



**BAB II**  
**TINJAUAN PUSTAKA DAN**  
**LANDASAN TEORI**

## BAB II TINJAUAN PUSTAKA DAN LANDASAN TEORI

### 2.1 Tinjauan Pustaka

Pada penelitian yang dilakukan sebelumnya [6] yaitu membangun sebuah sistem diagnosis penyakit tanaman padi menggunakan metode *promethee* bertujuan untuk membantu mendiagnosis penyakit tanaman padi sesuai gejala yang terjadi. Sistem ini menggunakan metode *Promethee* untuk mendiagnosis penyakit tanaman padi. Implementasi dari sistem ini menggunakan bahasa pemrograman PHP dan menggunakan data *MySQL*. Input pada sistem ini berupa gejala penyakit yang terjadi oleh data uji sedangkan output dari sistem adalah hasil diagnosis berdasarkan perhitungan menggunakan metode *Promethee*.

Pada penelitian selanjutnya [7] yang membangun sistem pakar diagnosis hama dan penyakit padi dengan metode bayesian berbasis *certainty factor* bertujuan untuk mempermudah para petani padi dalam menangani hama dan penyakit padi layaknya seorang pakar. Sistem ini menggunakan metode Bayesian Berbasis *Certainty Factor*. Proses pada sistem ini yaitu pengguna men-ceklist atau menginputkan data gejala yang kemudian akan dicocokkan dengan aturan pada database sistem. Setelah proses pencocokan gejala maka sistem akan menghitung nilai kepastiannya atau *certainty factor*.

Kemudian peneliti lainnya [8] yang membangun sistem pakar mendiagnosis penyakit dan hama tanaman padi menggunakan metode *theorema bayes* untuk mendiagnosis tanaman padi. Sistem ini menggunakan metode pengembangan menggunakan metode Siklus hidup pengembangan sistem atau *Sistem Development Life Cycle* atau yang biasa disebut SDLC. Output sistem ini berupa hasil dari perhitungan metode *theorema bayes* dan hasil diagnosis berupa nama penyakit tanaman padi.

Penelitian selanjutnya [9] yang membangun sebuah aplikasi sistem pakar untuk mendiagnosis kerusakan pada komputer PC menggunakan metode *certainty factor* dengan menggunakan aplikasi Visual Studio 2010 dan menggunakan *Microsoft Access 2013* sebagai database nya. Aplikasi ini berfungsi untuk membantu pengguna komputer PC dalam mendiagnosis kerusakan pada perangkat keras komputer PC. Dalam membangun aplikasi, aplikasi dibuat dengan menggunakan metode *Certainty Factor* (CF). Cara ini dianggap tepat dalam pengenalan gejala kerusakan, dan menentukan jenis kerusakan Komputer PC

Sehingga diharapkan aplikasi yang dibuat dapat digunakan kapan saja, dan dimana saja pada saat dibutuhkan.

Penelitian selanjutnya [10] yang membangun modelan spasial erosi kepala parit di Dataran Tinggi Loess menggunakan *certainty factor* dan model hutan acak. Tujuan dari penelitian ini adalah untuk membandingkan dan mengevaluasi model pembelajaran mesin canggih (RF) dan metode statisti bivariat (CF) untuk memetakan GHEM dan menentukan faktor pengkondisian pembentukan selokan yang paling penting di Dataran Tinggi Loess China pada skala regional.

**Tabel 2. 1** Tabel Perbandingan Penelitian

No	Judul	Peneliti (Tahun)	Metode	Hasil Penelitian
1	Diagnosis Penyakit Tanaman Padi Menggunakan Metode Promethee	Eny Maria, Feri Fadlin, Medi Taruk (2020)	<i>Promethee</i>	Membangun sistem untuk mendiagnosis penyakit tanaman padi sesuai gejala yang terjadi.
2	Penyakit Padi Dengan Metode Bayesian Berbasis Certainty Factor	Dwi Puji Raharjo, Andharini Dwi Cahyani, Bain Khusnul Khotimah (2019)	<i>Bayesian berbasis certainty factor</i>	Membangun sistem pakar diagnosis hama dan penyakit padi dengan metode bayesian berbasis <i>certainty factor</i> dengan men-ceklist atau menginputkan data gejala pada sistem.
3	Sistem Pakar Mendiagnosis Penyakit Dan Hama Tanaman Padi Menggunakan Metode Theorema Bayes	Husrina, Ismail Majid, dan Musyrifah (2021)	<i>Theorema bayes</i>	Hasil dari perhitungan metode <i>theorema bayes</i> dan hasil diagnosis berupa nama penyakit tanaman padi

4	Implementation of Certainty Factor Method for Expert System	Ade Setiawan Sembiring, Sulindawaty, Napitupulu, Penda Sudarto Hasugian, Simanjorang, Agustina Simangunsong, Sihotang	<i>Certainty factor</i>	Aplikasi untuk membantu pengguna komputer PC dalam mendiagnosis kerusakan pada perangkat keras komputer PC
5	Spatial modeling of gully head erosion on the Loess Plateau using a certainty factor and random forest model	Chengcheng Jiang, Wen Fan, Ningyu Yu, Enlong Liu	<i>Certainty factor</i>	Menentukan faktor pengkondisian pembentukan selokan yang paling penting di Dataran Tinggi Loess China pada skala regional
6	Sistem Pakar Diagnosis Penyakit Tanaman Padi Berbasis Website	Zaroh Khoerunisa	<i>Certainty Factor</i>	Penelitian ini akan menitikberatkan pada penyakit tanaman padi, dengan menggunakan perhitungan <i>certainty factor</i> berbasis <i>website</i> . Hasil yang keluaran penelitian ini meliputi nama penyakit, presentase, prediksi penyakit, dan saran untuk penyakit yang diprediksi oleh sistem.

Dari hasil penelitian sebelumnya dapat disimpulkan bahwa proses inputan pada sistem pakar berupa gejala – gejala yang terjadi dan output yang dihasilkan berupa hasil diagnosis perhitungan dari metode yang digunakan. Sehingga perbedaan dari penelitian sebelumnya dengan sistem pakar yang akan dibangun adalah sistem dapat menampilkan hasil diagnosis penyakit tanaman padi berupa jenis penyakit dan menampilkan persentase hasil dari perhitungan metode *certainty factor*. Sistem ini berbasis *website* dengan menggunakan bahasa pemrograman PHP, *framework laravel*, database yang digunakan yaitu *MySQL*, dan metode pengembangan menggunakan metode *waterfall*.

## **2.2 Landasan Teori**

### **2.2.1 Kecerdasan buatan**

Kecerdasan buatan berasal dari bahasa Inggris “*Artificial Intelligence*” atau disingkat AI, yaitu *intelligence* adalah kata sifat yang berarti cerdas, sedangkan *artificial* artinya buatan. Kecerdasan buatan yang dimaksud di sini merujuk pada mesin yang mampu berpikir, menimbang tindakan yang akan diambil, dan mampu mengambil keputusan seperti yang dilakukan oleh manusia. Berikut beberapa definisi kecerdasan buatan yang telah didefinisikan oleh beberapa ahli.

Herbert Alexander Simon mendefinisikan Kecerdasan Buatan (*artificial intelligence*) merupakan kawasan penelitian, aplikasi, dan instruksi yang terkait dengan pemrograman komputer untuk melakukan sesuatu hal yang dalam pandangan manusia cerdas [11].

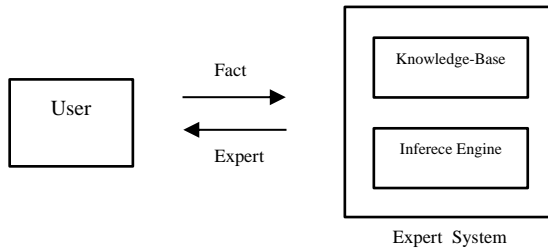
Rich and Knight mendefinisikan Kecerdasan Buatan (AI) sebagai sebuah studi tentang bagaimana membuat komputer melakukan hal-hal yang pada saat ini dapat dilakukan lebih baik oleh manusia [11].

Encyclopedia Britannica mendefinisikan Kecerdasan Buatan (AI) merupakan cabang dari ilmu komputer yang dalam merepresentasi pengetahuan lebih banyak menggunakan bentuk simbol – simbol daripada bilangan dan memproses informasi berdasarkan metode heuristik atau dengan berdasarkan sejumlah aturan [11].

Ada tiga tujuan kecerdasan buatan, yaitu: membuat komputer lebih cerdas, mengerti tentang kecerdasan, dan membuat mesin lebih berguna.

Yang dimaksud kecerdasan adalah kemampuan untuk belajar atau mengerti dari pengalaman, memahami pesan yang kontradiktif dan ambigu, menanggapi dengan cepat dan baik atas situasi yang baru, menggunakan penalaran dalam memecahkan masalah serta menyelesaikannya dengan efektif [12].

### 2.2.2 Sistem Pakar



**Gambar 2. 1** Konsep Dasar Sistem Pakar Berbasis Pengetahuan

Sistem pakar merupakan cabang dari *Artificial Intelligence* (AI) yang cukup tua karena sistem ini mulai dikembangkan pada pertengahan 1960. Sistem pakar yang muncul pertama kali adalah *General – purpose problem solver* (GPS) yang dikembangkan oleh Newel dan Simon. Sampai saat ini sudah banyak sistem pakar yang dibuat, seperti MYCIN untuk diagnosis penyakit, DENDRAL untuk mengidentifikasi struktur molekul campuran yang tak dikenal, XCON & XSEL untuk membantu konfigurasi sistem komputer besar, SOPHIE untuk menganalisis sirkuit elektronik, Prospector digunakan dibidang geologi untuk membantu mencari dan menemukan deposit, FOLIO digunakan untuk membantu memberikan keputusan bagi seorang manager dalam stok dan investasi, DELTA dipakai untuk pemeliharaan lokomotif listrik diesel, dan sebagainya.

Sistem adalah setiap kesatuan secara konseptual atau fisik yang terdiri dari bagian-bagian dalam keadaan saling tergantung satu sama lainnya [10]. Pakar adalah seseorang yang menguasai dalam bidang tertentu dengan sangat baik yang dapat memberikan respon dengan cepat tanpa harus berpikir panjang dan sesekali muncul tanpa kesadaran [11].

Sistem Pakar adalah sistem berbasis komputer yang menggunakan pengetahuan, fakta, dan teknik penalaran dalam memecahkan yang biasanya dapat diselesaikan oleh seorang ahli dalam bidang tertentu. Pengetahuan yang digunakan dalam sistem pakar merupakan serangkaian informasi mengenai gejala diagnosa, sebab-akibat, aksireaksi tentang suatu domain tertentu (misalnya, domain diagnosa medis). Secara umum, definisi tradisional sebuah program komputer biasa : Algoritma + Struktur data = PROGRAM. Dalam sistem pakar, definisi berubah menjadi mesin inferensi + Pengetahuan = Sistem Pakar [12].

Sistem pakar atau Expert System biasa disebut juga dengan *Knowledge Based System* yaitu suatu aplikasi computer yang ditujukan untuk membantu pengambilan keputusan atau pemecahan persoalan dalam bidang yang spesifik, dalam memecahkan persoalan pada bidang tertentu dan didukung mesin Inferensi/*Inferensi Engine* yang melakukan penalaran atau pelacakan terhadap sesuatu atau fakta – fakta dan aturan kaidah yang ada di basis pengetahuan setelah dilakukan pencarian, sehingga dicapai kesimpulan. Sistem ini bekerja dengan menggunakan pengetahuan dan metode analisis yang telah didefinisikan terlebih dahulu oleh pakar yang sesuai dengan bidang keahliannya. Sistem ini disebut sistem pakar karena fungsi dan perannya sama seperti seorang ahli yang harus memiliki pengetahuan, pengalaman dalam memecahkan suatu persoalan. Sistem biasanya berfungsi sebagai kunci penting yang akan membantu suatu sistem pendukung keputusan atau sistem pendukung eksekutif. Dengan adanya sistem pakar maka semua bidang ilmu bisa diwakili dengan namanya sistem pakar jika pakarnya bersedia menjelaskan dan memberikan pengetahuan tentang kepakarannya masing-masing disiplin ilmu [13]. Sistem pakar dapat digunakan oleh [12]:

1. Orang awam yang bukan pakar untuk meningkatkan kemampuan mereka dalam memecahkan masalah
2. Pakar sebagai asisten yang berpengetahuan.
3. Memperbanyak atau menyebarkan sumber pengetahuan yang semakin langka.



Sistem pakar merupakan program yang dapat menggantikan keberadaan seorang pakar. Alasan mendasar mengapa *expert system* dikembangkan untuk menggantikan seorang pakar [12]:

1. Dapat menyediakan kepakaran setiap waktu dan di berbagai lokasi.
2. Secara otomatis mengerjakan tugas - tugas rutin yang membutuhkan seorang pakar.
4. Seorang pakar akan pensiun atau pergi.
5. Menghadirkan/menggunakan jasa seorang pakar memerlukan biaya yang mahal
6. Kepakaran dibutuhkan juga pada lingkungan yang tidak bersahabat (*hostile environment*).

### 2.2.3 Metode Certainty Factor (CF)

Metode *Certainty factor* (CF) adalah sebuah metode yang membuktikan suatu fakta apakah pasti atau tidak yang berbentuk *metric* yang digunakan sistem pakar [14]. *Certainty Factor* merupakan bagian dari *certainty theory*, yang pertama kali dikenalkan oleh E. H. Shorliffe dan B. G. Buchanan dalam pembuatan MYCIN (aplikasi sistem pakar awal yang dirancang untuk mengidentifikasi infeksi di dalam darah) mencatat bahwa pakar sering kali menganalisis informasi [15]. Metode ini menggunakan ungkapan seperti mungkin, kemungkinan besar, dan hampir pasti sehingga metode ini menggambarkan tingkat kepercayaan pakar terhadap masalah yang dihadapi. *Certainty factor* didefinisikan sebagai berikut:

$$CF(H, E) = MB(H, E) - MD(H, E) \quad (1)$$

Keterangan :

- CF (H, E) : *Certainty Factor*  
 E : *Evidence* (Peristiwa atau fakta)  
 H : *Hipotesis* (Dugaan)  
 MB (H, E) : Ukuran kenaikan kepercayaan (*Measure of increased belief*) terhadap *hipotesis* H yang dipengaruhi oleh gejala E.  
 MD (H, E) : Ukuran kenaikan ketidakpercayaan (*Measure of increased disbelief*) terhadap *hipotesis* H dipengaruhi oleh gejala E [3].

Ada dua cara mendapatkan tingkat keyakinan dari sebuah *rule*, yaitu[14]:

- a. Metode Net Belief yang diusulkan oleh E.H. Shortliffe dan B.G. Buchanan.

$$CF (Rule) = MB (H, E) \quad (2)$$

$$MB (H, E) = \frac{\max[P(H|E), P(H)] - P(H)}{\max[1, 0] - P(H)} \dots P(H) = 1 \quad (3)$$

$$MD (H, E) = \min[P(H|E), P(H)] - P(H) \dots P(H) = 0 \quad (4)$$

Dimana :

CF (Rule) : *Certainty Factor*

E : *Evidence* (Peristiwa atau fakta)

H : *Hipotesis* (Dugaan)

MB (H, E) : Ukuran kenaikan kepercayaan (*Measure of increased belief*) terhadap *hipotesis* H yang dipengaruhi oleh gejala E (antara 0 dan 1).

MD (H, E) : Ukuran kenaikan ketidakpercayaan (*Measure of increased disbelief*) terhadap *hipotesis* H dipengaruhi oleh gejala E (antara 0 dan 1).

P(H) : Probabilitas kebenaran hipotesis H.

P(H|E) : Probabilitas bahwa H benar karena fakta E.

- b. Dengan cara mewawancarai pakar:

**Tabel 2. 2** Range Nilai *Certainty Factor*

Uncertain Term	CF
Tidak tahu	-0.2 s/d 0.2
Mungkin	0.4
Kemungkinan besar	0.6
Hampir pasti	0.8
Pasti	1.0

Pada tabel 2.2 tersebut menjelaskan *uncertain term* dan nilai CF. jika belum ada nilai CF untuk setiap gejala yang menyebabkan penyakit, maka digunakan formula dasar yang digunakan untuk mendiagnosa penyakit.

1. *Certainty factor* untuk kaidah dengan premis / gejala tunggal (*single premis rules*):

$$CF_{gejala} = CF[user] * CF [pakar] \quad (5)$$

2. Apabila terdapat kaidah dengan kesimpulan yang serupa (*similary concluded rules*) atau lebih dari satu gejala, maka CF selanjutnya dihitung dengan persamaan:

$$CF_{combine} = CF_{old} * CF_{gejala} * (1 - CF_{old}) \quad (6)$$

3. Sedangkan untuk menghitung persentase tergapad penyakit, digunakan persamaan:

$$persentase = CF_{combine} * 100 \quad (7)$$

Adapun beberapa kelebihan dari metode *certainty factor* yaitu sebagai berikut:

- a. Yang mengandung ketidak pastian, metode ini sangat cocok digunakan dalam sistem pakar.
- b. Keakuratan data dapat terjaga karena dalam sekali proses perhitungan hanya dapat mengolah 2 data saja.

Selain kelebihan, *certainty factor* juga memiliki kekurangan yaitu sebagai berikut:

- a. Masih diperdebatkan pemodelan ketidak pastian yang menggunakan metode *certainty factor*.
- b. Harus dilakukan beberapa kali pengolahan data untuk data lebih dari 2 buah.

#### 2.2.4 Penyakit

Penyakit adalah suatu keadaan abnormal dimana tubuh ataupun pikiran mengalami ketidaknyamanan atau disfungsi terhadap orang yang dipengaruhinya [16]. Penyakit tidak hanya menyerang manusia saja, tetapi dapat menyerang hewan dan juga tumbuhan. Penyakit dapat disebabkan oleh virus, bakteri, dan kuman yang di tandai dengan adanya gejala yang ditimbulkan.

#### 2.2.5 Tanaman Padi

Padi termasuk dalam suku padi-padian atau poaceae. Terna semusim, berakar serabut, batang sangat pendek, struktur serupa batang terbentuk dari rangkaian pelepah daun yang saling menopang daun sempurna dengan pelepah tegak, daun berbentuk lanset, warna hijau muda hingga hijau tua, berurat daun sejajar, tertutupi oleh rambut yang pendek

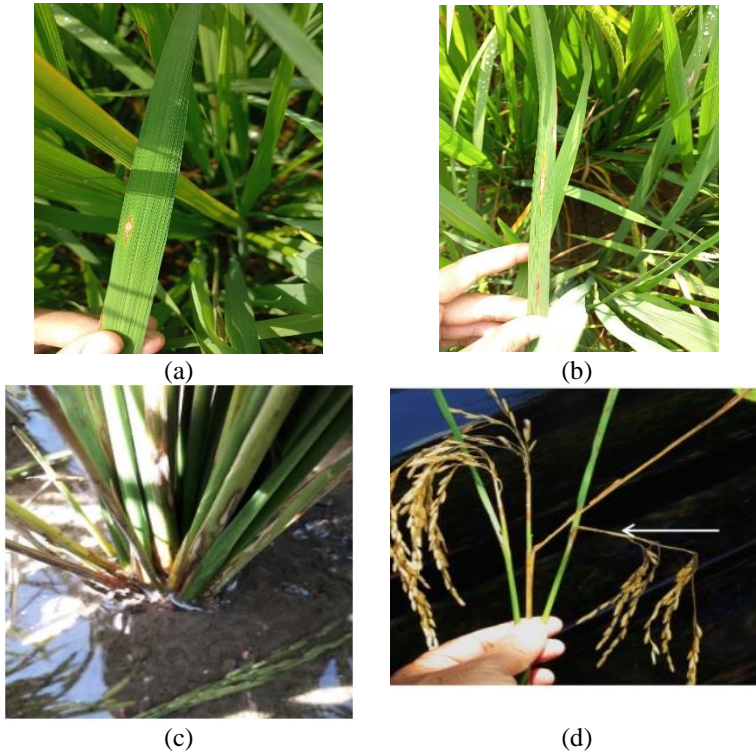
dan jarang, bagian bunga tersusun majemuk, tipe malai bercabang, satuan bunga disebut *floret* yang terletak pada satu spikelet yang duduk pada panikula, tipe buah bulir atau kariopsis yang tidak dapat dibedakan mana buah dan bijinya, bentuknya hampir bulat hingga *lonjong*, ukuran 3 mm hingga 15 mm, tertutup oleh palea dan lemma yang dalam bahasa sehari-hari disebut sekam, struktur dominan padi yang biasa dikonsumsi yaitu jenis endospermium [17]. Padi (*Oryza sativa* L.) merupakan tanaman pangan penting penghasil beras yang masih sebagian besar menjadi makanan pokok bagi lebih dari separuh penduduk di dunia. Bahkan di Indonesia, padi merupakan salah satu tanaman utama yang ditanam oleh petani sebagai makanan pokok masyarakat [18]. Klasifikasi tanaman padi sebagai berikut:

1. Kerajaan : *Plantae*
2. Divisi : *Magnoliophyta*
3. Kelas : *Liopsida*
4. Ordo : *Cyperales*
5. Famili : *Gramineae*
6. Marga : *Oryza*
7. Spesies : *O. Sativa*

Dalam budidaya tanaman padi tidak akan terlepas dengan adanya ancaman hama ataupun penyakit yang sering menyerang tanaman tersebut. Ada beberapa penyakit tanaman padi yaitu:

## a. Blas

Untuk lebih jelasnya gejala penyakit blas dapat dilihat pada Gambar 2.2 berikut ini:



**Gambar 2. 2** (a) Penyakit blas pada malai padi (b) Penyakit blas pada pelepah padi (c) Penyakit blas pada daun padi (d) Penyakit blas pada tangkai malai padi

Gejala penyakit blas yang biasanya banyak ditemukan pada pertanaman padi di daerah endemis. Bagian-bagian tanaman padi yang rentan terhadap penyakit blas adalah daun yang menimbulkan gejala bercak daun (*leaf blas*), buku batang (*node blas*), leher malai (*neck blas*),

bulir padi (*spikelet blas*), dan kolar daun (*collar rot*). Gejala penyakit blas yang parah terdapat pada bagian buku tanaman padi dapat menyebabkan batang patah dan kematian pada bagian batang di atas buku yang terinfeksi. Infeksi yang terjadi pada daun setelah fase anakan maksimum biasanya hanya menyebabkan sedikit kehilangan hasil, namun infeksi pada awal pertumbuhan sering menyebabkan tanaman puso, terutama jika ditanam varietas rentan terhadap penyakit blas [19].

Infeksi penyakit blas pada daun padi menyebabkan gejala bercak berbentuk belah ketupat dengan dua ujungnya agak meruncing. Gejala awal dapat dilihat dengan adanya bercak kecil berwarna hijau gelap keabu-abuan. Bercak cepat melebar pada varietas rentan, khususnya bila cuaca lembab dan hangat. Pada bagian tengah bercak berwarna putih keabu-abuan dengan tepi berwarna cokelat tua. Jika beberapa bercak daun tumbuh meluas dan menyatu maka daun-daun padi yang terinfeksi menjadi kering dan tanaman mati [19].

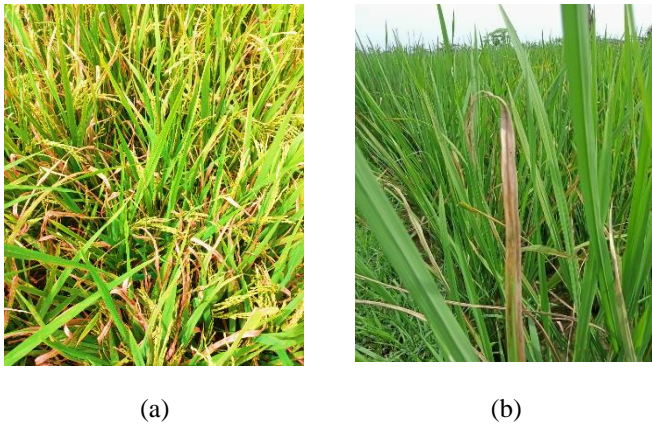
Gejala penyakit blas pada stadia generatif, terutama pada saat pengisian biji, sering ditemukan pada bagian leher malai. Gejala penyakit blas pada leher malai berwarna cokelat kehitaman (gosong) seperti terkena letupan api (blas). Kondisi penularan yang parah dilihat saat leher malai menjadi busuk, kering, dan mudah patah, aliran fotosintesis ke bulir terhambat. Apabila malai terinfeksi pada stadia masak susu maka bulir padi banyak yang hampa. Makin tinggi intensitas penyakit, makin banyak leher malai yang patah dan jatuh, sehingga makin banyak pengurangan hasil panen [19].

Penyakit blas disebabkan oleh infeksi jamur yang terdiri dari dua jenis, yaitu yang berkembangbiak secara seksual dan aseksual. Patogen blas yang mampu berkembang biak secara seksual adalah *Magnaporthe oryzae* Cav, sedangkan yang berkembang biak secara aseksual adalah *Pyricularia oryzae* Cav. Jamur *P. oryzae* menyebar di alam dengan membentuk spora berupa konidia yang berbentuk lonjong yang meruncing pada ujungnya. Suhu yang baik untuk perkecambahan konidium dan pembentukan apresorium yaitu 25 - 30 °C. Dibutuhkan waktu selama 6 - 10 jam jamur penyebab penyakit blas untuk dapat menginfeksi tanaman. Varietas padi yang semula tahan terhadap suatu ras

*P. oryzae*, dapat menjadi rentan terhadap ras baru yang tidak dikenali oleh gen ketahanannya, walaupun masih tetap tahan terhadap ras patogen semula. Pemantauan ras patogen blas di Indonesia dilakukan dengan menggunakan tujuh varietas deferensial [19].

b. Hawar Daun Bakteri (Kresek)

Untuk lebih jelasnya gejala penyakit hawar daun bakteri (kresek) dapat dilihat pada Gambar 2.3 berikut ini:



**Gambar 2. 3 Gejala Penyakit Hawar Daun Bakteri (Kresek)**

Hawar daun bakteri (*Bacterial Leaf Blight / BLB*) adalah penyakit yang disebabkan oleh infeksi bakteri (*Xanthomonas oryzae pv. oryzae*) pada tanaman padi. Penyakit ini menyerang daun, tangkai daun, dan bunga padi. Gejala awalnya berupa bercak air kecil pada daun yang kemudian berkembang menjadi bercak berwarna abu-abu kehijauan dengan tepi yang tak beraturan. Pada kondisi yang parah, daun padi bisa menjadi kuning dan mati. Hawar daun bakteri dapat menyebar melalui air, angin, dan serangga. Pencegahan dan pengendalian penyakit ini meliputi penggunaan bibit sehat, sanitasi lahan, rotasi tanaman, penggunaan pupuk yang tepat, dan aplikasi pestisida jika diperlukan [19]. Penyakit ini mengakibatkan kegagalan hasil panen sebesar 35,8%.

Hawar daun bakteri merupakan penyakit yang dapat menginfeksi bibit dan tanaman tua. Penyakit HDB menghasilkan 2 gejala yang khas, yaitu kresek dan hawar daun.

Penyakit kresek merupakan gejala terjadi pada tanaman mulai di pembibitan sampai berumur < 30 hari. Daun-daun berwarna hijau kelabu, melipat dan menggulung, layu dan mati, mirip tanaman terserang penggerek batang dan terkena air panas. Sedangkan gejala hawar dijumpai pada pertanaman tahap tumbuh anakan hingga tahap pemasakan. Setelah infeksi berlanjut, daun menguning hingga seperti jerami sampai bibit mati. Bibit yang sudah terinfeksi hingga layu (kresek) hampir sama dengan kerusakan awal penggerek batang. Pada tanaman yang lebih tua, memiliki ciri-ciri luka yang diawali dengan strip basah hingga kekuning-kuningan pada ujung daun atau helaian daun. Luka berubah dari warna kuning menjadi putih dan menyerang pada daun secara cepat sehingga mati dengan cepat. Luka ini kemudian berwarna keabuan akibat pertumbuhan berbagai jenis jamur saprofit. Malai menjadi hampa tapi pertumbuhan tanaman tidak akan terganggu walaupun dalam keadaan parah. Suhu tinggi, kelembapan tinggi, cuaca hujan dan pemakaian pupuk Nitrogen berlebihan mendorong perkembangan dan penyebaran Hawar Daun Bakteri [20].



c. Hawar Pelelah

Untuk lebih jelasnya gejala penyakit hawar pelelah dapat dilihat pada Gambar 2.4 berikut ini:



**Gambar 2. 4** Gejala Penyakit Hawar Pelelah

Penyakit hawar pelelah yang disebabkan oleh jamur *Rhizoctonia solani* Kuhn, merupakan penyakit padi yang sering ditemukan setiap musim tanam. Penyakit berkembang dengan tingkat keparahan bervariasi dan diduga berkaitan erat dengan asupan teknologi yang diterapkan petani. Pada pertanaman, varietas unggul padi biasanya memberikan respons yang kurang tahan terhadap penyakit hawar pelelah. Penyakit ini disebabkan oleh patogen yang mempunyai inang luas sehingga sifat ketahanan secara genetik sulit ditemukan. Pada varietas padi yang mempunyai tipe tanaman pendek beranakan banyak dan berdaun lebat penyakit hawar pelelah terlihat berkembang parah, hal ini diduga dipicu oleh kondisi lingkungan di sekitar tanaman yang lebih hangat dan lembap. Oleh karena itu, penyakit hawar pelelah perlu diperhatikan dalam praktik budidaya padi di daerah tropik karena dapat menurunkan hasil secara nyata [21].

d. Bercak Daun Cokelat

Untuk lebih jelasnya gejala penyakit bercak daun cokelat dapat dilihat pada Gambar 2.5 berikut ini:



**Gambar 2. 5** Gejala Penyakit Hawar Bercak Daun Cokelat

Gejala penyakit bercak coklat yaitu pada daun dan glumae (bagian bulir) dapat juga muncul pada pelepah daun, cabangcabang malai bibit yang muda dan batang. Bercak pada daun yang umumnya berbentuk oval, berukuran variatif serta serangannya seringkali tersebar di seluruh permukaan daun.

Bercak berwarna coklat, dilingkari dengan warna abu bagian tengah bercak bulat berwarna putih. Gejala yang masih muda berupa bintik-bintik coklat atau coklat keabuan. Pada varietas peka bercak akan lebih lebar, berukuran mencapai 1 cm atau lebih. Bercak daun seringkali memenuhi permukaan daun sehingga mengakibatkan daun layu.

Penyakit bercak coklat umumnya terdapat pada tanaman padi di Indonesia. Bahkan penyakit ini terdapat disemua negara penanam padi, baik di tropik maupun daerah yang iklim sedang. Penyakit ini disebabkan oleh *Drechslera oryzae* [19].

### **2.2.6 Website**

*Website* merupakan kumpulan halaman yang berisi informasi dan data tertentu yang dapat memudahkan surfer [22]. *Website* dapat diakses dengan menggunakan *browser* seperti *google chrome*, *mozilla firefox*, *safari*, *opera* dan *internet explorer*. *Website* memiliki banyak manfaat salah satunya sebagai wadah untuk membagikan sumber informasi.

### **2.2.7 Framework**

*Framework* adalah kumpulan intruksi-intruksi yang dikumpulkan dalam *class* dan *function-function* dengan fungsi masing-masing untuk memudahkan *developer* dalam memanggilnya tanpa harus menuliskan *syntax* program yang sama berulang-ulang serta dapat menghemat waktu [23]. *Framework* banyak digunakan oleh *developer* untuk memudahkan pengembangan *website*. Banyak sekali jenis *framework* yang saat ini digunakan oleh *developer* seperti *laravel*, *codeigniter*, *symfony*, *zend*, *phalcon*, *cakePHP*, *yii*, dan *fuelPHP*.

### **2.2.8 Laravel**

*Laravel* merupakan sebuah *Framework PHP (Hypertext Preprocessor)* yang dirilis dibawah lisensi MIT, dibangun dengan konsep MVC (*Model, View, Controller*). *Laravel* adalah *FrameworkPHP* yang dibuat oleh Taylor Otwell dan dirilis pertama kali pada tahun 2011. *Laravel* memiliki banyak fitur modern yang membantu dalam proses pengembangan *website* seperti *artisan*, *blade* *template engine*, *database migration*, *pagination*, dan *eloquent ORM (Object Relation Mapping)* [24].

### 2.2.9 XAMPP

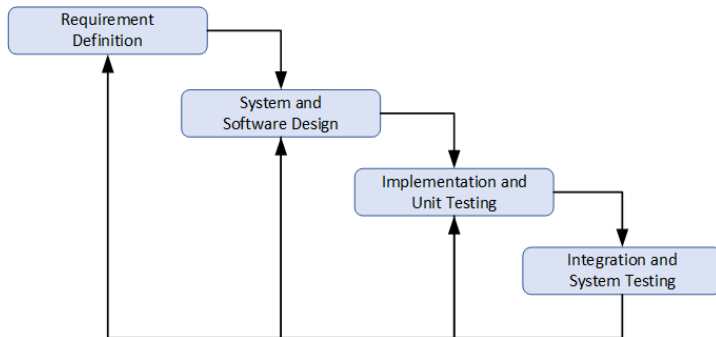
*XAMPP* adalah aplikasi yang berfungsi sebagai server yang berdiri sendiri (localhost), yang terdiri beberapa program antara lain : *Apache HTTP Server*, *MySQL database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP sendiri merupakan singkatan dari X empat sistem operasi, yang meliputi *Apache*, *MySQL*, *PHP* dan *Perl*. Program ini tersedia dalam GNU (*General Public License*), merupakan web server yang mudah untuk digunakan yang dapat menampilkan halaman web yang dinamis [22].

### 2.2.10 PHP

PHP (*Hypertext Preprocessor*) merupakan salah satu bahasa pemrograman yang berjalan dalam sebuah *web server* dan berfungsi sebagai pengolah data pada sebuah server. Data yang dikirim oleh *user client* akan diolah dan disimpan pada database web server dan dapat ditampilkan kembali apabila diakses. Untuk menjalankan kode-kode program PHP, file harus di *upload* kedalam server. *Upload* adalah proses mentransfer data atau file dari komputer *client* ke dalam *web server*. Untuk membuat *website* yang dinamis dan mudah di *update* setiap saat dari *browser*, dibutuhkan sebuah program yang mampu mengolah data dari komputer *client* atau dari komputer *server* itu sendiri sehingga mudah dan nyaman disajikan di *browser*. Salah satu program yang dapat dijalankan di server dan cukup andal adalah PHP [25].

### 2.2.11 Metode Pengembangan Sistem

Dalam penelitian ini metodologi yang akan digunakan dalam pengembangan sistem adalah metodologi waterfall dengan mengadopsi empat tahapan dari Sommerville 2011 lebih jelasnya dapat dilihat pada Gambar 2.6 sebagai berikut :



**Gambar 2. 6** Metode Waterfall [4]

#### 1. Analisis Kebutuhan dan Perangkat Lunak

Dalam langkah ini merupakan analisis terhadap kebutuhan sistem. Pengumpulan data dalam tahap ini bisa dilakukan dengan cara peneliti wawancara atau studi literatur. Seorang sistem analis akan menggali informasi sebanyak – banyak nya kepada pengguna sehingga sistem yang dibangun akan melakukan tugas – tugas sesuai dengan kemauan pengguna [26]. Pada tahap ini peneliti datang ke tempat observasi untuk melakukan wawancara kepada petugas pengamat organisme pengganggu tanaman di Balai Penyuluhan Pertanian Kecamatan Ajibarang untuk mengumpulkan informasi dan data yang digunakan dalam perancangan sistem, agar sistem yang dibangun yang sesuai dengan keinginan pengguna.

#### 2. Desain Sistem

Proses desain sistem merupakan tahap perancangan perangkat lunak yang diperkirakan sebelum ke tahap koding. Proses ini berfokus pada struktur data, representasi *interface*, dan detail algoritma prosedural. Tahap ini akan menghasilkan *software requirement*. Dokumen inilah

yang digunakan seorang programmer untuk melakukan aktifitas pembuatan sistem [26]. Tahap ini peneliti menerapkan tahap analisis kebutuhan (*requirement*) yang dirancang menggunakan perangkat keras maupun perangkat lunak.

### **3. Penulisan Kode Program**

Tahap ini merupakan tahap dimana seorang programmer melakukan koding berdasarkan data yang diperoleh dari tahap analisis kebutuhan dan desain sistem. Tahap inilah seorang programmer akan menghasilkan sebuah sistem nyata dan melakukan pengkodean sampai selesai. Setelah itu dilakukan pengujian koding per unit sebelum dilakukan pengujian sistem keseluruhan.




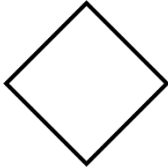

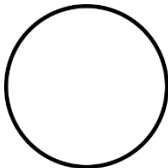
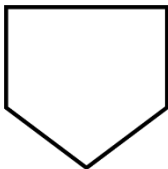
### **4. Pengujian Program**

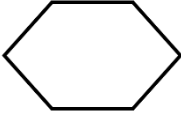
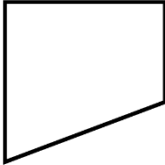
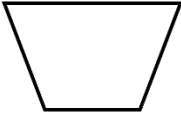

Tahap ini merupakan tahap akhir yaitu pengujian program. Setelah melakukan beberapa tahapan seperti analisis kebutuhan, desain sistem, kode program maka dilakukan pengujian program. Pengujian program menggunakan *blackbox testing*, pengujian dengan *blackbox testing* ini pengujian hanya melakukan pengujian interface tanpa mengetahui penulisan kode program.

#### **2.2.12 Flowchart**

*Flowchart* atau bagan alir adalah penggambaran secara grafik dari langkah-langkah dan urutan prosedur dari suatu program. Simbol *flowchart* dipakai sebagai alat untuk membantu dalam mendefinisikan sebuah proses. Proses dalam *flowchart* digambarkan dari halaman atas ke bawah dan dari kiri ke kanan. Aktifitas yang digambarkan kemudian didefinisikan dengan jelas sehingga dimengerti oleh pembacanya [27].

Tabel 2. 3 Simbol *Flowchart*

No	Gambar Simbol	Nama	Keterangan
1.		Terminal	Menyatakan permulaan atau akhir suatu program.
2.		Arus/ <i>Flow</i>	Untuk menyatakan jalanya arus suatu proses.
3.		Proses	Menunjukkan sebuah kegiatan atau proses yang dilakukan sistem.
4.		<i>Decision</i> /Logika	Menunjukan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan (ya/tidak).
5.		<i>Input</i> / <i>output</i>	Menyatakan proses <i>input/output</i> tanpa tergantung dengan jenis peralatannya.
6.		<i>Connector</i>	Menyatakan sambungan dari suatu proses ke proses lainnya dalam halaman/lembar yang sama.
7.		<i>Offline Connector</i>	Menyatakan sambungan dari suatu proses ke proses lainnya dalam halaman/lembar yang berbeda.

8.		<i>Predefined Proses</i>	Menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.
9.		Manual <i>input</i>	Memasukan data secara manual dengan menggunakan <i>online keyboard</i> .
10.		Manual	Menyatakan suatu Tindakan (proses) yang tidak dilakukan oleh komputer (manual).
11.		<i>Document</i>	Mencetak laporan ke printer.

### 2.2.13 UML (*Unified Modeling Language*)

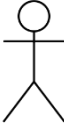
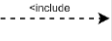



*Unified Modeling Language* merupakan salah satu metode pemodelan visual yang digunakan dalam perancangan dan pembuatan sebuah *software* yang berorientasikan pada objek. UML merupakan sebuah standar penulisan atau semacam *blueprint* dimana didalamnya termasuk sebuah bisnis proses, penulisan kelas-kelas dalam sebuah bahasa yang spesifik [27].

#### A. *Use Case Diagram*

*Use case diagram* digunakan untuk menggambarkan fungsi dasar dari sebuah sistem informasi. *Use case* mendeskripsikan cara sistem bisnis berinteraksi dengan lingkungannya [28]. Berikut adalah elemen-elemen dari use case diagram:



**Tabel 2.2** Simbol *Use Case Diagram*


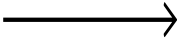



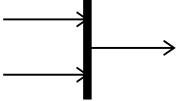
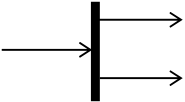
No	Simbol	Fungsi
1.		Actor/role adalah orang atau sistem lain yang berinteraksi dengan sistem saat ini.
2.		Memasukkan satu use case dalam use case lainnya. Perilaku (behavior) yang harus terpenuhi agar sebuah event dapat terjadi, di mana kondisi ini sebuah use case adalah bagian dari use case lainnya. Tanda panah mengarah dari base use case (pusat) menuju ke use case yang di-includ.
3		Menggambarkan navigasi antar class, berupa banyak objek lain yang berhubungan dengan satu objek, dan apakah suatu class menjadi bagian dari class lainnya.
4.		Berisi nama dari sistem yang diletakkan di dalam atau di bagian atas boundary. Mewakili ruang lingkup sistem. Actor berada di luar ruang lingkup sistem.
5.		Use case adalah bagian utama dari fungsionalitas sistem. Bisa extend (memperluas) use case lainnya. Ditempatkan di dalam system boundary (batasan sistem). Dilabeli dengan kata kerja – frase kata benda.

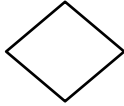
### B. *Activity Diagram*

*Activity diagram*, dalam bahasa Indonesia diagram aktivitas, yaitu diagram yang dapat memodelkan proses - proses yang terjadi pada sebuah sistem. Runtutan proses dari suatu sistem digambarkan secara vertikal. Activity diagram merupakan pengembangan dari *Use Case* yang memiliki alur aktivitas. Alur atau aktivitas berupa bisa berupa runtutan menu-menu atau proses bisnis yang terdapat di dalam sistem tersebut[29].

Berikut adalah elemen-elemen dari sequence diagram:

Tabel 2. 4 Simbol *Activity Diagram*


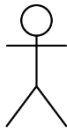
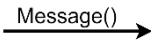
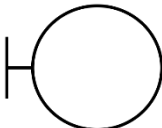
No	Gambar Simbol	Nama Simbol	Fungsi Simbol
1.		Activity	Menyatakan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
2.		Control Flow	Menunjukkan Urutan Eksekusi.
3.		Object Flow	Menunjukkan aliran objek dari sebuah action atau activity ke action.
4.		Start Point	Menyatakan bahwa sebuah objek dibentuk atau diawali.
5.		End Point	Menyatakan bahwa sebuah objek dibentuk atau diakhiri.
6.		Join/Penggabungan	Menyatakan untuk menggabungkan kembali activity atau action yang parallel.
7.		Fork	Menyatakan untuk memecah behavior menjadi activity atau action yang parallel.

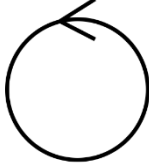
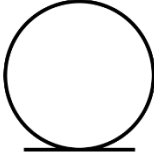
8.		Decision	Menunjukkan penggambaran suatu keputusan/tindakan yang harus di ambil pada kondisitertentu.
----	---	----------	---

### C. *Sequence Diagram*

*Sequence diagram* menunjukkan pesan yang lewat di antara objek untuk use case tertentu dari waktu ke waktu. *Sequence diagram* mengilustrasikan objek-objek yang berpartisipasi di dalam suatu *use case* [28]. Berikut adalah elemen-elemen dari *sequence diagram*:

**Tabel 2. 5** Simbol *Sequence Diagram*

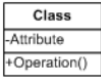
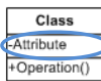
No	Simbol	Fungsi
1.		Merupakan persegi panjang sempit panjang yang ditempatkan di atas lifeline. Menunjukkan kapan suatu objek mengirim atau menerima pesan.
2..		Adalah orang atau sistem yang memperoleh manfaat dari dan berada di luar sistem. Berpartisipasi dalam suatu urutan dengan mengirim dan / atau menerima pesan. Ditempatkan di bagian atas diagram.
3.		Spesifikasi dari komunikasi antar objek yang memuat informasi – informasi tentang aktivitas yang terjadi.
4.		Menggambarkan tampilan program.

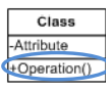
5.		Menggambarkan controller.
6.		Digunakan untuk menggambarkan hubungan kegiatan yang akan dilakukan.

#### D. *Class Diagram*

*Class diagram* adalah model statis yang menunjukkan kelas dan hubungan di antara kelas yang tetap konstan dalam sistem dari waktu ke waktu. *Class diagram* menggambarkan kelas, yang meliputi perilaku dan keadaan, dengan hubungan antar kelas [28]. Berikut adalah elemen-elemen dari class diagram:

**Tabel 2. 6** Simbol *Class Diagram*

No	Simbol	Fungsi
1.		Class, Mewakili orang, tempat, atau hal-hal yang dibutuhkan sistem untuk menangkap dan menyimpan informasi. Memiliki nama yang diketik dengan huruf tebal dan berpusat di bagian atas kotak. Memiliki daftar atribut di kotak tengahnya. Memiliki daftar operasi di kotak bawahnya. Tidak secara eksplisit menunjukkan operasi yang tersedia untuk semua kelas..
2.		Attribute, merupakan properti yang menggambarkan keadaan suatu objek. Dapat diturunkan dari atribut lain, ditampilkan dengan menempatkan garis miring sebelum nama atribut.

3	 <p>The diagram shows a rectangular box representing a class. Inside the box, the word 'Class' is at the top. Below it, '-Attribute' is written. At the bottom, '+Operation()' is written and circled with a blue oval.</p>	<p>Operation, mewakili tindakan atau fungsi yang dapat dilakukan oleh kelas. Dapat diklasifikasikan sebagai konstruktor, permintaan, atau operasi pembaruan. Termasuk tanda kurung yang mungkin berisi parameter atau informasi yang diperlukan untuk melakukan operasi.</p>
---	--	--

### 2.2.14 ERD (*Entity Relationship Diagram*)

ERD merupakan pemodelan awal basis data yang dikembangkan berdasarkan teori himpunan dalam bidang matematika untuk pemodelan basis data relasional(kutipan). Simbol-simbol dalam ERD (*Entity Relationship Diagram*) adalah sebagai berikut:

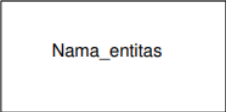
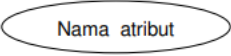

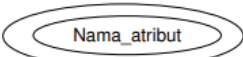
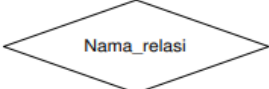
- Entitas: suatu yang nyata atau abstrak yang mempunyai karakteristik dimana kita akan menyimpan data.
- Atribut: ciri umum semua atau sebagian besar instansi pada entitas tertentu.
- Relasi: hubungan alamiah yang terjadi antara satu atau lebih entitas.
- Link: garis penghubung atribut dengan kumpulan entitas dan kumpulan entitas dengan relasi.


Kardinalitas Relasi:

- Satu ke satu (*One to One*)  
Setiap elemen dari Entitas A berhubungan paling banyak dengan elemen pada Entitas B. Demikian juga sebaliknya setiap elemen B berhubungan paling banyak satu elemen pada Entitas A.
- Satu ke banyak (*One to Many*)  
Setiap elemen dari Entitas A berhubungan dengan maksimal banyak elemen pada Entitas B. Dan sebaliknya setiap elemen dari Entitas B berhubungan dengan paling banyak satu elemen di Entitas A.
- Banyak ke satu (*Many to One*)  
Setiap elemen dari Entitas A berhubungan paling banyak dengan satu elemen pada Entitas B. Dan sebaliknya setiap elemen dari Entitas B berhubungan dengan maksimal banyak elemen di entitas A.
- Banyak ke banyak (*Many to Many*)

Setiap elemen dari Entitas A berhubungan maksimal banyak elemen pada Entitas B demikian sebaliknya.  
Berikut merupakan simbol-simbol yang digunakan pada ERD dengan notasi Chen [30] :

**Tabel 2. 7** Simbol pada ERD

No	Simbol	Deskripsi
1	Entitas / Entity 	Entitas merupakan data inti yang akan disimpan; bakal table pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi computer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel
2	Atribut 	Field atau kolom data yang butuh disimpan dalam suatu entitas.
3	Atribut Kunci Primer 	Field atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses record yang diinginkan; biasanya berupa id; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama).
4	Atribut Multinilai / Multivalued 	Field atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.
5	Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja.

6	Asosiasi / Association 	Penghubung antara relasi dan entitas dimana di kedua ujungnya memiliki multiplicity kemungkinan jumlah Pemakaian kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut dengan one to many menghubungkan entitas A dan entitas B.
---	---	---

### 2.2.15 Basis Data (*Database*)

Basis data adalah suatu kumpulan data terhubung yang disimpan secara bersama-sama pada suatu media, tanpa adanya suatu kerangkapan data, sehingga mudah untuk digunakan kembali, dapat digunakan oleh satu atau lebih program aplikasi secara optimal, data disimpan tanpa mengalami ketergantungan pada program yang akan menggunakannya, data disimpan sedemikian rupa sehingga apabila ada penambahan, pengambilan dan modifikasi data dapat dilakukan dengan mudah dan terkontrol [31]. Basis data berfungsi menyimpan dan mengelola data berupa tabel – tabel yang dapat diakses saat menggunakan sistem. Aplikasi database *open source* yang banyak diminati saat ini, contohnya *MySQL*, *Oracle*, *Microsoft Access*, *PostgreSQL*, *SQLite* dan lain sebagainya. Database dapat mempermudah dalam mengakses dan mengambil data karena semua data tertata dengan baik. Selain itu database digunakan untuk berbagi data dengan sesama pengguna lainnya.

### 2.2.16 Notifikasi Email

Email (*electronic mail*) adalah surat elektronik, merupakan layanan internet populer untuk mengirimkan berita, informasi, surat, dokumen, gambar, dan file. Email memudahkan pengguna dalam mengelola surat menyurat dan lampirannya secara elektronik, sangat praktis digunakan oleh semua orang [32]. Sedangkan notifikasi email merupakan pemberitahuan melalui surat elektronik tentang peristiwa penting terkait akun atau aktivitas email. Contohnya: email masuk baru, verifikasi akun, perubahan kata sandi, aktivitas masuk mencurigakan, pemberitahuan aktivitas, pengingat jadwal, kuota penyimpanan, dan

perubahan kebijakan layanan. Ini membantu pengguna tetap terhubung dengan informasi penting dan perubahan dalam akun email mereka.

### **2.2.17 Notifikasi Whatsapp**

Notifikasi WhatsApp merupakan pemberitahuan melalui aplikasi pesan WhatsApp tentang pesan baru, panggilan masuk, atau peristiwa penting lainnya. Ini membantu pengguna mengetahui tentang aktivitas baru di aplikasi tanpa perlu membukanya. Penggunaan notifikasi whatsapp ini dapat membantu pengguna dalam mendapatkan notifikasi mengenai berhasil buat akun dan atau menerima hasil diagnosis sistem.



**~ Halaman ini sengaja dikosongkan ~**