

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Sebelumnya

Pada bab ini, dilakukan studi literatur berupa pencarian sebuah jurnal penelitian yang telah dilakukan untuk mencari informasi yang relevan dengan penelitian supaya jurnal tadi dapat digunakan sebagai acuan dan pembanding penelitian yang akan dilakukan. Peneliti meninjau lima jurnal penelitian yang relevan yang dilakukan pada tahun 2017 hingga 2020.

Penelitian pertama berjudul “Sistem Pakar Diagnosis Penyakit yang disebabkan oleh Rokok dengan Metode *Forward Chaining*” yang dilakukan oleh H. Akik Hidayat dan Gilang Gumilang pada tahun 2017. Tujuan dari penelitian ini adalah membangun sebuah sistem pakar diagnosa penyakit yang ditimbulkan oleh rokok dengan metode *forward chaining*. Bahasa pemrograman yang digunakan dalam pengembangan adalah PHP dengan menggunakan *framework CodeIgniter*. Model pengembangan sistem yaitu menggunakan metode *waterfall*. Kesimpulan dari penelitian ini adalah dengan adanya aplikasi sistem pakar ini diharap angka perokok di Indonesia semakin menurun [9].

Penelitian kedua berjudul “Penerapan Metode *Forward Chaining* pada Sistem Pakar untuk Diagnosa Hama dan Penyakit Padi” yang dilakukan oleh Khurotul Aeni pada tahun 2018. Tujuan dari penelitian ini adalah membuat sebuah sistem pakar menggunakan metode *forward chaining* untuk diagnosa penyakit dan hama tanaman padi. Metode pengembangan sistem dari penelitian ini menggunakan metode *waterfall*. Kesimpulan dari penelitian adalah pengaplikasian metode *forward chaining* pada sistem pakar dapat memberikan informasi kepada pemakai khususnya petani mengenai jenis hama dan penyakit, gejala-gejala serangan berdasarkan pertanyaan-pertanyaan yang telah diajukan ahli atau pakar kepada pemakai, serta memberika informasi pengendaliannya [10].

Penelitian ketiga berjudul “Perancangan Sistem Pakar Diagnosis Penyakit pada Bayi dengan Metode *Forward Chaining* Berbasis *Android*” yang dilakukan oleh Viviliani dan Radius Tanone pada tahun 2019. Tujuan dari penelitian ini yaitu

melakukan perancangan sistem pakar untuk diagnosa penyakit pada bayi menggunakan metode *forward chaining* berbasis *android*. Metode perancangan yang digunakan adalah metode penelitian dan pengembangan atau *research and development*. Perancangan desain produk aplikasi ini menggunakan *UML* yang meliputi *use case diagram*, *activity diagram*, dan *class diagram*. Pengujian sistem dilakukan menggunakan metode *black box testing* yaitu pengujian yang dilakukan dengan menjalankan unit, kemudian diamati apakah hasil dari unit itu sesuai dengan *output* yang diinginkan. Kesimpulan dari penelitian yaitu aplikasi sistem pakar yang dibangun dapat mempermudah orang tua dalam memberikan informasi mengenai suatu penyakit, gejala, serta solusi [11].

Penelitian keempat berjudul “Penerapan Metode *Forward Chaining* untuk Mendiagnosa Penyakit Tanaman Sawi” dilakukan oleh Ajeng Savitri Puspaningrum, Erliyan Redy Susanto, dan Adi Sucipto pada tahun 2020. Tujuan dari penelitian ini adalah membangun sebuah sistem pakar bermetode *forward chaining* untuk mendiagnosa penyakit pada tanaman sawi berbasis *website*. Tahap penelitian yang diterapkan yaitu identifikasi masalah, studi pustaka, pengumpulan data, analisis kebutuhan sistem, perancangan aplikasi, analisis menggunakan metode *forward chaining*, pengujian sistem, serta validasi pakar. Kesimpulan dari penelitian ini adalah sistem pakar yang dibuat menghasilkan hasil diagnosa dengan akurasi sebesar 88,8% artinya sistem pakar layak digunakan [12].

Penelitian kelima berjudul “Implementasi Algoritma *Forward Chaining* untuk Sistem Pakar Diagnosis Hama Tanaman Kacang Kedelai pada Dinas Pertanian Pandeglang Provinsi Banten” dilakukan oleh Zaenal Hakim, Lili Sujai, Neli Nailul Wardah, Ervi Nuravlian Susanti, dan Robby Rizky pada tahun 2020. Tujuan dari penelitian ini adalah merancang sebuah sistem pakar menggunakan metode *forward chaining* untuk diagnosa hama tanaman kacang kedelai. Kesimpulan dari penelitian ini menyatakan bahwa metode *forward chaining* sangat cocok untuk diagnosa penyakit hama karena algoritma *forward chaining* menganut runut maju [13].

Pada Tabel 2.1 berisikan ringkasan penelitian terdahulu yang relevan dengan penelitian yang akan dilakukan oleh peneliti

Tabel 2.1 Ringkasan penelitian terdahulu

No	Peneliti	Judul	Metode	Kesimpulan	Perbedaan
1	H. Akik Hidayat dan Gilang Gumilang (2017)	Sistem Pakar Diagnosis Penyakit yang disebabkan oleh Rokok dengan Metode <i>Forward Chaining</i>	Metode sistem pakar yang diterapkan adalah <i>forward chaining</i> .	Kesimpulan dari penelitian ini adalah dengan adanya aplikasi sistem pakar ini diharap angka perokok di Indonesia semakin menurun	Bahasa pemrograman yang digunakan dalam pengembangan adalah <i>PHP</i> dengan menggunakan <i>framework CodeIgniter</i> . Model pengembangan sistem yaitu menggunakan metode <i>waterfall</i>
2	Khurotul Aeni (2018)	Penerapan Metode <i>Forward Chaining</i> Pada Sistem Pakar Untuk Diagnosa Hama dan Penyakit Padi	Metode sistem pakar yang digunakan adalah metode <i>forward chaining</i> .	Kesimpulan dari penelitian adalah pengaplikasian metode <i>forward chaining</i> pada sistem pakar dapat memberikan informasi kepada pemakai khususnya petani mengenai jenis hama dan penyakit, gejala-gejala serangan berdasarkan pertanyaan-pertanyaan yang telah diajukan ahli atau pakar kepada pemakai, serta memberika informasi pengendaliannya	Objek yang penelitian ini adalah padi, sedangkan penelitian yang akan dilakukan adalah kedelai. Sistem pakar pada penelitian terdahulu ini tidak dikembangkan ke bentuk <i>website</i> , sedangkan pada penelitian yang akan dilakukan, sistem pakar berbasis <i>website</i>

No	Peneliti	Judul	Metode	Kesimpulan	Perbedaan
3	Viviliani dan Radius Tanone (2019)	Perancangan Sistem Pakar Diagnosis Penyakit pada Bayi dengan Metode <i>Forward Chaining</i> Berbasis Android	Metode sistem pakar yang diterapkan adalah metode <i>forward chaining</i> .	Kesimpulan dari penelitian yaitu aplikasi sistem pakar yang dibangun dapat mempermudah orang tua dalam memberikan informasi mengenai suatu penyakit, gejala, serta solusi	Metode perancangan yang digunakan adalah metode penelitian dan pengembangan atau <i>research and development</i> Pengujian sistem dilakukan menggunakan metode <i>black box testing</i> yaitu pengujian yang dilakukan dengan menjalankan unit, kemudian diamati apakah hasil dari unit itu sesuai dengan <i>output</i> yang diinginkan
4	Ajeng Savitri Puspaningrum, Erliyan Redy Susanto, dan Adi Sucipto (2020)	Penerapan Metode <i>Forward Chaining</i> untuk Mendiagnosa Penyakit Tanaman Sawi	Metode sistem pakar yang diterapkan adalah metode <i>forward chaining</i> .	Kesimpulan dari penelitian ini adalah sistem pakar yang dibuat menghasilkan hasil diagnosa dengan akurasi sebesar 88,8 % artinya sistem pakar layak digunakan	Tahap penelitian yang diterapkan yaitu identifikasi masalah, studi pustaka, pengumpulan data, analisis kebutuhan sistem, perancangan aplikasi, analisis menggunakan

No	Peneliti	Judul	Metode	Kesimpulan	Perbedaan
					metode <i>forward chaining</i> , pengujian sistem, serta validasi pakar
5	Zaenal Hakim, Lili Sujai, Neli Nailul Wardah, Ervi Nuravlian Susanti, dan Robby Rizky (2020)	Implementasi Algoritma <i>Forward Chaining</i> untuk Sistem Pakar Diagnosis Hama Tanaman Kacang Kedelai pada Dinas Pertanian Pandeglang Provinsi Banten	Metode sistem pakar yang diterapkan adalah metode <i>forward chaining</i> .	Penelitian ini menghasilkan <i>website</i> sistem pakar diagnosa hama tanaman kacang kedelai menggunakan metode <i>forward chaining</i> yang dapat digunakan untuk melakukan edukasi kepada para petani tanaman kedelai.	Objek yang difokuskan pada penelitian yang akan dilakukan adalah penyakit pada tanaman kedelai, sedangkan pada penelitian terdahulu ini adalah hama tanaman kedelai.

2.2 Dasar Teori

2.2.1 Kedelai

Kedelai merupakan tanaman pangan unggulan setelah padi dan jagung. Kedelai merupakan sumber protein nabati yang populer bagi masyarakat Indonesia. Hal tersebut terbukti dengan banyaknya makanan maupun minuman olahan berbahan dasar kedelai seperti susu kedelai, tempe, tahu, tauco, dan lain-lain. Selain sebagai bahan pangan, komoditas ini juga digunakan untuk kepentingan industri dan benih [14].

Kandungan gizi pada kedelai terdiri dari sekitar 40% protein, 20% minyak, 35% karbohidrat larut (sukrosa, rafinosa, dan lain-lain) dan karbohidrat tidak larut (serat). Kedelai juga merupakan sumber vitamin B dibandingkan dengan komoditas golongan biji-bijian lainnya. Lemak pada kedelai mengandung antioksidan alami. Selain itu, kedelai mengandung

mineral yang kaya akan K, P, Ca, Mg, dan Fe, serta komponen nutrisi lainnya yang bermanfaat [15].

Pada tanamannya, batang tanaman kedelai bertekstur lembut dan berwarna hijau serta memiliki ketinggian sekitar 30 hingga 100 cm. Daun tanaman kedelai memiliki dua bentuk yaitu bulat atau oval dan lancip atau *lanceolate*. Biji tanaman kedelai mempunyai bentuk bulat, pipih atau lonjong dan memiliki berat sekitar enam hingga 30 gram setiap 100 biji [16].

Manfaat dari kedelai adalah antara lain menjaga kekuatan dan kesehatan tulang dikarenakan kedelai merupakan salah satu sumber kalsium, protein, dan antioksidan isoflavon yang bermanfaat untuk menjaga kesehatan tulang dan gigi. Selain itu, kedelai dapat mengurangi kadar kolesterol jahat (LDL) dan meningkatkan kadar kolesterol baik (HDL) dikarenakan kandungan serat dan lemak sehat yang tinggi pada kedelai. Lalu kedelai dapat mengurangi risiko kanker seperti kanker payudara dan kanker prostat dikarenakan adanya kandungan fitonutrien dan antioksidan isoflavon pada kedelai. Manfaat kedelai yang lainnya adalah dapat menjaga kesehatan jantung, mengontrol tekanan darah, dan mencegah penyumbatan pembuluh darah karena kedelai memiliki kandungan kalium, protein, lemak sehat, dan serat yang tinggi [17].

2.2.2 Penyakit Tanaman Kedelai

Berikut adalah jenis-jenis penyakit yang menyerang tanaman kedelai beserta gejala dan pengendaliannya:

1. Penyakit Layu Bakteri

Penyakit ini disebabkan oleh bakteri *pseudomonas solanacearum*. Tanaman kedelai rentan terkena penyakit ini saat berumur dua hingga tiga minggu setelah penanaman. Penyakit ini cepat menyebar pada kondisi kelembaban yang tinggi. Gejala pada tanaman kedelai yang terserang penyakit ini adalah layu tiba-tiba, tanaman mengering, dan tanaman mati. Pengendalian penyakit ini adalah dengan cara biji yang ditanam sebaiknya dari

varietas tahan layu, sanitasi, kebersihan lahan dan sekitar tanaman dijaga, elakukan pergiliran tanaman dengan yang bukan tanaman inang penyakit tersebut, mencabut dan memusnahkan tanaman yang terinfeksi, mengatur jarak tanam, drainase yang baik [18].

2. Penyakit Busuk Akar

Penyakit ini disebabkan oleh *rhizoctonia solani*. Penyakit ini lebih cepat berkembang pada suhu antara 28 hingga 32 derajat celsius. Gejala pada penyakit ini adalah batang keriput, tanaman mati, terdapat luka berwarna coklat kehitaman pada pangkal batang, dan terdapat bercak coklat kemerahan pada akar. Pengendalian penyakit ini yaitu dengan cara kultur teknis dengan membuat guludan dan mengatur drainase, menghindari penanaman kedelai saat curah hujan tinggi, rotasi dengan tanaman bukan inang, mengurangi sumber inokulum di dalam tanah dengan menjaga kebersihan lahan. Pengendalian kimiawi dengan perlakuan benih menggunakan fungisida berbahan aktif karboksini, triadimefon, dan iprodion. Pengendalian kimia di lapang menggunakan fungisida berbahan aktif kloranil, kloroneb, mankozeb, thiram, dan kaptan. Pengendalian hayati menggunakan jamur antagonis dari genus *Trichoderma* dan *Gliocladium spp* [19].

3. Penyakit Karat Daun

Penyakit ini disebabkan oleh jamur *phakopsora pachyrhizi*. Penyakit ini menyerang tanaman kedelai saat umurnya belum tua dan bisa menyebabkan polong kacang hampa. Gejala pada tanaman kedelai yang terserang penyakit ini adalah adanya bercak kecil berwarna colat kemerahan mirip karat pada daun, daun gugur, dan biji mengalami pemasakan lebih awal. Pengendalian penyakit ini yaitu dengan cara menanam varietas yang toleran dengan penyakit ini seperti Wilis atau Dena 1, tanam

serempak, rotasi tanaman dengan tanaman bukan inang, aplikasi fungisida berbahan aktif triadimefon dan mankozeb [20].

4. Penyakit Virus Kerdil

Penyakit ini disebabkan oleh virus *soybean dwarf virus*. Gejala pada penyakit ini adalah tanaman menjadi kerdil, daun mengecil, mosaik pada daun, dan menghasilkan polong dengan biji yang kecil disertai belang. Pengendalian penyakit ini dengan cara menanam benih kedelai bebas virus, sanitasi yaitu memusnahkan atau membakar gulma inang, menghindari menanam kedelai di daerah endemik vektor SDV, perlakuan benih dengan insektisida berbahan aktif tiametoxam, aplikasi insektisida berbahan aktif imidacloprid untuk mengendalikan serangga vektor, aplikasi cendawan hiperparasit *L. lecanii* dan *P. fumosoroseus* untuk membunuh serangga vektor dan telur aphid [18].

5. Penyakit Busuk Batang

Penyakit ini disebabkan oleh jamur *Phytium sp* dan menyerang batang tanaman kedelai dan membuat batang tanaman berubah warna menjadi kuning kecoklatan, dan tanaman mati. Untuk pengendalian penyakit ini adalah dengan cara mengatur jarak tanaman, membuat drainase yang baik, menggunakan benih yang toleran terhadap penyakit ini, penyemprotan fungisida [18].

6. Penyakit Bercak Daun

Penyakit ini disebabkan oleh *Corynespora cassiicola*. Tanaman kedelai rentan terserang penyakit ini disaat kondisi kelembaban sedang tinggi. Daun tanaman kedelai yang terserang penyakit ini akan timbul bercak-bercak berwarna coklat kemerahan. Pengendalian penyakit ini dilakukan dengan cara gunakan varietas unggul untuk menghindari kerugian secara ekonomi, lakukan penanaman lebih awal dengan varietas yang lebih cepat matang untuk menghindari serangan puncak, Bersihkan lahan secara menyeluruh dari sisa-sisa tanaman setelah panen, lakukan

rotasi dengan tanaman non-inang dan hindari tanaman sejenis (monokultur) di area yang sama [18].

7. Penyakit Antraknosa

Penyakit antraknosa disebabkan oleh jamur *Colletotrichum truncatum*. Gejala penyakit ini pada tanaman kedelai hanya adanya hifa berwarna putih, daun gugur, tanaman tumbuh lebih pendek, dan polong hampa. Pengendalian penyakit ini yaitu dengan mengusahakan tanaman tumbuh sehat agar terhindar dari infeksi, karena penyakit antraknosa umumnya merupakan penyakit sekunder, menjaga kebersihan lahan, perlakuan benih dengan fungisida berbahan aktif benomil [18][21].

8. Penyakit Embun Bulu

Penyebab penyakit embun bulu adalah jamur *Peronospora manshurica*. Gejala pada tanaman yang terserang penyakit ini adalah terdapat bercak kuning pucat di daun, biji mengecil, tanaman menjadi kerdil, terdapat benang-benang halus putih menyerupai bulu di permukaan bawah daun. Pengendalian yang dapat dilakukan adalah membersihkan lahan dari sisa-sisa tanaman sebelumnya, melakukan pengaturan waktu tanam keedelai yang ditanam pada akhir musim tanam lebih rentan terhadap infeksi patogen, perlakuan benih menggunakan fungisida berbahan aktif thiram atau kaptan, aplikasi fungisida yang mengandung bahan aktif metalaksil [21].

9. Penyakit Pustul Bakteri

Penyakit pustul atau bisul pada kedelai disebabkan oleh bakteri *Xanthomonas axonopodis pv. glycines*. Pada daun tanaman kedelai yang terinfeksi, muncul bisul kecil berwarna kecoklatan, dan daun gugur. Cuaca hangat dengan suhu 26°C – 33°C, dan lingkungan lembab karena hujan sangat mendukung perkembangan penyakit pustul pada kedelai. Pengendalian penyakit ini yaitu dengan cara menanam benih bebas patogen,

membenamkan sisa tanaman terinfeksi, hindari rotasi dengan buncis dan kacang tunggak, dan menggunakan bakterisida [22].

10. Penyakit Virus Mozaik

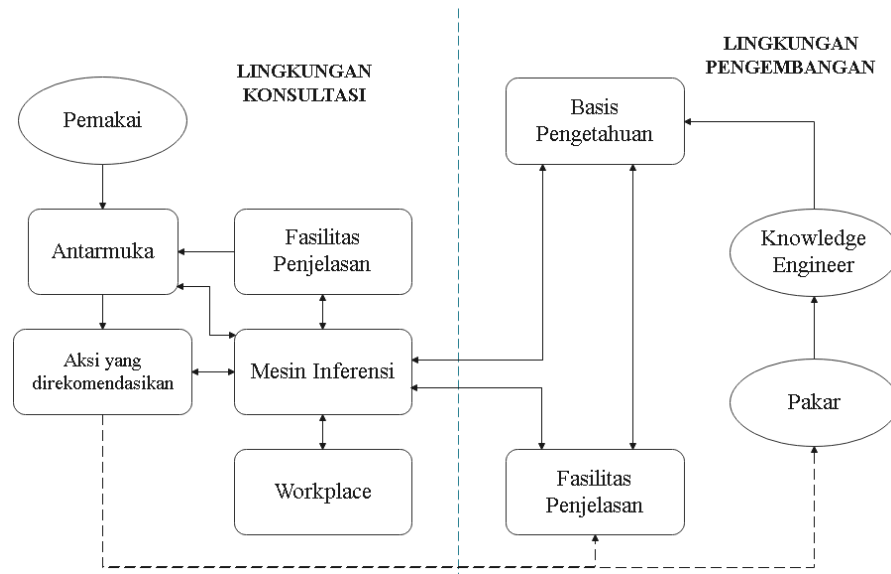
Penyakit ini disebabkan oleh partikel virus SMV (*Soybean mosaic virus*). Tanaman kedelai yang terkena virus ini mempunyai gejala permukaan daun tidak rata, daun mengecil, tepi daun melengkung, biji mengecil, dan ada gambaran mosaik dengan warna hijau gelap di sepanjang tulang daun. Pengendalian virus ini yaitu menanam benih kedelai bebas virus, menghindari menanam kedelai di daerah endemik vektor SMV, memusnahkan tanaman terinfeksi dengan cara mencabut dan membakar, dan aplikasi insektisida berbahan aktif imidakloprid untuk mengendalikan vektor [22].

2.2.3 Diagnosa

Diagnosa dapat diartikan sebagai upaya untuk menemukan kelemahan pada seseorang dengan cara melakukan studi yang disertai pengujian mengenai gejala-gejalanya. Manfaat diagnosa antara lain untuk menemukan kelemahan atau penyakit pada seseorang, untuk mencegah penyebaran suatu wabah, dan untuk pertimbangan dalam upaya pengendalian penyakit [23].

2.2.4 Sistem Pakar

Sistem pakar adalah sebuah program komputer yang mengandung pengetahuan tentang suatu bidang yang spesifik dari satu atau lebih pakar manusia. Sistem pakar yang baik dirancang agar dapat menyelesaikan suatu permasalahan tertentu dengan meniru cara kerja dan pengetahuan yang berasal dari para pakar manusia. Keberadaan sistem pakar sangat memudahkan pekerjaan para pakar manusia menjadi lebih baik dan efektif.



Gambar 2.1 Komponen sistem pakar

Pada Gambar 2.1 merupakan komponen penting dalam sistem pakar yang terdiri dari:

1. *Pemakai/Pengguna (User)*

User adalah seorang non ahli yang dapat melakukan interaksi dengan sistem pakar untuk berkonsultasi.

2. *Basis Pengetahuan (Knowledge Base)*

Basis pengetahuan berisikan pengetahuan-pengetahuan yang digunakan dalam memecahkan penyelesaian masalah.

3. *Mesin Inferensi (Inference Engine)*

Berisi metodologi yang digunakan untuk melakukan penalaran terhadap informasi-informasi dalam basis pengetahuan untuk merumuskan kesimpulan. Komponen ini merupakan bagian yang memuat mekanisme fungsi berpikir dan pola-pola penalaran sistem yang digunakan oleh seorang pakar.

4. *Fasilitas Penjelas*

Komponen ini berperan untuk merekam setiap aktivitas sistem pakar. Adanya keberadaan komponen ini diharapkan seorang user dapat melakukan penelusuran alur dari penalaran sistem pakar yang bersangkutan

5. *Antarmuka Pengguna (User Interface)*

Merupakan mekanisme yang menjembatani pengguna dan sistem pakar untuk melakukan komunikasi

6. *Akuisi Pengetahuan*

Merupakan akumulasi dan transformasi keahlian dalam menyelesaikan masalah dari sumber pengetahuan ke dalam komputer

7. *Workplace*

Workplace merupakan area dari sekumpulan memori kerja yang digunakan untuk merekam hasil dari kesimpulan yang dicapai oleh sistem [24].

Sebuah sistem pakar tentu saja memiliki beberapa ciri-ciri yaitu sebagai berikut:

1. Terbatas pada keahlian atau bidang tertentu
2. Keluaran sistem pakar berupa anjuran atau nasihat
3. Keluaran sistem pakar tergantung dengan interaksi antara sistem pakar dengan pengguna sistem pakar
4. Dirancang untuk dapat dikembangkan secara bertahap [25].

2.2.5 Metode *Forward Chaining*

Forward chaining adalah metode penarikan kesimpulan yang berdasarkan pada fakta yang ada [26]. Teknik pencarian yang diterapkan pada *forward chaining* adalah dengan cara menggunakan fakta yang diketahui lalu fakta-fakta tersebut dicocokkan dengan aturan *IF-THEN*. Bila ada *rule* yang cocok dengan bagian *IF*, maka *rule* tersebut dieksekusi. Saat *rule* dieksekusi maka ada fakta baru pada bagian *THEN* [27].

Gambar 2.2 merupakan gambar yang menjelaskan bagaimana cara kerja dari metode *forward chaining* yaitu kumpulan sebuah data berupa fakta dijadikan sebuah aturan menggunakan aturan *IF-THEN*, lalu dengan kumpulan aturan-aturan tersebut dapat ditarik sebuah kesimpulan.



Gambar 2.2 Cara kerja *forward chaining*

Contoh penggunaan metode *forward chaining* jika tujuannya adalah untuk menentukan nama hewan berdasarkan beberapa fakta adalah sebagai berikut : *IF* A adalah suka pisang dan A adalah mamalia, *THEN* A adalah monyet.

2.2.6 Website

Website adalah sekumpulan halaman yang terdiri dari beberapa lama yang berisikan konten berupa data berbentuk teks, gambar, audio, dan video yang dapat diakses menggunakan jaringan internet [28]. Fungsi dari *website* secara umum adalah sebagai tempat menyebarkan informasi guna edukasi, dan lain-lain. Berikut adalah unsur-unsur utama *website* yang sangat vital sehingga *website* dapat diakses oleh pengguna internet:

1. *Domain*

Domain adalah alamat sebuah *website*. Contoh dari *domain* adalah facebook.com, google.co.id, dan lain-lainnya.

2. *Hosting*

Hosting adalah *server* tempat dimana semua file *website* disimpan serta dapat diakses dan dikelola melalui internet. *Hosting* dapat diibaratkan sebuah gudang dan *website* adalah seluruh isi gudang tersebut.

3. Konten

Konten adalah isi dari *website* yang dapat berupa teks, video, audio, *image*, dan lain-lain.

4. Tampilan

Tampilan pada sebuah *website* harusnya baik karena sekitar 48 % pengguna internet menyebutkan bahwa desain *website* adalah faktor yang menentukan kredibilitas [29].

2.2.7 Bahasa Pemrograman

2.2.7.1 PHP

PHP atau singkatan dari PHP *Hypertext Preprocessor* merupakan sebuah bahasa pemrograman yang fleksibel yang digunakan untuk pengembangan *website*. Bahasa pemrograman yang diciptakan oleh Rasmus Lerdorf di tahun 1995 ini bersifat *open source*. Karakteristiknya yang fleksibel, sederhana, mudah dipahami, dan kompatibilitasnya yang baik dengan HTML membuat setidaknya 78% *website* di seluruh dunia menggunakan bahasa pemrograman ini [30].

Bahasa pemrograman PHP adalah bahasa *scripting server-side* yang berarti jenis bahasa pemrograman ini akan dijalankan oleh server. Secara umum, fungsi PHP adalah untuk membuat dan mengembangkan situs *website* statis dan dinamis serta aplikasi *website*. Pada *website* dinamis, PHP digunakan untuk menyimpan data ke basis data, membuat halaman lebih interaktif dengan pengguna, dan lain-lain [31].

2.2.7.2 Javascript

Javascript adalah bahasa pemrograman yang digunakan untuk membuat situs dengan konten *website* yang dinamis yang artinya konten dinamis adalah konten yang dapat bergerak atau berubah tanpa perlu memuat ulang halaman situs. *Javascript* biasanya dikolaborasikan dengan HTML dan CSS dimana HTML digunakan untuk membuat struktur dan CSS untuk merancang *style website*, sementara itu *javascript* digunakan untuk fungsionalitas *website* [32].

2.2.7.3 HTML (*HyperText Markup Language*)

HTML atau singkatan dari *HyperText Markup Language* merupakan bahasa *markup* yang digunakan untuk membuat struktur halaman *website* agar dapat ditampilkan di *web browser*. HTML

diibaratkan sebagai pondasi awal dalam menyusun kerangka halaman *website*. Yang dapat dilakukan menggunakan HTML adalah:

1. Mengontrol tampilan dari *web page* dan *content*.
2. Mempublikasikan dokumen secara *online* sehingga dapat diakses di seluruh dunia.
3. Menambahkan objek-objek seperti *image*, video, audio, dan lain-lainnya dalam dokumen HTML [33].

2.2.7.4 CSS

CSS ialah singkatan dari *Cascading Style Sheets* yang biasanya digunakan untuk mengatur tampilan elemen yang tertulis dalam bahasa *markup*, seperti HTML. CSS digunakan untuk mendesain tampilan *website* seperti warna teks, jenis *font*, ukuran kolom, warna latar belakang *website* dan lain-lain. Berikut merupakan kelebihan dari CSS:

1. Proses desain halaman *website* menjadi lebih cepat.
2. Halaman *website* lebih cepat dimuat.
3. Proses *maintenance* mudah.
4. *Style* lebih beragam dibanding HTML [34].

2.2.8 Perancangan Basis Data

Perancangan basis data merupakan proses untuk menentukan data yang akan dibutuhkan oleh sistem. Perancangan basis data dapat dilakukan dengan menggunakan UML, *Flowchart*, dan lain-lain.

2.2.8.1 UML (*Unified Modelling Language*)





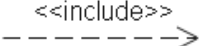
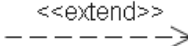
UML atau *Unified Modelling Language* adalah suatu metode dalam pemodelan secara visual yang digunakan sebagai sarana perancangan sistem berorientasi objek. UML merupakan standar penulisan atau semacam *blue print* yang dimana didalamnya termasuk sebuah bisnis proses, penulisan kelas-kelas dalam sebuah bahasa yang spesifik. Terdapat beberapa jenis diagram UML yang sering

digunakan dalam perancangan sebuah sistem, yaitu *use case diagram*, *class diagram*, *activity diagram*, dan *sequence diagram* [35].

2.2.8.2 Use Case Diagram

Use Case Diagram merupakan gambaran dari fungsionalitas yang diharapkan dari sebuah sistem, dan merepresentasikan sebuah interaksi antara *actor* dan sistem. *Actor* pada diagram ini adalah sebuah gambaran entitas dari manusia atau pengguna sistem [35]. Pada tabel 2.2 berisikan informasi simbol khusus yang ada pada *use case diagram* beserta penjelasannya [36].

Tabel 2.2 Simbol *use case*

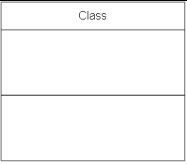


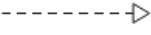
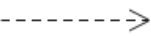
No	Simbol	Nama	Keterangan
1		<i>Actor</i>	Mewakili manusia yang berkomunikasi dengan <i>use case</i>
2		<i>Use Case</i>	Interaksi antara aktor dan sistem
3		<i>Association</i>	Penghubung antara aktor dan <i>use case</i>
4		<i>Generalization</i>	Hubungan dimana objek anak berbagai perilaku dari objek yang ada di objek induk
5		<i>Include</i>	Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya
6		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> asal pada suatu titik yang diberikan

2.2.8.3 Class Diagram

Merupakan diagram yang menggambarkan struktur dan deskripsi *class*, *package*, dan *object* beserta hubungan satu sama lain

seperti containment, pewarisan, asosiasi, dan lain-lain. *Class diagram* berfungsi untuk menjelaskan tipe dari objek sistem dan hubungannya dengan objek lain. Pada tabel 2.3 berisikan informasi simbol khusus yang ada pada *sequence diagram* beserta penjelasannya .


Tabel 2.3 Simbol *class diagram*




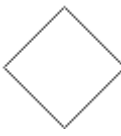
No	Simbol	Nama	Keterangan
1		<i>Class</i>	Himpunan objek-objek yang berbagi atribut serta operasi yang sama
2		<i>Association</i>	Penghubung objek satu dengan objek yang lain
3		<i>Generalization</i>	Hubungan dimana objek anak berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk.
4		<i>Realization</i>	Operasi yang dilakukan oleh suatu objek
5		<i>Depedency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri akan mempengaruhi elemen yang bergantung pada elemen yang tidak mandiri

2.2.8.4 Activity Diagram

Diagram ini menggambarkan aliran fungsional sistem atau menggambarkan gambaran alir dari aktivitas-aktivitas didalam sistem yang berjalan beserta manusia yang berperan sebagai pengguna yang berinteraksi dengan sistem [35]. Pada tabel 2.4 berisikan informasi simbol yang ada pada *activity diagram* beserta penjelasannya [37].

Tabel 2.4 Simbol *activity diagram*




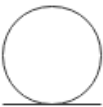

No	Simbol	Nama	Keterangan
1		<i>Activity</i>	Aktivitas yang dilakukan oleh sistem



2		<i>Initial Node</i>	Sebuah penanda dimulainya suatu aktivitas
3		<i>Final Node</i>	Sebuah penanda berakhirnya suatu aktivitas
4		<i>Line Connector</i>	Digunakan untuk menghubungkan satu simbol dengan simbol lainnya
5		<i>Decision</i>	Digunakan untuk menggambarkan suatu keputusan yang harus diambil pada kondisi tertentu

2.2.8.5 Sequence Diagram

Diagram ini digunakan untuk menjelaskan dan menampilkan interaksi antar objek-objek dalam sebuah sistem secara terperinci dan menampilkan perintah yang dikirim, beserta waktu pelaksanaannya pada suatu sistem [35]. Pada tabel 2.5 berisikan informasi simbol khusus yang ada pada *sequence diagram* beserta penjelasannya [38].

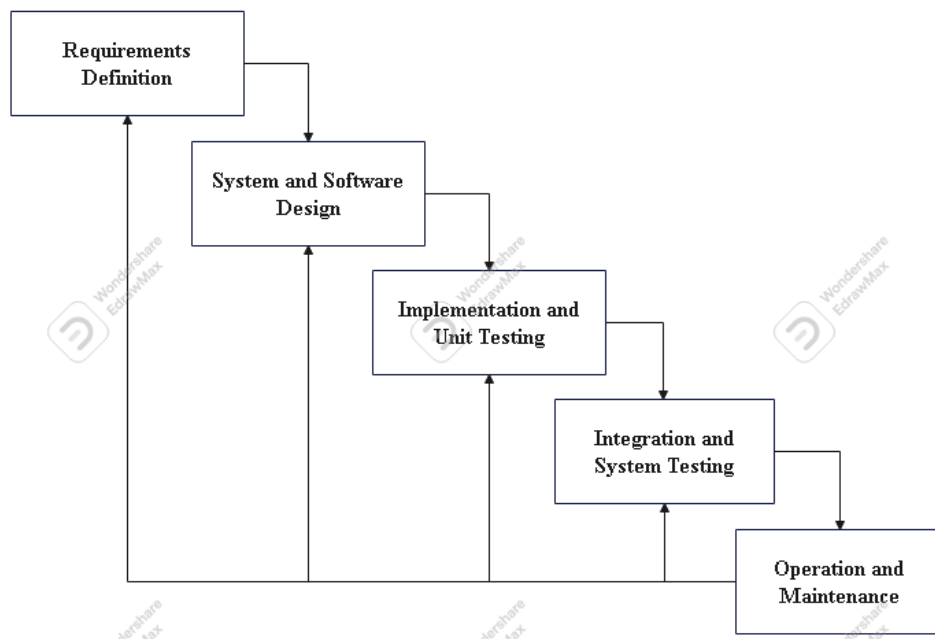
Tabel 2.5 Simbol *sequence diagram*

No	Simbol	Nama	Keterangan
1		<i>Actor</i>	Menggambarkan manusia yang berinteraksi baik di dalam maupun di luar sistem
2		<i>Message</i>	Menggambarkan pengiriman pesan
3		<i>A focus of Control and a life line</i>	Menggambarkan tempat mulai dan berakhirnya pesan
4		<i>Entity Class</i>	Menggambarkan sebuah hubungan yang akan dilakukan
5		<i>Boundary Class</i>	Menggambarkan antarmuka atau alat yang digunakan untuk berinteraksi antar sistem

6		<i>Control Class</i>	Untuk mengatur arus informasi dalam skenario sistem
7		<i>Object</i>	Menggambarkan sebuah kelas atau objek

2.2.9 Metode *Waterfall*

Metode *waterfall* merupakan metode yang dilakukan dengan pendekatan yang sistematis, mulai dari tahap kebutuhan sistem lalu tahap analisis, desain, pengkodean, *testing*, dan *maintenance* [39]. Langkah demi langkah yang dilalui harus diselesaikan satu per satu secara berurutan. Pada gambar 2.3 merupakan tahapan-tahapan pada metode *waterfall*.



Gambar 2.3 Tahap metode *waterfall*

Berikut adalah penjelasan tahapan-tahapan metode *waterfall* yaitu sebagai berikut:

1. *Requirements Definition*

Merupakan tahap untuk mengetahui dan memahami bagaimana informasi kebutuhan pengguna terhadap sebuah perangkat lunak.

Metode pengumpulan informasi dapat diperoleh dengan cara antara lain diskusi, survei, wawancara, observasi, dan sebagainya.

2. *System and Software Design*

Tahap ini dilakukan untuk memberikan gambaran lengkap mengenai apa yang harus dikerjakan.

3. *Implementation and Unit Testing*

Tahap ini merupakan tahap pemrograman. Selain pemrograman, pada tahap ini dilakukan pengujian dan pemeriksaan terhadap fungsionalitas program yang dibuat untuk melihat apakah program sudah memenuhi kriteria yang diinginkan atau belum.

4. *Integration and System Testing*

Setelah program yang dikembangkan telah diuji di tahap *Implementation and Unit Testing*, selanjutnya program diintegrasikan dalam sistem secara keseluruhan. Setelah integrasi selesai, selanjutnya dilakukan pemeriksaan program secara keseluruhan untuk mengidentifikasi apakah ada kesalahan pada program.

5. *Operation and Maintenance*

Pada tahap terakhir ini, program yang telah jadi dioperasikan pengguna dan dilakukan pemeliharaan oleh pengembang. Pemeliharaan meliputi perbaikan kesalahan, peningkatan, dan penyesuaian program dengan kebutuhan [40].

2.2.10 Black Box Testing

Black box testing merupakan metode pengujian suatu sistem program yang dilakukan dengan menjalankan dan mengeksekusi unit atau modul, kemudian diamati apakah hasil dari unit tersebut sesuai dengan keluaran yang diinginkan. Singkatnya, metode pengujian ini dilakukan terhadap kegunaan suatu aplikasi. *Black box testing* meninjau masukan dan keluaran sistem program tanpa pengetahuan tentang internal program. Pengujian ini penting

untuk menemukan kesalahan pada sistem program sebelum program disebarluaskan [41].

2.2.11 Confusion Matrix

Confusion Matrix merupakan alat ukur menggunakan matriks dua kali dua yang digunakan untuk mendapatkan jumlah ketepatan klasifikasi data uji atau *dataset* terhadap kelas tepat dan tidak tepat pada metode yang digunakan. Hasil dari penggunaan *confusion matrix* adalah nilai keakuratan suatu sistem. Pada tabel 2.6 berikut merupakan model *confusion matrix*.

Tabel 2.6 Model *confusion matrix*

Aktual	Prediksi	
	<i>TRUE</i>	<i>FALSE</i>
<i>TRUE</i>	TP (<i>True Positive</i>)	FP (<i>False Positive</i>)
<i>FALSE</i>	FN (<i>False Negative</i>)	TN (<i>True Negative</i>)

Berikut merupakan penjelasan istilah-istilah yang terdapat pada tabel 2.6, sebagai berikut:

1. TP (*True Positive*) yaitu dimana data prediksi positif dan itu benar
2. TN (*True Negative*) yaitu dimana data prediksi positif dan itu salah
3. FP (*False Positive*) yaitu dimana data prediksi negatif dan itu benar
4. FN (*False Negative*) yaitu dimana data prediksi negatif dan itu salah [42].

Nilai yang dihasilkan dari *confusion matrix* berupa hasil evaluasi akurasi sistem yang dilakukan dengan cara menggunakan persamaan sebagai berikut [43][44]:

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \%$$

2.2.12 Perangkat Lunak Yang Digunakan

2.2.11.1 MySQL

MySQL adalah sistem pengelola basis data yang menggunakan perintah *SQL* (*Structured Query Language*) yang banyak digunakan dalam pembuatan *website*. *MySQL* termasuk ke dalam RDBMS

(*Relational Database Management System*) yang artinya pada basis datanya menggunakan tabel, kolom, dan baris. Fungsi utama dari *MySQL* adalah untuk membuat dan mengelola basis data dari sisi *server* yang memuat berbagai informasi dengan menggunakan bahasa *SQL*. Kelebihan dari *MySQL* adalah sebagai berikut:

1. Bersifat *open source* atau dapat diakses oleh siapa saja dan kapan saja secara gratis.
2. *Multi User* yang artinya *MySQL* dapat digunakan secara bersamaan dalam satu waktu.
3. Terintegrasi dengan bahasa pemrograman lain seperti PHP.
4. Sistem keamanan yang telah terjamin
5. Perangkat lunak yang portabel karena dapat digunakan dan dijalankan dengan baik di sistem operasi *Windows*, *Linux*, maupun *MacOS*

Kelemahan terbesar dari *MySQL* adalah kurang mendukung pemakaian basis data dengan kapasitas besar karena *MySQL* dikhususkan pada perangkat yang memiliki spesifikasi rendah [45].

2.2.11.2 XAMPP

XAMPP adalah sebuah *tool* untuk membantu pengembangan *website* dan aplikasi lainnya. XAMPP berperan sebagai *server* yang berdiri sendiri (*localhost*) dan sebuah *web server* yang mudah digunakan untuk dapat menampilkan halaman *website* yang dinamis. Singkatnya, XAMPP adalah perangkat lunak yang memungkinkan pengguna untuk menjalankan *web server* dan basis data pada *localhost* komputer. XAMPP biasanya digunakan menguji kinerja fitur dari *website* yang sedang dikembangkan [46].

2.2.11.3 Figma

Figma adalah *tool* berupa *website* dan aplikasi *desktop* untuk desain yang terhubung dengan *cloud* sehingga bisa digunakan

kapanpun dan dimanapun melalui *internet*. Aplikasi ini biasanya digunakan untuk mendesain suatu tampilan *website*, *mobile app*, dan lain-lain. Figma memiliki fungsi sebagai *tools prototyping*, membuat grafik vektor, dan aplikasi desain UI/UX [47].

2.2.11.4 Visual Studio Code

Visual Studio Code atau *VS Code* merupakan teks editor atau IDE (*Integrated Development Environment*) platform ganda yang sangat ringan. Teks editor ini mendukung banyak sekali bahasa pemrograman antara lain *JavaScript*, *PHP*, *Python*, dan lain-lain [48].