensi atau rujukan penelitian selanjutnya agar penelitian lebih berkembang.

Penelitian yang dilakukan oleh Demas Matias L Tobing, Elvis Pawan, Friden E Neno dan Kusriani, dalam jurnal yang berjudul “Sistem Pakar Mendeteksi Penyakit Pada Tanaman Padi Menggunakan Metode *Forward chaining*” (2019). Masalah yang diangkat pada penelitian tersebut adalah penyakit pada tanaman padi, penyebab dan solusinya. Data yang digunakan pada penelitian ini adalah data gejala, data penyakit dan *rules*. Metode yang digunakan yaitu Metode *Forward chaining*. Dilakukan *crosscheck* sebanyak 15 kali percobaan dengan hasil persentase akhir 93% [11].

Penelitian selanjutnya dilakukan oleh Putri Taqwa Prasetyaningrum dan Juwita, dalam jurnal yang berjudul “Implementasi Sistem Pakar Berbasis *Website* Untuk Mengidentifikasi Hama Tanaman Padi Beserta Penanganannya” (2020). Masalah yang diangkat pada penelitian tersebut adalah pendeteksian hama pada tanaman padi dan cara penanganannya. Data yang digunakan pada penelitian ini adalah data kumpulan hama tanaman padi dan gejalanya. Metode yang digunakan yaitu Theorema Bayes untuk menentukan hama tanaman padi dan *Forward chaining* untuk menentukan penanganan hama yang telah terdeteksi. Dilakukan pengujian dengan 20

sampel data dengan hasil persentase akhir 85% , dengan demikian *Theorema Bayes* dan Algoritma *Forward chaining* dapat digunakan untuk Mengidentifikasi Hama Pada Tanaman Padi Beserta Penanganannya [14].

Penelitian selanjutnya dilakukan oleh Sri Wulandari, Muhammad Fajrian Noor, Ajie Kusuma Wardhana dan Kusrini, dalam jurnal yang berjudul “Sistem Pakar Diagnosis Hama Dan Penyakit Tanaman Padi Dengan Metode *Bayes*” (2019). Masalah yang diangkat pada penelitian tersebut adalah diagnosis hama dan penyakit pada tanaman padi. Data yang digunakan pada penelitian ini adalah data gejala hama dan penyakit pada batang, daun, bulir, dan malai. Metode yang digunakan pada penelitian ini yaitu *Theorema Bayes*. Dilakukan Uji coba pertanyaan dengan hasil akhir tanaman padi terserang hama dan penyakit bercak coklat mempunyai *nilai bayes* sebesar 68,28%. Dengan demikian *Theorema Bayes* bisa digunakan untuk Diagnosis Hama Dan Penyakit Tanaman Padi [15].

Penelitian Huzainsyahnoor Aksad dan Fitri Aditiya, dalam jurnal yang berjudul “Model Sistem Pakar Diagnosis Penyakit Tanaman Padi Menggunakan Metode *Certainty factor*” (2019). Masalah yang diangkat pada penelitian tersebut adalah diagnosis penyakit tanaman padi. Data yang digunakan adalah data gejala, data penyakit pada padi, dan data konsultasi dari para pakar. Metode penelitian ini yaitu *Certainty factor*. Dilakukan pengujian terhadap 25 sampel data dengan hasil persentase sebesar 100%. Dengan demikian Metode *Certainty factor* bisa digunakan untuk Diagnosis Penyakit Tanaman Padi [16].

Penelitian Dianmita Ayu Putri dan Arik Aranta, yang berjudul “Sistem Pakar Diagnosis Penyakit Tanaman Padi Menggunakan *Forward chaining* Dan *Dempster Shafer*” (2020). Masalah yang diangkat pada penelitian tersebut adalah diagnosis penyakit tanaman padi. Data yang digunakan adalah data gejala, data penyakit, dan data penanggulangan penyakit. Metode yang digunakan yaitu *Forward chaining* dan *Dempster Shafer*. Dilakukan

pengujian terhadap 13 sampel data dengan hasil *presentase* sebesar 81,11%. Dengan demikian *Forward chaining* dan *Dempster Shafer* bisa digunakan untuk Diagnosis Penyakit Tanaman Padi [17].

Tabel 2.1 Penelitian Terdahulu

No	Peneliti dan Tahun Penelitian	Judul Penelitian	Masalah	Metode Penelitian	Hasil
1.	Dema Matias L Tobing, Elvis Pawan, Friden E Neno dan Kusrini (2019).	Sistem Pakar Mendeteksi Penyakit Pada Tanaman Padi Menggunakan Metode <i>Forward chaining</i>	Banyaknya petani yang masih tidak tau bagaimana cara mengelola dan merawat tanaman padi agar memberikan hasil yang melimpah. Hal yang biasanya menjadi kendala yaitu kurangnya pengetahuan dalam penanganan penyakit pada padi yang menjadi sumber gagal panen yang membuat petani merugi. Dengan adanya penelitian ini diharapkan memberikan sebuah solusi yang baru kepada para petani dengan menggunakan sistem pakar agar petani dapat segera mencegah penyebaran penyakit atau hama tanaman	Metode <i>Forward chaining</i>	Dari hasil pengujian yang telah dilakukan menggunakan sistem pakar, 15 kasus yang berbeda dilapangan selanjutnya di <i>cross check</i> dengan hasil Analisa pakar dan memiliki kesesuaian sebanyak 93%. Dengan presentase tersebut bisa dikatakan metode <i>Forward chaining</i> cukup akurat untuk diimplementasikan pada sistem.

No	Peneliti dan Tahun Penelitian	Judul Penelitian	Masalah	Metode Penelitian	Hasil
2.	Putri Taqwa Prasetyaningrum dan Juwita (2020).	Implementasi Sistem Pakar Berbasis <i>Website</i> Untuk Mengidentifikasi Hama Tanaman Padi Beserta Penanganannya	Dampak hama padi seperti tikus, wereng, walang sangit dan hama lainnya sering dirasakan oleh petani, yang paling parah adalah terjadinya gagal panen. Dari permasalahan diatas, maka diperlukan suatu sistem secara komputerisasi akan membantu dalam pengambilan keputusan yang lebih mudah dan cepat, oleh karena itu perlu adanya penelitian untuk mengakusisi kemampuan pakar secara komputerisasi.	<i>Theorema Bayes</i> dan <i>Forward chaining</i>	Dilakukan pengujian dengan <i>sampel</i> sebanyak 20 data, hasil presentase yang diperoleh sebesar 85%, dengan presentase tersebut bisa dikatakan bahwa metode <i>teorema bayes</i> dan <i>forward cahining</i> dapat digunakan untuk membantu dalam identifikasi hama padi dan penanganannya
3.	Sri Wulandari, Muhammad Fajrian Noor, Ajie Kusuma Wardhana dan Kusrini (2019).	Sistem Pakar Diagnosis Hama Dan Penyakit Tanaman Padi Dengan Metode <i>Theorema Bayes</i>	Banyaknya petani yang masih salah dalam mendiagnosis jenis hama dan penyakit yang menyerang tanaman padi mereka membuat para petani salah dalam melakukan penanganan yang tepat. Dalam hal ini	<i>Theorema Bayes</i>	Hasil uji coba pada sistem menunjukkan bahwa tanaman padi terserang hama dan penyakit bercak coklat mempunyai <i>nilai bayes</i> sebesar 68,28%. Dengan presentase tersebut bisa dikatakan

No	Peneliti dan Tahun Penelitian	Judul Penelitian	Masalah	Metode Penelitian	Hasil
			pendapat para ahli dibutuhkan sebagai pemecahan masalah, oleh karenanya dibangunlah sebuah sistem pakar untuk membantu para ahli dalam memberikan saran yang tepat terhadap masalah yang dikeluhkan terkait masalah hama dan penyakit pada padi.		<i>Theorema Bayes</i> cukup akurat untuk diimplementasikan pada sistem.
4.	Huzainsyahnoor Aksad dan Fitri Aditiya (2019).	Model Sistem Pakar Diagnosis Penyakit Tanaman Padi Menggunakan Metode <i>Certainty Factor</i>	Kurang pengetahuan para petani dalam hal mengatasi hama dan penyakit tanaman padi, sehingga dibutuhkan bantuan orang ahli pertanian untuk mengatasinya, maka digunakanlah metode teorema bayes untuk mengidentifikasi penyakit tanaman padi dengan menggunakan gejala tanaman sebagai parameternya.	<i>Certainty factor</i>	Hasil diagnosis sistem pakar menunjukkan dari 25 sampel, seluruh sampel diagnosis sama dengan pendapat pakar, sehingga dapat dikatakan bahwa sistem pakar dengan metode <i>certainty factor</i> memiliki tingkat akurasi yang tinggi yaitu 100% .

No	Peneliti dan Tahun Penelitian	Judul Penelitian	Masalah	Metode Penelitian	Hasil
5.	Dianmita Ayu Putri dan Arik Aranta (2020).	Sistem Pakar Diagnosis Penyakit Tanaman Padi Menggunakan <i>Forward chaining</i> Dan <i>Dempster Shafer</i>	Salah satu penghambat pertumbuhan produksi panen di Indonesia adalah penyakit tanaman padi. Penyakit padi menjadi penyebab kegagalan produksi atau gagal panen yang dialami petani padi. Perlunya pengetahuan dan penanganan yang tepat dari seorang ahli dalam bidangnya untuk memecahkan masalah.	<i>Forward chaining</i> dan <i>Dempster Shafer</i>	Data penyakit yang diuji yaitu sebanyak 13 <i>sampel</i> , dengan pengujian yang dilakukan pada percobaan sistem sebanyak 30 kali, hasil akhir persentasenya yaitu 81,11%, bisa dikatakan gabungan antara Metode <i>Forward chaining</i> dan <i>Dempster Shafer</i> cukup akurat dan bisa diimplementasikan pada sistem.

2.2 Landasan Teori

Landasan Teori Berisi beberapa pengetahuan yang berkaitan dengan penelitian ini untuk membantu peneliti dalam melakukan penelitian, berikut adalah landasan teori yang berkaitan dengan penelitian ini.

2.2.1. Pengertian Penyakit Pada Padi dan Jenis-jeni Penyakit Pada Padi

Penyakit pada tanaman padi biasanya disebabkan oleh Bakteri dan Jamur. Penyakit pada padi biasanya menyerang batang, daun, dan biji padi. Penyakit pada padi yang diakibatkan oleh bakteri salah satunya adalah penyakit Blas (*pyricularia grisea*). Penyakit blas dapat menyerang tanaman padi pada semua stadia pertumbuhan dan beresiko menyebabkan kegagalan panen. Gejala penyakit ini membentuk bercak pada daun, leher malai, dan cabang malai. Bentuk khas dari bercak Blas yaitu elips dan runcing pada kedua ujungnya [18]. Penyakit pada padi yang diakibatkan oleh jamur adalah penyakit layu daun yang diduga disebabkan oleh jamur anggota spesies *Fusarium sp.* Ditandai dengan daun mulai menguning dan daun terpelintir, walaupun terkadang tidak terlalu kelihatan dan pangkal batang membusuk [19].

2.2.2. Pengertian Hama dan Jenis-Jenis Hama

Hama adalah organisme yang merugikan dan tak diinginkan. Dalam pertanian, hama adalah organisme pengganggu tanaman yang menimbulkan kerusakan secara fisik, dan semua hewan yang menyebabkan kerugian dalam pertanian.

Ditemukan 12 jenis hama pada habitat tanaman padi sawah di Kelurahan Makalonsow Kecamatan Tondano Timur Kabupaten Minahasa [20].

1. Penggerek batang padi putih (*Tryporyza innotata*)
2. Penggerek Batang Padi Bergaris (*Chilo suppressalis*)
3. Penggerek Batang Padi Ungu (*Sesamia inferens*)
4. Hama Putih (*Nymphula depunctalis*)
5. Wereng Coklat (*Nephotettix virescens*)

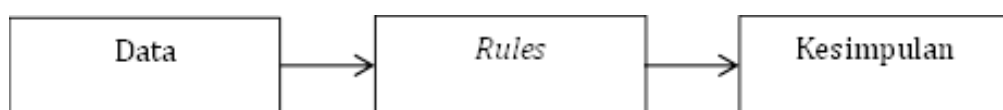
6. Wereng Hijau (*Nilaparvata lugens*)
7. Walang Sangit (*Leptocorisa acuta*)
8. Kepik Hitam (*Pareaucosmetus sp*)
9. Bubuk Beras (*Sitophilus oryzae*)
10. Keong Emas (*Pomacea caniculata*)
11. Hama Burung Padi Sawah (*Passer spp*)
12. Hama Tikus Sawah (*Ratus argentiventer*)

2.2.3. Pengertian Sistem Pakar

Sistem Pakar merupakan suatu kecerdasan buatan yang berisi ilmu pengetahuan dari satu atau lebih pakar mengenai suatu bidang spesifik. Sistem pakar digunakan sebagai media untuk membantu pakar dalam memecahkan masalah berbasis *Android* atau *Website*, dalam ranah pertanian sendiri sistem pakar berguna sebagai pemecah masalah yang dialami oleh petani. Dengan adanya sistem pakar petani bisa dengan mudah menemukan solusi terbaik dengan bantuan dari pakar bidang pertanian, tanpa harus konsultasi secara langsung cukup dengan menggunakan *Gadget* atau Laptop [21].

2.2.4. Pengertian Metode *Forward chaining*

Dalam sistem pakar, metode *Forward chaining* merupakan metode yang melakukan pelacakan ke depan, dengan melihat fakta yang dieksekusi pada *rules* dan berakhir di kesimpulan. Hasil dari pemrosesan fakta akan menghasilkan fakta baru yang digunakan untuk melanjutkan proses dan mendapatkan kesimpulan akhir setelah tidak ada lagi aturan yang premisnya cocok dengan fakta. Penerapan metode *Forward chaining* yaitu hasil kesimpulan yang diketahui akan di *training* dalam suatu sistem pakar. [22]. Untuk lebih Jelasnya berikut alur dari metode *Forward chaining* dapat dilihat pada Gambar 2.1 *Rules Forward chaining*.



Gambar 2.1 Cara 1 *Forward chaining*

Contoh penerapan *Forward chaining* :

Mencari jenis hama dan penyakit yang menyerang pada tanaman padi, data awal adalah Daun Mengering dan hawar memanjang. Terdapat 4 aturan :

Rules 1:

If Daun Mengering *Then* Batang Membusuk

If Batang Membusuk *Then* Penyakit Layu Daun

If Bercak Abu Pada Tepi Daun *And* Daun Mengering *Then* Hawar Memanjang

If Hawar Memanjang *Then* Penyakit Kresek

Rules 2:

R1 = *If* A *Then* C

R2 = *If* C *Then* L

R3 = *If* (B *And* A) *Then* D

R4 = *If* D *Then* K

Fakta-fakta :

A dan D bernilai benar

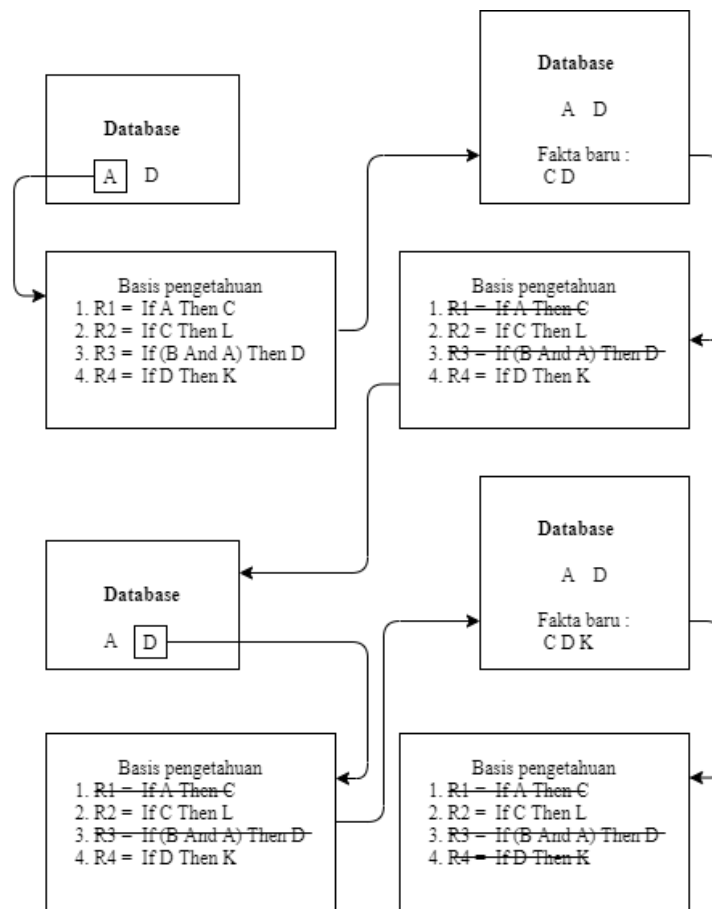
L bernilai Salah

Goal : menentukan apakah K bernilai benar atau salah?

Cara 1 :

Pertama akan mencari aturan 1 dan 3 (sesuai dengan tujuan kita mencari Daun Mengering), maka kedua anteseden (*If* daun Mengering *Then* Batang Membusuk, *If* Bercak Abu Pada Tepi Daun *and* Daun Mengering *Then* Hawar Memanjang) ditambahkan ke tujuan. Konsekuen (*Then* Hawar Memanjang) ditambahkan ke data yang dimiliki. *If* Hawar Memanjang *Then* Penyakit Kresek (Tujuan).

Cara 2:



Gambar 2.2 Cara 2

Berikut alur pencarian menggunakan cara 2 *forward chaining*:

1. Memeriksa fakta A dengan *rules* yang ada pada basis pengetahuan, lalu fakta C dan D ditambahkan ke fakta baru pada *database*. Karena pada R1 dan R3 terdapat A dengan hasil C dan D.
2. Setelah fakta A diperiksa selanjutnya memeriksa fakta D dengan *rules* yang ada pada basis pengetahuan. Lalu fakta K ditambahkan ke fakta baru pada *database*. Karena R4 terdapat D dengan hasil K.
3. Proses dihentikan karena sudah tidak ada lagi *rules* yang bisa dieksekusi. Hasil pencarian adalah K bernilai benar (lihat *database* di bagian fakta baru).

2.2.5 Pengertian Metode *Certainty factor*

Metode *Certainty factor* menunjukkan ukuran kepastian terhadap suatu fakta atau aturan dengan menghitung probabilitasnya. *Certainty factor* merupakan nilai parameter yang diberikan *MYCIN* untuk menunjukkan tingkat kepercayaan terhadap hasil diagnosis. Kelebihan dari metode *Certainty factor* adalah dapat mengukur tingkat kepastian dalam pengambilan keputusan pada sistem pakar diagnosis penyakit [8]. Rumus dasar *Certainty factor* :

$$CF(h, e) = MB(h, e) - MD(h, e) \dots\dots\dots(1)$$

$$MB(h,e) = \left\{ \frac{\max[P(h|e),P(h)]-P(h)}{\max(1,0)-P(h)} \right\} \dots\dots\dots(2)$$

$$MD(h,e) = \left\{ \frac{\min[P(h|e),P(h)]-P(h)}{\min(1,0)-P(h)} \right\} \dots\dots\dots(3)$$

Keterangan:

CF(h,e) : *Certainty factor* (faktor keyakinan) dalam hipotesis h dipengaruhi evidence e.

MB(h,e) : *Measure of Belief* (tingkat keyakinan), merupakan ukuran keyakinan dari hipotesis h dipengaruhi oleh *evidence* (gejala) e.

MD(h,e) : *Measure of Disbelief* (tingkat ketidak yakinan), merupakan ukuran ketidak yakinan dari hipotesis h dipengaruhi oleh *evidence* (gejala) e.

P(h) : probabilitas kebenaran hipotesis h.

P(h|e) : probabilitas bahwa h benar dengan fakta e.

Rumus kombinasi dua atau lebih *rule* dengan *evidence* berbeda tetapi dalam hipotesis yang sama:

$$Rule\ 1\ CF(h, e1) = CF1 = C(e1) \times CF(Rule1) \dots\dots\dots(4)$$

$$Rule\ 2\ CF(h, e2) = CF2 = C(e2) \times CF(Rule2) \dots\dots\dots(5)$$

$$CF_{kombinasi}\ [CF1, CF2] = CF1 + CF2 (1 - CF1) \dots\dots\dots(6)$$

Tabel 2.2 Interpretasi Term Menjadi Nilai CF

<i>Uncertain Term</i>	CF
<i>Definetely not</i> (pasti tidak)	-1.0
<i>Almost certainly not</i> (hampir pasti tidak)	-0.8

<i>Probably not</i> (kemungkinan besar tidak)	-0.6
<i>Maybe not</i> (mungkin tidak)	-0.4
<i>Unknown</i> (tidak tahu)	-0.2 sampai 0.2
<i>Maybe</i> (mungkin)	0.4
<i>Probably</i> (kemungkinan benar)	0.6
<i>Almost certainly</i> (hampir pasti)	0.8
<i>Definetely</i> (pasti)	1.0

Contoh penerapan :

Diketahui probabilitas tanaman padi terkena penyakit kresek adalah 0.2 dan dikatakan bahwa memiliki gejala hawar memanjang. Dengan menganggap $h =$ penyakit kresek dan $e =$ hawar memanjang. Dan diketahui $P(h|e)$ adalah 1.0, hitung faktor kepastian bahwa penyakit kresek memiliki gejala hawar memanjang.

Jawab :

$P(\text{penyakit kresek}) : 0.2$

$P(\text{kresek|hawar memanjang}) : 1.0$

$$MB(h,e) = \left\{ \frac{\max[1.0,0.2]-0.2}{\max(1,0)-0.2} = \frac{1.0-0.2}{1-0.2} = 1 \dots\dots\dots(7) \right.$$

$$MD(h,e) = \left\{ \frac{\min[1.0,0.2]-0.2}{\min(1,0)-0.2} = \frac{0.2-0.2}{0-0.2} = 0 \dots\dots\dots(8) \right.$$

$$CF = 1-0 = 1$$

Rules IF (gejala hawar memanjang) *THEN* penyakit = Kresek
(CF=1).

Hasil CF adalah 1 merujuk pada Tabel 2.2 Interpretasi Term Menjadi Nilai CF bahwa nilai CF 1 adalah pasti maka dapat disimpulkan bahwa penyakit kresek memiliki gejala hawar memanjang.

2.2.6 XAMPP

XAMPP ialah perangkat lunak bebas yang mendukung banyak sistem operasi, merupakan campuran dari beberapa program. Yang mempunyai fungsi sebagai *server* yang berdiri sendiri (*localhost*), yang terdiri dari program *MySQL database*, *Apache HTTP Server*, dan penerjemah ditulis dalam bahasa pemrograman *PHP* dan *Perl*. Program ini tersedia di bawah *GNU General Public License* dan bebas menggunakan *web server* yang dapat melayani tampilan halaman *web* yang *dinamis*.

Mengenal bagian *XAMPP* yang biasa digunakan pada umumnya: *htdocs* adalah folder tempat meletakkan berkas-berkas yang akan dijalankan, seperti berkas *PHP*, *HTML* dan *skrip* lain. *phpMyAdmin* merupakan bagian untuk mengelola basis data *MySQL* yang ada dikomputer. Kontrol Panel yang berfungsi untuk mengelola *layanan service XAMPP*. *XAMPP* merupakan *tool* yang menyediakan paket perangkat lunak ke dalam satu buah paket. Dengan *menginstall XAMPP* maka tidak perlu lagi melakukan *instalasi* dan *konfigurasi web server Apache*, *PHP* dan *MySQL* secara manual. *XAMPP* akan menginstalasi dan mengkonfigurasikannya secara otomatis atau *auto konfigurasi* [23].

2.2.7 Visual Studio Code

Visual Studio Code (*VS Code*) merupakan sebuah teks editor ringan untuk sistem operasi *multiplatform*, tersedia *versi Linux*, *Mac*, dan *Windows*. Teks editor ini mendukung bahasa pemrograman *JavaScript*, *Typescript*, dan *Node.js*, serta bahasa pemrograman lainnya dengan bantuan plugin yang dapat dipasang melalui *marketplace Visual Studio*.

Teks editor *VS Code* bersifat *open source*, yang mana kode sumbernya dapat dilihat dan dapat berkontribusi untuk pengembangannya. Kode sumber dari *VS Code* dapat dilihat di *link Github* dan *Gitlab*. Hal ini yang membuat *VS Code* menjadi favorit para pengembang aplikasi, karena para pengembang aplikasi bisa ikut serta dalam proses pengembangan *VS Code* ke depannya [24].

2.2.8 Bahasa Pemrograman *PHP*

PHP adalah bahasa pemrograman dasar yang juga dapat digunakan untuk mengembangkan halaman web dinamis. Inti *PHP* bekerja tanpa perpustakaan atau *built-in fungsionalitas* dan biasanya pengembang harus membuat *skrip* kode menggunakan keterampilan dan logika mereka.

Di sisi lain, kerangka kerja *PHP* adalah kerangka kerja yang terdiri dari kumpulan kode sumber terorganisir dalam sebuah arsitektur tertentu untuk mendukung perkembangan dinamis aplikasi dan layanan situs web.

Alat Aplikasi Web adalah sebuah kerangka kerja yang biasanya didistribusikan bersama dengan alat yang mendukung pengembangan elemen kode sumber. Alat-alatnya dapat mempercepat proses pengembangan sumber kode aplikasi, mengurangi beban kerja programmer dan mengurangi kesalahan pemrograman [25]. Alat-alat yang mendukung pola *MVC* harus memiliki yang elemen berikut:

1. Model, pola *MVC* mewakili masalah tunduk pada logika bisnis di mana modelis digunakan untuk pemetaan tabel.
2. *CRUD*, (Buat, Baca, Perbarui, dan Hapus catatan) alat memungkinkan untuk melakukan operasi dasar pada catatan *database*.
3. *Controller*, kelas bertanggung jawab atas fungsionalitas pengontrol dalam pola *MVC*.
4. Tampilan, elemen pola *MVC*.
5. Bentuk, kelas tunggal atau kelompok kelas bertanggung jawab untuk merender dan memvalidasi logika dari data masuk.
6. *Modul*, fungsi terpisah dan memastikan portabilitas kode sumber antar proyek.

2.2.9 *MySQL Workbench*

MySQL Workbench sebuah *software* yang di desain untuk keperluan *database* yang terintegrasi dengan *SQL Development, administration, database design, creation* dan *maintance*. Menggunakan *mysql workbench* memudahkan

pengguna dalam pembuatan *basisdata* sistem dengan tampilan *GUI (Graphical User Interface)* sehingga komponen visual interkatif dan *user friendly* [26].

2.2.10 Bahasa Pemrograman HTML

Html merupakan bahasa pemrograman yang digunakan untuk membangun halaman *website*. *Html* digunakan untuk membuat elemet teks atau gambar. *Html* menyediakan struktur dasar halaman dan menggunakan *css* sebagai desain tampilannya. *Html* bisa dikombinasikan dengan bahasa pemrograman lain dalam pembuatan *website* sehingga hasil *website* yang dibuat lebih *interaktif* [27].

2.2.11 Cascading Style Sheet (CSS)

Css merupakan sebuah *style sheet* yang digunakan untuk mengatur tampilan halaman *website* mulai dari warna, *font*, hingga tata letak *layout*. *Css* biasanya dikombinasikan dengan *html* atau bahasa pemrograman lain untuk membuat sebuah *website*. Adanya *css* membantu pengembang *web* dalam mendesain halaman *website* dengan cepat. Proses pengkodean singkat dan sederhana sehingga pembukaan halaman *web* lebih cepat dan *css* bersifat adaptif sehingga *css* kompatibel diberbagai perangkat atau *browser* [28].

2.2.12 Bahasa pemrograman Javascript

Javascript merupakan bahasa pemrograman yang digunakan untuk membangun *website* dari sisi *client (client side)*. *Javascript* memungkinkan tampilan *website* menjadi dinamis dan interaktif. *Javascript* merupakan bahasa tingkat tinggi karena memiliki aturan penulisan yang menyerupai bahasa manusia. *Javascript* ditulis secara dinamis, satu variable pada program dapat berisi tipe data yang berbeda. *Javascript* harus diterjemahkan kedalam bahasa computer sebelum dijalankan dan dilakukan otomatis oleh *web browser* [29].

2.2.13 *Framework*

Framework merupakan sebuah kerangka kerja yang biasanya digunakan untuk membantu pengembangan sebuah *website*. *Framework* tercipta untuk membantu para *web developer* untuk membuat tampilan *website* dan *database* dari *website* [30]. Terdapat berbagai macam jenis *framework* yang digunakan sebagai pengembangan *website* dua diantaranya yang sering digunakan *web developer* dalam pengembangan *website* yaitu *Bootstrap* dan *Laravel*.

Bootstrap merupakan *framework* dari *css* yang digunakan sebagai pembuatan *front end* sebuah *website*. *Bootstrap* menyediakan berbagai *template user interface* untuk *login*, hingga tampilan *admin* untuk *website* [31]. *Laravel* merupakan *template* yang biasa digunakan *back end* untuk mengembangkan *website*. *Laravel* menggunakan bahasa pemrograman *php* yang banyak digunakan oleh *programmer*, migrasi data yang mudah, efisien dan hemat biaya. Oleh karenanya, *laravel* banyak digunakan *web developer* dalam mengembangkan *website* [32].

2.2.14 *Blackbox Testing*

Blackbox testing merupakan metode pengujian pada fungsionalitas atau kegunaan dari *software*. Pengujian dilakukan untuk menilai kebutuhan spesifikasi dari *software*. *Blackbox testing* hanya menjangkau *input* dan *output* sistem *software* tanpa melihat kode dari internal program. Terdapat tiga jenis pengujian menggunakan *blackbox testing* yaitu *functional testing*, *non-functional testing*, dan *regression testing* [33].

Functional testing digunakan untuk pengujian fitur spesifik dari *software*. *Non-functional testing* digunakan untuk mengukur kemampuan *software* dalam menjalankan perintah. *Regression testing* digunakan untuk mengukur kinerja aplikasi setelah proses *upgrade*.

2.2.15 *Unified Modeling Language (UML)*

Uml merupakan pemodelan sistem yang memudahkan programmer dalam memahami dan menganalisa program yang akan dibuat. *uml* pemodelan secara visual yang digunakan sebagai perancangan sistem berorientasi objek [34]. *Uml* memiliki 13 jenis diagram yang memiliki peranan masing-masing:

1. *Use case diagram*, abstraksi dari interaksi sistem dan aktor.
2. *Activity diagram*, alur kerja model atau logika keputusan yang terkandung dalam metode individual dan paralel.
3. *Package diagram*, mengelompokkan elemen diagram *uml* yang berbeda kedalam sebuah paket.
4. *State diagram*, penggolongan satu atau lebih nilai *attribute* pada kelas.
5. *Sequence diagram*, tahapan yang menghasilkan *use case diagram*.
6. *Class diagram*, pendeskripsian *property*, operasi, dan relasi.
7. *Communication diagram*, gambaran interaksi antar objek yang menekankan peranan masing-masing objek.
8. *Composite structure diagram*, menunjukkan struktur internal *classifier* sebagian lain dari sistem.
9. *Object diagram*, gambaran objek dalam sebuah sistem pada satu titik waktu.
10. *Timing diagram*, bentuk khusus dari *sequence diagram*.
11. *Component diagram*, visualisasi komponen sistem dan hubungan antar komponen.
12. *Deployment diagram*, visualisasi hubungan antara *software* dan *hardware*.
13. *Interaction overview diagram*, visualisasi kerjasamayang terjadi antara *sequence diagram* dengan *activity diagram*.