

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Landasan Teori**

Bagian ini menjelaskan dasar-dasar teori yang relevan dan menjadi pijakan dalam perancangan dan pengembangan sistem ini.

##### **2.2.1 PT. Perkebunan Nusantara IV Regional III**

PT. Perkebunan Nusantara IV (PTPN IV) Regional III merupakan salah satu bagian dari holding BUMN Perkebunan Nusantara, yang memiliki fokus utama pada budidaya dan pengolahan komoditas kelapa sawit. Regional III meliputi beberapa unit kebun dan pabrik pengolahan yang tersebar di wilayah operasionalnya. Dalam menjalankan aktivitas bisnis dan non-bisnisnya, PTPN IV Regional III memiliki struktur organisasi yang tersusun secara hierarkis, mulai dari manajemen pusat regional, hingga unit-unit di bawahnya seperti kebun dan pabrik. Salah satu elemen penting dalam struktur organisasi ini adalah Unit Sumber Daya Manusia (SDM). Unit ini berperan sebagai pusat administrasi dan pengelolaan personel, termasuk mengoordinasikan kegiatan yang berkaitan dengan pengembangan sumber daya manusia internal maupun eksternal, seperti penerimaan peserta magang, penelitian, dan kunjungan akademik. Karena keterbatasan sistem yang terpusat, unit SDM di tingkat regional saat ini masih menangani seluruh pengajuan akademik dari berbagai subarea, yang mencakup kebun dan pabrik berbeda. Hal ini menimbulkan beban kerja tinggi karena tidak adanya desentralisasi pengelolaan melalui sistem digital.

Setiap subarea (Distrik petani Mitra, kebun sei galuh, kebun sei garo, dan lain lain.) memiliki aktivitas operasional yang berbeda dan dalam beberapa kasus, lokasi yang berjauhan secara geografis. Tanpa adanya sistem informasi manajemen yang memfasilitasi pembagian beban kerja antar subarea, koordinasi antar unit dan kecepatan dalam pelayanan kegiatan akademik menjadi kurang efisien.

### 2.2.2 Sistem Informasi

Menurut Jonny Seah (2020), sistem informasi dapat dipahami sebagai suatu integrasi dari berbagai komponen teknologi informasi yang dirancang untuk bekerja secara sinergis dalam menghasilkan informasi yang relevan dan bermanfaat. Tujuan utama dari sistem ini adalah untuk membangun dan memfasilitasi jalur komunikasi yang efektif di dalam suatu organisasi atau kelompok, sehingga proses penyampaian dan pertukaran informasi dapat berjalan secara terstruktur dan efisien.

### 2.2.3 Sistem Informasi Manajemen

Sistem Informasi Manajemen (SIM) merupakan sistem yang terstruktur dan terorganisir, terdiri dari berbagai elemen seperti prosedur kerja, manusia, teknologi, dan informasi yang saling berinteraksi untuk mendukung proses pengambilan keputusan dalam organisasi. Menurut James Alter (1992), SIM mengintegrasikan proses kerja, aliran informasi, teknologi informasi, dan peran manusia guna mencapai tujuan strategis organisasi. Sementara itu, Bodnar & Hopwood (1993) menyatakan bahwa SIM merupakan gabungan perangkat keras dan lunak yang bertugas mengolah data menjadi informasi yang relevan bagi manajemen. Adapun struktur hirarki manajemen dalam SIM dapat dilihat pada gambar 2.1 Hirarki Manajemen SIM.



Gambar 2. 1 Hirarki Manajemen SIM

Dalam model ini, SIM berfungsi sebagai alat komunikasi dan pengambilan keputusan yang efektif, di mana setiap lapisan manajemen memperoleh akses terhadap informasi yang sesuai: mulai dari rekap data operasional harian di level bawah hingga ringkasan statistik dan laporan strategis di tingkat atas.

#### 2.2.4 Sistem Manajemen Pengajuan

Berdasarkan jurnal yang ditulis oleh Alfarisy dan Sutabri, Sistem manajemen pengajuan merupakan struktur terintegrasi yang dirancang untuk mengoptimalkan proses pengajuan dengan komponen utama meliputi antarmuka pengguna, manajemen data, dan mekanisme notifikasi. Sistem ini bertujuan meningkatkan efisiensi operasional dengan mengotomatisasi proses yang sebelumnya dilakukan secara manual, seperti penggunaan formulir kertas yang rentan terhadap kesalahan dan memerlukan waktu lama dalam pemrosesan. Komponen penting lainnya mencakup modul pelaporan dan audit yang menyimpan seluruh data pengajuan untuk keperluan analisis berkelanjutan.

#### 2.2.5 Prinsip Desain Antarmuka (UI/UX)

Prinsip desain antarmuka pengguna (*User Interface*) dan pengalaman pengguna (*User Experience*) merupakan aspek fundamental dalam merancang sistem informasi berbasis web. Desain antarmuka yang baik tidak hanya menampilkan elemen visual yang menarik, namun juga memastikan kenyamanan, kemudahan penggunaan, serta efektivitas interaksi pengguna terhadap sistem. Menurut Fathurrahman dan Sumarsono (2024), prinsip-prinsip desain antarmuka yang umum digunakan dalam evaluasi UI/UX mengacu pada teori dari para pakar seperti Ben Shneiderman, Donald Norman, dan Jakob Nielsen. Prinsip-prinsip tersebut meliputi lima aspek utama berikut:

1. Konsistensi (*Consistency*)

Konsistensi mengacu pada penggunaan elemen desain yang seragam, seperti warna, tipografi, ikon, dan struktur navigasi.

2. Tampilan (*Appearance*)

Prinsip tampilan menekankan pentingnya estetika dalam antarmuka. Pemilihan warna, jenis huruf, serta elemen grafis yang tepat meningkatkan keterbacaan dan kenyamanan visual.

### 3. Umpan Balik (*Feedback*)

Sistem yang baik harus mampu memberikan respons yang cepat dan jelas terhadap setiap tindakan yang dilakukan oleh pengguna. Umpan balik ini dapat berupa notifikasi, pesan kesalahan, ataupun status *progres* yang menunjukkan bahwa sistem merespons *input* pengguna dengan tepat.

### 4. Efisiensi (*Efficiency*)

Efisiensi berkaitan dengan seberapa cepat dan mudah pengguna dapat mencapai tujuannya saat berinteraksi dengan sistem. Desain yang efisien meminimalisasi proses berpikir yang rumit dan menyediakan fitur-fitur yang mudah ditemukan serta digunakan oleh berbagai kalangan pengguna.

### 5. Keamanan dan Pencegahan Kesalahan (*Security and Prevent Errors*)

Prinsip ini mencakup kemampuan sistem dalam mencegah terjadinya kesalahan pengguna serta menyediakan solusi atau navigasi untuk keluar dari kondisi yang tidak diinginkan.

Selain itu, beberapa aspek visual penting dalam UI/UX yang juga menjadi pertimbangan utama dalam desain antarmuka adalah:

#### 1. Warna (*Color*)

Pemilihan warna dalam antarmuka pengguna (*UI*) harus mempertimbangkan aspek psikologi warna, yaitu bagaimana warna memengaruhi persepsi, emosi, dan perilaku pengguna dalam konteks *digital* (Karja, 2021). Warna berfungsi sebagai penanda visual (*visual signifier*) yang membantu pengguna memahami informasi, membedakan status, serta mengarahkan tindakan secara lebih cepat dan intuitif (Hartadi et al., 2020).

Dalam sistem berbasis web, makna warna bersifat kontekstual, namun terdapat konvensi umum yang dapat dimanfaatkan untuk meningkatkan efektivitas desain *UI/UX*. Beberapa warna yang umum digunakan beserta implikasinya dalam sistem pengajuan kegiatan akademik adalah sebagai berikut:

- a. Biru (*Blue*) sering dipilih sebagai warna primer dalam aplikasi korporat karena merepresentasikan kepercayaan, ketenangan, profesionalisme, dan keamanan (Patrycia Z, 2020). Warna ini membantu menanamkan rasa aman dan andal, yang sangat sesuai untuk sistem administratif seperti pengajuan kegiatan. Nuansa biru tua (*dark blue*) dapat menyampaikan stabilitas dan otoritas, sementara biru muda lebih bersifat menenangkan dan ramah.
- b. Hijau (*Green*) secara universal dikaitkan dengan alam, pertumbuhan, dan kesuksesan. Dalam *UI*, hijau berfungsi sebagai penanda konfirmasi, keberhasilan, dan tindakan positif (Gusia, 2024). Oleh karena itu, warna ini sangat efektif untuk tombol "Ajukan" atau "Submit", notifikasi "Berhasil", atau status "Disetujui/Diterima", karena memberikan umpan balik yang positif dan meyakinkan.
- c. Merah (*Red*) adalah warna dengan daya tarik perhatian paling kuat. Merah melambangkan bahaya, urgensi, atau tindakan kritis (Patrycia Z, 2020). Dalam sistem pengajuan, penggunaan merah harus dibatasi dan strategis, misalnya untuk: tombol "Hapus" atau "Batalkan", pesan error validasi, peringatan deadline, atau status pengajuan "Ditolak". Warna ini mendorong pengguna untuk berhenti sejenak dan mempertimbangkan tindakannya dengan hati-hati.
- d. Kuning & Oranye (*Yellow & Orange*) merupakan warna hangat yang menyampaikan optimisme, peringatan (tanpa bahaya ekstrim), dan kehangatan (Patrycia Z, 2020). Oranye sering digunakan untuk tombol *call-to-action* sekunder atau untuk menyorot elemen yang memerlukan perhatian. Kuning umumnya digunakan sebagai penanda status "Dalam Proses" atau "Menunggu Tinjauan" (*pending*).
- e. Warna Netral (Abu-abu, Hitam, Putih) membentuk fondasi hierarki visual. Abu-abu (*Grey*) digunakan untuk teks sekunder, garis pemisah, dan latar belakang untuk menciptakan struktur dan keteraturan tanpa mengganggu fokus. Hitam (*Black*) merupakan pilihan standar untuk teks utama karena keterbacaan tertinggi. Putih (*White*) atau ruang negatif (*negative space*) sangat penting untuk memberikan kesan

bersih, tidak padat, dan memandu mata pengguna ke elemen interaktif (Gusia, 2024).

Berdasarkan teori psikologi warna, perancangan *front-end* sistem ini menerapkan skema warna yang tidak hanya estetis tetapi juga fungsional.

## 2. Tipografi (*Typography/Font*)

Menurut studi oleh Alsudani & Casey (2022), tipografi membantu pengguna dalam memahami struktur informasi dengan lebih baik melalui elemen-elemen seperti jenis huruf (*font family*), ukuran huruf (*font size*), berat huruf (*font weight*), gaya huruf (*style*), serta jarak antar karakter dan baris. Misalnya, penggunaan huruf *sans-serif* seperti *Helvetica* atau *Open Sans* direkomendasikan untuk antarmuka digital karena memiliki bentuk yang bersih dan mudah dibaca di berbagai resolusi layar. Selain itu, konsep hirarki visual juga penting dalam tipografi UI. Variasi ukuran dan berat huruf digunakan untuk menandai perbedaan tingkat informasi, seperti membedakan antara judul, subjudul, dan isi. Tipografi yang baik memandu pengguna dari satu bagian ke bagian lain dengan lancar tanpa membingungkan.

## 3. Ikon (*Icons*)

Menurut Alsudani (2022), ikon yang efektif adalah ikon yang familiar, konsisten, dan mudah dikenali. Ikon sebaiknya tidak berdiri sendiri, melainkan didukung oleh label teks (*label-supporting icons*) untuk menghindari ambiguitas, terutama pada sistem yang digunakan oleh beragam pengguna dengan latar belakang berbeda. Ikon yang tidak familiar atau terlalu abstrak dapat menimbulkan kebingungan dan justru memperlambat proses navigasi. Selain itu, konsistensi gaya ikon juga merupakan aspek penting. Semua ikon dalam sistem harus menggunakan gaya visual yang seragam misalnya *outline*, *filled*, atau *duotone* untuk menjaga harmonisasi antarmuka. Ketidakkonsistenan dalam penggunaan ikon dapat mengganggu kesan profesionalitas dan menurunkan usability dari sistem. Dalam sistem ini, ikon digunakan untuk menunjang akses navigasi (seperti ikon "beranda", "tambah", "hapus", dan "unggah"), serta memperjelas status atau tindakan tertentu. Pemilihan ikon yang tepat

mendukung prinsip desain yang berpusat pada pengguna (*user-centered design*) dan berkontribusi terhadap efisiensi serta kepuasan pengguna dalam menggunakan sistem.

#### 4. Tata Letak (*Layout*)

Menurut Feriadi dan Safitri (2023), prinsip utama dalam merancang layout adalah mengelompokkan informasi secara hierarkis dan konsisten sehingga memudahkan pengguna dalam menemukan fungsi-fungsi penting dari sistem. *Layout* yang baik mampu membimbing perhatian pengguna dari satu elemen ke elemen lain secara alami dan terstruktur, misalnya melalui penggunaan grid system, pemilihan posisi tombol aksi yang strategis, serta pemisahan visual antara bagian navigasi, konten utama, dan *footer*. Prinsip keselarasan (*alignment*) dan keseimbangan (*balance*) juga harus diterapkan untuk menciptakan antarmuka yang bersih dan profesional. Elemen yang sejajar dengan rapi memberikan kesan tertata, sedangkan distribusi ruang yang seimbang menghindarkan antarmuka dari kesan berat sebelah atau berantakan. Selain itu, penting pula untuk memperhatikan tata letak responsif, agar sistem tetap optimal diakses melalui berbagai ukuran perangkat, seperti *desktop*, *tablet*, atau ponsel pintar.

#### 5. Ruang Kosong (*White Space*)

Menurut The Power of White Space in Design (Super.so, 2023), white space terbagi menjadi dua jenis utama:

- a. *Micro white space*: jarak antar huruf, baris, atau elemen kecil seperti ikon dan label.
- b. *Macro white space*: ruang di antara blok konten besar, seperti margin, padding, dan jarak antar komponen utama.

*White space* membantu mengarahkan fokus pengguna ke informasi yang penting dan memperjelas struktur hierarki konten. Ruang kosong juga mengurangi beban kognitif dengan mencegah antarmuka terlihat terlalu padat atau penuh sesak. Dalam sistem pengajuan kegiatan akademik ini, *white space* digunakan untuk:

- a. Memberikan pemisahan visual antara bagian *form*, tombol aksi, dan navigasi.

- b. Memastikan antarmuka tetap bersih dan profesional.
- c. Meningkatkan keterbacaan dan mempermudah interaksi, terutama pada perangkat *mobile* dengan ruang layar terbatas.

Penerapan *white space* secara strategis menjadi bagian dari prinsip desain minimalis dan *user-centered*, yang bertujuan untuk meningkatkan fokus, efisiensi, dan estetika keseluruhan dari sistem.

Dengan menerapkan prinsip-prinsip dan elemen *UI/UX* tersebut, diharapkan sistem yang dikembangkan tidak hanya fungsional tetapi juga mampu memberikan pengalaman pengguna yang baik, meningkatkan kepuasan, serta mendorong penggunaan sistem secara optimal di lingkungan organisasi.

#### 2.2.6 *Front-end Development*

Dalam pengembangan front-end, arsitektur *Model-View-Controller* (MVC) digunakan untuk memisahkan data (*model*), tampilan (*view*), dan kontrol aplikasi (*controller*). Konsep ini diperkenalkan oleh Trygve Reenskaug dalam pengembangan *Smalltalk* untuk meningkatkan modularitas sistem dan mengurangi ketergantungan antar komponen (Deacon, 2009). Dalam konteks *front-end*, *view* berperan dalam menampilkan informasi kepada pengguna berdasarkan instruksi dari *controller*, yang bertindak sebagai penghubung antara tampilan dan logika bisnis (*model*).

Komponen utama dalam pengembangan front-end meliputi *User Interface* (UI) dan *User Experience* (UX). UI berfokus pada elemen visual dan interaksi langsung dengan pengguna, mencakup desain warna, animasi, serta tata letak komponen agar lebih menarik dan intuitif. Sementara itu, UX mencakup keseluruhan pengalaman pengguna saat berinteraksi dengan sistem, termasuk kemudahan penggunaan, nilai manfaat, aksesibilitas, serta daya tarik emosional dari desain sistem. Frank Guo mengidentifikasi empat elemen utama UX, yaitu *usability* (kemudahan dalam menyelesaikan tugas), *value* (kesesuaian fitur dengan kebutuhan pengguna), *adoptability* (kemudahan akses terhadap sistem), dan *desirability* (kesan positif yang ditimbulkan oleh sistem).

#### 2.2.7 *Alpine.js*

Alpine.js merupakan *framework front-end ringan* yang dirancang untuk memberikan kemudahan dalam menambahkan interaktivitas langsung ke dalam elemen HTML melalui atribut, dengan sintaks yang menyerupai Vue.js versi 2. *Framework* ini memungkinkan pengembangan antarmuka pengguna yang reaktif dengan pendekatan deklaratif dan minim manipulasi DOM secara manual, sehingga menghasilkan kode yang lebih bersih, terorganisir, dan mudah dikelola. *Framework* ini pertama kali dirilis sebagai versi 0.1.0 oleh Caleb Porzio pada 29 November 2019, dan versi stabil 1.0 dirilis pada 19 Desember 2019. Contoh penerapan alpine.js dalam kode program dapat dilihat pada Gambar 2.2 Alpine.js.

```
<script src="//unpkg.com/alpinejs" defer></script>

<div x-data="{ open: false }">
  <button @click="open = true">Expand</button>

  <span x-show="open">
    Content...
  </span>
</div>
```

Gambar 2. 2 Alpine.js

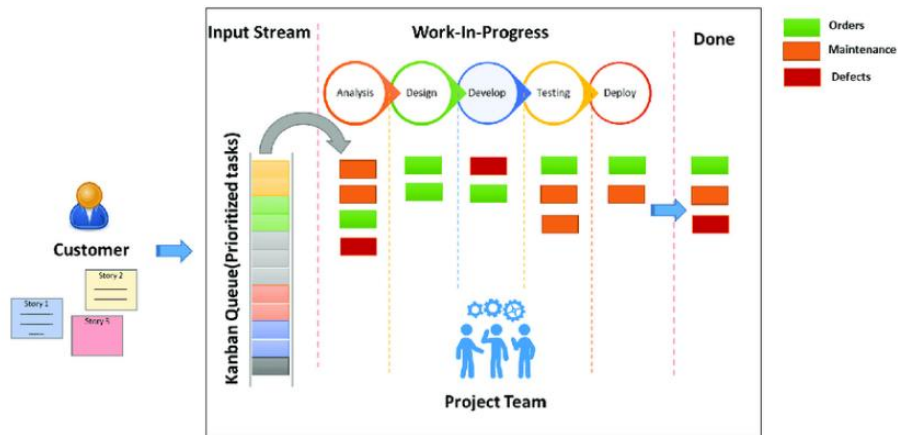
Dibandingkan dengan jQuery, Alpine.js menawarkan pendekatan yang lebih sederhana dan efisien, dengan mengadopsi paradigma pemrograman reaktif, di mana *binding* dan pembaruan data terjadi secara otomatis. Dalam perbandingannya dengan react, alpine.js menawarkan pendekatan yang lebih ringan dan sederhana dalam pengembangan *frontend*. Jika react merupakan pustaka javascript yang digunakan untuk membangun antarmuka pengguna berbasis komponen dengan manajemen state yang kompleks, alpine.js lebih berfokus pada kemudahan integrasi langsung ke dalam HTML tanpa perlu konfigurasi yang rumit. (<https://medium.com/@ssbhattarai/alpine-js-a-modern-lightweight-javascript-framework-a-jquery-replacement-68faa791edbb> dan <https://alpinejs.dev/> diakses pada 16 Juni 2025).

Selain efisiensi sintaks dan ringan dalam performa, pemilihan Alpine.js dalam proyek ini juga didasarkan pada konteks lingkungan operasional PT. Perkebunan Nusantara IV Regional III. Sistem yang dibangun lebih berfokus pada proses pengisian *form*, validasi data, dan tampilan informasi secara *real-time* dalam skala ringan. Dengan infrastruktur teknologi perusahaan yang masih dalam tahap *semi-digital* dan belum sepenuhnya berbasis SPA (*Single Page Application*), Alpine.js memberikan keunggulan dalam integrasi ke sistem legasi yang telah ada. Selain itu, karena Alpine.js tidak membutuhkan dependensi besar seperti *framework modern* lainnya (misalnya React atau Angular), sistem ini menjadi lebih hemat *bandwidth* dan dapat diakses lebih cepat, khususnya di area operasional perusahaan yang memiliki keterbatasan konektivitas atau perangkat dengan spesifikasi rendah.

#### 2.2.8 Metodologi Kanban

Menurut Darwana (2024) Kanban adalah suatu alur kerja digunakan pada proses manufaktur dan pengembangan *software* dalam meningkatkan efisiensi dan produktivitas. Selain itu, kanban bisa digunakan secara efektif untuk menjaga proyek agar tetap pada alur kerjanya dan dapat juga berfungsi sebagai alat manajemen waktu pribadi. Kanban adalah sistem untuk memvisualisasikan pekerjaan. Inti dari metode kanban yaitu membuat "papan" dan di atasnya "kartu" atau kartu dengan tugas yang perlu dilakukan. Papan kanban dalam bentuknya yang paling sederhana berisi tiga kolom: yang harus dikerjakan, yang sedang dilakukan dan yang sudah dilakukan.

Hal yang penting untuk sistem kanban yaitu konsep pembatasan jumlah pekerjaan yang sedang berlangsung untuk memusatkan perhatian serta kerja tim. Ini membantu membatasi jumlah waktu yang dihabiskan untuk penyelesaian tugas yang dibawa ke kolom Tugas yang diselesaikan dan menyoroti potensi kemacetan beban kerja. Ketika merancang sistem informasi menurut metode kanban, perlu untuk menentukan dan menghitung jumlah baris dan kolom papan kanban, karena model kanban merupakan metode yang sederhana serta dapat dijalankan dalam proyek kecil atau besar. Contoh *header* dari kanban *board* dapat dilihat pada gambar 2.3. Kanban *Board*.



Gambar 2. 3 Kanban Board

Gambar 2.3 merupakan *header* dari Kanban board yang terdiri dari *task*, *analysis*, *design*, *evaluate* & *done*. Jumlah *header* bisa disesuaikan dengan kebutuhan dalam menjalankan sebuah proyek.

### 2.2.9 Pengujian Sistem

Menurut jurnal yang ditulis oleh Mintarsih (2023), Pengujian sistem perangkat lunak merupakan salah satu tahapan penting dalam proses pengembangan sistem informasi. Tujuan utama dari pengujian ini adalah untuk menjamin kualitas perangkat lunak yang dihasilkan, memastikan bahwa sistem berjalan sesuai dengan kebutuhan pengguna, serta meminimalkan potensi kesalahan (*error*) sebelum perangkat lunak digunakan secara luas.

### 2.2.10 5 Second Testing

5 Second Testing adalah metode evaluasi antarmuka yang digunakan untuk mengukur kesan pertama pengguna terhadap tampilan halaman dalam waktu 5 detik. Menurut Triputra dan Fitriasia (2020), metode ini bertujuan mengetahui apakah fungsi utama dan elemen penting dari halaman dapat langsung dipahami oleh pengguna. Dalam pengujian ini, pengguna diberikan waktu 5 detik untuk melihat tampilan (seperti *dashboard* atau *login*), lalu menjawab pertanyaan seperti:

- a. Apa yang diingat dari halaman tersebut?
- b. Apa fungsi halaman tersebut?

c. Elemen apa yang paling menarik perhatian?

Hasil penelitian menunjukkan bahwa metode ini efektif untuk mengukur kejelasan desain, struktur informasi, dan daya tarik visual, serta sangat berguna dalam pendekatan User Centered Design. Untuk menghitung rata-rata keberhasilan pemahaman pengguna, dapat dihitung dengan:

$$\text{Rata – rata tingkat pemahaman} = \frac{\text{total nilai pemahaman}}{\text{jumlah responden}}$$

### 2.2.11 Usability Testing

*Usability testing* merupakan metode evaluasi yang digunakan untuk mengukur sejauh mana suatu sistem dapat digunakan oleh pengguna secara efektif, efisien, dan memuaskan dalam mencapai tujuan tertentu (ISO 9241-11). *Usability testing* menekankan observasi langsung terhadap interaksi pengguna dengan sistem untuk mengidentifikasi hambatan penggunaan dan area yang perlu diperbaiki. Menurut Utami et al. (2020), *usability testing* mencakup tiga aspek utama:

1. Efektivitas: seberapa akurat dan lengkap pengguna dapat menyelesaikan tugas.
2. Efisiensi: jumlah sumber daya (terutama waktu) yang dibutuhkan untuk menyelesaikan tugas.
3. Kepuasan pengguna: persepsi pengguna terhadap kenyamanan dan kemudahan penggunaan sistem.

Salah satu instrumen yang umum digunakan dalam *usability testing* adalah *System Usability Scale* (SUS). SUS dikembangkan oleh John Brooke (1986) sebagai alat ukur kuantitatif untuk menilai persepsi pengguna terhadap kemudahan penggunaan sistem. SUS terdiri dari 10 item pernyataan dengan skala Likert 1–5. Penilaian dilakukan sebagai berikut:

1. Untuk item positif: skor = jawaban – 1
2. Untuk item negatif: skor = 5 – jawaban
3. Total skor dari seluruh item kemudian dikalikan 2.5 untuk menghasilkan skor akhir (0–100)

$$Skor\ SUS = \sum_{i=1}^{10} Si \times 2.5$$

Selain menghasilkan skor global, dalam penelitian ini dilakukan analisis tambahan pada setiap aspek *usability*, yaitu *learnability*, *efficiency*, *memorability*, *errors*, dan *satisfaction*. Analisis ini dilakukan dengan menghitung persentase skor tiap aspek berdasarkan hasil konversi skor SUS. Rumus yang digunakan adalah:

$$Persentase\ Aspek = \frac{Total\ Skor\ Aspek}{Skor\ Maksimum} \times 100\%$$

Tabel 2. 1 Interpretasi Skor SUS

Skor SUS	Kriteria
0 - 25	Sangat buruk
26 - 50	Buruk
51 - 68	Cukup
69 - 84	Baik
85 -100	Sangat Baik

## 2.2 Penelitian Terdahulu

Penelitian terdahulu merupakan landasan penting dalam perancangan dan pengembangan *front-end* sistem pengajuan magang berbasis alpine.js (studi kasus : PT. Perkebunan Nusantara IV Regional III) yang dibangun dalam penelitian ini. Dengan menganalisis berbagai studi yang telah dilakukan sebelumnya, dapat diperoleh wawasan mengenai metode, teknologi, dan pendekatan yang paling efektif dalam membangun sistem informasi manajemen yang sesuai dengan kebutuhan pengguna. Penelitian sebelumnya juga memberikan gambaran mengenai tantangan yang dihadapi dalam implementasi sistem serupa, sehingga dapat dijadikan sebagai bahan evaluasi dan perbandingan dalam pengembangan sistem ini. Oleh karena itu, berikut beberapa penelitian yang relevan sebagai referensi dan bahan perbandingan dalam proyek akhir ini.

Penelitian oleh Purwaningrum (2021) membahas sistem informasi magang lapangan berbasis web. Penelitian ini bertujuan untuk mempermudah proses pengajuan dan penerimaan magang serta manajemen berkas. Namun, pada praktiknya, pengelolaan pengajuan magang di Diskominfo Jatim masih dilakukan secara manual dan *website* yang ada jarang digunakan secara optimal. Selain itu, penelitian ini belum mencakup fitur otomatisasi pemantauan kehadiran peserta magang dan proses pembuatan email penerimaan yang masih manual. Kelemahan pada penelitian terdahulu ini, terutama dalam aspek efektivitas penggunaan sistem web yang sudah ada dan kurangnya fitur otomatisasi untuk pemantauan *real-time* serta manajemen kehadiran, menunjukkan adanya celah penting yang perlu diperbaiki dan menjadi fokus utama penelitian ini.

Penelitian oleh Emirzaki, Ajie, dan Nurhidayat (2022) mengembangkan modul *front-end website* untuk sistem manajemen aset di UPT TIK Universitas Negeri Jakarta. Tujuannya adalah memudahkan pengguna dalam mengakses sistem manajemen aset, dan hasilnya berupa *prototype high-fidelity* yang siap diimplementasikan. Namun, penelitian ini menggunakan *library react.js* dan metode *waterfall*; berbeda dengan proyek ini yang memilih *framework* yang lebih ringan seperti *alpine.js* untuk performa optimal dan fleksibilitas *kanban*, penelitian terdahulu tersebut mungkin memiliki *overhead* yang lebih besar dan kurang adaptif terhadap perubahan kebutuhan.

Penelitian yang dilakukan oleh Nugroho (2024) menciptakan *website* pemasaran produk *Non Fungible Token (NFT)*. Penelitian ini bertujuan menarik pelanggan baru dan membangun hubungan melalui platform NFT yang aman, dengan hasil pengembangan *website* pemasaran *digital* yang memudahkan seniman *digital*. Meskipun penelitian ini relevan karena penggunaan *alpine.js*, fokusnya pada *website* pemasaran NFT sangat berbeda dengan kompleksitas alur proses bisnis pengajuan kegiatan akademik yang berjenjang dan kebutuhan fitur spesifik seperti presensi yang menjadi inti proyek ini.

Penelitian dilakukan oleh Pratama dan Ismail (2024) merancang dan mengimplementasikan sistem manajemen cuti pegawai berbasis web. Tujuannya adalah memfasilitasi pengajuan, persetujuan, dan pemantauan cuti

dalam organisasi, dan hasilnya menunjukkan peningkatan efisiensi serta transparansi dalam manajemen cuti. Akan tetapi, penelitian ini menggunakan PHP dan *database* MySQL serta berfokus pada manajemen cuti; berbeda dengan proyek ini yang mengoptimalkan *front-end* dengan *alpine.js* dan menangani kebutuhan khusus pengajuan akademik yang lebih variatif, penelitian terdahulu tersebut tidak mencakup fitur pemantauan kehadiran secara detail.

Penelitian dilakukan oleh Pradana dan Wibowo (2025) mengimplementasikan *website* penerimaan peserta didik baru berbasis *agile* dan *waterfall*. Penelitian ini bertujuan meningkatkan efisiensi operasional, transparansi data, serta kemudahan komunikasi antara calon peserta didik dan institusi, dan hasilnya menunjukkan pengurangan kesalahan *input* data, percepatan proses pendaftaran, dan peningkatan kepuasan pengguna. Meskipun menggunakan *alpine.js* untuk interaktivitas, fokusnya pada penerimaan peserta didik baru tidak mencakup kebutuhan spesifik sistem manajemen pengajuan kegiatan akademik yang lebih kompleks, seperti pengelolaan presensi, dan otomatisasi surat balasan yang menjadi fitur inovatif pada proyek ini.

Tabel 2. 2 Perbandingan Penelitian Terdahulu

Judul	Masalah	Metode	Hasil
Sistem Informasi Pengajuan Magang Berbasis Web (Studi Kasus : Dinas Komunikasi dan Informatika Provinsi Jawa Timur	masalah dalam efektivitas penggunaan sistem web dan belum menyediakan fitur otomatisasi seperti pemantauan kehadiran dan pembuatan email penerimaan.	<i>Prototype</i>	Membantu proses pengajuan maupun penerimaan magang agar menjadi lebih efisien dan terkoordinasi

Judul	Masalah	Metode	Hasil
Pengembangan Modul <i>Front-end Website</i> Sistem Manajemen Aset Unit Pelayanan Teknik Teknologi Informasi dan Komunikasi Universitas Negeri Jakarta	menghadapi keterbatasan dari segi metode <i>waterfall</i> yang kurang fleksibel serta potensi <i>overhead</i> dari penggunaan React.js.	<i>Waterfall</i>	Memudahkan pengguna dalam mengakses sistem manajemen aset UPT TIK.
Membuat <i>Website</i> Pemasaran Produk <i>Non Fungible Token</i> Menggunakan Django Dan Alpine.js	kurang relevan terhadap kebutuhan sistem akademik karena fokus pada pemasaran NFT dan tidak menangani proses bisnis akademik yang kompleks.	<i>Prototype</i>	pengembangan website pemasaran digital yang memudahkan seniman <i>digital</i> dalam menggunakan fitur-fitur dalam <i>website</i> tersebut.
Perancangan Dan Implementasi Sistem Manajemen Cuti Pegawai Berbasis Web Menggunakan Pendekatan <i>Agile</i>	tidak mencakup fitur pemantauan kehadiran secara detail dan hanya berfokus pada manajemen cuti pegawai.	<i>Agile</i>	meningkatkan efisiensi dan transparansi dalam

Judul	Masalah	Metode	Hasil
Implementasi <i>Website</i> Penerimaan Peserta Didik Baru berbasis <i>Agile</i> Dan <i>Waterfall</i> Untuk Kemudahan Pendaftaran Dan Komunikasi	proses penerimaan peserta didik baru tanpa menyertakan fitur penting seperti presensi dan otomatisasi surat balasan yang dibutuhkan dalam sistem pengajuan kegiatan akademik.	<i>Hybrid agile</i> dan <i>waterfall</i>	manajemen cuti, memberikan solusi efektif bagi organisasi.