

BAB 2

LANDASAN TEORI

2.1 Platform Sosial

Platform sosial merupakan aplikasi berbasis web atau *mobile* yang memudahkan interaksi antar pengguna dalam jaringan digital di internet. *Platform* sosial pada intinya memiliki fitur-fitur spesifik, maka diperlukan untuk menggunakan instrumen yang spesifik juga. Sebuah *platform* sosial digunakan dengan tujuan yang berbeda-beda, namun para pengguna akan berekspektasi untuk memiliki interaksi yang alami, menyenangkan, intuitif, digunakan tanpa instruksi, dan tanpa masalah [5].

Setiap *platform* sosial memiliki target pengguna yang berbeda-beda, maka ekspektasi pengalaman pengguna (*user experience*) akan berbeda-beda juga. *Platform-platform* seperti *Discord* dan *Tumblr* memiliki pengalaman pengguna yang jauh berbeda dari *platform-platform* ternama seperti *Instagram*, *Facebook*, dan *X (Twitter)*. Namun, hal ini juga karena interaksi *platform-platform* tertentu sudah dirancang sesuai target penggunanya, seperti *Discord* dan *Twitch* yang difokuskan untuk komunitas *gaming* [19] [20]. Begitu juga dengan *Tumblr* yang berkaitan erat dengan komunitas *fandom* [21], sehingga memiliki navigasi yang sulit dipelajari bagi pengguna biasa [22] dan memiliki pola waktu penggunaan yang unik [23].

Evaluasi pengalaman pengguna menjadi keperluan untuk mendeteksi elemen-elemen yang menghindar atau menggaggu pengguna, lalu memperbaiki masalahnya. *Usability heuristics* adalah salah satu prinsip untuk desain interaksi, maka bisa digunakan untuk mendesain jaringan sosial [24] [5].

2.2 Pesan Kesalahan (*Error Messages*)

Pesan kesalahan (*error message*) adalah elemen yang penting dalam aspek *UI/UX*, dari tampilan penyajian pesannya hingga penyampaian kesalahannya kepada pengguna. Pada umumnya, *error message* berfungsi untuk menjadi pertanda bagi pengguna akan adanya masalah pada sistem. Namun, yang dirasakan oleh pengguna adalah keberadaan *error message* yang tidak membantu atau menyembunyikan masalahnya.

Telah terbukti bahwa beberapa faktor yang menghambat pengguna untuk tidak memahami *error message* antara lain adalah kalimat panjang, jargon, struktur

kalimat, dan kosakata yang rumit. Setiap faktor tersebut memiliki hubungan dengan mudah dibacanya dan mudah dipahamnya *error message*, yang dibuktikan oleh Denny et al. secara statistik, hingga dapat membuat kesimpulan bahwa yang terpenting adalah pemilihan kata untuk pesannya bisa dikomunikasikan ke pembaca atau pengguna. [9]

Penelitian oleh Naveed menganalisa dampaknya presentasi *error message* pada pembelajaran pemrograman dan *debugging* menggunakan metode dengan perhitungan statistik. Hasil *t-test* yang dilakukan untuk menguji nilai *debugging* menyatakan bahwa *error message* sangatlah penting untuk performasi *debugging* [25]. Hal tersebut mengindikasikan bahwa kualitas *error message* mempengaruhi pemahaman teknis pengguna serta efisiensi masalahnya.

2.3 A/B Testing

Penelitian ini melibatkan evaluasi desain antarmuka, maka metode yang digunakan untuk menganalisis pengalaman pengguna adalah *A/B testing* atau *split testing*. Dalam metode *A/B testing*, dua versi sistem perangkat lunak akan dibandingkan dari sudut pandang pengguna akhir, yang memungkinkan pengambilan keputusan dari pengguna berbasis data [16]. *A/B testing* memberikan kerangka kerja sistematis untuk membandingkan efektivitas dua desain alternatif secara objektif, mengukur dampak perubahan desain terhadap perilaku pengguna, meminimalkan bias subjektif dalam evaluasi desain, dan mengoptimalkan pengalaman pengguna berdasarkan data aktual.

A/B testing sendiri adalah metode yang eksperimental, yang berarti cara implementasinya bisa berbeda-beda. Pengujinya dapat dilakukan secara langsung atau *real-time*, dengan lingkungan yang terkontrol di sebuah laboratorium, atau berbasis survei [8]. Contohnya, penelitian oleh Hoffmann et al. pada tahun 2022 mengeksplorasi metode *A/B testing* dengan menggunakan rumus algoritma Bayes untuk membandingkan dua proporsi yang menghasilkan kesimpulan berbeda untuk masing-masing hasil [26]. Di sisi lain, Gullberg mengaplikasikan *A/B testing* untuk mengevaluasikan desain [8]. Pada penelitian ini, akan dilakukan survei dengan evaluasi komparatif yang diadaptasi dari prinsip-prinsip *A/B testing*.

2.3.1 Perhitungan Statistika

Penelitian ini menggunakan pendekatan evaluasi dengan perhitungan statistika, yang sudah pernah dilakukan di penelitian sebelumnya. Pendekatan ini dilakukan pada penelitian oleh Naveed & Sarim yang menghitung pengaruh presentasi *error message* terhadap kinerja *debugging* menggunakan uji *t-test* [25], dan pada penelitian oleh Denny et al. yang menerapkan metode ANOVA untuk mengevaluasi faktor-faktor yang mempengaruhi persepsi *error message* [9]. Metode-metode statistika yang digunakan pada penelitian ini adalah sebagai berikut.

1. Korelasi Pearson

Perhitungan ini dilakukan untuk mencari tahu hubungan antara dua variabel numerik. Hasil dari perhitungan korelasi ini menunjukkan seberapa kuatnya hubungan kedua variabel (x dan y) serta kemaknaan atau signifikansinya. Jika nilai korelasinya (r) berjumlah lebih dari 0.05, kedua variabel memiliki memiliki korelasi tinggi [27].

$$r = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \cdot \sum_{i=1}^n (y_i - \bar{y})^2}} \quad (2.1)$$

r = korelasi

x_i = variabel x

y_i = variabel y

\bar{x}, \bar{y} = rata-rata dari masing-masing nilai x dan y

n = ukuran sampel

2. *t-test* Dua Sampel Welch (*Welch's Two Sample t-test*)

Perhitungan ini mempertimbangkan adanya perbedaan signifikan atau tidak dari dua kelompok numerik. Hal ini dilakukan dengan menghitung rata-rata dari dua kelompok tersebut terlebih dahulu. Penelitian ini menggunakan perhitungan Welch ketika memiliki sampel yang tidak dilakukan dari distribusi normal [28].

$$t = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}} \quad (2.2)$$

\bar{x}_1 = rata-rata variabel 1

\bar{x}_2 = rata-rata variabel 2

s_1^2, s_2^2 = simpangan baku (*standard deviation*)

n_1, n_2 = ukuran sampel

3. Regresi Linear Berganda

Perhitungan fungsi ini digunakan untuk menganalisis pengaruh dua variabel independen (x_1 dan x_2) terhadap satu variabel dependen (\hat{y}). Hasilnya menunjukkan seberapa banyak kedua variabel independen memprediksi hasil dari variabel dependen tersebut. Signifikansi kedua variabel juga ditunjukkan melalui perhitungan ini [29].

$$t = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}} \quad (2.3)$$

\hat{y} = variabel dependen yang diprediksi

x_1 = variabel independen 1

x_2 = variabel independen 2

$\beta_0, \beta_1, \beta_2$ = koefisien regresi

2.3.2 *User Persona*

User persona adalah representasi pengguna yang dibuat dengan tujuan untuk mempermudah menjabarkan dan menyediakan keperluan pengguna dalam perancangan sebuah sistem. Pembuatan persona sendiri juga harus dipertimbangkan sesuai target audiens untuk sistemnya, karena kegunaan sistem terpengaruh oleh hasil dari data persona yang membantu menguatkan strategi penarikan terhadap target audiens. Persona dapat dibuat dengan pendekatan kualitatif, kuantitatif, campur (*mixed method*). Walaupun *mixed method* merupakan metode yang paling kuat secara teoritis, pendekatan kualitatif dan kuantitatif lebih direkomendasikan karena *mixed method* sulit untuk dikonstruksi [30].

Dalam penelitian oleh Petsani yang termasuk dari proyek perancangan asisten virtual untuk memonitor kebiasaan fisik serta nutrisi para lansia, *proto-persona* dibuat terlebih dahulu sebelum diberi timbal balik dari komunitas lansianya sendiri. Data dari timbal balik para lansia digunakan untuk investigasi jika persona yang dibuat sudah merepresentasikan informasi yang diperlukan oleh

target audiens (lansia). Setelah itu, persona yang sudah ada diperbaiki dan dibandingkan menggunakan algoritma penyesuaian dan dievaluasi menggunakan *Persona Perception Scale*. Persona yang diperbaiki memiliki literasi digital yang sebelumnya tidak ada. Dari hasil *mapping* algoritmanya, persona yang diperbaiki sangat berhasil merepresentasikan pengguna lansia yang akurat [31].

2.3.3 Desain Prototipe

Dalam penelitian UX (*user experience*), desain prototipe menjadi bagian dari tahap metodologi sebagai sebuah proses iteratif yang menjadi titik tengah antara ide-ide pengembangan sistem dan evaluasi yang praktis. Pembuatan prototipe bersifat iteratif karena dengan munculnya timbal balik yang konsisten, prototipe harus dipastikan untuk tetap didukung dengan data yang sudah ada [6] sekaligus mementingkan personalisasi dan adaptabilitas sesuai keperluan penggunanya [32]. Hal ini juga menyoroti pentingnya dilakukan penelitian terhadap ekspektasi pengguna, biasa dengan pembuatan *user persona*, sehingga mereka dapat lebih memiliki kepercayaan terhadap sistem yang dibuat [31].

Ketika adanya evaluasi yang konsisten, keperluan pengguna yang berbeda-beda dapat dipenuhi. Faktor lain yang mempengaruhi evaluasi adalah prinsip-prinsip atau heuristik yang harus disesuaikan dengan identitas serta kegunaan dari masing-masing domain atau platform [5], dan pengujian yang mencakup interaksi-interaksi kecil [11] seperti *error message*. Dari perspektif kreatif, dibuatnya prototipe dapat menjadi ruang yang eksperimental, sehingga kesalahan yang terjadi menjadi pusat inovasi [33].

UNIVERSITAS
MULTIMEDIA
NUSANTARA