

BAB 2

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Perkembangan teknologi digital mendorong inovasi dalam desain *UI/UX*, khususnya melalui adopsi alat desain kolaboratif seperti Figma dan integrasi kecerdasan buatan (*AI*). *AI* memungkinkan desainer menciptakan antarmuka yang lebih efisien, adaptif, dan berbasis data pengguna.

Dalam penelitian nasional yang berjudul “Analisis Perilaku Desainer dalam Memanfaatkan *Software* Figma untuk Mendesain” Responden yang terlibat dalam penelitian ini adalah para desainer *UI/UX* dan praktisi desain digital yang secara aktif menggunakan aplikasi Figma dalam aktivitas profesional mereka. Pemilihan profesi ini didasarkan pada keterlibatan langsung mereka dalam proses desain antarmuka pengguna yang membutuhkan penggunaan tools seperti Figma [5].

Penelitian internasional juga pernah dilakukan mengenai pengembangan *AI-Driven Design Thinking Framework* untuk membandingkan efektivitas prototipe desain *UI/UX* yang dibuat oleh manusia dan yang dihasilkan oleh *AI*. Hasilnya menunjukkan bahwa *AI* memiliki keunggulan dalam efisiensi dan eksplorasi visual, namun masih kurang dalam empati dan pemahaman konteks pengguna [6].

Dalam penelitian yang melibatkan dua kelompok data, pemilihan uji statistik yang tepat sangat bergantung pada distribusi data dan ukuran sampel. Dua metode statistik yang sering digunakan untuk membandingkan dua kelompok adalah uji *t* (*Student's t-test*) dan uji *Wilcoxon* (uji non-parametrik). Uji *t* mengasumsikan distribusi normal dan *homogenitas varians*, sedangkan uji *Wilcoxon* digunakan sebagai alternatif non-parametrik ketika asumsi normalitas tidak terpenuhi. Studi yang dilakukan secara eksperimental membandingkan efisiensi kedua uji tersebut pada data yang mengikuti distribusi *arcsine* — yakni distribusi non-normal. Hasil penelitian menunjukkan bahwa Hal ini menandakan bahwa uji *t* masih efisien dan valid digunakan pada ukuran sampel kecil hingga menengah (di bawah 45), bahkan dalam distribusi data yang tidak sepenuhnya normal. Dengan demikian, untuk penelitian eksperimental seperti ini yang melibatkan desain manual dan *AI* dalam

UI/UX, penentuan jumlah sampel sebanyak 30–50 responden masih dapat dibenarkan secara ilmiah [7].

Sementara itu, dalam penelitian yang berjudul “Pengembangan Aplikasi Pengujian *UI/UX* Dengan Metode *SUS* dan *TAM*” menggunakan pendekatan *Technology Acceptance Model (TAM)* yang dikembangkan oleh Davis (1986) dan telah digunakan secara luas dalam berbagai studi penerimaan teknologi. *TAM* mengukur dua konstruk uTAMa yaitu *Perceived Usefulness (PU)* dan *Perceived Ease of Use (PEOU)*, yang dipercaya mempengaruhi sikap terhadap penggunaan (*Attitude Toward Using – ATU*) dan niat perilaku untuk menggunakan teknologi (*Behavioral Intention – BI*). Dalam konteks penelitian ini, *TAM* dipilih karena kemampuannya dalam mengevaluasi penerimaan teknologi desain antarmuka pengguna (*UI/UX*) berbasis *AI* pada aplikasi Figma. Model ini telah dibuktikan relevan dalam studi serupa seperti yang dilakukan oleh peneliti [8].

Tingkat kepuasan pengguna berkaitan erat dengan persepsi personal mereka terhadap kemampuan sistem dalam memenuhi ekspektasi. Kepuasan ini muncul sebagai hasil dari kesesuaian antara harapan awal pengguna dan performa aktual sistem yang mereka alami. Salah satu unsur yang berkontribusi besar terhadap tercapainya kepuasan tersebut adalah mutu layanan yang diberikan. Dalam proses pengumpulan data, instrumen berupa *skala likert* digunakan untuk menangkap opini responden dalam bentuk numerik, baik dalam bentuk persetujuan maupun ketidaksetujuan terhadap sejumlah pernyataan. Data tersebut memberikan representasi kuantitatif atas pengalaman pengguna terhadap sistem yang diuji. Penelitian ini mengacu pada model *EUCS (End-User Computing Satisfaction)* dan menggunakan metode regresi untuk menguji pengaruh lima variabel independen—yakni isi informasi, ketepatan data, format *TAM*pilan, ketepatan waktu, serta kemudahan penggunaan—terhadap variabel dependen berupa tingkat kepuasan pengguna terhadap aplikasi Figma [9].

2.2 Landasan Teori

2.2.1 Pengertian *Artificial Intelligence (AI)*

Artificial Intelligence (AI) adalah cabang ilmu komputer yang berfokus pada penciptaan sistem yang dapat meniru kecerdasan manusia, seperti pengambilan keputusan, pembelajaran dari data, dan pemrosesan bahasa alami. Dalam konteks desain *UI/UX*, *AI* berperan dalam menghasilkan saran desain otomatis, prediksi perilaku pengguna, serta percepatan iterasi desain. *AI* mampu bertindak sebagai *co-creator* dalam proses desain *UI/UX* dengan mendukung seluruh tahapan *Design Thinking* mulai dari *Empathize* hingga *test*. Namun, *AI* masih menghadapi tantangan dalam memahami empati pengguna dan konteks budaya yang kompleks [6].

2.2.2 Pemanfaatan *Artificial Intelligence (AI)* Secara Umum

Kecerdasan Buatan (*Artificial Intelligence/AI*) kini telah menjadi elemen penting dalam berbagai aspek kehidupan sehari-hari. Kehadirannya merevolusi interaksi manusia dengan teknologi, serta meningkatkan efisiensi dalam banyak aktivitas rutin. Salah satu penerapan yang paling terlihat adalah pada asisten virtual seperti *Siri*, *Alexa*, dan *Google Assistant*. Asisten digital berbasis *AI* ini membantu pengguna dalam mengatur pengingat, menjawab pertanyaan, mengendalikan perangkat rumah pintar, hingga memutar musik, semuanya melalui komunikasi berbasis bahasa alami.

Di bidang media sosial, *AI* digunakan untuk menyesuaikan konten berita dan iklan berdasarkan perilaku serta minat pengguna, sehingga menciptakan pengalaman digital yang lebih personal. Dalam sektor kesehatan, *AI* berperan dalam meningkatkan akurasi diagnosis dan efektivitas perencanaan pengobatan. Dengan kemampuannya menganalisis data medis dalam jumlah besar, *AI* membantu tenaga medis dalam mengidentifikasi pola penyakit serta memprediksi risiko kesehatan secara lebih tepat.

Pada ranah pendidikan, sistem pembelajaran berbasis *AI* memungkinkan pendekatan belajar yang dipersonalisasi sesuai dengan kekuatan dan kelemahan siswa, sehingga dapat mengoptimalkan pencapaian belajar. *AI* juga berkontribusi signifikan di dunia hiburan, di mana algoritma pada platform seperti Netflix dan

Spotify memberikan rekomendasi konten berdasarkan kebiasaan pengguna. Di bidang gim, *AI* berperan dalam menciptakan pengalaman bermain yang adaptif, dengan tingkat kesulitan yang dapat menyesuaikan kemampuan pemain. Teknologi seperti *Augmented Reality (AR)* dan *Virtual Reality (VR)* pun memperkaya pengalaman bermain dengan simulasi yang lebih nyata dan interaktif [10].

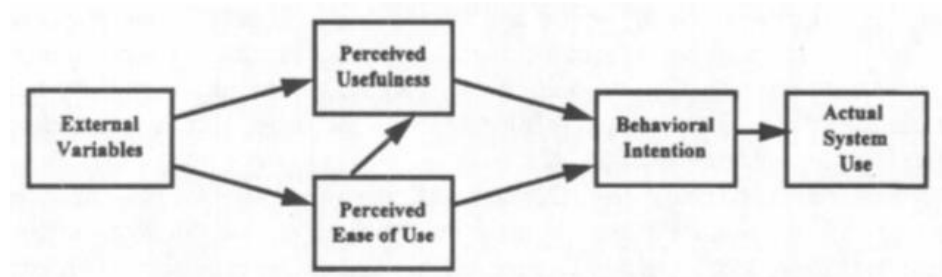
2.2.3 *Machine Learning*

Machine Learning (ML) merupakan cabang dari *Artificial Intelligence (AI)* yang memungkinkan sistem komputer memperoleh kemampuan belajar secara mandiri melalui pemrosesan data dan pengenalan pola tertentu. Teknologi ini dirancang agar komputer dapat meningkatkan kinerjanya dalam menyelesaikan tugas-tugas spesifik berdasarkan pengalaman yang diperoleh dari data historis. Secara umum, *ML* terbagi menjadi dua pendekatan *uTAMa*, yakni *supervised learning* (pembelajaran dengan pengawasan) dan *unsupervised learning* (pembelajaran tanpa pengawasan). Pada metode *supervised*, algoritma dilatih menggunakan dataset yang sudah dilengkapi dengan label jawaban, sehingga sistem dapat mengenali hubungan antara input dan output untuk membuat prediksi yang akurat. Sementara itu, *ML* juga banyak diterapkan untuk melakukan analisis prediktif serta menghasilkan rekomendasi berdasarkan kumpulan data dalam skala besar. Beberapa algoritma yang populer untuk tujuan ini antara lain *regresi linier*, *XGBoost*, *random forest*, dan *decision tree*, yang semuanya berperan dalam mengidentifikasi pola dan kecenderungan dari informasi yang tersedia secara efisien [11].

2.2.4 *Technology Acceptance Model (TAM)*

Technology Acceptance Model (TAM) merupakan model teoritis yang dikembangkan oleh Davis untuk menjelaskan faktor-faktor yang memengaruhi penerimaan teknologi informasi oleh pengguna. Model ini terdiri atas dua variabel *uTAMa*, yaitu *Perceived Usefulness (PU)* dan *Perceived Ease of Use (PEOU)*, yang kemudian memengaruhi sikap terhadap penggunaan (*Attitude Toward Using/ATU*) dan niat perilaku untuk menggunakan sistem (*Behavioral Intention/BI*). *TAM* telah banyak diterapkan dalam berbagai bidang di Indonesia, mulai dari sektor pendidikan, *e-commerce*, hingga sistem pelayanan publik. Hal ini menunjukkan

bahwa *TAM* merupakan pendekatan yang fleksibel dan relevan untuk menganalisis penerimaan teknologi, termasuk dalam konteks penggunaan kecerdasan buatan (AI) dalam proses desain *UI/UX* melalui aplikasi Figma [12].



Gambar 2.1 *Technology Acceptance Model (TAM)*

Technology Acceptance Model mempunyai 5 komponen uTAMa, yaitu:

- a. Persepsi Kegunaan
- b. Persepsi Kemudahan Penggunaan
- c. Sikap Penggunaan
- d. Intensi Dalam Menggunakan
- e. Penggunaan Teknologi Sesungguhnya.

Komponen persepsi kegunaan menunjukkan sejauh mana seseorang meyakini bahwa penggunaan teknologi meningkatkan kualitas pekerjaan. Penelitian sebelumnya memperlihatkan bahwa persepsi kegunaan adalah komponen yang terpenting untuk memengaruhi sikap penggunaan, intensi, dan tindakan. Dalam mengukur konstruk kegunaan ada 6 indikator yang digunakan yaitu pekerjaan cepat terselesaikan, meningkatkan kinerja, meningkatkan produktivitas, meningkatkan efektivitas pekerjaan, mempermudah pekerjaan dan bermanfaat [13].

Berdasarkan uraian diatas, dapat disimpulkan bawah *TAM* merupakan model penerimaan teknologi yang paling banyak digunakan, dikarenakan model *TAM* lebih sederhana. Sedangkan menurut pengertiannya *TAM* adalah sebuah teori yang dikembangkan untuk mengetahui pengaruh penggunaan teknologi informasi.

1. Persepsi Kemudahan Penggunaan (*Perceived Ease of Use*)

Kemudahan penggunaan merupakan suatu tingkatan dimana seseorang percaya bahwa suatu sistem digunakan karena sistem tersebut mudah dipahami dan digunakan, sehingga tidak diperlukan usaha apapun. Kemudahan penggunaan teknologi informasi dapat membantu usaha atau pekerja seseorang, kemudahan tersebut di tunjukan dari seseorang yang bekerja dengan menggunakan teknologi informasi lebih mudah atau lebih cepat selesai dibandingkan dengan orang yang bekerja tanpa menggunakan teknologi informasi (manual) [14].

Persepsi kemudahan penggunaan memiliki hubungan dengan beberapa konstruk diantaranya persepsi kegunaan, eksternal dan intensi, dimana setiap konstruk memiliki pengaruh masing-masing terhadap persepsi kemudahan penggunaan. Persepsi kemudahan penggunaan telah beberapa kali teruji keandalannya oleh beberapa peneliti yang menemukan bahwa persepsi kemudahan penggunaan mempengaruhi persepsi kegunaan, sikap, minat perilaku dan penggunaan nyata.

2. Persepsi Kegunaan (*Perceived Usefulness*)

Persepsi kegunaan (*perceived usefulness*) merupakan suatu tingkatan dimana seseorang percaya bahwa pengguna suatu sistem tertentu akan dapat meningkatkan prestasi kerja orang tersebut. Kemanfaatan teknologi informasi merupakan manfaat yang diharapkan oleh pengguna teknologi informasi dalam melaksanakan tugas. Individu akan menggunakan teknologi informasi jika orang tersebut mengetahui manfaat atau kegunaan dari teknologi informasi itu sendiri [14].

3. Intensi Penggunaan (*Behavioral Intention to Use*)

Perilaku penggunaan sebagai kecenderungan perilaku untuk tetap menggunakan suatu teknologi. Tingkat penggunaan sebuah teknologi pada seseorang dapat diprediksi penggunaannya melalui sikap perhatiannya terhadap teknologi tersebut, misalnya berkeinginan menambah peripheral pendukung dan memotivasi orang lain untuk menggunakan teknologi tersebut.

Berdasarkan uraian di atas dapat disimpulkan bahwa perilaku penggunaan adalah keinginan pengguna untuk tetap menggunakan suatu teknologi. Minat perilaku penggunaan memiliki hubungan antara persepsi kegunaan dan persepsi kemudahan penggunaan. minat perilaku penggunaan dipengaruhi oleh kemudahan penggunaan dan kegunaan [14].

4. Pengguna Nyata (*Actual Use*)

Pengguna nyata merupakan kondisi nyata penggunaan sistem. Dikonsepkan dalam bentuk pengukuran terhadap frekuensi dan durasi waktu penggunaan teknologi. Seseorang akan puas menggunakan sistem jika mereka meyakini bahwa sistem tersebut mudah digunakan dan akan meningkatkan produktifitas mereka, yang tercermin dari kondisi nyata penggunaan.

Berdasarkan uraian di atas dapat disimpulkan bahwa pengguna nyata adalah seseorang yang menjadi pengguna suatu sistem dan meyakini suatu sistem tersebut dapat meningkatkan produktifitas mereka. Bentuk pengukuran pengguna nyata adalah jumlah dan waktu penggunaan terhadap teknologi tersebut. Penggunaan nyata teknologi dapat diukur dengan jumlah waktu yang digunakan oleh pengguna dalam berinteraksi dengan teknologi.

Pengguna nyata memiliki hubungan dengan intensi penggunaan. seseorang yang menjadi pengguna suatu sistem dan meyakini suatu sistem tersebut dapat meningkatkan produktifitas mereka yang dipengaruhi oleh intensi pengguna [14].

2.2.5 Uji Validitas & Reliabilitas

Dalam penelitian kuantitatif, pengujian *validitas* dilakukan untuk mengetahui apakah instrumen benar-benar mampu mengukur apa yang seharusnya diukur. instrumen dinyatakan *valid* apabila nilai *r-hitung* > *r-tabel* pada tingkat signifikansi 5%. Uji *validitas* biasanya dilakukan menggunakan korelasi *Pearson Product Moment* dengan rumus:

$$r = (n\sum XY - \sum X \sum Y) / \sqrt{[(n\sum X^2 - (\sum X)^2)(n\sum Y^2 - (\sum Y)^2)]}$$

Sementara itu, *reliabilitas* digunakan untuk mengukur konsistensi atau keajegan suatu instrumen. Pada penelitian yang sama, *reliabilitas* diukur menggunakan *Cronbach's Alpha*, dan instrumen dikatakan reliabel jika nilai $\alpha > 0,60$. Nilai *Cronbach's Alpha* yang tinggi menunjukkan bahwa butir-butir pertanyaan dalam kuesioner memiliki tingkat konsistensi yang baik antar item [15]. Dilakukan dengan menggunakan rumus *Cronbach's Alpha* sebagai berikut:

$$\alpha = (k / (k - 1)) * [1 - (\sum \sigma^2_i / \sigma^2_t)]$$

2.2.6 Pengertian *User interface (UI)*

User interface (UI) atau antarmuka pengguna merupakan istilah untuk mendeskripsikan tampilan dari komputer yang berinteraksi langsung dengan pengguna. Kualitas dari *user interface (UI)* ditunjukkan dari kesulitan atau kemudahan pengguna untuk dapat memahami sebuah program melalui tampilan antarmuka program tersebut. Tujuan dari *user interface (UI)* adalah merancang *interface* yang efektif untuk sistem perangkat lunak. Efektif artinya siap digunakan, dan hasilnya sesuai dengan kebutuhan. Kebutuhan disini adalah kebutuhan penggunanya. Pengguna sering menilai sistem dari interface bukan dari fungsinya.

Apabila *user interface (UI)* dari sistem buruk, maka hal tersebut sering menjadi alasan pengguna untuk tidak menggunakan produk tersebut. Tujuan dari *user interface (UI)* adalah merancang *interface* yang efektif untuk sistem perangkat lunak. Efektif artinya siap digunakan, dan hasilnya sesuai dengan kebutuhan. Kebutuhan disini adalah kebutuhan penggunanya. Pengguna sering menilai sistem dari interface bukan dari fungsinya. Apabila *user interface (UI)* dari sistem buruk, maka hal tersebut sering menjadi alasan pengguna untuk tidak menggunakan produk tersebut [16].

2.2.7 Komponen Berpengaruh pada *User interface (UI)*

Ada pedoman untuk membuat suatu desain aplikasi yang mudah digunakan yaitu dengan dibagi menjadi bagian-bagian penting:

a. *Concistency*

*TAM*pilan antarmuka pengguna yang konsisten.

b. *Hierarchy*

Hirarki yang disusun berdasarkan kepentingan dari sekumpulan obyek yang terdapat pada aplikasi.

c. *Personality*

Personality mengacu pada impresi yang ditunjukkan oleh identitas dari sebuah sistem.

d. *Layout*

Posisi dari elemen-elemen yang ada pada aplikasi.

e. *Type*

Bentuk tipografi yang digunakan untuk *TAM*pilan aplikasi.

f. *Warna*

Penggunaan warna yang tepat digunakan oleh sebuah sistem.

g. *Imagery*

Imagery berguna untuk mengkomunikasi informasi di dalam aplikasi. *Imagery* dapat berupa gambar, icon, dan lain sebagainya.

h. *Control and affordances*

Keduanya merupakan elemen dari antarmuka pengguna yang dapat digunakan oleh orang untuk berinteraksi dengan sistem melalui suatu layar [16].

2.2.8 Pengertian *User experience (UX)*

Pengertian *User experience (UX)* adalah bagaimana perasaan atau pengalaman pengguna terhadap produk, pengembangan produk, 39 dan saat menggunakan produk atau sebuah layanan pada sebuah *website*. *User experience* merupakan totalitas efek yang dirasakan oleh pengguna sebelum, selama, dan setelah interaksi dengan produk atau sistem dalam suatu lingkungan. Ketika seorang pengguna menggunakan suatu produk, dia dapat mengembangkan perasaan baik atau buruk tentang produk tersebut.

Tugas seorang desainer adalah meneliti pengalaman pengguna menggunakan suatu produk dan mencari tahu sisi desain mana yang mengganggu atau membuat pengguna stres dan memperbaikinya. Desain *UX* mengarah pada optimalisasi kepuasan pengguna setelah menggunakan produk atau layanan. Dan itulah alasan mengapa desain *UX* sangat penting dalam mendesain. Desain *UX* adalah tentang memberikan pengalaman yang bermakna dan relevan kepada pengguna untuk meningkatkan interaksi dan retensi dengan cara melakukan perbaikan secara terus menerus dalam desain untuk kemudahan penggunaan dan kualitas yang dirasakan [17].

2.2.9 Pengertian *User Centered Design (UCD)*

User Centered Design (UCD) adalah sebuah proses desain yang iteratif yang berfokus pada kebutuhan dan keinginan pengguna. Melalui pendekatan *UCD* sebuah desain aplikasi akan dirancang dengan adaptasi terhadap karakteristik atau perilaku pengguna dalam menggunakan aplikasi sehingga aplikasi yang dikembangkan tidak memaksa pengguna untuk mengubah perilakunya ketika menggunakan produk tersebut. Tujuannya adalah agar produk yang dikembangkan dapat bermanfaat serta mudah digunakan bagi pengguna. *UCD* menekankan bahwa di dalam proses perancangan, harus berhubungan langsung atau melibatkan pengguna sesungguhnya atau calon pengguna dengan wawancara, survei, dan *usability testing* [18].

2.2.10 *Skala Likert*

Skala Likert merupakan alat ukur yang digunakan untuk menilai sikap, pandangan, serta persepsi individu atau kelompok terhadap suatu fenomena sosial. Fenomena sosial yang dimaksud dalam konteks penelitian ini telah dirumuskan secara jelas oleh peneliti dan dikenal sebagai variabel penelitian. Melalui penggunaan *skala likert*, variabel yang ingin diukur diuraikan menjadi indikator-indikator. Indikator-indikator tersebut kemudian digunakan sebagai dasar dalam menyusun butir-butir instrumen, baik dalam bentuk pernyataan maupun pertanyaan [19].

Dalam *skala likert*, setiap item pernyataan diberi nilai dengan rentang skor antara 1 sampai 5. Untuk pernyataan yang bernada positif, nilai tertinggi (5) diberikan pada jawaban "sangat setuju", kemudian berturut-turut "setuju" (4), "netral" (3), "tidak setuju" (2), dan "sangat tidak setuju" (1). Sebaliknya, untuk pernyataan yang bernada negatif, penilaian dibalik, di mana skor tertinggi diberikan pada jawaban "sangat tidak setuju" (1), diikuti oleh "tidak setuju" (2), "netral" (3), "setuju" (4), dan "sangat setuju" (5) [20].

Pengukuran User Interface	Pengukuran User Experience
Sistem kendali pada visual yang digunakan	Aplikasi berjalan sesuai fungsi
Interaksi di dalam aplikasi	Aplikasi dapat memenuhi kebutuhan
Penyajian informasi dalam UI	Aplikasi mudah digunakan
Konsistensi visual dalam UI	Aplikasi nyaman digunakan
	Aplikasi menyenangkan untuk digunakan kembali
Pengukuran Motivasi	Aplikasi sangat berarti untuk kemudahan hidup pengguna
Pengguna termotivasi untuk menggunakan aplikasi	

Gambar 2.2 Instrumen pengukuran UI, UX, dan Motivasi [21].

2.2.11 Prototype

Model *prototyping* adalah suatu teknik untuk mengumpulkan informasi spesifik tentang kebutuhan informasi pengguna secara cepat. *Prototyping* merupakan metode pengembangan perangkat lunak, yang berupa model fisik kerja sistem dan berfungsi sebagai versi awal dari sistem. Dengan metode *prototyping* ini akan dihasilkan *prototype* sistem sebagai perantara pengembang dan pengguna agar dapat berinteraksi dalam proses kegiatan pengembangan sistem informasi. Agar proses pembuatan *prototype* ini berhasil dengan baik adalah dengan mendefinisikan kebutuhan awal. *Prototype* akan dihilangkan atau ditambahkan pada bagiannya sehingga sesuai dengan perencanaan dan analisis yang dilakukan oleh pengembang sampai dengan uji coba dilakukan secara *simultan* seiring dengan proses pengembangan [19].

Ada 4 metodologi *prototyping* yang paling utama, yaitu :

1. *Illustrative*, menghasilkan contoh laporan dan tampilan layar.
2. *Simulated*, mensimulasikan beberapa alur kerja sistem tetapi tidak menggunakan data *real*.
3. *Functional*, mensimulasikan beberapa alur sistem yang sebenarnya dan menggunakan data *real*.
4. *Evolutionary*, menghasilkan model yang menjadi bagian dari operasional system.

Dibuatnya sebuah *prototyping* bagi pengembang sistem bertujuan untuk mengumpulkan informasi dari pengguna sehingga pengguna dapat berinteraksi dengan model *prototype* yang dikembangkan, sebab *prototype* menggambarkan versi awal dari sistem untuk kelanjutan sistem sesungguhnya yang lebih besar. Manfaat lain dari penggunaan *prototyping* adalah :

1. Mewujudkan sistem sesungguhnya dalam sebuah replika sistem yang akan berjalan, menampung masukan dari pengguna untuk kesempurnaan sistem.
2. Pengguna akan lebih siap menerima setiap perubahan sistem yang berkembang sesuai dengan berjalannya *prototype* sampai dengan hasil akhir pengembangan yang akan berjalan nantinya.
3. *Prototype* dapat ditambah maupun dikurangi sesuai dengan berjalannya proses pengembangan. Kemajuan tahap demi tahap dapat diikuti langsung oleh pengguna.
4. Penghematan sumberdaya dan waktu dalam menghasilkan produk yang lebih baik dan tepat guna bagi pengguna.

2.2.12 Figma

Figma adalah editor grafik vektor dan alat pembuatan prototipe dengan kemampuan offline berbasis *web* tambahan yang diaktifkan oleh aplikasi desktop untuk Mac OS dan Windows. Aplikasi Figma *mirror* yang disertakan untuk Android dan iOS memungkinkan Anda melihat prototipe Figma di perangkat seluler Anda.

Kumpulan fitur Figma berfokus pada penggunaan dalam antarmuka pengguna dan desain pengalaman pengguna, dengan fokus pada kolaborasi waktu nyata [22].



Gambar 2.3 Logo Figma [23]

Beberapa kelebihan menggunakan Figma [24], sebagai sarana pembuatan desain *UI* adalah :

1. Saat ini Figma masih gratis tanpa dipungut biaya bagi penggunanya.
2. Siapa saja dapat membuka dokumen yang ada di Figma tanpa harus melakukan instalasi. Orang lain juga dapat meninggalkan komentar terhadap file yang baru dia buka.
3. Figma dapat memaksimalkan penggunanya dari segi performa kecepatan saat membuka atau membagi file.
4. Figma bersifat agnostik dan dapat dibuka menggunakan *browser* apapun.
5. Kolaborasi file secara *real time* dan pembaruan file.

Sederhananya, Figma adalah desain digital dan alat *prototyping*. Ini adalah aplikasi desain *UI* dan *UX* yang dapat digunakan untuk membuat situs *web*, aplikasi, atau komponen antarmuka pengguna yang lebih kecil yang dapat diintegrasikan ke dalam proyek lain. Dengan alat berbasis vektor yang hidup di *cloud*, Figma memungkinkan para penggunanya untuk bekerja di mana saja dari browser. Cara ini termasuk alat *zippy* yang dibuat untuk desain, pembuatan prototipe, kolaborasi, dan sistem desain organisasi

Figma memiliki banyak fitur, fitur-fitur utama dalam Figma antara lain sebagai berikut :

1. Alat pena modern yang memungkinkan Anda menggambar ke segala arah dan desain busur instan.
2. Fitur *font Open Type*.
3. Tugas otomatis melalui *plugin* untuk elemen berulang untuk mempercepat proyek.
4. Alat seleksi cerdas dengan penyesuaian otomatis untuk pengaturan jarak, pengaturan, dan pengorganisasian.
5. Metode dan cara fleksibel yang menghemat hak untuk proyek
6. Kemampuan membuat sistem dan komponen desain
7. Perpustakaan dan cara *drag and drop* yang dapat diakses
8. Kemampuan untuk mawadahi banyak audien tanpa batas pada suatu proyek sehingga seluruh pembagian progress kerja tim
9. Kemampuan memeriksa file desain dan *grab code snippets*
10. Fitur ekspor mudah yang memiliki tautan langsung (bukan hanya format PDF datar)
11. Desain interaksi dan prototipe yang dioptimalkan untuk seluler
12. Animasi cerdas untuk menghubungkan objek dan transisi
13. Komentar tersemat (*embedded comment*)
14. Kemampuan untuk mengedit bersama dengan tim secara waktu nyata (*real-time*) di ruang desain bersama.
15. Riwayat versi yang memungkinkan untuk mengetahui segala sesuatu atau *progress* yang telah berubah dan siapa saja yang melakukannya.
16. Buat sistem desain dengan aset yang dapat dicari, gaya dan desain yang dapat dibagikan dan semuanya dalam satu lokasi.
17. Kemampuan membuat komponen yang dapat digunakan kembali dengan kemampuan untuk menyimpannya jika diperlukan.
18. Satu *platform* untuk segalanya, mulai dari desain hingga prototipe.

2.2.13 *Plugin Figma*

Dalam konteks aplikasi desain Figma, *plugin* dapat diartikan sebagai fitur tambahan yang berfungsi untuk memperluas kemampuan dan fungsionalitas perangkat lunak tersebut. *plugin* ini memberikan kesempatan kepada pengguna Figma untuk meningkatkan efisiensi dalam bekerja dan mengeksplorasi berbagai fitur tambahan yang tidak tersedia secara *default* di dalam *platform* Figma. Dengan menggunakan *plugin*, pengguna dapat dengan mudah mengintegrasikan alat dan layanan dari pihak ketiga ke dalam lingkungan Figma, memperkaya pengalaman desain dan meningkatkan produktivitas. Dengan begitu, para pengguna memiliki fleksibilitas tambahan untuk menyesuaikan dan mengoptimalkan alur kerja mereka sesuai dengan kebutuhan dan preferensi pribadi.

Dari perspektif pengguna, *plugin* dapat menjadi aset berharga karena memungkinkan mereka untuk mengakses fitur-fitur khusus dan spesifik yang dirancang oleh pihak ketiga untuk memenuhi kebutuhan desain yang lebih kompleks. Pengguna juga dapat menggabungkan *plugin* yang berbeda untuk menciptakan pengalaman desain yang disesuaikan dengan keinginan dan gaya kerja masing-masing. Dengan adanya fleksibilitas dan potensi peningkatan yang ditawarkan oleh *plugin* dalam Figma, pengguna memiliki akses yang lebih luas ke berbagai alat desain inovatif, integrasi dengan platform eksternal, dan pilihan sumber daya tambahan yang dapat meningkatkan kreativitas dan kualitas hasil akhir proyek desain mereka [25].

2.2.14 *Plugin Figma Iconify*

Plugin ini menyediakan ribuan ikon dari berbagai pustaka dalam satu antarmuka terpadu, yang sangat membantu dalam proses eksplorasi dan perancangan desain *UI/UX*. *Iconify* menjadi contoh nyata bagaimana *plugin* dapat memperkaya proses eksplorasi visual secara signifikan. Dengan mengintegrasikan *Iconify* ke dalam alur kerja desain menggunakan Figma, pengguna dapat dengan mudah mengakses berbagai macam ikon tanpa perlu keluar dari aplikasi. Hal ini memberikan kemudahan dalam pencarian ikon yang sesuai dengan konsep desain yang sedang dibangun, sehingga mempercepat proses kerja dan meminimalkan hambatan teknis.

Selain itu, variasi ikon yang tersedia dalam *plugin* Iconify mencakup berbagai gaya dan kategori, seperti *flat*, *line*, *filled*, hingga *duotone*, yang memungkinkan desainer memilih ikon yang paling relevan sesuai dengan kebutuhan tampilan antarmuka yang diinginkan. Kemampuan ini mendukung proses iterasi dan eksperimentasi desain dengan lebih fleksibel dan efisien [26].

2.2.15 Plugin HTML to Design untuk Figma

Plugin seperti *StarryStudioAI* mampu mengonversi *HTML* dan *CSS* menjadi desain visual secara otomatis di Figma. Teknologi ini sangat membantu dalam menjembatani kesenjangan antara pengembang dan desainer. *Plugin* ini mencapai *BERT F1 score* sebesar 0.79 dan memperoleh tingkat kepuasan pengguna lebih dari 80% dalam eksperimen lintas latar belakang teknis [27].

2.2.16 Plugin Text-to-Design untuk Figma

Plugin text-to-design berbasis *AI* di Figma menggunakan input teks (deskripsi) untuk membangun antarmuka visual secara otomatis. Ini mempermudah kolaborasi lintas fungsi, terutama saat tim non-desain perlu menyampaikan ide produk secara cepat. *StarryStudioAI* juga menawarkan fitur ini melalui pemrosesan natural language prompt yang dikonversi menjadi layout *pixel-perfect*. Dengan memanfaatkan *GPT-4o* dan teknik *RAG*, *plugin* ini mempercepat iterasi awal desain dari perintah teks ke tampilan *UI* yang dapat langsung diedit di Figma [27].

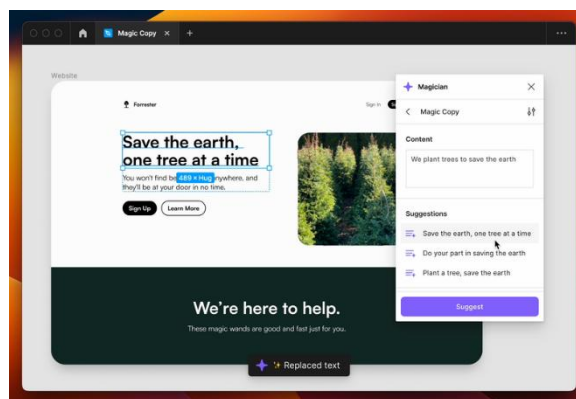
2.2.17 Plugin Magician untuk Figma

Magician adalah *plugin* *AI* untuk Figma yang memungkinkan desainer menghasilkan teks, ikon, dan ilustrasi hanya dari deskripsi teks (*text prompt*). *Plugin* ini dikenal karena kemampuannya membantu proses *copywriting*, *icon generation*, dan *layer renaming* secara otomatis. Meskipun belum tersedia jurnal ilmiah *peer-reviewed* dengan *DOI* yang membahas *plugin* ini, sumber resmi menyatakan bahwa *Magician* menggunakan model *AI* generatif untuk mempercepat eksplorasi ide dan meningkatkan efisiensi desain [28].

2.2.18 Plugin *MagicoPy* Figma

Plugin MagicoPy merupakan bagian dari ekosistem *Magician*, sebuah *AI tool* yang terintegrasi dengan Figma untuk membantu desainer mempercepat proses kreatif dengan dukungan teknologi kecerdasan buatan. Dalam pengembangannya, *Magician* memanfaatkan *Figma Text Review API* untuk mengakses dan mengolah teks dalam *canvas* desain secara *real-time*. Hal ini memungkinkan *AI* memberikan saran perbaikan konten berbasis konteks serta membuat prediksi desain yang lebih sesuai dengan kebutuhan pengguna. Teknologi ini tidak hanya mempercepat iterasi desain, tetapi juga meningkatkan kualitas desain dengan mempertimbangkan data dan preferensi pengguna secara adaptif.

Selain kemampuan analisis teks secara dinamis, *MagicoPy* juga mengimplementasikan model *language-based AI* untuk menghasilkan elemen desain seperti *copywriting* dan *visual suggestion* yang sesuai konteks. *Plugin* ini mengurangi kebutuhan berpindah aplikasi saat mengembangkan ide desain, karena integrasi langsung dalam Figma memungkinkan desainer mendapatkan umpan balik instan tanpa mengganggu alur kerja kreatif. Kemampuan ini menjadikan *MagicoPy* sebagai alat bantu yang mendukung produktivitas serta menghemat waktu dalam proses desain antarmuka pengguna [29].



Gambar 2.4 *Magicopy in Magician*