

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 PENELITIAN TERDAHULU**

Penelitian yang dilakukan oleh Zaenuddindan Mirza yogi Kurniawan dengan judul SISTEM PENUNJANG KEPUTUSAN PEMILIHAN VARIETAS JAGUNG BERDASARKAN KESESUAIAN LAHAN DENGAN METODE PROFILE MATCHING bertujuan untuk produksi jagung dalam negeri masih bisa meningkat melalui perluasan areal tanam jagung utamanya di luar daerah jawa. Profile matching dapat memberikan rekomendasi berdasarkan variabel kesesuaian lahan dengan varietas. Sistem penunjang keputusan yang dihasilkan telah memiliki fitur yang dibutuhkan untuk merekomendasikan varietas jagung yang sesuai dengan lahan yang akan ditanami. Perbedaan antara penelitian terdahulu sama yang diteliti ama si penulis yaitu terletak di kriteria tanaman yang ditelitinya

##### **2.1.1 Konsep Dasar Sistem**

Winanti, Marlina Budhiningtias, and Dewan Rahadyan dalam Jurnal Teknologi dan Informasi (2014), Konsep Dasar Sistem informasi merupakan kumpulan perangkat lunak komputer serta perangkat manusia yang akan mengolah data menggunakan perangkat keras dan perangkat keras tersebut.[1]

##### **2.1.2 Definisi Sistem**

Andri Kristanto (2008) “Sistem merupakan kumpulan elemen - elemen yang saling terkait dan bekerja sama untuk memproses masukan (input) yang ditujukan kepada sistem tersebut dan mengubah masukan tersebut sampai menghasilkan keluaran (output) yang diinginkan.[2]

##### **2.1.3 Sistem Pendukung Keputusan**

Menurut Annisa Paramitha Fadillah (2018) Sistem pendukung Keputusan (SPK) adalah sebuah sistem yang mampu memberikan kemampuan pemecahan

masalah maupun kemampuan pengkomunikasian untuk masalah dengan kondisi semi terstruktur dan tak terstruktur.[3]

#### **2.1.4 Manfaat Sistem Pendukung Keputusan**

Tujuan dari pengembangan aplikasi pengambilan keputusan seperti yang dibahas diatas, yaitu untuk membantu menentukan keputusan yang akan diambil. Berikut ini beberapa manfaat membuat dan menggunakan sistem pendukung keputusan atau spk

1. **Membantu menentukan keputusan**
2. **Mengurangi kesalahan pengambilan keputusan**
3. **Penghematan waktu**
4. **Meningkatkan efektifitas**

## **2.2 Pengertian Lahan**

Djaenudin, e. A (2019) Lahan merupakan bagian dari bentang alam (landscape) yang mencakup pengertian lingkungan fisik termasuk iklim, topografi/relief, tanah, hidrologi dan keadaan vegetasi alami (natural vegetation). Lahan dalam pengertian yang lebih luas termasuk yang telah dipengaruhi oleh berbagai aktivitas flora, fauna dan manusia, baik di masa lalu maupun saat sekarang, seperti lahan rawa dan pasang surut yang telah direklamasi atau tindakan konservasi tanah pada suatu lahan tertentu.

Karakteristik lahan adalah sifat lahan yang dapat diukur atau diestimasi. Dari beberapa pustaka disebutkan bahwa penggunaan karakteristik lahan untuk keperluan evaluasi lahan bervariasi. Karakteristik dalam menilai lahan adalah temperatur rata-rata tahunan, curah hujan (tahunan atau pada masa pertumbuhan), kelembaban udara, drainase, tekstur, bahan kasar, kedalaman efektif, kematangan dan ketebalan gambut, KTK, KB, pH, C organik, total N, P<sub>2</sub>O<sub>5</sub>, K<sub>2</sub>O, salinitas, alkalinitas, kedalaman sulfidik, lereng, batuan di permukaan, singkapan batuan, bahaya longsor, bahaya erosi serta tinggi dan lama genangan.[4] Uraian masing – masing disajikan pada tabel 2.1

Tabel 2.1 karakteristik lahan

No.	Karakteristik Lahan	Uraian
1	Temperatur Rata-rata Tahunan	Suhu udara rata – rata tahunan ( $^{\circ}\text{C}$ )
2	Curah Hujan	jumlah curah hujan tahunan atau curah hujan pada masa pertumbuhan (mm)
3	Kelembaban Udara	merupakan tingkat kebasahan udara atau jumlah uap air yang di udara (%).
4	Drainase	merupakan pengaruh laju perkolasi air ke dalam tanah terhadap aerasi udara dalam tanah
5	Tekstur	perbandingan butir-butir pasir (0,05 - 2,0 mm), debu (0,002 - 0,05 mm) dan liat (< 0,002 mm)
6	Bahan Kasar	bahan yang berukuran > 2 mm (%)
7	Kedalaman Efektif	kedalaman lapisan tanah yang dapat dimanfaatkan untuk perkembangan perakaran tanaman (cm)
8	Kematangan Gambut	tingkat kandungan serat, dimana semakin tinggi kandungan serat, maka semakin rendah tingkat kematangan gambut. Tingkat kematangan gambut dibedakan atas: saprik (matang), setengah matang (hemik), dan belum matang (fibrik)
9	Ketebalan Gambut	tebal lapisan gambut (cm)
10	KTK Tanah	kemampuan tanah mempertukarkan kation (me/100 g tanah)
11	Kejenuhan Basa (KB)	jumlah basa-basa terekstrak $\text{NH}_4\text{OAc}$ pada setiap 100 g contoh tanah
12	PH Tanah	merupakan $[\text{H}^+]$ di dalam larutan tanah, semakin tinggi $[\text{H}^+]$ , maka nilai pH semakin masam, sebaliknya semakin rendah $[\text{H}^+]$ , maka pH semakin basis
13	C Organik	kandungan karbon organik di dalam tanah (%)
14	Total N	total kandungan N dalam tanah (%)
15	$\text{P}_2\text{O}_5$	kandungan $\text{P}_2\text{O}_5$ terekstrak HCl 25% dalam tanah (mg/100 g tanah)
16	$\text{K}_2\text{O}$	kandungan $\text{K}_2\text{O}$ terekstrak HCl 25% dalam tanah (mg/100 g tanah)

### 2.3 Tanaman Pangan

Selviani Kusnadi & Lalan Jaelani (2019) Di dalam ilmu pengetahuan khususnya pertanian ada pengertian tanaman pangan dan ciricirinya yang harus diketahui dan dibedakan dengan jenis tanaman lain. Dalam pengelompokan tanaman terdapat pengelompokan jenis salah satunya adalah jenis tanaman pangan. Pengertian tanaman pangan sendiri adalah tanaman yang mengandung karbohidrat dan protein utama sebagai sumber makanan pokok sumber energi manusia sehari-hari. [5]

## 2.4 Kesesuaian Lahan

Kesesuaian lahan adalah kecocokan sebidang lahan untuk penggunaan tertentu. Sebagai contoh lahan sangat sesuai untuk sawah irigasi, lahan cukup sesuai untuk pertanian tanaman tahunan atau pertanian tanaman semusim. Kesesuaian lahan tersebut dapat dinilai untuk kondisi saat ini (present) atau setelah diadakan perbaikan (improvement). Secara spesifik, kesesuaian lahan adalah kesesuaian sifat-sifat fisik lingkungan, yaitu iklim, tanah, topografi, hidrologi atau drainase untuk usaha tani.

## 2.5 Metode Profile Matching

Menurut Handojo (2011), Proses *Profile Matching* secara garis besar merupakan proses membandingkan antara nilai data aktual dari suatu profil yang akan dinilai dengan nilai profil yang diharapkan, sehingga dapat diketahui perbedaan kompetensinya (disebut juga *gap*), semakin kecil nilai *gap* yang dihasilkan maka bobot nilainya akan semakin besar.[6]

Tahapan-tahapan yang terdapat pada metode profile matching antara lain:

1. Menentukan kriteria-kriteria yang dibutuhkan
2. Menentukan sub kriteria-kriteria yang digunakan untuk penilaian
3. Menghitung nilai gap untuk masing-masing kriteria, proses perhitungan nilai gap yaitu dengan menentukan selisih antar atribut dengan standar profil target rumus menghitung nilai gap:

$$\text{GAP} = \text{NILAI Profil} - \text{NILAI Sample}$$

4. Pembobotan nilai gap. Setelah nilai *gap* dihitung, setiap nilai atribut akan diberikan bobot nilai dengan patokan pada tabel bobot nilai *gap*. Tabel 2.2 merupakan penjelasan mengenai tabel bobot nilai *gap* pada metode *Profile Matching*.

Tabel 2.2 Bobot Nilai GAP

No	Selisih GAP	Bobot Nilai	Keterangan
----	-------------	-------------	------------

1	0	5	Kompetensi sesuai dengan yang dibutuhkan (tidak ada selisih)
2	1	4.5	Kompetensi individu kelebihan 1 tingkat/level
3	-1	4	Kompetensi individu kekurangan 1 tingkat/level
4	2	3.5	Kompetensi individu kelebihan 2 tingkat/level
5	-2	3	Kompetensi individu kekurangan 2 tingkat/level
6	3	2.5	Kompetensi individu kelebihan 3 tingkat/level
7	-3	2	Kompetensi individu kekurangan 3 tingkat/level
8	4	1.5	Kompetensi individu kelebihan 4 tingkat/level
9	-4	1	Kompetensi individu kekurangan 4 tingkat/level

5.

Pengelompokan Core Factor dan *Secondary Factor*. Setelah menentukan bobot nilai *gap*, langkah selanjutnya adalah menghitung *Core Factor* dan *Secondary Factor* dari masing-masing kriteria. Sebelum dilakukan perhitungan, terlebih dahulu tentukan faktor mana yang termasuk ke dalam Core Factor (CF) dan *Secondary Factor* (SF)

- a. *Core Factor* (Faktor Utama), yaitu merupakan kriteria (kompetensi) yang paling penting atau menonjol atau paling dibutuhkan oleh suatu penilaian yang diharapkan dapat memperoleh hasil yang optimal.
- b. *Secondary Factor* (Faktor Pendukung), yaitu merupakan item-item selain yang ada pada *core factor* atau dengan kata lain merupakan faktor pendukung yang kurang dibutuhkan oleh suatu penilaian.

Tabel 2.3 rumus menghitung *Core Factor* (CF) dan *Secondary Factor* (SF)

<p>Core factor</p> $NCF = \frac{\sum NC}{\sum IC}$	<p>Secondary Factor</p> $NSF = \frac{\sum NS}{\sum IS}$
<p><b>Keterangan:</b>  NCF: Nilai rata-rata <i>core factor</i>  NC : Jumlah total nilai <i>core factor</i>  IC : Jumlah item <i>core factor</i></p>	<p><b>Keterangan:</b>  NSF: Nilai rata-rata <i>secondary factor</i>  NS: Jumlah total nilai <i>secondary factor</i>  IS: Jumlah item <i>secondary factor</i></p>

6. Perhitungan nilai total. Nilai total diperoleh dari perhitungan *Core Factor* dan *Secondary Factor* dari masing-masing aspek (kriteria) kemudian

menghitung nilai total dari masing-masing kriteria yang berpengaruh pada rekomendasi penempatan masing-masing atribut. Rumus menghitung nilai total dari masing-masing kriteria:

$$Ni = ((x)\% \times NCF) + ((x)\% \times NSF)$$

Keterangan:

Ni : Nilai total dari masing-masing kriteria

NCF : Nilai rata-rata *Core Factor*

NSF : Nilai rata-rata *Secondary Factor*

(x)% : Nilai persen untuk masing-masing *Factor*

7. Perhitungan penentuan nilai akhir atau ranking. Hasil akhir dari proses perhitungan profil matching adalah menentukan nilai ranking atribut-atribut yang ditawarkan dimana menentukan ranking sebagai berikut:

$$\text{Ranking} = \sum (X) \times Ni$$

**Keterangan:**

Ni: Nilai setiap (kriteria) penilaian

(x)%: Nilai persen yang di inputkan

## 2.6 UML (Unified Modeling Language)

Rosa dan Shalahuddin (2014), berpendapat bahwa UML (Unified Modeling Language) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisa & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. UML (Unified Modeling Language) memiliki diagram-diagram yang digunakan dalam pembuatan aplikasi berorientasi objek, diantaranya [7]:

### 2.6.1 Use Case Diagram

*Use Case Diagram* merupakan pemodelan untuk melakukan (*behavior*) sistem informai yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja

yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

#### 2.6.2 Activity Diagram

*Activity Diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Perlu diperhatikan bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

#### 2.6.3 Sequence Diagram

*Sequence diagram* atau diagram urutan adalah sebuah diagram yang digunakan untuk menjelaskan dan menampilkan interaksi antar objek-objek dalam sebuah sistem secara terperinci. Selain itu *sequence diagram* juga akan menampilkan pesan atau perintah yang dikirim, beserta waktu pelaksanaannya.

#### 2.6.4 Class Diagram

Diagram kelas adalah diagram UML yang menggambarkan kelas-kelas dalam sebuah sistem dan hubungannya antara satu dengan yang lain, serta dimasukkan pula atribut dan operasi.

#### 2.6.5 Deployment Diagram

*Deployment Diagram* adalah salah satu model diagram dalam UML untuk menguraikan artefak dalam node. Deployment diagram digunakan untuk memvisualisasikan hubungan antara *software* dan *hardware*. Secara spesifik deployment diagram dapat membuat *physical model* tentang bagaimana komponen perangkat lunak (artefak) digunakan pada komponen perangkat keras, yang dikenal sebagai node. Ini adalah salah satu diagram paling penting dalam tingkat implementasi perangkat lunak dan ditulis sebelum melakukan *coding*

## 2.7 Konsep Dasar Basis Data

### 2.7.1 Pengertian Database

Sutarman (2019), Database sekumpulan file yang saling berhubungan dan terorganisasi atau kumpulan record- record yang menyimpan data dan hubungan diantaranya[8]

Minarni dan Susanti (2018) Database adalah kumpulan dari item data yang saling berhubungan satu dengan yang lainnya yang diorganisasikan berdasarkan sebuah skema atau struktur tertentu, yang kelak dapat dimanfaatkan kembali dengan cepat dan mudah.[9]

### 2.7.2 MySQL

Sumiyati (2018) MySQL sebuah perangkat lunak sistem manajemen basis data SQL.MySQL merupakan bahasa pemrograman database terbuka yang memungkinkan untuk bisa membuat, merubah, dan mengakses beberapa database sekaligus dalam server,MySQL juga merupakan sebuah program database server yang mampu menerima dan mengirim data cepat [10]

### 2.7.3 Database Management System (DBMS)

Andi Kristanto (2018) dalam bukunya yang berjudul “Perancangan Sistem Informasi Dan Aplikasinya” Database Management System (DBMS) adalah perangkat lunak yang didesain untuk membantu dalam hal pemeliharaan dan utilitas kumpulan data dalam jumlah besar dan merupakan bagian perangkat lunak yang didesain untuk memudahkan pekerjaan pengelolaan data. Dengan menyimpan data kedalam suatu DBMS, akan lebih mudah dari pada menyimpannya ke dalam sekumpulan sistem file. DBMS dapat digunakan untuk mengolah data secara efisien.[11]

## 2.8 Konsep Dasar Pemrograman

### 2.8.1 PHP (PHP Hypertext Preprocessor)<sup>yiii</sup>



Betha Sidik (2018) menyebutkan bahwa PHP merupakan secara umum dikenal dengan sebagai bahasa pemrograman *script-script* yang membuat dokumen HTML secara *on the fly* yang dieksekusi di server web, dokumen HTML yang dihasilkan dari suatu aplikasi bukan dari dokumen HTML yang dibuat dengan menggunakan *editor text* atau editor HTML, dikenal juga sebagai bahasa pemrograman *server side*. [12]

(Janner,20019) PHP singkatan dari PHP *Hypertext Preprocessor*. Ia merupakan bahasa berbentuk skrip yang ditempatkan dalam server dan diproses di server. Hasilnyalah yang dikirim ke klien, tempat pemakai menggunakan *browser*. [13]

#### 2.8.2 Hyper Text Markup Language (HTML)

Setiawan (2018), Hypertext Markup Language (HTML) adalah sebuah bahasa markup yang digunakan untuk membuat sebuah halaman web, menampilkan berbagai informasi di dalam sebuah penjelajah web internet dan formatting hypertext sederhana yang ditulis kedalam berkas format ASCII agar dapat menghasilkan tampilan wujud yang terintegrasi. [14]

#### 2.8.3 JavaScript

Suryana dan Koesheryanti (2019) JavaScript adalah bahasa script berdasarkan pada objek yang memperbolehkan pemakai untuk mengendalikan baanyak aspek interaksi pemaikaian pada suatu dokumen HTML. Dimana objek tersebut dapat berupa suatu window, frame, URL, dokumen, form, button, atau item yang lainnya. Yang semuanya itu mempunyai properti yang saling berhubungan dengannya, dan masing-masing memiliki nama, lokasi, warna, nilai, dan atribut. [15]

#### 2.8.4 CSS (Cascading Style Sheet)

Didik Setiawan (2018) dalam bukunya yang berjudul “Buku Sakti Pemrograman Web: HTML, CSS, PHP, MySQL& Javascript.Cascading Style Sheet (CSS) merupakan salah satu kode pemrograman yang bertujuan untuk menghias dan mengatur gaya 22 tampilan / layout halaman web supaya lebih elegan dan menarik. CSS adalah sebuah teknologi internet yang direkomendasikan oleh World Wide Web Consortium atau W3C pada tahun 1996. CSS digunakan oleh web programmer dan juga web designer untuk menentukan warna, tataletak font dan semua aspek lain dari persentasi dokumen di situs mereka.[16]

## 2.9 Konsep Dasar Perangkat Lunak Pendukung

### 2.9.1 Visual Studio Code

Edy Winarno dan Ali Zaki (2018) Visual Studio Code adalah kode editor sumber yang dikembangkan oleh microsoft untuk Windows, Linux dan MacOS. Ini termasuk dukungan untuk *debugging*, kontrol git yang tertanam dan Github, penyorotan sintaksis, penyelesaian kode cerdas, snippet dan *refactoring kode*. ini sangat dapat disesuaikan memungkinkan pengguna untuk mengubah tema pintasan keyboard, preferensi dan menginstal ekstensi yang menambah fungsionalitas tambahan.[17]

### 2.9.2 XAMPP

Menurut Palit (2019), XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem informasi yang merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri.[18]