

BAB II

KAJIAN PUSTAKA

A. Kajian Teorits

1. Rancang Bangun

Menurut Surahman et al., (2022) rancang bangun sistem adalah proses menerjemahkan hasil analisis ke dalam bahasa pemrograman untuk mendesain dan mengimplementasikan komponen sistem. Proses ini mencakup pembuatan, penggantian, atau perbaikan sistem, serta mengubah analisis menjadi perangkat lunak secara menyeluruh atau parsial. Pengertian lainnya menurut Aprilia & Rudi, (2020) Rancang bangun adalah proses menerjemahkan hasil analisis ke dalam bahasa pemrograman untuk mendeskripsikan implementasi sistem secara rinci. Pembangunan sistem mencakup pembuatan, penggantian, atau perbaikan sistem, baik secara menyeluruh maupun parsial. Pendapat lainnya dari Gunawan et al., (2021) Rancang bangun adalah proses menggambarkan, merancang, dan menyusun sketsa atau mengatur berbagai elemen yang terpisah menjadi satu kesatuan yang utuh dan berfungsi. Dengan kata lain, rancang bangun melibatkan penerjemahan hasil analisis ke dalam bentuk paket perangkat lunak, serta pembuatan atau perbaikan sistem yang sudah ada.

Berdasarkan pengertian di atas, rancang bangun adalah proses menerjemahkan hasil analisis ke dalam bahasa pemrograman untuk

merancang dan mengimplementasikan komponen sistem. Proses ini mencakup pembuatan, penggantian, atau perbaikan sistem secara menyeluruh atau parsial, dengan tujuan mengubah analisis menjadi perangkat lunak yang utuh dan berfungsi.

2. *Game RPG (Role Playing Game)*

Menurut Riyan Rinaldi et al., (2023) *game* adalah suatu permainan yang digunakan untuk bermain, yang pada umumnya bertujuan untuk menghibur ataupun mendapatkan kesenangan. Pendapat lainnya dari Ashari et al., (2023), selain untuk hiburan mencari kesenangan, *game* juga bisa menjadi sarana edukasi tergantung bagaimana kita menggunakannya. Pendapat dari Remington Huan & Anggriani Adnas, (2023) *game* memiliki banyak sekali genre yang bisa dimainkan seperti contohnya menyusun strategi, *game* bertahan hidup, tembak-menembak, horror, aksi, *RPG* dan masih banyak lagi. Dan salah satu yang populer di kalangan penikmat *game* adalah genre *RPG*. Menurut penjelasan dari Irkinovich, (2022) *Game RPG* adalah genre permainan di mana pemain memerankan karakter tertentu dan menjalani alur cerita yang kaya serta penuh pilihan yang memengaruhi hasil akhir



Gambar 2.1 Contoh Game *RPG*

Sumber : Chusyairi et al., (2020)

Berdasarkan pengertian di atas, *game* adalah media hiburan yang memberikan kesenangan, dan tergantung dari cara memakai atau memanfaatkannya, *game* juga bisa memberikan edukasi bagi penggunanya. *Game* juga memiliki beberapa genre yang dapat dimainkan, dan dalam genre tersebut pemain akan memerankan peran sebagai karakter dalam *game* tersebut.

3. *Turn-Based Battle*

Menurut Turangan D.V (2020) *turn-based battle* adalah salah satu subgenre dari genre aksi sekaligus bagian dari genre strategi. Sistem ini menghadirkan mekanisme permainan di mana pertarungan berlangsung secara bergiliran antara pemain dan AI atau pemain lain. Saat giliran pemain tiba, mereka memiliki kesempatan untuk memanfaatkan waktu yang tersedia dengan cermat dalam merencanakan strategi terbaik untuk mengalahkan lawannya. Selanjutnya menurut Marjuni et al., (2021) menjelaskan *turn-based battle* adalah sistem pertarungan di mana setiap karakter mengambil tindakan secara bergantian, dengan urutan yang dipengaruhi oleh atribut seperti kecepatan. Ketika giliran karakter pemain tiba,

menu tindakan seperti serangan atau pergerakan akan muncul, dan waktu akan berhenti sementara hingga pemain menentukan langkah yang diinginkan. Dalam sistem ini, baik lawan maupun rekan tim menunggu hingga keputusan dibuat oleh pemain.

Berdasarkan pengertian di atas, Turn-based battle adalah sistem pertarungan bergiliran yang merupakan bagian dari genre aksi dan strategi. Dalam sistem ini, pemain atau karakter mengambil tindakan secara bergantian, dengan urutan yang sering dipengaruhi oleh atribut seperti kecepatan. Saat giliran pemain tiba, waktu berhenti sementara, memungkinkan mereka untuk merencanakan langkah terbaik, baik itu menyerang, bergerak, atau mengambil tindakan lainnya, sehingga memberikan peluang untuk strategi yang matang dalam menghadapi lawan.

4. Cerita Rakyat

Menurut Darihastining, (2022) cerita rakyat adalah cerita dari masa lampau yang muncul dan tersebar di masyarakat secara lisan. Cerita rakyat selalu hidup di tengah-tengah masyarakat dan sudah ada sejak zaman dahulu dan tersebar secara lisan dari mulut ke mulut. Sedangkan menurut Retnasari et al., (2023) cerita rakyat adalah serangkaian peristiwa yang menunjukkan sebuah kisah pada masa lampau yang bersumber dan berkembang pada masyarakat saat itu. Cerita rakyat merupakan suatu cerita atau kisah yang bersifat asli dan berasal dari masyarakat. Cerita rakyat juga memiliki fungsi sebagai hiburan dan fungsi lainnya. Pendapat lain dari Uswatun Khasanah et al., (2022), cerita rakyat adalah budaya yang melekat pada masyarakat. Oleh karena itu alur, bahasa, tradisi, dan budaya yang ada dalam cerita rakyat kerap memiliki kedekatan dalam kehidupan masyarakat.

Berdasarkan pengertian di atas, cerita rakyat adalah kisah masa lampau yang tersebar secara lisan dalam masyarakat dan menjadi bagian penting dari budaya.

Cerita ini tidak hanya berfungsi sebagai hiburan tetapi juga mencerminkan tradisi, bahasa, dan nilai-nilai budaya yang dekat dengan kehidupan masyarakat, menjadikan suatu warisan yang berkembang dan melekat pada komunitas sejak zaman dahulu.

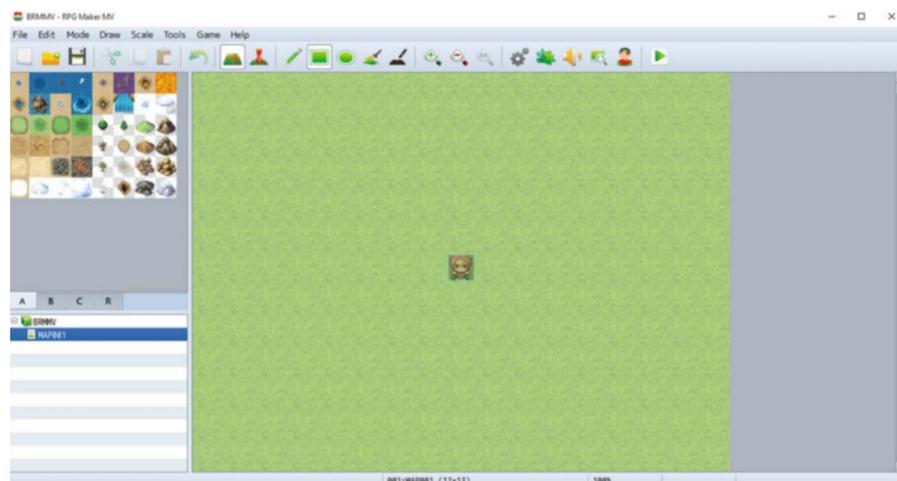
5. Hanoman

Dikutip dari buku karya Farobi Z, (2019). Hanoman adalah sebuah perwujudan dari seekor kera. Hanoman mempunyai ayah bernama Batara Bayu (Dewa Angin) dan ibu bernama Dewi Anjani. Dikarunia kekuatan yang luar biasa, Hanoman memiliki kemampuan terbang, kecepatan, dan kekebalan terhadap senjata. Ia memiliki karakter yang setia, cerdas, dan berani. Dalam kisahnya, Hanoman juga dikenal memiliki sifat pengabdian tinggi, rela berkorban, serta memiliki kebijaksanaan dalam menghadapi berbagai rintangan. Hanoman membuat namanya dihormati dan menjadikan dia sebagai symbol kekuatan dan kesetiaan.

6. *RPG Maker MV*

Berdasarkan buku yang ditulis oleh Perez, (2016) *RPG Maker MV* sebuah *game engine* yang dipublis oleh *Degica* dan di developer oleh *Kadokawa Games*. Diluncurkan secara internasional pertama kali pada 23 Oktober 2015 dan 17 Desember 2015 di Jepang. Menurut Fitriyani & Nita, (2021) *RPG Maker MV* adalah sebuah *tools* atau *game engine* yang umumnya digunakan untuk membuat sebuah *game* dengan genre *RPG* yang menekankan pada grafik 2 dimensi. Pendapat Wahyuni & Andiyoko, (2020) menjelaskan *RPG Maker MV* adalah

sebuah program yang dimana pengguna bisa membuat *game* RPG dengan ide kreativitas mereka sendiri.



Gambar 2.2 Tampilan *Software RPG Maker MV*

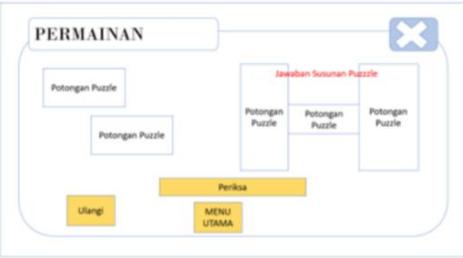
Sumber : Perez, (2016)

Berdasarkan pengertian di atas, *RPG Maker MV* adalah sebuah *game engine* yang dipublikasikan oleh *Kadokawa Games* dengan *tools* yang umumnya digunakan untuk game bergenre *RPG* yang menekankan pada grafik 2 dimensi.

7. *Storyboard*

Menurut Angga & Widodo, (2021) *storyboard* adalah serangkaian ilustrasi berurutan yang memberikan gambaran awal tampilan film, membantu sutradara dan sinematografer memvisualisasikan adegan, memeriksa detail, dan mengantisipasi potensi masalah. Pendapat lainnya dari Sari N, Yaniawati P, Firmansyah E et al., (2023) *storyboard* adalah tahap awal dalam visualisasi sebelum produksi game interaktif, yang memberikan rincian mengenai urutan, adegan, dan alur cerita untuk memudahkan proses produksi. *Storyboard* berfungsi sebagai representasi visual yang menunjukkan hasil akhir yang diinginkan dari sebuah cerita. Menurut Novita et al., (2020) *Storyboard* adalah kegiatan pra-

penulisan yang berfokus pada penjelasan rinci, perkiraan atau prediksi, pengembangan ide, serta penyusunan urutan.

Papan Gambar	Waktu	Keterangan
		<p>Rancangan Halaman Menu Utama</p> <p>Terdapat <i>Button</i>:</p> <ol style="list-style-type: none"> 1. Batik 2. Permainan 3. Bantuan <p>Musik <i>Slow</i></p>
		<p>Rancangan Halaman Menu Permainan</p> <p>Terdapat <i>Button</i>:</p> <ol style="list-style-type: none"> 1. Ulangi 2. Periksa 3. Menu Utama 4. Keluar <p>Terdapat potongan puzzle batik yang harus disusun</p> <p>Musik <i>Slow</i></p>

Gambar 2.3 Contoh *storyboard*

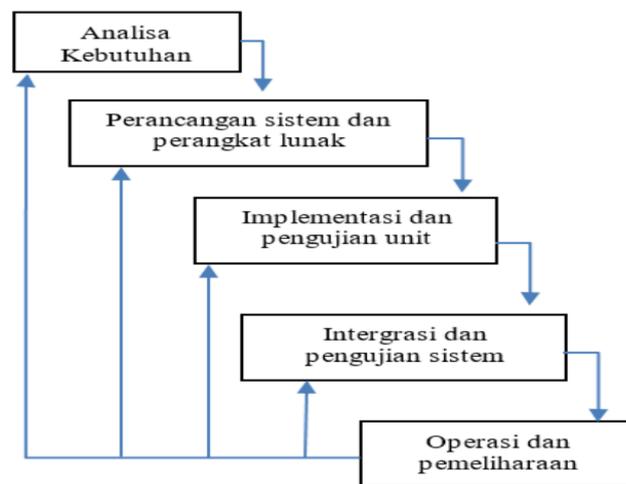
Sumber :Ariyana et al., (2022)

Berdasarkan pengertian di atas, *storyboard* adalah serangkaian gambar yang disusun berurutan untuk memberikan gambaran awal tentang film atau game. Ini berfungsi sebagai panduan bagi tim kreatif untuk memvisualisasikan alur cerita, memeriksa detail, dan mengatasi potensi masalah, serta membantu mengembangkan ide dan urutan adegan secara terstruktur.

8. *Waterfall*

Menurut Rafi & Indahyanti, (2022) *waterfall* juga dikenal dengan sebutan model linier sekuensial atau siklus hidup klasik. Model ini menawarkan pendekatan perangkat lunak yang dilakukan secara berurutan, dimulai dari tahap analisis, desain, pengodean, pengujian, hingga tahap-tahap pendukung lainnya. Pendapat Putra et al., (2022) menjelaskan *Waterfall*, yang juga dikenal sebagai

metode sekuensial linier, adalah pendekatan pengembangan perangkat lunak yang dilakukan secara berurutan, dimulai dari analisis, desain, pengodean, pengujian, hingga pemeliharaan. Ciri utama dari metode ini adalah sifatnya yang sekuensial, artinya setiap tahap hanya dapat dilanjutkan setelah tahap sebelumnya selesai. Hal ini membuat metode waterfall mudah untuk dipahami. Pendapat lainnya dari Yanti & Nurhayati, (2023) *waterfall* fokus pada urutan langkah-langkah dalam proses pengembangan perangkat lunak. Pendekatan ini cocok diterapkan untuk pengembangan perangkat lunak yang relatif sederhana dan melibatkan sumber daya manusia yang terbatas.



Gambar 2.4 Metode *waterfall*

Sumber : Yauma et al., (2021)

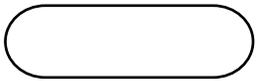
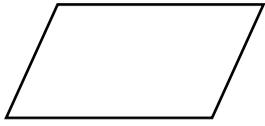
Berdasarkan pengertian di atas, *waterfall* adalah pendekatan pengembangan perangkat lunak yang dilakukan secara berurutan, dimulai dari analisis hingga pemeliharaan. Setiap tahap harus selesai sebelum melanjutkan ke tahap berikutnya. Metode ini cocok untuk pengembangan perangkat lunak sederhana dengan sumber daya manusia terbatas dan mudah dipahami.

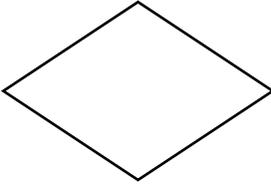
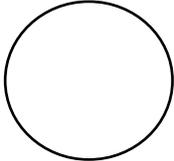
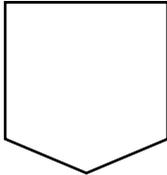
9. *Flowchart*

Menurut Zulfa & Wanda, (2023) *flowchart* adalah diagram yang menggambarkan sebuah sistem otomatis, manual, atau kombinasi keduanya,

dengan menyusun komponen-komponen sistem yang saling terhubung sesuai dengan aturan yang berlaku. Menurut pendapat lainnya dari Dalimunthe, (2022) *flowchart* adalah diagram grafis yang menunjukkan alur langkah-langkah penyelesaian masalah, menggambarkan urutan prosedur dalam program, serta membantu memecah masalah dan menganalisis alternatif operasional. Pendapat dari Zalukhu et al., (2023) *flowchart* adalah representasi grafis langkah-langkah dan prosedur dalam program. *Flowchart* sistem menunjukkan urutan proses, media input, output, dan penyimpanan data, sedangkan flowchart program menggunakan simbol untuk menggambarkan hubungan antar proses secara rinci. Berikut di bawah ini tabel pengertian *flowchart* yang ada dalam penelitian ini :

Table 2.1 Fungsi simbol pada *flowchart*

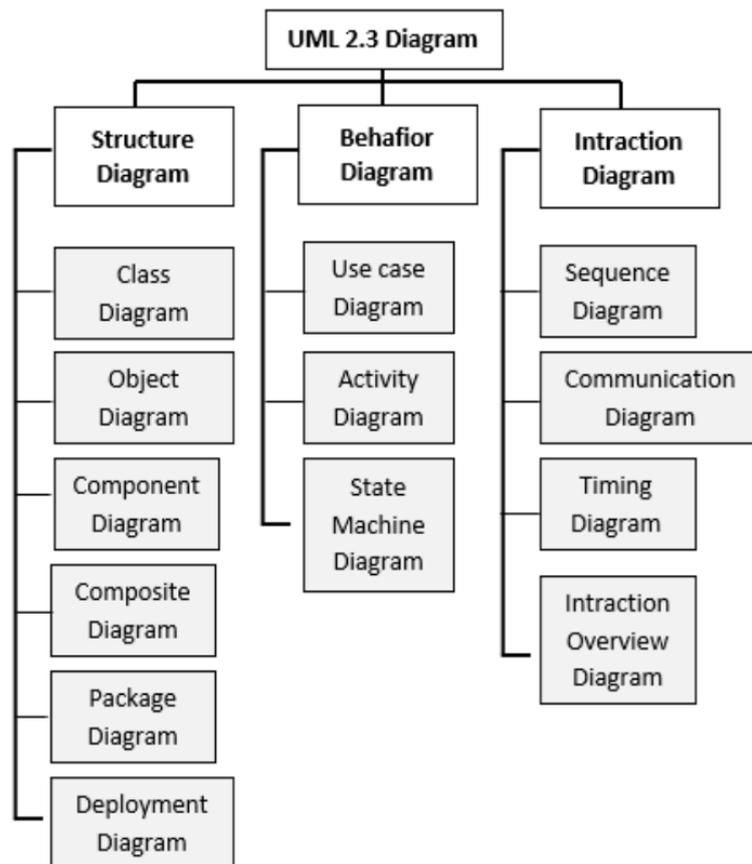
Simbol	Nama	Fungsi
	<i>Terminator</i>	Permulaan/akhir program
	Garis alir (<i>flow line</i>)	Arah aliran program
	<i>Preparation</i>	Proses inisialisasi
	<i>Process</i>	Proses perhitungan/proses pengolahan data
	<i>Input/output data</i>	Proses input/output data, parameter, informasi

	<p><i>Predefines process</i> (sub program)</p>	<p>Permulaan sub program/proses menjalankan sub program</p>
	<p><i>Decision</i></p>	<p>Perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya</p>
	<p><i>On page connector</i></p>	<p>Penghubung bagian-bagian <i>flowchart</i> yang berada dalam satu halaman</p>
	<p><i>Off page connector</i></p>	<p>Penghubung bagian-bagian <i>flowchart</i> yang berada pada halaman berbeda</p>

Berdasarkan pengertian di atas, *flowchart* adalah diagram visual yang menggambarkan urutan langkah atau prosedur dalam sebuah sistem atau program, baik otomatis, manual, maupun kombinasi keduanya. Diagram ini membantu memecah masalah menjadi bagian yang lebih mudah dianalisis. *Flowchart* sistem menunjukkan alur proses dalam sistem, sedangkan *flowchart* program menggambarkan urutan dan hubungan antar proses dalam program menggunakan simbol khusus.

10. UML

Menurut Mubarak, (2019) *UML (Unified Modeling Language)* adalah bahasa berbasis grafis yang digunakan untuk memvisualisasikan, merancang, membangun, dan mendokumentasikan sistem pengembangan perangkat lunak berbasis *Object-Oriented (OO)*. *UML* menyediakan standar dalam pembuatan cetak biru sistem, mencakup proses bisnis, definisi kelas dalam bahasa pemrograman tertentu, skema basis data, serta berbagai komponen yang diperlukan dalam sistem perangkat lunak. Menurut Sumiati et al., (2021) *UML* adalah bahasa pemodelan perangkat lunak yang telah distandarisasi untuk digunakan sebagai alat dalam menyusun cetak biru perangkat lunak. *UML* dapat dimanfaatkan untuk memvisualisasikan, merancang, membangun, dan mendokumentasikan berbagai komponen sistem dalam perangkat lunak. Menurut Voutama, (2022) *UML (Unified Modeling Language)* adalah alat bantu atau metode pemodelan yang sangat andal dalam pengembangan sistem berorientasi objek. *UML* merupakan hasil penggabungan dari beberapa bahasa pemodelan, yaitu *Booch*, *Object Modeling Technique (OMT)*, dan *Object-Oriented Software Engineering (OOSE)*.



Gambar 2.5 Pembagian kategori diagram *UML*

Sumber : Voutama, (2022)

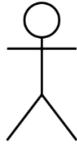
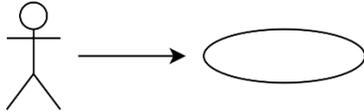
Berdasarkan pengertian di atas, *UML* adalah bahasa grafis standar untuk memvisualisasikan, merancang, membangun, dan mendokumentasikan sistem perangkat lunak berorientasi objek. *UML* mengintegrasikan metode *Booch*, *Object Modeling Technique*, dan *Object-Oriented Software Engineering*, sehingga menjadi alat pemodelan yang andal dalam pengembangan sistem.

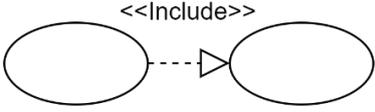
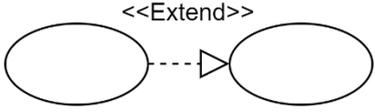
11. Use Case Diagram

Menurut Rohmanto & Setiawan, (2022) *use case diagram* adalah diagram yang menggambarkan fungsi sistem dari perspektif pengguna atau aktor di luar sistem. Menurut Ashari et al., (2023) *use case diagram* memodelkan perilaku sistem dengan menunjukkan interaksi antara aktor dan sistem, sementara detail

prosesnya dapat digambarkan lebih rinci melalui *activity diagram*.. Pendapat lainnya dari Aliman, (2021) *use case diagram* adalah langkah awal untuk memodelkan kebutuhan fungsional sistem, merangkum skenario aktor dalam batasan sistem dengan elemen yang terhubung melalui garis notasi. Dikutip dari Saputra & Fahrizal, (2021), berikut pengertian mengenai simbol dari *use case diagram* :

Table 2.2 Simbol *use case diagram*

Simbol	Keterangan
	<p><i>Use case</i> : menggambarkan bagaimana seseorang akan menggunakan/memanfaatkan sistem.</p>
	<p>Aktor : seseorang/sesuatu yang berinteraksi dengan sistem yang sedang kita kembangkan.</p>
	<p>Relasi : sebagai penghubung antara aktor-<i>use case</i>, <i>use case-use case</i>, dan lain-lain.</p>
	<p>Relasi asosiasi : relasi terjadi antara aktor dengan <i>use case</i> biasanya berupa garis lurus dengan kepala panah disalah satu ujungnya.</p>

	<p><i>Include relationship</i> (relasi cakupan) : memungkinkan suatu <i>use case</i> untuk menggunakan fungsionalitas yang disediakan oleh <i>use case</i> yang lainnya.</p>
	<p><i>Extend relationship</i> : memungkinkan <i>use case</i> memiliki kemungkinan untuk memperluas fungsionalitas yang disediakan oleh <i>use case</i> yang lainnya.</p>

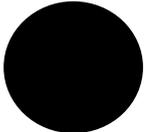
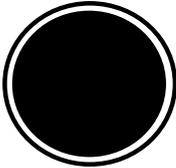
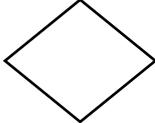
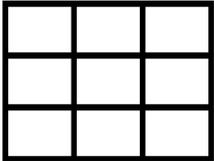
Berdasarkan pengertian di atas, *use case diagram* adalah diagram yang memodelkan fungsi sistem dari perspektif pengguna atau aktor di luar sistem. Diagram ini menggambarkan interaksi antara aktor dan sistem untuk memvisualisasikan kebutuhan fungsional sistem.

12. Activity Diagram

Menurut Musthofa & Adiguna, (2022) *activity diagram*, atau diagram aktivitas, adalah diagram yang memodelkan urutan proses dalam suatu sistem secara vertikal. Diagram ini termasuk dalam *UML* dan digunakan untuk mendukung pembuatan *use case*. Pendapat lainnya dari Arianti et al., (2022) *activity diagram* adalah diagram yang memvisualisasikan alur data atau kontrol, serta aksi terstruktur yang dirancang secara sistematis dalam sebuah sistem. Menurut penelitian Fajarwati et al., (2023) Activity diagram lebih berfokus pada eksekusi dan alur sistem daripada pada cara sistem dibangun. Diagram ini tidak hanya digunakan untuk memodelkan perangkat lunak, tetapi juga untuk memodelkan proses bisnis. Diagram ini menggambarkan sistem sebagai kumpulan

aksi-aksi. Dikutip dari Saputra & Fahrizal, (2021), berikut mengenai pengertian dan simbol *activity diagram* :

Table 2.3 Simbol *activity diagram*

Simbol	Keterangan
	Lingkaran penuh. Memulai proses dalam suatu diagram aktivitas.
	Mata banteng. Akhir dari proses.
	Segi empat panjang, kejadian aktivitas, atau pemicu.
	Berlian, sebuah cabang.
	Table. Suatu <i>file</i> komputer dari mana data bisa dibaca atau direkam selama kejadian bisnis.
	Dokumen. Menunjukkan dokumen sumber atau laporan.
	Garis putus-putus. Alur informasi antar kejadian.
	Garis tidak putus-putus. Urutan dari suatu kejadian atau aktivitas.

	<p>Catatan. Memberikan acuan bagi pembaca pada diagram atau dokumen lain untuk memperincinya.</p>
---	---

Berdasarkan pengertian di atas, *activity diagram* adalah diagram *UML* yang memodelkan urutan proses dalam suatu sistem, berfokus pada alur data, kontrol, dan aksi terstruktur. Selain mendukung pembuatan *use case*, diagram ini digunakan untuk memvisualisasikan eksekusi sistem, baik dalam perangkat lunak maupun proses bisnis.

13. *Beta Testing*

Menurut Mujiyanto et al., (2022) *beta testing* adalah tahap di mana penulis melibatkan pihak eksternal untuk menguji apakah game berfungsi dengan baik atau masih terdapat *bug*. Pada tahap ini, pengujian dilakukan langsung oleh pengguna dengan bantuan kuesioner, yang bertujuan untuk mengevaluasi apakah sistem yang dikembangkan telah memenuhi kebutuhan pengguna. Menurut pendapat Riyandana et al., (2022) *beta* adalah tahap pengujian eksternal yang melibatkan pihak ketiga. Hasilnya berupa laporan *bug* dan masukan pengguna. Sesi *beta* berakhir ketika fase selesai atau semua penguji telah memberikan laporan. Berdasarkan hasil ini, produk dapat ditingkatkan lebih lanjut atau diluncurkan jika hasilnya memuaskan. Menurut Mustofa, (2021) *beta testing* dibagi menjadi dua jenis, yaitu *beta testing* terbuka dan *beta testing* tertutup. *Beta testing* tertutup hanya melibatkan peserta yang diundang secara khusus, sementara *beta testing* terbuka terbuka bagi siapa saja yang mendaftar. Kualitas versi *beta* sangat bergantung pada tingkat pengembangan prototipe saat ini.

Berdasarkan pengertian di atas, *beta testing* adalah tahap pengujian eksternal yang melibatkan pengguna untuk mengidentifikasi *bug* dan mengevaluasi apakah sistem telah memenuhi kebutuhan. Pengujian ini terbagi menjadi dua jenis, beta tertutup yang melibatkan peserta undangan, dan beta terbuka yang dapat diikuti oleh siapa saja. Hasil pengujian berupa laporan bug dan masukan pengguna, yang digunakan untuk memperbaiki atau meluncurkan produk jika sudah memenuhi standar.

B. Kajian Empiris

Berbagai penelitian terdahulu yang membahas terkait *game RPG* telah dilakukan. Penelitian tentang *game RPG* edukasi mengenai kisah pahlawan Indonesia bernama Tuanku Imam Bonjol karya Luluhan et al., (2022). Penelitian ini menggunakan genre *RPG* yang mengisahkan sejarah, awal mula, dan perjalanan seorang pahlawan nasional bernama Tuanku Imam Bonjol. Penelitian ini menggunakan *game engine RPG Maker MV* dalam pembuatannya.

Penelitian lainnya dari Nursadewa, (2022) yang mengangkat kisah dari Reog Ponorogo yang digunakan sebagai *game* edukasi. Dalam penelitian ini, penulis membahas bagaimana cara membuat kisah Reog Ponorogo yang akan dijadikan sebuah *game* edukasi dengan menggunakan *game engine RPG Maker MV*. Hasil dari penelitian ini menyatakan bahwa penulis berhasil menciptakan *game* tersebut dengan berbasis android yang digunakan sebagai sarana pengenalan budaya Reog Ponorogo.

Penelitian berikutnya yaitu dari Tresnawati & Setyawan, (2021). Penelitian ini berfokus pada perancangan dan pengembangan *game RPG* bertema cerita rakyat. *Game* yang dibuat mengambil kisah Sangkuriang dan dirancang dalam bentuk aplikasi android. Metode yang diterapkan dalam penelitian ini adalah *Digital Based Game*

Learning-Instructional Design, dengan proses pengujian dilakukan melalui alpha testing.

Penelitian terdahulu selanjutnya adalah dari Kuswanto et al., (2021). Penelitian tersebut mengambil tema rakyat dengan judul Roro Jonggrang. Cerita melegenda dalam kalangan masyarakat yang dibungkus kedalam media modern berbentuk *role playing game*. Penelitian ini menggunakan *RPG Maker MV* sebagai *game engine*-nya. Hasil penelitian ini mendapatkan persentase positif sebanyak 82% dari responden yang menyukai hasil penelitian ini karena memberikan dampak positif.

Penelitian terdahulu berikutnya dari Latifah et al., (2022) yang membuat *game* cerita rakyat berjudul Asal Usul Pulomas yang berasal dari Jawa Barat. Penulis mengambil tema tersebut untuk membuat *game* bergenre *Role Playing Game* yang dapat mengenalkan cerita rakyat khususnya pada Asal Usul Pulomas. Hasil penelitiannya sebuah *game* yang mengadopsi cerita rakyat yang menggunakan *game engine RPG Maker MV*, penelitian ini mendapatkan respon positif dari masyarakat karena inovasinya.

Berdasarkan penelitian yang telah dilakukan dan dijelaskan di atas, kita dapat mengetahui kesamaan dan perbedaan penelitian yang telah dilakukan. Untuk persamaannya adalah penelitian terdahulu menggunakan *RPG Maker* sebagai *game engine* atau alat pembuatannya. Setiap penelitian mendapatkan hasil yang memuaskan, dari hasil tersebut bisa kita simpulkan bahwa *game* merupakan media yang multifungsi karena bisa kita gunakan sebagai media hiburan dan edukasi. Dan pada perbedaannya adalah pada penelitian terdahulu terdapat berbagai macam konsep dan tema yang digunakan, baik itu media pembelajaran, media edukasi, maupun sekedar media hiburan. Dari hal tersebut bisa disimpulkan bahwa *game* bisa diimplementasikan sebagai bentuk media.

C. Kerangka Berfikir

Game RPG “Petualangan Hanoman : Kobaran Api Perang” akan menggunakan konsep gabungan antara cerita rakyat dan teknologi modern dalam bentuk *game RPG*. Metode pengembangan yang digunakan dalam pembuatan *game RPG* “Petualangan Hanoman : Kobaran Api Perang” adalah *waterfall*. *Waterfall* sendiri adalah metode pengembangan sistem yang berfokus pada alur yang terurut, terdiri dari analisis kebutuhan, perancangan, implementasi, pengujian, dan perilisan sekaligus pemeliharaan. Untuk perancangan penelitian ini akan menggunakan *flowchart* dan *UML*, dimana nanti di dalam *UML*-nya terdapat *use case* dan *activity diagram*.

Dalam membangun *game RPG* “Petualangan Hanoman : Kobaran Api Perang”, penulis akan membangunnya menggunakan *game engine RPG Maker MV*. *Game engine* tersebut dipilih karena pembuatan *game* bergenre *RPG* dengan tampilan 2D di *game engine RPG Maker MV* memiliki fitur-fitur yang memadai.

Untuk rincian kerangka penelitian dapat dilihat pada gambar berikut :

Table 2.4 Kerangka berfikir

