

BAB 3 METODOLOGI PENELITIAN

3.1 Studi Identifikasi Masalah

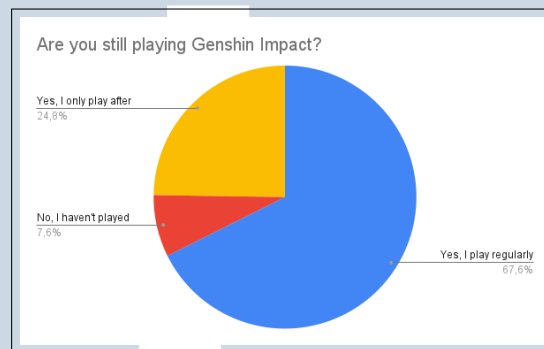
Studi identifikasi masalah dilakukan dengan melakukan survei pada permainan RPG bernama *Genshin Impact*. Survei ini dilaksanakan mulai dari tanggal 8 hingga 12 Mei 2024. Survei disebarakan dalam bentuk *Google Forms* kepada forum global komunitas *Genshin Impact*. Masing-masing *email* responden hanya dapat memberikan satu respon. Terdapat 105 responden pemain *Genshin Impact*, dan terdapat tiga pertanyaan survei.

Pertanyaan pertama menanyakan tanggal responden tersebut mulai bermain *Genshin Impact*. Pertanyaan ini diperlukan karena permainan *Genshin Impact* dapat berubah dengan berjalannya waktu. Tanggal yang lebih tua merupakan pemain yang memainkan versi *Genshin Impact* dengan variasi konten yang lebih minimalis. Terdapat 74% responden yang mulai bermain *Genshin Impact* setelah 1 Januari 2023, sementara *Genshin Impact* telah diterbitkan sejak 28 September 2020. Gambar 3.1 menunjukkan distribusi jawaban pada pertanyaan pertama.



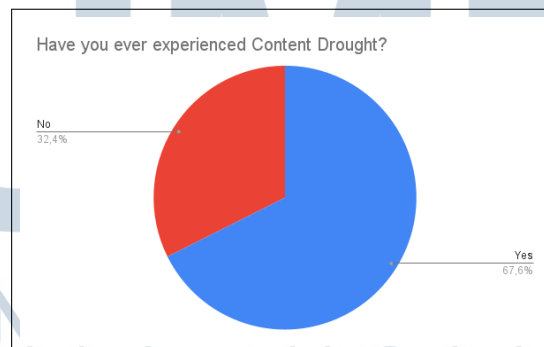
Gambar 3.1. Distribusi jawaban survei *Genshin Impact* pada pertanyaan pertama.

Pertanyaan kedua menanyakan jika responden tersebut masih aktif dalam bermain *Genshin Impact*. Pertanyaan ini diperlukan karena frekuensi bermain dapat menentukan kecepatan pemain mencapai akhir cerita dalam permainan. Terdapat 67% responden yang masih aktif bermain *Genshin Impact*, terdapat 24% responden yang bermain ketika terdapat konten baru, dan sisanya adalah responden yang sudah tidak bermain *Genshin Impact*. Gambar 3.2 menunjukkan distribusi jawaban pada pertanyaan kedua.



Gambar 3.2. Distribusi jawaban survei *Genshin Impact* pada pertanyaan kedua.

Pertanyaan ketiga menanyakan jika responden tersebut telah mencapai akhir cerita. Terdapat 67% responden yang telah mencapai akhir cerita. Data ini membuktikan bahwa terdapat keterbatasan konten dalam *Genshin Impact*. Gambar 3.3 menunjukkan distribusi jawaban pada pertanyaan ketiga.



Gambar 3.3. Distribusi jawaban survei *Genshin Impact* pada pertanyaan ketiga.

3.2 Landasan Teori

Mempelajari cara kerja dari *Large-Language-Model*, *Transformer*, mekanisme *Attention*, dan *Stable Diffusion*. Mempelajari hubungan antara mekanisme *Attention* dan *Transformer* dalam *Large-Language-Model*.

Mempelajari teknik-teknik *prompt* yang dapat dipakai dalam *Large-Language-Model*.

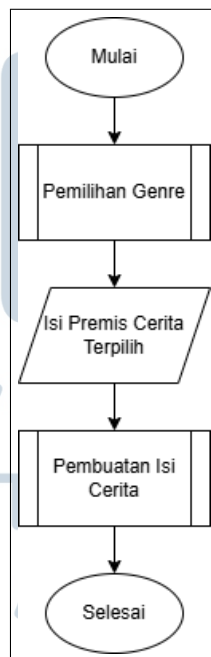
3.3 Perancangan Prototipe

3.3.1 Flowchart

Flowchart dibuat melalui *draw.io* untuk merencanakan fitur teknis dalam aplikasi *visual novel*. Berikut adalah ilustrasi *flowchart* yang dipakai dalam membuat alur kerja sistem *visual novel*.

A Alur Utama

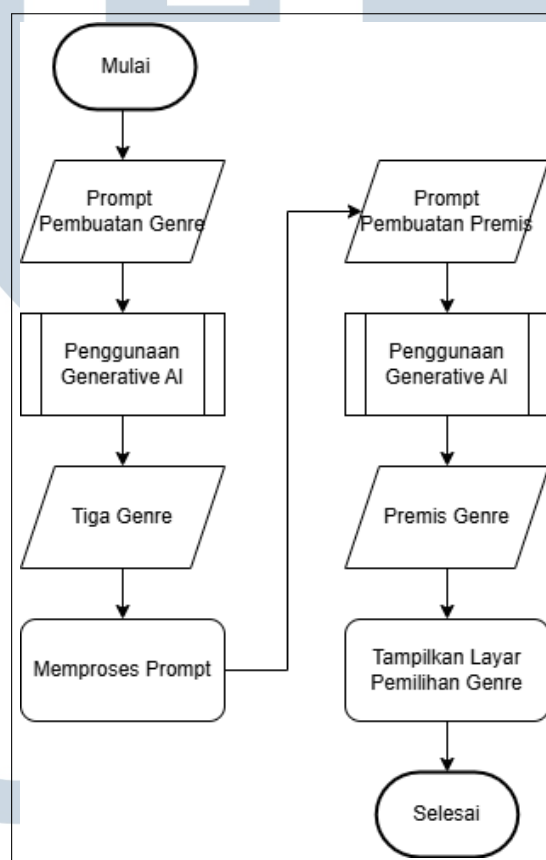
Aplikasi *visual novel* akan dibagi menjadi dua bagian utama yaitu bagian Pemilihan *Genre* Cerita dan bagian Pembuatan Isi Cerita. Dalam bagian Pemilihan *Genre* Cerita, aplikasi menyediakan pilihan *genre* cerita yang akan dibuat. Setelah sebuah *Genre* dipilih, aplikasi menjalankan bagian Pembuatan Isi Cerita. Dalam bagian Pembuatan Isi Cerita, aplikasi menyediakan isi cerita dalam bentuk teks dan gambar. Alur Utama *visual novel* dapat diilustrasikan melalui *flowchart* dalam Gambar 3.5.



Gambar 3.4. *Flowchart* Alur Utama

B Pemilihan Genre Cerita

Sebelum pembuatan cerita, pemain akan diberikan pilihan *Genre* cerita untuk diproduksi. Pada awal pembukaan aplikasi, LLM ditugaskan untuk memberikan tiga ide *genre* untuk cerita pendek. Setelah itu, masing-masing *genre* diberikan kepada LLM dengan tugas membuat *premis* menarik sepanjang satu kalimat. Setelah semua *premis* selesai dibuat, aplikasi berpindah dari halaman memuat menjadi halaman pemilihan *genre*. Proses pembuatan pilihan *genre* dapat diilustrasikan melalui *flowchart* dalam Gambar 3.5.

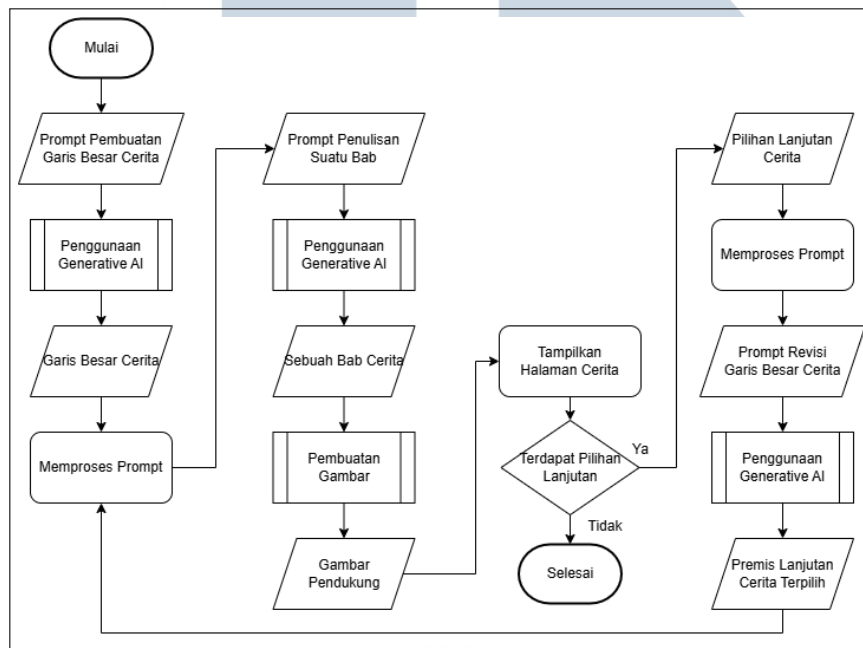


Gambar 3.5. *Flowchart* Pemilihan Genre

C Pembuatan Isi Cerita

Setelah pemain memilih *genre*, cerita dibuat secara perlahan. Dimulai dari pembuatan garis besar isi masing-masing bab. Lalu bab pertama mulai ditulis, gambar pendukung dibuat, dan pada akhir cerita dapat disediakan dua pilihan. Setelah terpilih sebuah lanjutan cerita, garis besar bab selanjutnya diperbarui

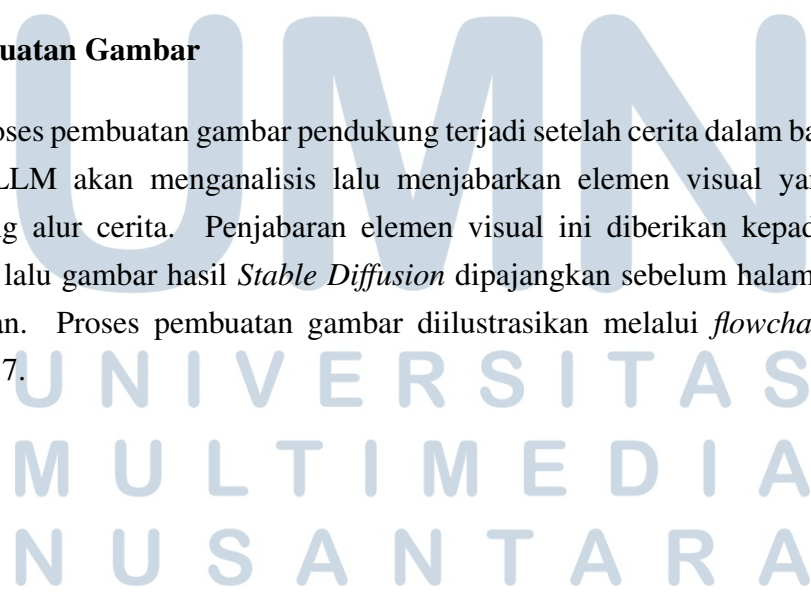
dengan alur cerita yang dikehendaki. Lalu proses pembuatan cerita berulang untuk bab selanjutnya, hingga tidak terdapat lanjutan cerita. Jika tidak terdapat lanjutan, proses pembuatan cerita sesi ini telah selesai. Proses pembuatan cerita dapat diilustrasikan melalui *flowchart* dalam Gambar 3.6.

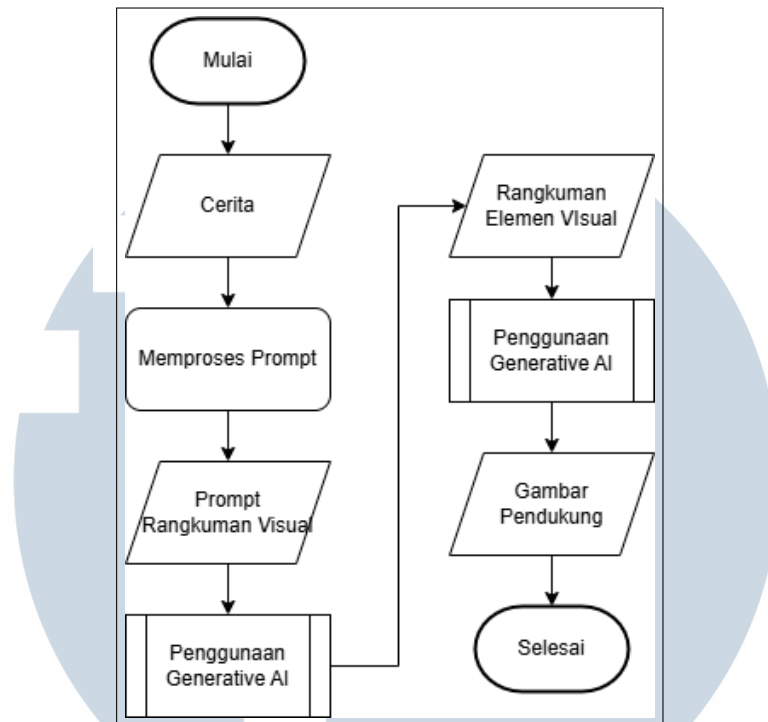


Gambar 3.6. *Flowchart* Pembuatan Cerita

D Pembuatan Gambar

Proses pembuatan gambar pendukung terjadi setelah cerita dalam bab selesai ditulis. LLM akan menganalisis lalu menjabarkan elemen visual yang dapat mendukung alur cerita. Penjabaran elemen visual ini diberikan kepada *Stable Diffusion*, lalu gambar hasil *Stable Diffusion* dipajangkan sebelum halaman cerita ditampilkan. Proses pembuatan gambar diilustrasikan melalui *flowchart* dalam Gambar 3.7.



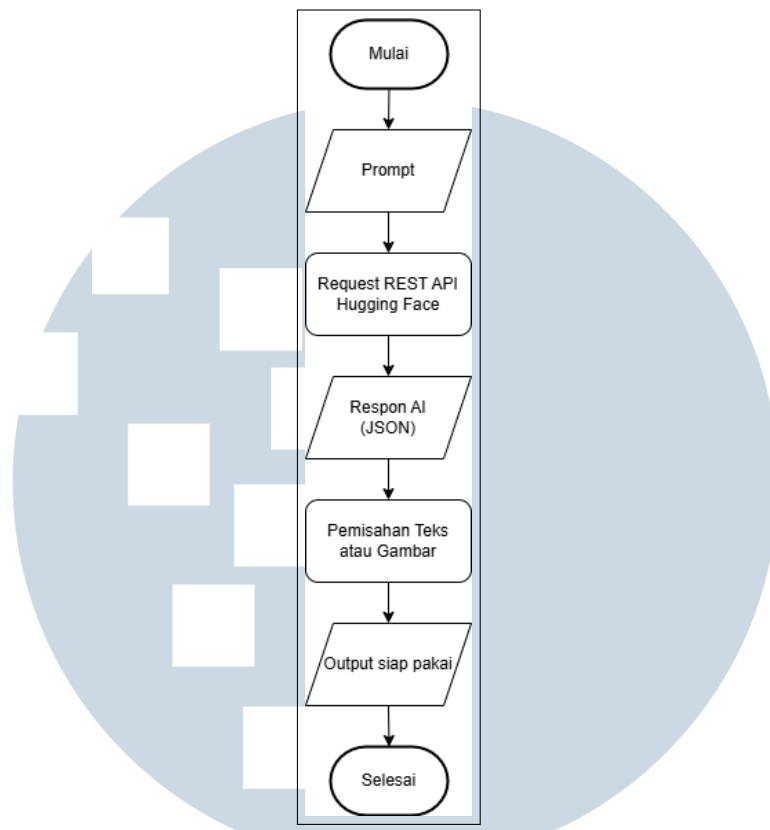


Gambar 3.7. Flowchart Pembuatan Gambar

E Penggunaan *Generative AI*

Setiap penggunaan *Generative AI* untuk *output* teks atau gambar memiliki alur proses yang sama. Pertama dilakukan pembuatan *prompt* serta pengaturan parameter. Lalu aplikasi lokal melakukan *request POST* dengan isi data sebelumnya, serta menggunakan *link* model dan *API key*. *Request POST* akan dibalas dengan hasil data buatan *Generative AI*. Data ini berupa *JSON*, sehingga terdapat proses pemisahan data agar dapat dipakai komponen *GameObject* dalam *Unity3.8*.

UNIVERSITAS
MULTIMEDIA
NUSANTARA



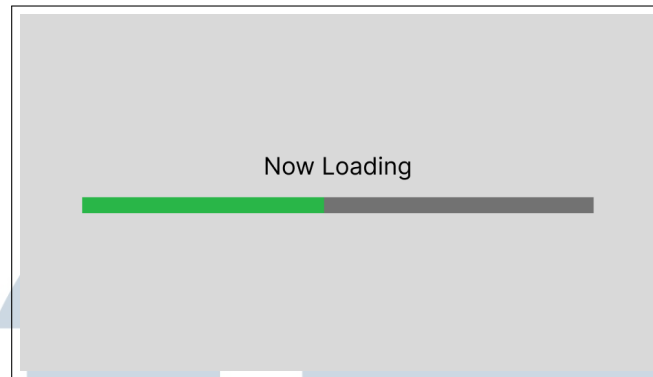
Gambar 3.8. *Flowchart* Penggunaan *Generative AI*

3.3.2 Mockup

Mockup atau desain antarmuka dibuat melalui *Figma* untuk memandu pembuatan aplikasi *visual novel*. Berikut adalah *mockup* yang dipakai dalam pembuatan *visual novel*.

A Halaman Memuat

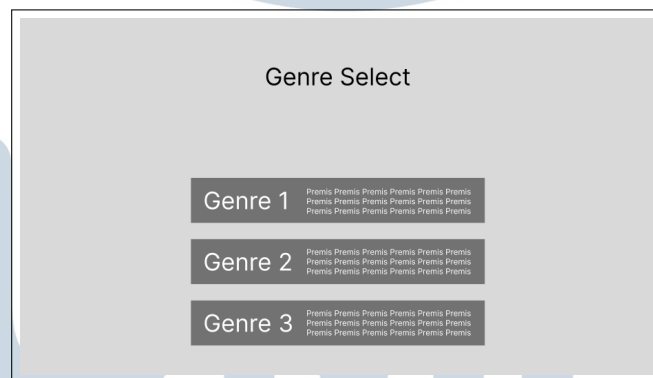
Halaman memuat adalah halaman yang dipakai untuk menggantikan halaman yang aktif ketika aplikasi sedang melakukan proses yang berhubungan dengan *Generative AI*. Halaman ini hanya berisi sebuah *slider* yang menunjukkan jumlah proses yang selesai dibandingkan dengan jumlah seluruh proses yang ditunggu. *Mockup* halaman memuat dapat dilihat pada Gambar 3.9.



Gambar 3.9. *Mockup* halaman memuat

B Halaman Pemilihan *Genre*

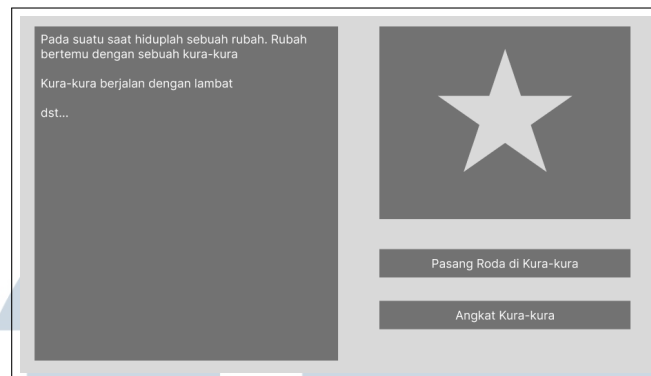
Halaman pemilihan *genre* adalah halaman yang dipakai untuk menampilkan pilihan *genre* serta masing-masing premisnya. Halaman ini berisi tiga tombol dengan teks *genre* serta premis. Setelah salah satu *genre* terpilih, halaman ini berpindah menjadi halaman cerita setelah melalui proses halaman memuat. *Mockup* halaman pemilihan *genre* dapat dilihat pada Gambar 3.10.



Gambar 3.10. *Mockup* halaman pemilihan *genre*

C Halaman Cerita

Halaman cerita adalah halaman yang dipakai untuk menampilkan konten cerita dalam bentuk teks dan gambar. Halaman ini berisi sebuah *textbox* berisi konten cerita teks, sebuah *sprite* berisi konten cerita gambar, dan dua buah tombol berisi deskripsi pilihan alur cerita. *Mockup* halaman cerita dapat dilihat pada Gambar 3.11.



Gambar 3.11. Mockup halaman cerita

3.4 Implementasi

Penelitian ini menggunakan model *Generative AI Large-Language-Model* bernama *mistralai/Mixtral-8x7B-Instruct-v0.1* serta model AI *Text-To-Image* bernama *stabilityai/stable-diffusion-xl-base-1.0*, keduanya di-host dalam cloud pada website *Hugging Face*. Aplikasi diterapkan dalam *Unity Engine* versi 2022.3.26f1. Model *sentence similarity* yang digunakan untuk mengukur keserupaan premis bernama *sentence-transformers/all-MiniLM-L6-v2*. Akses antara aplikasi *Unity* dan website *Hugging Face* diterapkan melalui *REST API*.

Hugging Face merupakan sebuah forum AI yang menyediakan berbagai sumber daya *Dataset* untuk melatih AI, serta berbagai model AI siap pakai. *Hugging Face* juga menyediakan berbagai model *Generative AI* dipakai dalam penelitian ini. Model AI yang disediakan *Hugging Face* dapat di akses melalui *REST API* tanpa dipungut biaya.

3.5 Uji Coba

Dalam pengujian, alur kerja sistem diubah untuk memungkinkan aplikasi menciptakan berbagai cerita berturut-turut. Data *genre* dan *premis* hasil LLM disimpan dalam *file format CSV*, Aplikasi tidak menggunakan *stable diffusion*. Untuk mendapatkan pengukuran yang akurat, *prompt* yang dipakai dalam pengujian tidak diubah dari *prompt* yang dipakai oleh penggunaan normal.

Pada saat aplikasi dibuka secara normal, terdapat tiga pilihan *genre*. Ketiga *genre* ini selalu unik karena bersumber dari satu *prompt*. *Prompt* dapat secara mudah diubah untuk membuat lebih banyak *genre* sekaligus, namun untuk mensimulasikan pemakaian normal, *genre* tetap dibuat sebanyak tiga sekaligus,

namun diulang berkali-kali sesuai kebutuhan skenario uji coba.

3.6 Dokumentasi

Proses dan hasil penelitian ditulis dalam laporan ini untuk kepentingan Skripsi. Laporan ini telah melalui proses bimbingan dan pengujian Skripsi.

