

BAB II LANDASAN TEORI

2.1 Tinjauan Pustaka

Tinjauan pustaka yang serupa dan relevan dengan penelitian bertujuan untuk membuktikan keaslian penelitian, untuk tinjauan pustaka dapat dilihat pada tabel 2.1 dibawah ini:

Tabel 2.1 Studi Pustaka

1	Judul Penelitian	Sistem Informasi Geografis Pemetaan Wilayah Kelayakan Tanam Tanaman Jagung Dan Singkong Pada Kabupaten Lampung Selatan
	Nama Peneliti (Tahun)	Alita et al., (2020)
	Permasalahan	Butuhnya pengetahuan lahan dari segi luas tanah, lokasi, potensi yang ada, dan ekosistem yang berkembang sebagai syarat yang baik untuk pertanaman jagung dan singkong
	Metode	Kualitatif
	Hasil	<i>Website</i> sistem kelayakan tanaman singkong dan jagung yang dapat menampilkan data hasil panen dan lokasi layak tanam dalam bentuk peta
	Kekurangan	Peta yang ditampilkan tidak bersifat tematik interaktif, sehingga sulit mengira seberapa luas lahan yang layak untuk ditanami

2	Judul Penelitian	Sistem Informasi Geografis Pemetaan Lahan Tanaman Melon Khususnya Di Wilayah Kota Binjai Berbasis <i>Web</i>
	Nama Peneliti (Tahun)	Putri, (2022)
	Permasalahan	Petani kesulitan dalam mencari informasi tentang lahan tanaman melon yang cocok untuk ditanami karena kurangnya akses informasi yang memadai.
	Metode	Kualitatif
	Hasil	sistem pemetaan wilayah yang menggunakan parameter indeks potensi lahan yaitu kemiringan lereng, jenis tanah, litologi (jenis batuan), curah hujan dan kerawanan bencana
	Kekurangan	Visualisasi masih ditampilkan dalam bentuk <i>wireframe</i>
3	Judul Penelitian	Sistem Informasi Geografis Untuk Pemetaan Lahan Kakao Menggunakan <i>Leaflet Js</i> Dan <i>Geojson</i>
	Nama Peneliti (Tahun)	Arifin & Supriyatna, (2023)
	Permasalahan	<i>Web portal</i> yang digunakan sebagai sistem informasi komoditas kakao di Kabupaten Pesawaran dianggap belum berhasil dalam memberikan efek yang diharapkan
	Metode	<i>Rapid Application Development (RAD)</i>

	Hasil	Aplikasi yang dapat memudahkan masyarakat, terutama dalam mencari informasi mengenai lokasi lahan kakao, serta melakukan monitoring dan evaluasi terhadap komoditas kakao yang ada di Kabupaten Pesawaran
	Kekurangan	Peta yang ditampilkan belum bersifat tematik dan interaktif, tidak ada informasi terkait komoditas
4	Judul Penelitian	Aplikasi Sistem Informasi Geografis (SIG) Pemetaan Lahan Pertanian dan Komoditas Hasil Panen di Kabupaten Sidrap Berbasis Web
	Nama Peneliti (Tahun)	Masnur et al., (2022)
	Permasalahan	Lokasi lahan pertanian belum terlalu dikenal oleh masyarakat lokal maupun pengunjung.
	Metode	<i>Prototype</i>
	Hasil	aplikasi SIG pemetaan lahan pertanian dan komoditas hasil panen di kabupaten tersebut yang berisi informasi lahan dan hasil panen
	Kelemahan	Pengunjung harus melakukan login untuk menggunakan sistem, tidak ada informasi tentang komoditas
5	Judul Penelitian	Sistem Kesesuaian Lahan Bawang Putih berdasarkan <i>Spatial Decision Tree</i>
	Nama Peneliti (Tahun)	Sitanggang et al., (2020)

Permasalahan	Keterbatasan lahan yang tersedia untuk menanam bawang putih dan mengakibatkan rendahnya produksi bawang putih.
Metode	Algoritma spasial ID3
Hasil	Aplikasi berbasis web yang menggambarkan kesesuaian lahan bawang putih di Magetan
Kekurangan	Tidak ada informasi tentang komoditas

Dari tinjauan pustaka yang telah dipaparkan, menunjukkan beberapa kekurangan dalam beberapa aspek seperti peta yang tidak interaktif, kesulitan akses bagi pengguna dan kurangnya informasi terkait komoditas yang ada. Sehingga pada penelitian ini, penting untuk mengatasi kekurangan yang ada dan akan menambahkan fitur terkait informasi benih yang unggul dan artikel terkini tentang komoditas kedelai.

2.2 Sistem Informasi Geografis

Sistem Informasi Geografi (SIG) adalah suatu sistem yang berhubungan dengan data spasial. SIG merupakan sistem informasi yang terkomputerisasi yang melibatkan berbagai prosedur terkait penyimpanan, pengolahan, dan penyajian data yang berkaitan dengan sumber daya lahan. Manfaat SIG sangat luas, dan aplikasinya mencakup bidang-bidang seperti pertanian, kehutanan, hidrologi, dan lainnya. Saat ini, SIG telah banyak digunakan dalam berbagai aplikasi di bidang pertanian, kehutanan, hidrologi, dan lainnya (Rahmawati et al., 2013).

2.3 Kedelai

Menurut Nurkholis & Styawati, (2021) Kedelai adalah salah satu sumber utama protein yang bisa dimanfaatkan untuk dikonsumsi dalam berbagai bentuk

seperti tempe, tahu, susu kedelai, dan lain-lain. Kedelai juga salah satu jenis biji-bijian yang memiliki kandungan lemak nabati dan protein yang tinggi (Setyawan & Huda, 2022). Kedelai ini telah menjadi salah satu bahan makanan pokok yang penting bagi masyarakat Indonesia.

2.4 Kabupaten Lampung Timur

Kabupaten Lampung Timur memiliki luas wilayah 5.325,03 km². Dilihat dari segi geografis, Kabupaten Lampung Timur berada di antara 105⁰15' Bujur Timur dan 105⁰15' Lintang Selatan. Kedudukan geografis Kabupaten Lampung Timur di bagian utara berbatasan dengan Kecamatan Rumbia (Kabupaten Lampung Tengah) dan Kecamatan Dente Teladas (Kabupaten Tulang Bawang). Bagian timur berbatasan dengan Laut Jawa. Bagian selatan berbatasan dengan Kecamatan Jati Agung, Candipuro dan Way Sulan (Lampung Selatan). Serta di bagian barat berbatasan dengan Kota Metro.

Kabupaten Lampung Timur memiliki iklim kategori B yang dicirikan dengan bulan basah selama enam bulan yaitu Desember – Juni dengan suhu rata – rata 24 – 34°C. Curah hujan rata-rata setiap tahun berkisar antara 2.000 hingga 2.500 mm. dan bulan kering selama tiga bulan.

2.5 Website

Menurut Feri Efendi (2017) Website adalah suatu bentuk media publikasi elektronik yang terdiri dari serangkaian halaman web yang saling terhubung melalui link pada elemen teks atau gambar. Istilah website atau situs merujuk pada kumpulan halaman yang memuat informasi berupa teks, gambar diam atau gerak, animasi, suara, video atau kombinasinya, baik dalam bentuk statis maupun dinamis,

yang saling terhubung membentuk sebuah struktur yang terdiri dari jaringan-jaringan halaman (Suprianto & Matsea, 2018).

Berdasarkan pemaparan tersebut, dapat disimpulkan bahwa website atau situs adalah bentuk media publikasi elektronik yang terdiri dari kumpulan halaman web yang saling terhubung melalui link pada element teks atau gambar, yang memuat informasi berupa teks, gambar diam atau gerak, animasi, suara, video atau kombinasinya, baik dalam bentuk statis maupun dinamis, yang saling terhubung membentuk sebuah struktur yang terdiri dari jaringan-jaringan halaman.

2.6 Leaflet

Leaflet adalah sebuah *library JavaScript Open Source* yang mempermudah proses pembuatan peta pada halaman web (Tanjaya et al., 2016). *Leaflet* menyediakan banyak fitur pemetaan salah satunya *Interactive Choropleth Map* atau peta tematik yang interaktif dan bisa menampilkan ruangan pada tempat tertentu. Selain fitur-fitur yang kuat, *Leaflet* juga memiliki dokumentasi yang lengkap dan komunitas yang aktif. Pengembang dapat dengan mudah menemukan bantuan, tutorial, dan contoh penggunaan *Leaflet* untuk membantu mereka dalam mengimplementasikan peta interaktif sesuai dengan kebutuhan proyek mereka.

2.7 Framework Laravel

Framework merupakan suatu kerangka kerja yang disediakan untuk mempermudah pembangunan suatu sistem sehingga pembangunan sistem tidak perlu melakukan perancangan dari nol (Sinaga & Samsudin, 2021). Salah satu *Framework PHP* yang terkenal yaitu *Laravel*. *Laravel* merupakan kerangka

aplikasi web yang memiliki sintak yang indah dan mudah dipahami (Otwell, 2023). Laravel juga sudah mendukung sistem *Model View Controller* (MVC).

2.8 Konsep *Model View Controller* (MVC)

Model View Controller (MVC) terdiri dari tiga bagian yang masing masing memiliki peran unik, yaitu *Model* sebagai pengelola data aplikasi, *View* sebagai penyajian antarmuka pengguna, dan *Controller* sebagai pengatur proses *inputan user* untuk ditampilkan di *View* (Pratama & Paramita, 2020).

2.9 *Unified Model Language* (UML)

UML adalah sebuah bahasa standar dalam industri untuk visualisasi, merancang, dan mendokumentasikan sistem perangkat lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem dan telah menjadi pilihan utama dalam industri untuk membantu menggambarkan dan memahami sistem perangkat lunak (Dharwiyanti & Wahono, 2003). Berikut adalah beberapa contoh jenis diagram dalam UML:

2.10 Metode Pengembangan Sistem

Waterfall (air terjun) merupakan metode pengembangan sistem di mana analis dan pengguna bekerja sama untuk merumuskan suatu kegiatan secara berurutan dari satu fase ke fase berikutnya (Bahar et al., 2011). *Waterfall* memiliki keunggulan yang terletak pada kemampuannya untuk mengidentifikasi kebutuhan pengguna jauh sebelum kegiatan pemrograman dimulai dan membatasi perubahan kebutuhan pengguna saat proyek sudah dimulai.

Kelemahan metode *waterfall* yaitu cenderung sulit untuk menangani perubahan kebutuhan pengguna yang muncul selama proses pengembangan.

Karena setiap fase harus diselesaikan sebelum melanjutkan ke fase berikutnya, perubahan yang dibutuhkan seringkali sulit atau mahal untuk diimplementasikan setelah fase-fase awal selesai. Maka rancangan sistem harus benar-benar dipersiapkan secara matang sebelum tahap pengkodean dimulai.

Menurut Bahar et al. (2011) waterfall memiliki tahapan utama dalam pemrosesannya yaitu:

1. Analisis

Tahap ini melibatkan pemahaman mendalam tentang kebutuhan pengguna dan tujuan proyek. Analisis mengumpulkan informasi yang diperlukan untuk merumuskan persyaratan sistem dengan jelas dan lengkap.

2. Perancangan

Pada tahap ini, perancangan sistem secara keseluruhan dilakukan berdasarkan persyaratan yang telah didefinisikan sebelumnya. Perancangan ini mencakup desain arsitektur, desain antar muka pengguna, dan perencanaan modul dan komponen sistem.

3. Implementasi

Tahap ini melibatkan pembuatan kode program berdasarkan rancangan yang telah dibuat. Tim pengembang mulai mengimplementasikan modul-modul yang telah direncanakan dan melakukan pengujian unit untuk memastikan keberhasilan implementasi.



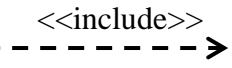
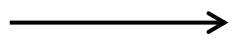
4. Pengujian

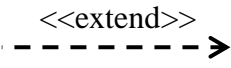
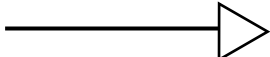
Tahap pengujian dilakukan untuk mengidentifikasi dan memperbaiki kesalahan atau bug yang ada.

2.10.1 Use Case Diagram

Use Case Diagram merupakan gambaran dari fungsionalitas yang diharapkan dari suatu sistem, yang menekankan pada "apa" yang dilakukan oleh sistem, bukan "bagaimana" melakukannya. Diagram ini merepresentasikan interaksi antara aktor dengan sistem, dimana setiap *use case* merepresentasikan suatu skenario atau fungsionalitas tertentu dari perspektif pengguna

Tabel 2.2 Simbol *Use Case Diagram*

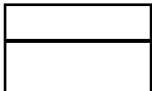
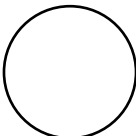

No	Simbol (Nama)	Deskripsi
1	Aktor 	Aktor dalam sebuah sistem adalah entitas yang terlibat dalam interaksi dengan sistem, dan dapat berupa pengguna, sistem lain, atau perangkat lain yang terlibat dalam penggunaan sistem yang sedang dirancang.
2	<i>Use case</i> 	<i>Use case</i> dalam UML menggambarkan interaksi antara aktor dengan sistem dan menunjukkan fungsionalitas yang diharapkan dari sistem, bukan cara kerja sistem.
3	<i>Association</i> 	Dalam <i>use case diagram</i> , hubungan antara aktor dan <i>use case</i> ditunjukkan untuk menggambarkan bagaimana aktor terlibat dalam sebuah <i>use case</i> .
4	<i>Include</i> 	<i>Include</i> menunjukkan bahwa sebuah <i>use case</i> termasuk bagian dari <i>use case</i> lain, artinya <i>use case</i> yang disertakan (<i>included</i>) akan selalu dieksekusi jika <i>use case</i> induknya dijalankan

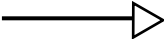
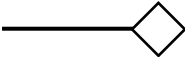
5	<i>Extend</i> 	<i>Extend</i> menunjukkan bahwa sebuah <i>use case</i> memiliki aksi opsional atau alternatif yang hanya dieksekusi pada kondisi tertentu, dan bisa menambahkan atau memperluas fungsionalitas dari <i>use case</i> utama.
6	<i>Generalisasi</i> 	<i>Generalisasi</i> adalah jenis hubungan antara dua <i>use case</i> di mana satu <i>use case</i> (yang disebut <i>child use case</i>) mewarisi sifat dan perilaku dari <i>use case</i> yang lain (yang disebut <i>parent use case</i>).

2.10.2 Class Diagram

Class diagram adalah suatu bentuk diagram pada pemodelan objek yang digunakan untuk menggambarkan struktur dan deskripsi dari *class*, *package*, objek, dan hubungan antara elemen-elemen tersebut, termasuk di dalamnya adalah hubungan *containment*, pewarisan, asosiasi, dan lain-lain

Tabel 2.3 *Class Diagram*



No	Jenis	Simbol	Deskripsi
1	<i>Class</i>		Merepresentasikan suatu objek dengan atribut, operasi dan relasi
2	<i>Interface</i>		Mirip dengan kelas tetapi hanya memiliki tanda tangan metode dan tidak ada implementasi
3	<i>Association</i>		Menunjukkan hubungan antara dua kelas yang tidak terikat



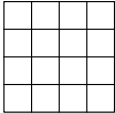


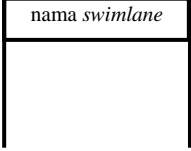
4	<i>Generalization</i>		hubungan antara dua class di mana satu class (subclass) diwarisi sifat dan perilaku dari class yang lain (superclass)
5	<i>Aggregation</i>		Menunjukkan hubungan antara kelas induk dan kelas anak, di mana kelas anak dapat terikat ke beberapa kelas induk

2.10.3 Activity Diagram

Activity diagram adalah sebuah *state diagram* yang khusus digunakan untuk menggambarkan sebagian besar *state* sebagai *action* dan sebagian besar transisi dipicu oleh selesainya *state* sebelumnya melalui *internal processing* (Dharwiyanti & Wahono, 2003). *Activity diagram* menggambarkan aliran aktivitas dalam sistem yang dirancang, termasuk awal, kemungkinan keputusan yang terjadi, dan akhir dari setiap aliran. Diagram ini juga mampu menggambarkan proses paralel yang terjadi pada beberapa aktivitas.

Tabel 2.4 *Activity Diagram*

No	Nama	Simbol	Deskripsi
1	Satatus Awal		Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2	Aktivitas		Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.

3	Percabangan		Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
4	Penggabungan		Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
5	Tabel		Suatu file komputer dari mana data bisa dibaca atau direkam selama kejadian bisnis
6	Dokumen		Menunjukkan dokumen sumber atau laporan
7	Status Akhir		Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
8	<i>Swimlane</i>		Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

2.11 Pengujian ISO 25010

ISO 25010 adalah sebuah standar internasional yang digunakan untuk mengevaluasi kualitas perangkat lunak dan sistem (Peters & Aggrey, 2020). ISO 25010 ini juga dikenal sebagai model *Systems and Software Quality Requirements*

and Evaluation (SQuaRE). Model ISO 25010 mencakup delapan karakteristik kualitas yang secara visual diperlihatkan dalam gambar 2.1 berikut:



Gambar 2.1 Visualisasi Karakteristik ISO 25010

(Sumber: ISO 25010 *software and data quality*)

Dari delapan karakteristik kualitas aplikasi di atas, penulis menetapkan hanya tiga karakteristik yang dijadikan sebagai variabel pengujian pada penelitian ini antara lain: *Functional Suitability*, *Performance Efficiency*, dan *Usability*. Penjelasan masing-masing karakteristik tersebut akan dijabarkan pada tabel 2.5

Tabel 2.5 Karakteristik ISO 25010

No	Karakteristik	Sub Karakteristik	Deskripsi
1	<i>Functional Suitability</i>	<i>Functional completeness</i>	Kemampuan sistem untuk mencakup fungsi mengakomodasi semua tugas dan tujuan pengguna yang telah ditetapkan.
		<i>Functional correctness</i>	Kemampuan sistem sebagai akurasi dan presisi hasil yang diberikan oleh produk atau sistem sesuai dengan tingkat kebutuhan yang diinginkan.

		<i>Functional appropriateness</i>	Kemampuan sistem untuk mengukur seberapa efektifitas fungsi-fungsi mempermudah pencapaian tugas dan tujuan yang spesifik
2	<i>Performance Efficiency</i>	<i>Time behaviour</i>	Kemampuan sistem untuk mengukur seberapa baik respons dan waktu pemrosesan, serta kecepatan throughput dari suatu produk atau sistem, saat digunakan dalam menjalankan fungsinya, memenuhi persyaratan yang telah ditetapkan.
		<i>Resource utilization</i>	Kemampuan sistem untuk mengukur seberapa sesuai jumlah dan jenis sumber daya yang digunakan oleh suatu produk atau sistem saat beroperasi, dengan memenuhi persyaratan yang telah ditetapkan.
		<i>Capacity</i>	Kemampuan sistem untuk mengukur seberapa besar kapasitas maksimum suatu produk atau parameter sistem dalam memenuhi persyaratan yang telah ditetapkan.
3	<i>Usability</i>	<i>Appropriateness recognizability</i>	Kemampuan sistem untuk mengukur sejauh mana kesesuaian produk atau

		sistem dengan kebutuhan pengguna dapat dikenali oleh mereka.
	<i>Learnability</i>	Kemampuan sistem untuk membantu pengguna mencapai tujuan pembelajaran dengan efektivitas, efisiensi, keamanan, dan kepuasan dapat dievaluasi.
	<i>Operability</i>	Kemampuan sistem untuk mengukur sejauh mana suatu produk atau sistem memiliki karakteristik yang memungkinkan pengoperasian dan pengendalian yang mudah.
	<i>User error protection</i>	Kemampuan sistem untuk mengukur seberapa jauh sistem memberikan perlindungan kepada pengguna agar terhindar dari kesalahan yang mungkin terjadi.
	<i>User interface aesthetics</i>	Kemampuan sistem untuk mengukur seberapa efektif antarmuka pengguna dalam menciptakan pengalaman interaktif yang menyenangkan dan memuaskan bagi para pengguna.
	<i>Accessibility</i>	Kemampuan sistem untuk mengukur sejauh mana kemampuan dan

			karakteristik yang paling luas dari pengguna dapat dimanfaatkan dalam mencapai tujuan tertentu dalam suatu konteks penggunaan, dengan menggunakan suatu produk atau sistem.
--	--	--	---

2.11.1 Skala Likert

Skala Likert merupakan jenis skala yang sangat sederhana dan mudah digunakan. Skala ini menggunakan beberapa pernyataan untuk mengukur perilaku individu dengan memberikan respon pada 5 pilihan yang tersedia, yaitu sangat setuju, setuju, tidak memutuskan (netral), dan tidak setuju, serta sangat tidak setuju (Budiaji, 2013).