

BAB 1

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

1.1 Pendahuluan

Pemrograman merupakan kemampuan yang memiliki peran penting bagi mahasiswa informatika maupun bidang studi semacamnya. Mata kuliah ini menjadi prasyarat utama yang harus dilalui mahasiswa sebelum melanjutkan ke mata kuliah berikutnya yang berkaitan dengan pemrograman. Melalui pembelajaran tersebut, mahasiswa dibimbing untuk mengembangkan kemampuan berpikir logis, menyelesaikan permasalahan, serta berinteraksi dengan komputer secara efektif (Prasetya et al., 2025). Penguasaan logika pemrograman merupakan dasar penting dalam pembentukan kemampuan berpikir komputasional mahasiswa di bidang Informatika. Di kehidupan nyata, banyak mahasiswa baru belum memiliki pengetahuan yang memadai untuk memahami konsep logika pemrograman secara efektif. Kondisi ini dapat menimbulkan hambatan dalam proses pembelajaran awal dan berpotensi mengurangi motivasi belajar apabila tidak didukung oleh media pembelajaran yang sesuai dengan karakteristik dan kebutuhan mahasiswa pemula (Maryono, 2016; Sophan & Kurniawati, 2018). Sebagai inovasi ada pengembangan gim edukasi yang bernama CodeLogicCraft 1.0. Gim tersebut berisi berbagai elemen permainan interaktif. Gim edukasi yang dikembangkan dengan beberapa fitur utama, seperti sistem darah, sertifikat pencapaian, dan alur cerita yang saling terintegrasi. Sistem darah berfungsi sebagai indikator ketahanan karakter, di mana setiap kesalahan pemain akan mengurangi jumlah darah yang dimiliki. Pada akhir permainan pemain akan mendapatkan sertifikat pencapaian. Hal tersebut sebagai bentuk penghargaan bagi pemain yang berhasil menyelesaikan seluruh level permainan. Sementara itu, alur cerita berperan sebagai pengikat antarlevel, memberikan konteks dan kesinambungan terhadap setiap tantangan yang dihadapi pemain selama permainan berlangsung. Permainan terdiri dari tiga tingkat kesulitan, yaitu *Beginner*, *Intermediate*, dan *Expert*, yang masing-masing terbagi menjadi tiga sublevel dengan tantangan yang meningkat secara bertahap. Level *Beginner* memperkenalkan mekanisme dasar permainan, *Intermediate* menggabungkan

elemen dasar dengan pola perulangan, sedangkan *Expert* menyatukan seluruh elemen sebelumnya dalam tantangan yang lebih kompleks.

Melalui struktur yang terarah dan sistem interaktifnya, CodeLogicCraft 1.0 menghadirkan pengalaman bermain yang menarik sekaligus menantang bagi para pemain. Sebagai tindak lanjut dari permasalahan tersebut, dilakukan pengembangan CodeLogicCraft 2.0 dengan fitur dan struktur permainan yang berbeda dari versi sebelumnya. Pada versi ini, ditambahkan fitur animasi alur cerita yang menampilkan urutan peristiwa di setiap tahap permainan secara visual, sehingga pemain dapat mengikuti jalan cerita dengan lebih jelas. CodeLogicCraft 2.0 terdiri atas empat level utama, yaitu Pergerakan-Dasar, Perulangan-Kode Berulang, Percabangan-Pilihan Jalan, dan Method atau Fungsi-Panggil Bantuan. Masing-masing level mekanisme tantangan yang berbeda sesuai dengan fokus level tersebut. Setiap level juga terbagi menjadi lima sublevel, yang menjadikan jumlah tantangan dalam CodeLogicCraft 2.0 lebih banyak dibandingkan versi sebelumnya. Selain itu, gim ini dilengkapi dengan sistem bintang sebagai bentuk penilaian terhadap hasil permainan. Pemain akan memperoleh jumlah bintang tertentu berdasarkan tingkat keberhasilan dalam menyelesaikan setiap sublevel. Dengan penambahan fitur animasi alur cerita, empat level utama, lima sublevel di setiap level, serta sistem bintang, CodeLogicCraft 2.0 menghadirkan susunan permainan yang lebih lengkap dan terstruktur dibandingkan versi sebelumnya. Sebagai pelengkap dari pengembangan tersebut, gim ini juga memiliki struktur yang kuat dalam penyusunan elemen permainan edukasi. Setiap komponennya dikembangkan untuk membentuk alur permainan yang konsisten, seimbang, dan mendukung peningkatan kualitas pengalaman bermain. Isi dari CodeLogicCraft 2.0 mencakup tiga konsep utama dalam logika pemrograman, yaitu percabangan(*conditional*), perulangan(*looping*), dan metode atau fungsi (*method/function*). Gim ini dikembangkan dalam bentuk dunia tiga dimensi bertema petualangan di pulau terpencil, di mana pemain berperan sebagai penjelajah yang harus memecahkan berbagai tantangan logika untuk bertahan hidup. Terdapat empat level utama dalam permainan, Level Pergerakan Dasar berfokus pada pengenalan urutan instruksi dan

kontrol karakter untuk mencapai tujuan tertentu. Level Perulangan mengajarkan efisiensi dalam penggunaan perintah melalui penerapan pola berulang. Level Percabangan menuntut pemain mengambil keputusan berdasarkan kondisi tertentu sesuai dengan logika percabangan. Level Metode atau Fungsi menekankan pentingnya penyusunan perintah secara modular dengan membuat dan memanggil fungsi untuk menyelesaikan tantangan yang lebih kompleks. Sebelum memulai permainan utama, pemain akan melalui tutorial interaktif untuk memahami mekanisme permainan dan alur logika yang digunakan pada setiap misi. Setelah itu, pemain dapat memilih salah satu dari tiga karakter yang tersedia, masing-masing memiliki gaya visual dan animasi yang berbeda namun tidak memengaruhi kemampuan permainan, sehingga pemain dapat menyesuaikan pilihan sesuai preferensi mereka. Setiap level terdiri dari lima sublevel dengan tingkat kesulitan yang meningkat secara bertahap dan dilengkapi animasi alur cerita yang menggambarkan perkembangan perjalanan karakter. Gim ini juga memiliki sistem bintang untuk menilai performa pemain yang dimana setiap sublevel diberikan maksimal 3 bintang. Pemain yang berhasil meraih 60 bintang penuh (60/60) akan memperoleh sertifikat “Master Logika”, sedangkan pemain dengan total bintang di bawah 60 akan mendapatkan sertifikat “Petarung Algoritma” sebagai bentuk apresiasi untuk mereka yang berhasil menyelesaikan semua level. Setiap misi dalam gim dikembangkan untuk merepresentasikan penerapan konsep tersebut dalam bentuk tantangan pemecahan masalah yang disajikan secara bertahap.

Berbagai penelitian media pembelajaran interaktif telah dilakukan untuk mendukung proses belajar yang lebih efektif dan menarik. Meski telah memberikan kontribusi positif, upaya tersebut masih perlu dikembangkan lebih lanjut agar dapat menciptakan pengalaman belajar yang lebih bermakna bagi seluruh pihak yang terlibat dalam proses pembelajaran. Penelitian oleh Adipat et al. (2021) yang berjudul “*Engaging Students in the Learning Process with Game-Based Learning: The Fundamental Concepts*” menyoroti pentingnya penerapan *game-based learning* (GBL) sebagai pendekatan inovatif dalam meningkatkan keterlibatan dan motivasi mahasiswa selama proses pembelajaran. Studi ini menjelaskan bahwa

GBL berperan dalam mengembangkan kemampuan berpikir kritis, pemecahan masalah, serta kolaborasi antar mahasiswa melalui lingkungan belajar yang interaktif dan kontekstual. Selain itu, penelitian ini juga menegaskan bahwa elemen-elemen permainan seperti tantangan (*challenge*), rasa ingin tahu (*curiosity*), dan kontrol (*control*) mampu menstimulasi keterlibatan kognitif maupun emosional mahasiswa, sehingga menciptakan pengalaman belajar yang lebih menarik dan bermakna. Sebagai implikasinya, penerapan GBL dinilai efektif dalam menumbuhkan *growth mindset*, karena memungkinkan mahasiswa untuk belajar dari kesalahan dan terus berusaha hingga mencapai keberhasilan. Dengan demikian, penelitian ini menegaskan bahwa GBL tidak hanya berfungsi sebagai sarana hiburan, tetapi juga sebagai media pembelajaran yang mendorong partisipasi aktif, kreativitas, serta refleksi terhadap konsep yang dipelajari.

Penelitian juga dilakukan “*Kesulitan Mahasiswa dalam Memahami Konsep Dasar Pemrograman*” oleh Cahya Karima dkk. (2025), diketahui bahwa banyak mahasiswa mengalami hambatan dalam memahami logika pemrograman, sintaksis, serta struktur data akibat pendekatan pembelajaran yang masih bersifat teoritis dan kurangnya kesempatan praktik langsung. Hasil penelitian tersebut menunjukkan perlunya media pembelajaran yang mampu menjembatani kesenjangan antara teori dan praktik agar mahasiswa dapat memahami konsep dasar pemrograman secara lebih efektif. Kondisi ini menjadi dasar relevansi pengembangan gim edukasi CodeLogicCraft 2.0, yang dirancang untuk memberikan pengalaman belajar interaktif melalui penerapan konsep logika pemrograman dalam bentuk tantangan dan skenario permainan. Dengan struktur level yang terarah, sistem bintang sebagai evaluasi kinerja, serta penyajian animasi alur cerita yang memperjelas konteks tantangan, CodeLogicCraft 2.0 berupaya menghadirkan sarana pembelajaran yang lebih menarik dan aplikatif. Melalui pendekatan tersebut, gim ini diharapkan dapat membantu mahasiswa memahami konsep pemrograman secara bertahap dan kontekstual, sekaligus menjadi solusi inovatif terhadap permasalahan kesulitan belajar pemrograman yang telah diidentifikasi dalam penelitian tersebut.

Berdasarkan hasil studi literatur yang berjudul “Strategi Pembelajaran untuk Meningkatkan Keterampilan Pemrograman dan Berpikir Komputasi” oleh Anistyasari, Ekohariadi, dan Munoto (2020), penelitian ini menekankan pentingnya berpikir komputasi sebagai kemampuan dasar yang perlu dimiliki oleh mahasiswa di era digital. Studi tersebut mengungkap bahwa berpikir komputasi bukan hanya keterampilan teknis dalam pemrograman, tetapi juga merupakan cara berpikir sistematis dalam memecahkan masalah melalui proses dekomposisi, abstraksi, dan perancangan algoritma. Hasil telaah menunjukkan bahwa masih terdapat kesenjangan antara pemahaman konseptual mahasiswa dan kemampuan mereka dalam menerapkan logika pemrograman ke dalam praktik. Banyak mahasiswa mengalami kesulitan dalam mentransfer konsep logika ke dalam sintaksis bahasa pemrograman yang konkret, sehingga berdampak pada rendahnya kemampuan debugging dan pengembangan solusi algoritmik yang efisien.

Selain itu, penelitian ini menyoroti bahwa sebagian besar pendekatan pembelajaran yang efektif dalam meningkatkan keterampilan berpikir komputasi dan pemrograman melibatkan model *Project-Based Learning*, *Problem-Based Learning*, *Collaborative Learning*, dan *Game-Based Learning*. Pendekatan-pendekatan ini terbukti mampu meningkatkan keterlibatan aktif mahasiswa, mendorong kolaborasi, serta mengasah kemampuan berpikir kritis dan kreatif dalam merancang solusi komputasional. Namun demikian, studi ini juga menekankan bahwa penerapan strategi pembelajaran tersebut masih menghadapi tantangan, terutama dalam konteks pendidikan tinggi di Indonesia, di mana pendekatan pengajaran masih dominan bersifat teoritis dan belum sepenuhnya memanfaatkan media interaktif seperti simulasi, animasi, atau gim edukatif.

Dengan demikian, artikel ini menegaskan perlunya transformasi pedagogis dalam pembelajaran pemrograman dasar—dari yang berorientasi pada transfer pengetahuan menjadi berorientasi pada eksplorasi dan praktik langsung. Integrasi media interaktif dan pendekatan berbasis proyek dinilai sebagai langkah strategis untuk menumbuhkan kemampuan berpikir komputasi mahasiswa secara lebih mendalam dan aplikatif dalam konteks dunia nyata.

Penelitian ini juga didasarkan pada temuan studi literatur “Pemahaman Mahasiswa tentang Logika Informatika dalam Pemrograman” (Juliantara dkk., 2025). Studi tersebut menunjukkan bahwa sebagian besar mahasiswa mengalami kesulitan dalam memahami konsep logika dasar, meliputi proposisi, inferensi, logika predikat, dan aljabar Boolean yang menjadi fondasi dalam pemrograman. Hasil penelitian mengungkap bahwa lebih dari 70% mahasiswa tidak mampu mengaitkan konsep logika dengan penerapannya pada struktur kontrol program seperti *if-else* dan *looping*, sehingga sering terjadi kesalahan berulang dalam proses penulisan kode.

Selain itu, ditemukan bahwa kemampuan *computational thinking* mahasiswa masih tergolong rendah, khususnya dalam aspek dekomposisi masalah, perancangan algoritma, dan generalisasi solusi. Kondisi tersebut diperparah oleh pendekatan pembelajaran yang masih berorientasi pada teori tanpa dukungan media visual atau interaktif yang dapat menjembatani pemahaman antara konsep logika dan implementasi praktisnya. Temuan ini menegaskan perlunya inovasi pembelajaran berbasis media interaktif yang mampu meningkatkan kemampuan berpikir logis dan sistematis mahasiswa dalam memecahkan masalah pemrograman.

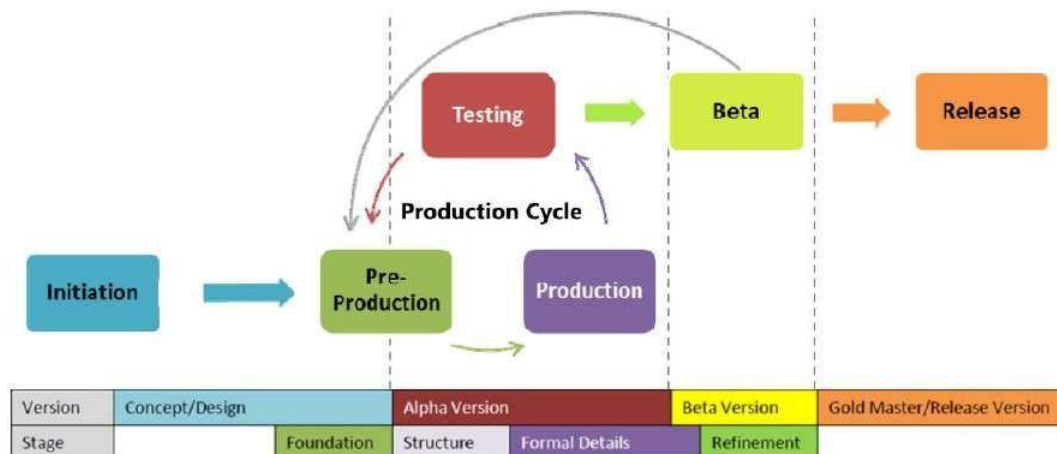
Berdasarkan penelitian “*Analisis Motivasi Belajar Pemrograman Mahasiswa JTIK Berbasis UTAUT Terhadap Model Gamification*” oleh Hidayat dkk. (2023), studi ini menegaskan bahwa gamifikasi memiliki pengaruh positif terhadap motivasi dan pemahaman mahasiswa dalam mempelajari pemrograman. Melalui pendekatan UTAUT, penelitian ini menunjukkan bahwa mahasiswa merasakan gamifikasi sebagai metode yang lebih mudah, menyenangkan, dan membantu mereka memahami alur logika pemrograman secara lebih konkret. Mayoritas responden juga mengakui bahwa game edukasi memberikan pengalaman belajar yang lebih santai namun efektif dibandingkan metode tradisional.

Meski sebagian kecil mahasiswa masih merasa gamifikasi cukup rumit akibat kurangnya pembiasaan, temuan penelitian ini secara umum menunjukkan bahwa pendekatan berbasis game merupakan strategi pembelajaran yang potensial untuk

meningkatkan fokus, keterlibatan, dan pemahaman konsep dasar pemrograman. Dengan demikian, artikel ini menegaskan pentingnya integrasi media interaktif dalam pembelajaran pemrograman sebagai upaya mengatasi kejenuhan dan meningkatkan motivasi belajar.

1.2 Siklus Hidup Pengembangan Aplikasi

Dalam pengembangan aplikasi dengan judul CodeLogicCraft 2.0 ini, tim kami melakukan pengembangan menggunakan siklus hidup pengembangan perangkat lunak **GDLC** dengan tahapan, i.e: *initiation*, *pre-production*, *production*, *testing*, dan *release*. Pada bab ini akan diuraikan khusus tahapan awal pengembangan yaitu *initiation* perangkat lunak/aplikasi yaitu membuat spesifikasi kebutuhan perangkat lunak.



Gambar 1 Model GDLC

1.2.1 Initiation

Pada tahap ini, kami menentukan eksplorasi ide awal untuk mengembangkan menjadi sebuah game edukatif. Dengan tujuan utama yang meliputi:

- Tujuan : Menentukan ide dasar dari game.
- Kegiatan : Merancang ide awal, fitur berdasarkan observasi, diskusi kelayakan fitur dengan dosen, dan referensi gim sejenis

1.2.1.1 Ide Awal

Pada tahap ini tim merancang ide awal untuk mengembangkan gim edukatif yang dapat menciptakan pengalaman belajar yang lebih menarik dan interaktif, sekaligus mendorong peningkatan minat pelajar tingkat menengah dan mahasiswa terhadap bidang pemrograman.

1.2.1.2 Analisis kebutuhan

Analisis kebutuhan merupakan langkah untuk mengidentifikasi, merumuskan, dan mendokumentasikan berbagai kebutuhan serta harapan pemain terhadap suatu produk. Proses ini dilakukan dengan menghimpun data dari berbagai sumber guna memahami tujuan yang ingin dicapai pemain, permasalahan yang mereka hadapi, serta ekspektasi terhadap fungsionalitas, pengalaman bermain, dan hasil yang diinginkan dari produk.

1.2.1.3 Target Pemain

Target pemain merupakan kelompok sasaran yang ingin dijangkau oleh suatu produk atau layanan. Pada tahap ini tim menentukan target pemain dan menentukan fitur serta fungsi yang relevan, merancang pengalaman pemain yang sesuai dengan preferensi mereka, serta menyusun strategi pemasaran yang tepat sasaran agar pesan dan kampanye yang disampaikan dapat menarik minat dan memenuhi kebutuhan kelompok tersebut.

1.2.2 Pre-Production

Pada tahap ini sangat penting dalam pengembangan gim yang berfungsi untuk menyusun perencanaan secara rinci sebelum proses produksi dimulai. Semua konsep awal yang telah dirumuskan pada tahap initiation dikembangkan lebih dalam menjadi dokumen teknis, desain visual, serta struktur logika permainan. Tujuan dalam tahap ini adalah:

- Tujuan : Menyusun rancangan lengkap dan terstruktur dalam pengembangan gim
- Kegiatan :
 - Menulis *Game Design Document* (GDD)
 - Menentukan teknologi dan alat yang akan digunakan

- Merancang sistem *gameplay* dan mekanik dalam gim

1.2.2.1 Identifikasi Kebutuhan Klien

Identifikasi kebutuhan klien merupakan tahap awal yang krusial dalam proses *pre-production*, karena berfungsi sebagai dasar dalam perancangan dan pengembangan gim. Tujuannya adalah untuk memahami secara menyeluruh apa yang diharapkan oleh klien atau pengguna akhir dari produk gim yang akan dikembangkan.

Proses ini dilakukan melalui beberapa metode, antara lain wawancara dan diskusi dengan klien atau calon pengguna untuk menggali kebutuhan dan ekspektasi mereka, serta observasi langsung terhadap aktivitas atau permasalahan yang relevan dengan konteks gim yang akan dikembangkan.

1.2.2.2 Desain Umum Gim

Setelah kebutuhan pengguna dipahami, tim menentukan elemen desain umum gim yang mencakup genre, platform, mekanisme permainan, dll. Tujuan dari bagian ini untuk memberikan gambaran secara menyeluruh mengenai bagaimana gim akan berfungsi dan terasa ketika gim dimainkan oleh pengguna.

Desain umum berfungsi dari konsep awal gim yang secara abstrak dengan implementasi teknis di tahap produksi. Aspek utama dalam desain umum gim yaitu:

1. Tujuan pengembangan gim

Tujuan pengembangan gim merupakan aspek penting, karena menjadi landasan utama dalam menentukan arah dan fokus gim yang akan dikembangkan.

2. Genre

Pada aspek ini menentukan genre yang relevan untuk gim, guna menambah estetika yang membentuk pengalaman bermain.

3. Jumlah pemain

Jumlah pemain dalam gim memiliki kaitan yang erat dengan sasaran pembelajaran yang hendak dicapai. Gim yang dirancang untuk satu pemain umumnya berfokus pada pendalaman konsep dan pengulangan materi secara individual, sedangkan gim

yang melibatkan banyak pemain cenderung menumbuhkan kemampuan bekerja sama, berkomunikasi, serta menyelesaikan masalah secara kolektif.

4. Deskripsi target pemain

Target pemain adalah kelompok mana yang menjadi sasaran utama dalam perancangan dan pengembangan gim. Mendeskripsikan target pemain dimulai dari aspek usia, latar belakang pendidikan, minat dan kebutuhan yang sesuai agar dapat menentukan kategori target pemain.

5. Kategori target pemain

Pada tahap ini, tim menentukan target pemain berdasarkan deskripsi target pemain untuk menyesuaikan karakteristik tertentu yang relevan dengan desain dan tujuan utama gim. Pengelompokan ini penting guna memastikan bahwa seluruh elemen dalam gim mulai dari *gameplay*, tampilan visual, hingga tingkat kesulitan disesuaikan dengan kebutuhan dan kemampuan pemain yang dituju.

6. Jenis pengalaman bagi pemain

Jenis pengalaman bagi pemain merujuk pada kesan, emosi, dan interaksi yang akan diciptakan dan dirasakan pemain saat memainkan gim ini. Tujuannya untuk memastikan bahwa tujuan utama gim baik dari segi edukatif, hiburan, atau eksplorasi tersampaikan dengan baik melalui pengalaman yang menyenangkan.

7. Platform distribusi

Platform distribusi adalah media yang digunakan untuk menyebarkan dan mengakses gim oleh pemain. Pemilihan platform distribusi ini penting karena akan mempengaruhi jangkauan pengguna, kompatibilitas perangkat, serta strategi penyebaran gim ke target pemain.

8. Bahasa pengantar

Bahasa pengantar adalah bahasa utama yang akan digunakan dalam gim guna mempertahankan komunikasi yang baik. Bahasa pengantar akan digunakan di dalam gim seperti teks naratif, instruksi, dan antarmuka pengguna karena akan mempengaruhi kemudahan pemahaman, kenyamanan pemain, serta efektivitas penyampaian informasi dan pembelajaran.

9. Gaya gambar (*Art Style*)

Gaya gambar adalah pendekatan visual yang digunakan untuk membentuk tampilan dunia permainan, karakter, latar, antarmuka, dan elemen-elemen visual lainnya. Pemilihan gaya gambar yang tepat akan membantu menciptakan suasana, identitas visual, dan pengalaman yang menarik bagi pemain.

10. Sudut pandang (*Point of View*)

Sudut pandang dalam gim merujuk pada cara pemain melihat dunia permainan, baik dari segi visual maupun interaksi yang dilakukan pemain. Sudut pandang mempengaruhi terhadap pengalaman bermain, persepsi ruang, dan kedekatan pemain terhadap karakter atau lingkungan dalam gim.

11. Ringkasan alur gim

Ringkasan umum adalah gambaran pemain mengenai jalannya permainan dari awal hingga akhir, termasuk bagaimana pemain memainkan gim, tantangan yang harus diselesaikan, dan bagaimana permainan berakhir. Alur gim diciptakan untuk memastikan pemain mengalami pengalaman permainan yang terstruktur, logis, dan progresif, terutama dalam gim edukatif.

12. Interaksi

Interaksi dalam gim mengacu bagaimana cara pemain merespon dan berkomunikasi pada lingkungan permainan, termasuk bagaimana pemain memberikan input dan gim memberikan umpan balik. Interaksi yang baik akan menciptakan suasana pengalaman bermain yang lebih imersif, menyenangkan, dan bermakna.

13. Penggunaan audio

Penggunaan audio sangat penting berfungsi untuk mendukung suasana, memberikan umpan balik, serta meningkatkan pengalaman bermain. Dalam konteks gim ini audio juga membantu memperjelas tindakan dan hasil dari interaksi yang dilakukan pemain.

14. Tema cerita

Tema cerita adalah ide pokok atau garis besar narasi yang menjadi dasar dunia, latar belakang, dan motivasi dalam gim. Tema ini membantu menciptakan konteks dan suasana yang mendukung tujuan permainan, serta memberikan motivasi naratif bagi pemain untuk menyelesaikan tantangan yang diberikan.

15. Prolog

Prolog adalah bagian pembuka dalam sebuah gim yang berfungsi untuk memperkenalkan latar belakang cerita, kondisi karakter, dan motif utama permainan. Dalam gim ini, prolog ditampilkan dalam bentuk cutscene dan teks naratif.

16. Ending

Ending adalah bagian penutup dalam sebuah gim yang menandai cerita atau permainan sudah selesai, dalam gim ini ending juga disajikan dalam bentuk cutscene dan teks naratif.

1.2.2.3 Lingkaran Permainan (Core Loop)

Pada tahap ini lingkaran permainan (*Core Loop*) adalah rangkaian aktivitas utama yang terus dilakukan pemain sepanjang permainan. *Core loop* mewakili struktur dasar dari alur gameplay yang berulang dan menjadi inti dari pengalaman bermain.

1.2.2.4 Dunia, Cerita, dan Karakter

Dunia, cerita, dan karakter dirancang untuk menciptakan suasana bermain yang lebih nyata dan membangun keterlibatan emosi pemain. Ketiganya sangat penting dalam memberikan konteks, motivasi, dan arah tujuan didalam permainan.

1. Dunia

Dalam *Game Design Document* (GDD), dunia dalam gim berperan sebagai ruang naratif, sarana pembelajaran, sekaligus area eksplorasi bagi pemain. Di dalamnya, pemain dapat menjalankan aksi, berinteraksi dengan elemen-elemen gim, menyelesaikan tantangan, dan merasakan dampak dari setiap keputusan yang dibuat. Dunia ini dapat diwujudkan dalam bentuk lingkungan yang realistis seperti kelas virtual atau hutan digital, maupun dalam bentuk imajinatif seperti planet

futuristik atau dunia fantasi (rincian deskripsi dunia dapat dilihat pada sub-bab 2.2.5.1).

2. Alur Cerita

Cerita di dalam gim ini dirancang sebagai kerangka naratif yang mendukung pembelajaran logika pemrograman melalui situasi bertahan hidup. Cerita berfungsi untuk memberikan konteks dan arah bagi pemain untuk menyelesaikan tantangan yang disajikan, sehingga pemain merasa relevan dan rasa ingin tahu yang lebih untuk mengetahui dari jalan akhir cerita. Alur cerita berkembang sejalan dengan progres yang telah pemain lalui, di mana setiap zona atau *level* menghadirkan tantangan baru yang berhubungan dengan konsep logika tertentu.

Cerita disusun secara bertahap untuk memperkuat motivasi mereka untuk terus bermain. Tujuan utama dari cerita adalah memberi alasan mengapa pemain harus menyelesaikan setiap tantangan dan bagaimana setiap keberhasilan membawa mereka ke tujuan akhir. Dengan ini cerita bukan hanya sekedar hiasan tetapi menjadi bagian dari pengalaman belajar dalam bermain game (alur cerita dapat dilihat pada sub-bab 2.2.5.2).

3. Karakter

Karakter dirancang untuk merepresentasikan pemain sebagai individu yang sedang belajar dan berkembang. Tokoh utama merupakan seorang pemula yang belum memahami logika pemrograman, namun secara bertahap mengalami perkembangan melalui tantangan yang dihadapi dalam permainan. Karakter mencerminkan proses belajar pemain, dari tidak tahu apa-apa lalu mengalami perkembangan (karakter dapat dilihat pada sub-bab 2.2.5.5 dan 2.2.5.6).

1.2.2.5 Analisis Alur dan Tantangan

Analisis alur dan tantangan dalam gim dilakukan untuk memastikan bahwa setiap tahapan permainan secara logis, terstruktur, dan mampu memberikan pengalaman belajar yang bertahap bagi pemain. Tahap ini sangat penting karena menentukan bagaimana pemain memahami mekanisme permainan, menghadapi rintangan

secara progresif, serta tetap termotivasi untuk menyelesaikan permainan. Dengan alur dan tantangan yang diciptakan secara bertahap, game dapat memberikan keseimbangan antara kesulitan dan kepuasan (analisis alur dan tantangan dapat dilihat pada sub-bab 2.2.4).

1.2.2.6 Daftar Aset

Daftar aset merupakan elemen yang akan digunakan dalam proses pengembangan gim, baik dari segi visual, audio, maupun elemen fungsional lainnya. Aset-aset ini menjadi komponen penting dalam membentuk pengalaman bermain yang menarik. Penyusunan aset bertujuan untuk memudahkan dalam proses manajemen konten selama pengembangan dan mempermudah integrasi ke dalam unity (daftar aset dapat dilihat pada sub-bab 2.2.7).

1.2.2.7 Storyboard

Storyboard adalah representasi visual dari alur permainan yang menggambarkan urutan kejadian atau interaksi penting yang terjadi dalam gim. Storyboard digunakan sebagai alat bantu untuk memvisualisasikan bagaimana pemain akan mengalami game dari awal hingga akhir (storyboard dapat dilihat pada sub-bab 2.3.3).

1.2.2.8 Diagram Alir

Diagram alir adalah representasi visual dari alur logika atau proses interaksi dalam gim, yang menggambarkan bagaimana pemain berinteraksi dengan sistem dari langkah ke langkah selanjutnya. Diagram alir digunakan untuk menjelaskan urutan tindakan, kondisi, keputusan, dan hasil yang terjadi selama permainan berlangsung (diagram alir dapat dilihat pada sub-bab 2.3.1).

1.2.2.9 Kontrol Pemain

Kontrol pemain dalam gim merupakan elemen penting yang memungkinkan pemain menjalankan berbagai tindakan, seperti menggerakkan karakter, melompat, menavigasi menu, menyelesaikan tantangan, atau memberikan perintah tertentu. Bentuk kontrol ini bisa berupa perangkat fisik seperti keyboard, mouse, joystick, layar sentuh, hingga sensor gerak, tergantung pada jenis gim dan media yang digunakan (kontrol pemain dapat dilihat pada sub-bab 2.3.4).

1.2.3 Production

Pada tahap ini, kami mulai mengimplementasikan seluruh rancangan yang telah dituangkan dalam GDD ke dalam bentuk permainan yang dapat dimainkan. Proses *production* mencakup pembuatan aset visual, penyusunan *level*, pemrograman sistem *gameplay*, serta penggabungan seluruh elemen ke dalam unity. Tahap ini inti dari pengembangan teknis, yang dimana sebelumnya bersifat konseptual lalu mulai diimplementasikan secara fungsional dan interaktif. Tujuan utama meliputi:

- Tujuan : Membangun gim sesuai rancangan
- Kegiatan :
 - Mengimplementasikan fitur *gameplay*
 - Pembuatan aset (misalnya karakter, background, tombol, animasi, dll)
 - Desain *level*
 - Mengimplementasikan audio dan musik
 - Integrasi semua elemen

1.2.3.1 Pembuatan aset visual

Pada tahap ini, semua elemen visual yang digunakan dalam gim mulai dibuat. Aset visual ini meliputi dari karakter, tombol antarmuka pengguna, aset pendukung seperti pohon, perahu, dll.

1.2.3.2 Penyusunan level

Setelah aset siap, level permainan disusun berdasarkan tingkat kesulitan secara bertahap. Setiap zona dirancang memperkenalkan satu atau lebih konsep dasar pemrograman, seperti:

- Tingkat kesulitan Dasar-Pergerakan : Pengenalan gerakan dasar (maju, belok kiri, belok kanan, dll).
- Tingkat kesulitan Perulangan-Kode Berulang : Pengenalan logika pemrograman dasar perulangan.
- Tingkat kesulitan Percabangan-Pilih Jalan : Pengenalan logika pemrograman dasar percabangan.

- Tingkat kesulitan Methode-Panggil Bantuan : Pengenalan logika pemrograman dasar *method/function*.

1.2.3.3 Pemrograman gameplay

Pada tahap ini, logika permainan mulai diimplementasikan menggunakan unity dan bahasa pemrograman C#. Sistem kontrol berbasis logika dibuat agar pemain dapat menyusun urutan perintah dan karakter akan menjalankan perintah tersebut secara otomatis.

1.2.4 Testing

Pengujian pada gim, merupakan tahapan krusial dalam proses pengembangan untuk menjamin kualitas, kinerja fungsional, serta pengalaman bermain yang maksimal sebelum gim dipublikasikan. Tahapan ini mencakup pendeteksian kesalahan (bug), evaluasi kesesuaian dengan berbagai perangkat, serta peninjauan elemen gameplay secara menyeluruh.

1.2.5 Beta Testing

Beta testing merupakan tahap akhir dari pengujian produk yang dilakukan oleh pengguna akhir di lingkungan yang nyata sebelum dirilis secara resmi. Tujuan utamanya adalah mengumpulkan masukan dari pengguna terkait performa, kemudahan penggunaan, dan stabilitas produk, serta mengidentifikasi bug atau kendala lain yang mungkin tidak terdeteksi selama proses pengujian awal.

1.2.6 Release

Tahap *release* merupakan fase di mana gim yang telah dikembangkan dan diuji akan dirilis ke publik. Pada bagian ini, gim dikemas dalam *versi final* dan diunggah ke platform distribusi (seperti Play Store, Steam, App Store, atau itch.io).

1.3 Tujuan Penulisan Spesifikasi Kebutuhan Perangkat Lunak

Spesifikasi Kebutuhan Perangkat Lunak (SKPL) bertujuan sebagai acuan teknis bagi pengembang perangkat lunak pada setiap tahap pengembangan untuk memenuhi kebutuhan yang diharapkan oleh pengguna perangkat lunak. Sedangkan dari sisi pengguna perangkat lunak, SKPL bertujuan untuk mencatat harapan dan semua kebutuhan yang diinginkan dari perangkat lunak yang akan dikembangkan

oleh pengembangan perangkat lunak. Dokumen SKPL memuat spesifikasi kebutuhan dari pengembangan gim CodeLogicCraft 2.0.

1.4 Cakupan Produk

CodeLogicCraft 2.0 adalah gim edukatif berbasis permainan (*Game-Based Learning*) yang dirancang untuk menjadi media pembelajaran yang seru, menantang, dan mendidik. Gim ini memberikan tantangan visual dan logika secara bertahap, sehingga pemain dapat mempelajari dasar-dasar pemrograman seperti urutan instruksi, perulangan (*looping*), pengambilan keputusan (*conditional branching*), dan fungsi (*function*) dengan cara yang menyenangkan. Pemain diajak berpikir seolah seperti programmer dengan menyusun strategi, mencoba solusi, dan memperbaiki kesalahan sambil mengasah keterampilan berpikir tingkat tinggi.

Gim CodeLogicCraft 2.0 juga dapat menjadi solusi untuk mahasiswa baru dan pelajar tingkat menengah ketika Universitas YARSI open house dengan memperkenalkan gim CodeLogicCraft 2.0 ini dan memainkannya secara langsung. Masalah tersebut dapat timbul ketika saat mahasiswa baru dan pelajar tingkat menengah untuk memperkenalkan Prodi Teknik Informatika secara visual dan interaktif sehingga saat open house akan lebih menyenangkan dibandingkan dengan memperkenalkannya secara teoritis dan kesulitan yang sering dihadapi saat mulai belajar pemrograman dengan anggapan bahwa materi tersebut kompleks dan membingungkan, terutama saat menyampaikan dengan pendekatan yang minim visualisasi dan kurang interaktif. CodeLogicCraft 2.0 membantu memahami logika dari pemrograman dasar dari awal secara 3 dimensi (3D). Selain itu, CodeLogicCraft 2.0 akan memberikan sebuah hadiah berupa sertifikat bagi siapapun yang berhasil menyelesaikan sampai selesai sebagai tanda pencapaian dalam memainkan gim CodeLogicCraft 2.0.

Ide-ide dalam pengembangan gim CodeLogicCraft 2.0 berasal dari ibu Herika Hayurani, S.Kom, M.Kom. sebagai Wakil Dekan 1 Prodi Teknik Informatika Fakultas Teknologi Informasi. Pengembang mengambil beberapa data dan

informasi yang dibutuhkan dalam pengembangan gim CodeLogicCraft sebelumnya dan diimplementasikan ke dalam gim CodeLogicCraft 2.0.

Dalam Islam, mencari ilmu merupakan kewajiban dan keutamaan yang sangat ditekankan oleh Allah SWT karena ilmu menjadi kunci untuk mengenal-Nya, memahami ciptaan-Nya, dan menjalankan tugas sebagai khalifah di muka bumi dengan bijak. Penegasan ini terlihat dalam wahyu pertama yang diturunkan kepada Nabi Muhammad SAW, yaitu Surat Al-‘Alaq ayat 1–5, yang dimulai dengan perintah "Iqra'" (bacalah), menandakan bahwa proses belajar, membaca, dan memahami adalah langkah awal menuju kedekatan dengan Allah. Ayat tersebut juga menunjukkan bahwa Allah mengajarkan manusia melalui pena dan mengaruniakan ilmu yang sebelumnya tidak diketahui, sehingga menuntut ilmu dalam Islam tidak hanya bernilai duniawi, tetapi juga menjadi bagian dari ibadah yang mengangkat derajat manusia di sisi-Nya. Al-‘Alaq ayat 1–5:

أَقْرَأْ بِاسْمِ رَبِّكَ الَّذِي خَلَقَ ١

Bacalah dengan (menyebut) nama Tuhanmu yang menciptakan. (QS: Al-‘Alaq (96):1)

خَلَقَ الْإِنْسَانَ مِنْ عَلَقٍ ٢

Dia telah menciptakan manusia dari segumpal darah. (QS: Al-‘Alaq (96):2)

أَقْرَأْ وَرَبُّكَ الْأَكْرَمُ ٣

Bacalah, dan Tuhanmulah Yang Maha Pemurah. (QS: Al-‘Alaq (96):3)

الَّذِي عَلَّمَ بِالْقَلَمِ ٤

Yang mengajar (manusia) dengan pena. (QS: Al-‘Alaq (96):4)

عَلَّمَ الْإِنْسَانَ مَا لَمْ يَعْلَمْ ٥

Dia mengajarkan manusia apa yang tidak diketahuinya. (QS: Al-‘Alaq (96):5)

Ibnu Katsir menjelaskan bahwa ayat pertama dalam Surah Al-‘Alaq merupakan wahyu pertama yang diturunkan kepada Nabi Muhammad SAW. Perintah "bacalah" (iqra’) tidak hanya berarti membaca teks, tetapi juga mencakup membaca dengan hati serta menyebut nama Allah. Ayat ini menandai awal kenabian dan menunjukkan pentingnya ilmu dalam Islam. Allah SWT mengajarkan manusia dengan pena simbol alat untuk menyampaikan, menyimpan, mewariskan pengetahuan. Manusia pada dasarnya tidak mengerti apa-apa menjadi makhluk berilmu karena bimbingan dan karunia dari Allah SWT.

Nilai-nilai tersebut selaras dengan apa yang ada di dalam gim edukasi CodeLogicCraft 2.0. Gim ini dirancang tidak hanya sebagai sarana hiburan atau latihan teknis pemrograman, tetapi juga sebagai media pembelajaran yang mendorong pemain untuk membaca, memahami, dan berpikir kritis terhadap situasi yang mereka hadapi saat memainkan gim ini. Gim CodeLogicCraft 2.0 tidak hanya sejalan dengan syariat islam, tetapi juga mencerminkan “iqra” di era digital. Gim ini menjadi media pembelajaran visual untuk melatih intelektual yang dimana dalam gim CodeLogicCraft 2.0 terdapat tantangan yang menjadi sebuah pembelajaran dan solusi untuk menuju pembelajaran lebih dalam.

وَمَا كَانَ الْمُؤْمِنُونَ لِيَنفِرُوا كَآفَّةً ۚ فَلَوْلَا نَفَرَ مِن كُلِّ فِرْقَةٍ مِّنْهُمْ طَائِفَةٌ لِّيَتَفَقَّهُوا فِي الدِّينِ وَلِيُنذِرُوا قَوْمَهُمْ إِذَا رَجَعُوا إِلَيْهِمْ لَعَلَّهُمْ يَحْذَرُونَ

"Tidak sepatutnya bagi seluruh orang mukmin itu pergi (berjihad) semuanya. Mengapa tidak pergi dari tiap-tiap golongan di antara mereka beberapa orang untuk memperdalam pengetahuan mereka tentang agama dan untuk memberi peringatan kepada kaumnya apabila mereka telah kembali kepadanya, supaya mereka itu dapat menjaga diri." (QS: At-Taubah (9):122)

Tafsir Al-Misbah Prof. Dr. M. Quraish Shihab

Quraish Shihab menekankan bahwa ayat ini menunjukkan bahwa jihad fisik bukan satu-satunya bentuk kontribusi dalam Islam. Menuntut ilmu dan mendidik umat juga merupakan bentuk jihad yang sangat penting. Dalam gim CodeLogicCraft 2.0 mengajak pemain menggunakan cara berpikir logis untuk menyelesaikan tantangan yang ada di dalam gim CodeLogicCraft 2.0 ini mencerminkan terdapat jihad intelektual yang dimana disebutkan dalam ayat tersebut. Gim ini implementasi modern untuk mendalami ilmu agama dan dunia sebagai kesatuan, yang kemudian di terapkan dalam teknologi pembelajaran saat ini.

يَا أَيُّهَا الَّذِينَ ءَامَنُوا إِذَا قِيلَ لَكُمْ تَفَسَّحُوا فِي الْمَجَالِسِ فَافْسَحُوا يَفْسَحِ اللَّهُ لَكُمْ وَإِذَا قِيلَ أَنْشُرُوا فَأَنْشُرُوا يَرْفَعِ اللَّهُ الَّذِينَ ءَامَنُوا مِنْكُمْ وَالَّذِينَ أُوتُوا الْعِلْمَ دَرَجَاتٍ وَاللَّهُ بِمَا تَعْمَلُونَ خَبِيرٌ

"Wahai orang-orang yang beriman! Apabila dikatakan kepadamu, 'Berilah kelapangan di dalam majelis', maka lapangkanlah, niscaya Allah akan memberi kelapangan untukmu. Dan apabila dikatakan, 'Berdirilah kamu', maka berdirilah, niscaya Allah akan meninggikan orang-orang yang beriman di antaramu dan orang-orang yang diberi ilmu beberapa derajat. Dan Allah Maha Teliti terhadap apa yang kamu kerjakan." (QS: Al-Mujadalah (58):11)

Tafsir Al-Misbah Prof. Dr. M. Quraish Shihab

1. Adab Sosial dan Etika Majelis:

Quraish Shihab menekankan bahwa ayat ini mengajarkan etika sosial dan kesopanan dalam pertemuan. Umat Islam diajarkan untuk tidak egois dalam majelis dan memberi ruang kepada orang lain. Ini mencerminkan akhlak Islam yang inklusif dan empatik.

2. Ketaatan terhadap Arahan Sosial:

Kata "ansyuzū" dimaknai sebagai perintah untuk bertindak aktif, misalnya berpindah tempat, mengatur barisan, atau taat pada perintah yang bersifat sosial dan terorganisir. Kepatuhan ini adalah bagian dari kedisiplinan umat.

3. Ketinggian Derajat Orang Beriman dan Berilmu:

Tafsir Al-Misbah menekankan bahwa ilmu yang dimaksud adalah ilmu yang diamalkan dan bermanfaat, bukan sekadar hafalan. Orang-orang yang berilmu dan beriman akan ditinggikan derajatnya di dunia dan akhirat, baik secara moral, sosial, maupun spiritual. "Allah meninggikan orang-orang berilmu bukan karena ilmunya semata, tetapi karena fungsi ilmu itu dalam memberi manfaat kepada sesama dan membimbing masyarakat."

قَالَ لَهُ مُوسَىٰ هَلْ أَتَّبِعُكَ عَلَىٰ أَنْ تُعَلِّمَنِي مِمَّا عُلِّمْتَ رُشْدًا

"Musa berkata kepadanya (Khidr), 'Bolehkah aku mengikutimu agar engkau mengajarkan kepadaku sebagian dari ilmu yang telah diajarkan kepadamu berupa petunjuk (yang benar)?'" (QS: Al-Kahfi (18):66)

Berikut Tafsiran Ibnu Katsir untuk surat Al-Kahfi Ayat 66

Tujuan Belajar:

Musa berkata: *"agar engkau mengajarkan kepadaku dari apa yang telah diajarkan kepadamu sebagai petunjuk (rushdan)"* — ini menunjukkan niat yang ikhlas dan benar, yaitu untuk mendapatkan petunjuk dan kebenaran, bukan untuk berdebat atau membanggakan ilmu.

شَهِدَ اللَّهُ أَنَّهُ لَا إِلَهَ إِلَّا هُوَ وَالْمَلَائِكَةُ وَأُولُوا الْعِلْمِ قَائِمًا بِالْقِسْطِ ۚ لَا إِلَهَ إِلَّا هُوَ
الْعَزِيزُ الْحَكِيمُ

“Allah menyatakan bahwa tidak ada Tuhan (yang berhak disembah) melainkan Dia, yang menegakkan keadilan. Para malaikat dan orang-orang yang berilmu (juga menyatakan demikian). Tak ada Tuhan (yang berhak disembah) selain Dia, Yang Mahaperkasa, Mahabijaksana.” (QS: Al-Imran (3):18)

Tafsiran Ibnu Katsir untuk surat Ali Imran ayat 18 menunjukkan kemuliaan ilmu dan orang-orang berilmu, karena ayat ini menyandingkan kesaksian para malaikat dengan kedudukan tinggi ahli ilmu di sisi Allah, sehingga ulama menjadikan ayat ini sebagai dalil tentang keutamaan ilmu dan kedudukan mulia para ahli ilmu di hadapan Allah.

Hadis tentang Kewajiban Menuntut Ilmu

طَلَبُ الْعِلْمِ فَرِيضَةٌ عَلَى كُلِّ مُسْلِمٍ

“Menuntut ilmu itu wajib atas setiap Muslim.” (HR. Ibnu Majah, no. 224; dihasankan oleh Al-Albani)

Hadis ini menunjukkan bahwa menuntut ilmu, terutama ilmu agama seperti tauhid, salat, zakat, dan ilmu yang berkaitan dengan kewajiban individu (fardhu ‘ain), merupakan kewajiban bagi setiap Muslim, termasuk wanita, karena lafaz "Muslim" mencakup pula Muslimah. Menuntut ilmu bukanlah pilihan, melainkan kewajiban agar seseorang dapat beribadah kepada Allah dengan benar.

Hadis tentang Keutamaan Penuntut Ilmu

مَنْ سَلَكَ طَرِيقًا يَلْتَمِسُ فِيهِ عِلْمًا، سَهَّلَ اللَّهُ لَهُ بِهِ طَرِيقًا إِلَى الْجَنَّةِ

“Barang siapa menempuh jalan untuk mencari ilmu, maka Allah akan memudahkan baginya jalan menuju surga.” (HR. Muslim, no. 2699)

Hadis ini menjelaskan bahwa menuntut ilmu merupakan jalan menuju surga, karena ilmu membimbing seseorang untuk beramal saleh. Baik melalui perjalanan fisik, seperti datang ke majelis, maupun cara nonfisik, seperti membuka kitab dan

bertanya, semuanya termasuk dalam usaha menuntut ilmu. Hadis ini juga menjadi motivasi bahwa ilmu bukan sekadar urusan dunia, tetapi merupakan sarana keselamatan di akhirat.

Lalu sebagaimana gim CodeLogicCraft 2.0 dibuat yang mana tujuannya yaitu untuk memperkenalkan dan melatih yang ingin belajar pemrograman dasar dari awal untuk semua kalangan. Tak hanya itu di dalam gim CodeLogicCraft 2.0 memberikan sebuah hadiah berbentuk sertifikat sebagai tanda bahwa sudah menyelesaikan gim.

1.5 Definisi, Singkatan, dan Akronim

Tabel 1 Akronim, Singkatan, dan Definisi

Akronim	Singkatan	Definisi
SKPL	Spesifikasi Kebutuhan Perangkat Lunak	dokumen yang menjelaskan kebutuhan fungsional dan non-fungsional dari perangkat lunak, serta menjadi dasar acuan dalam proses perancangan dan pengembangan.
3D	3 Dimensi	3 dimensi (3D) adalah representasi objek yang memiliki panjang, lebar, dan tinggi, sehingga tampak memiliki kedalaman dan terlihat nyata.
Prodi TI	Program Studi Teknik Informatika	program studi yang fokus pada pengembangan sistem komputer dan pemrograman untuk memecahkan berbagai masalah menggunakan teknologi.
JSON	JavaScript Object Notation	merupakan format pertukaran data yang ringan dan mudah dibaca, digunakan untuk

		menyimpan dan mentransmisikan data antara server dan aplikasi web.
<i>Method</i>		penyimpanan utama dalam perangkat untuk menyimpan data sistem, aplikasi, dan file pengguna.
<i>Internal Storage</i>		penyimpanan utama dalam perangkat untuk menyimpan data sistem, aplikasi, dan file pengguna.
<i>GDLC</i>	<i>Game Development Life Cyle</i>	merupakan suatu metode yang digunakan untuk mengatur tahapan pengembangan gim secara menyeluruh, mulai dari perumusan ide dan konsep, perencanaan dan perancangan, proses produksi, tahap pengujian, hingga gim siap diluncurkan kepada pengguna akhir (Prasetyo et al., 2021).
<i>C#</i>		C# adalah bahasa pemrograman berorientasi objek yang modern , inovatif , sumber terbuka , lintas platform dan salah satu dari 5 bahasa pemrograman teratas di GitHub. (https://dotnet.microsoft.com/en-us/languages/csharp)

<i>GBL</i>	<i>Game-Based Learning</i>	Game-based learning (GBL) adalah pendekatan pembelajaran yang memanfaatkan gim sebagai media untuk menyampaikan materi, meningkatkan pemahaman, dan membangun keterampilan tertentu melalui pengalaman bermain yang interaktif dan menyenangkan (Wong et al., 2025).
------------	----------------------------	--

1.6 Deskripsi Umum Bab

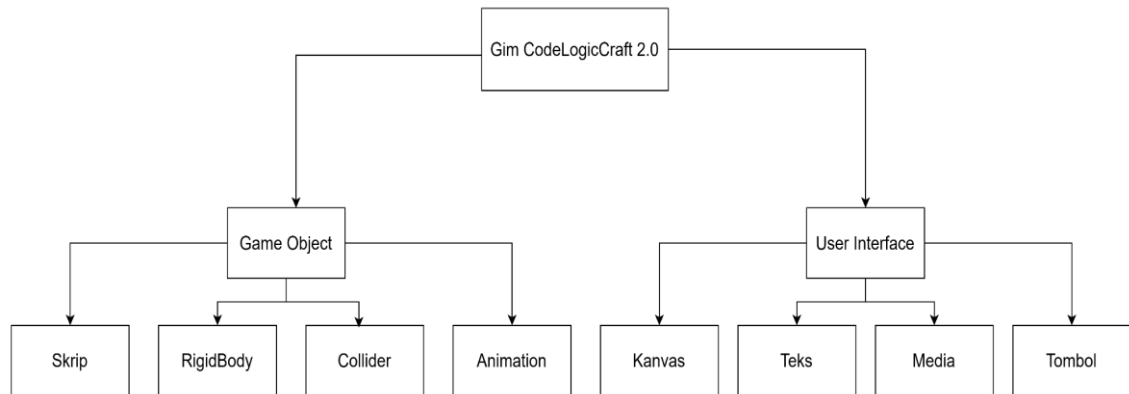
SKPL ini menjelaskan sistematika dari perangkat lunak sebagai berikut:

1. Pendahuluan
2. Deskripsi umum perangkat lunak
3. Kebutuhan fungsional
4. Fitur sistem
5. Kebutuhan nonfungsional

1.7 Deskripsi Umum Perangkat Lunak

Gim CodeLogicCraft 2.0 merupakan perangkat lunak edukatif yang dirancang untuk digunakan dalam lingkungan Universitas YARSI maupun oleh masyarakat umum dengan usia minimal 6 tahun. Gim ini bertujuan untuk mempermudah pemain dalam memahami dan mengenal konsep dasar logika pemrograman. Pengembangan CodeLogicCraft 2.0 dilakukan sebagai penyempurnaan dan pengganti dari versi sebelumnya.

Diagram komponen utama dari gim sebagai berikut:



Gambar 2 Deskripsi Umum Perangkat Lunak

1.7.1 Perspektif Produk

Gim CodeLogicCraft 2.0 adalah gim berbasis edukatif (*game-based learning*) yang dirancang untuk menjadi media pembelajaran yang menarik dan menantang. Gim ini memberikan tantangan visual dan logika secara bertahap, sehingga pemain dapat mempelajari dasar-dasar pemrograman seperti urutan instruksi, perulangan (*looping*), pengambilan keputusan (*conditional branching*), dan fungsi (*function*) dengan cara yang menyenangkan.

CodeLogicCraft 2.0 merupakan kelanjutan dari proyek gim yang dikembangkan dalam mata kuliah pengembangan gim, melanjutkan versi sebelumnya yaitu CodeLogicCraft yang dibuat oleh mahasiswa pada mata kuliah trek game. Pada versi terdahulu, gim hanya mencakup beberapa *level* dengan tingkat kesulitan dasar hingga percabangan, serta masih menggunakan tampilan dua dimensi (2D). Pengembangan pada versi 2.0 difokuskan pada peningkatan tingkat kesulitan, mencakup materi perulangan (*looping*) dan fungsi (*function*), serta peningkatan visual berupa tampilan tiga dimensi (3D) dan penambahan animasi sebagai elemen latar belakang. Produk ini dirancang untuk mendukung proses pembelajaran logika pemrograman secara lebih interaktif, menyenangkan, dan menarik bagi pengguna serta menerapkan prinsip-prinsip nilai islam yang terkait dengan menuntut ilmu dan merupakan ibadah di jalan Allah SWT.

CodeLogicCraft 2.0 menggunakan metode *Game Development Life Cycle (GDLC)*, serta didukung teknologi seperti bahasa pemrograman C# untuk Gim, Figma untuk desain UI/UX, Trello untuk manajemen tugas, dan Github sebagai dokumentasi kode program.

1.7.2 Fungsi Produk

Gim CodeLogicCraft 2.0 dibuat untuk membantu pemain untuk mengenalkan dan melatih konsep logika dasar pemrograman. Fungsi utama dari gim ini sebagai berikut:

1. Memfasilitasi pemain dalam memahami konsep-konsep dasar pemrograman, seperti perulangan, percabangan, dan metode, melalui penyajian tantangan yang dirancang secara menarik dan interaktif. Pemain dapat melatih logika untuk belajar dasar pemrograman.
2. Mendorong pemain untuk mengasah kemampuan berpikir logis secara terstruktur, yang merupakan fondasi utama dalam pembelajaran pemrograman dasar.
3. Berperan sebagai media pembelajaran alternatif bagi dosen dan mahasiswa guna menciptakan proses belajar yang lebih interaktif, menarik, dan tidak monoton.
4. Mengembangkan keterampilan pemain dalam memecahkan masalah dan melakukan debugging, dengan membiasakan mereka merancang tahapan pemrograman, mengidentifikasi kesalahan, serta menyusun solusi logis terhadap permasalahan yang muncul dalam permainan.
5. Mendorong ketertarikan terhadap bidang teknologi dan pemrograman, melalui pengalaman belajar yang tidak membosankan dan penuh tantangan.

1.7.3 Kelas dan Karakteristik Pengguna

CodeLogicCraft 2.0 ditujukan untuk pelajar tingkat menengah, mahasiswa, dan masyarakat umum yang ingin mempelajari dasar pemrograman melalui pendekatan *game-based learning*. Pengguna utama gim ini adalah individu dengan rentang usia dari pelajar tingkat menengah hingga mahasiswa awal yang memiliki minat pada

logika, teknologi, dan keterampilan berpikir komputasional. Sebagian besar pengguna menggunakan perangkat komputer yang mendukung media belajar yang interaktif, menarik, dan tidak membosankan. Mereka dapat mengakses penuh fitur yang tersedia, seperti latihan logika, tantangan pemrograman, umpan balik, dan umpan balik interaktif.

Tabel 2 Kelas dan Karakteristik Pengguna

Pengguna	Karakteristik	Hak Akses
Pelajar tingkat menengah dan Mahasiswa	Memainkan gim CodeLogicCraft 2.0	Menggunakan aplikasi gim CodeLogicCraft 2.0 dan mendapat sertifikat penghargaan.

1.7.4 Lingkungan Operasi

Gim CodeLogicCraft 2.0 dirancang untuk digunakan oleh pemain dengan kategori pelajar tingkat menengah dan mahasiswa khususnya, yang ingin belajar dasar-dasar logika pemrograman. Gim ini dapat dimainkan di perangkat desktop dengan sistem operasi minimal Windows 7 atau lebih tinggi.

Aplikasi ini dikembangkan dengan menggunakan Unity sebagai game engine utama dengan bahasa pemrograman C# dan desain antarmuka pengguna yang disesuaikan agar ramah dengan pengguna, khususnya pemain yang ingin belajar logika pemrograman.

1.7.5 Batasan Perancangan dan Implementasi

Gim CodeLogicCraft 2.0 ditujukan untuk masyarakat umum dan digunakan sebagai media pembelajaran bagi siapa saja yang ingin mempelajari logika pemrograman. Gim ini memiliki beberapa batasan, antara lain:

1. Hanya dapat digunakan pada *platform desktop*,
2. Dikembangkan menggunakan Unity dan bahasa C#,
3. Hanya berjalan secara independen (*standalone*),
4. Hanya tersedia dalam Bahasa Indonesia,

1.7.6 Dokumentasi Pengguna

Daftar dokumentasi pengguna yang disertakan bersama dengan gim CodeLogicCraft 2.0 adalah:

1. *Tutorial* gim (dapat diakses digim CodeLogicCraft 2.0)
2. Manual pengguna (dapat diakses digim CodeLogicCraft 2.0)

1.7.7 Asumsi dan Dependensi

Asumsi:

1. Pengguna memiliki perangkat yang memenuhi spesifikasi minimum agar dapat menjalankan gim CodeLogicCraft 2.0 dengan optimal.
2. Pengguna memiliki minat terhadap pembelajaran pemrograman.
3. pengguna memiliki pengalaman belajar yang menyenangkan dan tidak membosankan.
4. Pengembang menggunakan perangkat laptop dengan spesifikasi minimum yang sesuai dengan ketentuan dari Unity.

Dependensi:

Pengembangan ini bersifat lanjutan dari versi produk yang telah dirilis sebelumnya, dengan memanfaatkan fondasi yang sudah ada untuk menghadirkan pengalaman bermain yang lebih baik bagi pengguna.

1.8 Kebutuhan Fungsional

Dari hasil wawancara, beberapa kebutuhan fungsional yang telah diidentifikasi untuk pengembangan Gim CodeLogicCraft 2.0 adalah sebagai berikut:

1. Logo berfungsi sebagai identitas utama dari permainan.
2. Fitur *gameplay* harus mencakup konsep dasar logika pemrograman seperti urutan dasar, perulangan, percabangan, dan method.
3. Animasi karakter wajib menampilkan aksi seperti diam, berjalan, dan mengambil objek.
4. Pada tingkat kesulitan dasar, pemain ditugaskan untuk mengumpulkan seluruh objek dalam satu *level* dan mengarahkan karakter ke garis akhir.
5. Di tahap perulangan, pemain harus menyusun kode yang berulang menggunakan struktur pengulangan.
6. Saat mencapai tantangan percabangan, pemain perlu menggunakan kode bercabang untuk memilih arah pergerakan karakter.
7. Pada tahap method, pemain diminta untuk menyimpan kode ke dalam kotak method agar lebih terorganisir.
8. Bintang dapat diperoleh jika pemain menyelesaikan *level* dengan kode yang efisien dan optimal.
9. *Tutorial* dasar disediakan untuk menjelaskan penggunaan fitur *gameplay* pada tahap awal serta hubungan tombol kode dengan konsep pemrograman.
10. *Tutorial* perulangan menjelaskan pengertian dan cara menggunakan struktur perulangan dalam permainan.
11. *Tutorial* percabangan memberikan penjelasan mengenai fungsi dan penerapan logika percabangan.
12. Penjelasan tentang method dan cara menggunakannya terdapat pada *tutorial* khusus method.
13. Halaman utama menampilkan tombol seperti lanjut, main, tentang, keluar, serta opsi aktif/nonaktif suara dan pemilihan karakter.
14. Nama pemain dapat dimasukkan dan akan tercantum di sertifikat setelah semua *level* diselesaikan.

15. Halaman *level* memungkinkan pemain untuk memilih tingkat kesulitan dan *level* yang ingin dimainkan.
16. Animasi dan suara kemenangan digunakan untuk memberi rasa senang, menunjukkan keberhasilan, dan memotivasi pemain.
17. Animasi dan suara kekalahan digunakan untuk memberi tanda kegagalan, menciptakan ketegangan, dan mendorong pemain untuk belajar dari kesalahan.
18. Halaman sertifikat disediakan sebagai pengakuan atas pencapaian pemain yang telah menyelesaikan semua tantangan, sekaligus menjadi bentuk motivasi dan penghargaan.
19. Data seperti nama, jumlah bintang, dan *level* yang terbuka akan tersimpan otomatis di perangkat.
20. Tersedia opsi untuk memulai permainan baru atau melanjutkan progres terakhir.
21. Pemain dapat memilih salah satu dari tiga *skin* karakter yang tersedia melalui tombol ganti karakter.
22. Tombol suara berfungsi untuk mengaktifkan atau menonaktifkan efek suara dalam permainan.
23. Musik latar belakang membantu menciptakan suasana bermain yang menyenangkan dan hidup.
24. Cerita yang disediakan membantu membangun keterikatan emosional antara pemain dan karakter, sehingga meningkatkan semangat bermain dan rasa peduli terhadap alur permainan.

1.8.1 Antarmuka Pengguna

Pengguna berinteraksi dengan gim CodeLogicCraft 2.0 melalui antarmuka grafis atau *Graphical User Interface* (GUI), yang dirancang untuk menyajikan elemen-elemen visual seperti menu, gambar, tombol, dan animasi secara langsung pada layar monitor. Antarmuka ini memungkinkan pemain untuk dengan mudah memahami dan mengakses fitur-fitur yang tersedia dalam gim.

Interaksi dalam gim dilakukan menggunakan perangkat **input** utama, yaitu mouse dan keyboard. Mouse digunakan untuk melakukan *drag and drop* elemen-elemen tombol ke area eksekusi, serta untuk menggerakkan karakter dan berinteraksi dengan lingkungan sekitar maupun menu yang tersedia. Sementara itu, keyboard digunakan sebagai alat **input** teks, khususnya untuk mengisi nama pemain yang nantinya akan dicantumkan pada sertifikat digital sebagai bentuk penghargaan setelah berhasil menyelesaikan seluruh *level* dalam gim. Dengan rancangan antarmuka yang intuitif dan responsive CodeLogicCraft 2.0 diharapkan dapat memberikan pengalaman belajar yang menyenangkan dan mudah diakses oleh pengguna.

1.8.2 Antarmuka Perangkat Keras

Gim CodeLogicCraft 2.0 memiliki kebutuhan minimum perangkat keras yang digunakan pengguna untuk berinteraksi berupa:

- CPU : Intel Core I3 Gen 7 atau (APU) AMD Ryzen Gen 3
- RAM : 4 GB
- GPU : *Integrated GPU*
- Hard Drive : 10 GB *free disk space*
- Sistem Operasi : Windows 7
- Papan Kunci (*keyboard & mouse*)

1.8.3 Antarmuka Perangkat Lunak

CodeLogicCraft 2.0 merupakan gim edukatif berbasis komputer yang dirancang untuk membantu pengguna, khususnya mahasiswa dan mahasiswa pemula, dalam memahami dasar-dasar pemrograman secara interaktif dan menyenangkan melalui pendekatan *game-based learning*. Gim ini dapat dijalankan pada sistem operasi Windows 7 atau yang lebih baru, serta dikembangkan menggunakan bahasa pemrograman C# dalam platform Unity.

Untuk mendukung performa dan kenyamanan antarmuka pengguna, CodeLogicCraft 2.0 menggunakan sejumlah library dan komponen tambahan, antara lain:

1. Unity UI Toolkit untuk pengaturan elemen antarmuka seperti tombol, dialog, dan navigasi antar *level*.
2. Cinemachine dan Timeline untuk animasi transisi dan efek visual yang dinamis.
3. PlayerPrefs dan JSON File Handling untuk menyimpan data progres pengguna secara lokal, seperti *level* yang telah diselesaikan dan skor akhir.
4. In-App Event System untuk mengatur interaksi antar objek dan memberikan umpan balik otomatis terhadap aksi pemain.

Input dalam gim ini berasal dari interaksi langsung pemain, seperti memilih instruksi, menyusun blok logika, atau menyelesaikan tantangan pemrograman dasar. Data yang diperoleh dari aktivitas ini kemudian diproses untuk menghasilkan:

1. Penilaian logika pemain berdasarkan ketepatan dan efisiensi solusi kode.
2. Pemberian umpan balik otomatis berupa animasi, bintang, atau pesan perbaikan.
3. Pelacakan progres belajar yang ditampilkan dalam bentuk level.

Dengan pendekatan visual, naratif, dan interaktif, CodeLogicCraft 2.0 diharapkan menjadi media pembelajaran alternatif yang tidak hanya efektif, tetapi juga menarik untuk meningkatkan minat belajar pemrograman di kalangan pemula.

1.8.4 Antarmuka Komunikasi

Gim CodeLogicCraft 2.0 tidak memanfaatkan antarmuka komunikasi karena seluruh proses dijalankan secara mandiri (*standalone*). Sebagai aplikasi yang beroperasi secara independen tanpa koneksi ke sistem eksternal, gim ini tidak memerlukan penerapan standar komunikasi apa pun dalam fungsionalitasnya

1.9 Fitur Sistem

1.9.1 Fitur Penyimpanan Data

1.9.1.1 Deskripsi

Fitur penyimpanan data pada gim CodeLogicCraft 2.0 berfungsi untuk menyimpan progres permainan seperti jumlah bintang yang diperoleh dan *level* yang telah dimainkan oleh pengguna. Penyimpanan dilakukan secara lokal di *internal storage* perangkat, dengan menggunakan format berkas JSON sebagai acuan struktur data. Format ini dipilih karena fleksibel, ringan, dan mudah diolah untuk kebutuhan penyimpanan dan pemanggilan data dalam gim.

1.9.1.2 Trigger

Fitur ini akan dieksekusi secara otomatis ketika pemain telah menyelesaikan beberapa *level* dalam gim CodeLogicCraft 2.0. Data progres *level* yang telah dimainkan akan tersimpan, sehingga *level* berikutnya akan terbuka secara otomatis berdasarkan data yang telah disimpan.

1.9.1.3 Input

Input dari fitur penyimpanan data berupa data progres permainan yang dihasilkan selama pemain menjalankan gim. Data ini meliputi *level* yang telah diselesaikan, jumlah bintang yang diperoleh, serta status terbukanya *level* berikutnya. Seluruh data ini akan diolah dan disimpan dalam format JSON di penyimpanan internal perangkat untuk memastikan progres pemain dapat dilanjutkan di sesi permainan berikutnya tanpa harus mengulang dari awal.

1.9.1.4 Output

Output dari fitur penyimpanan data berupa file berformat JSON yang tersimpan di *internal storage* perangkat. File ini berisi informasi progres pemain seperti *level* yang telah diselesaikan, jumlah bintang yang diperoleh, dan *level* yang telah terbuka. Data yang tersimpan ini akan digunakan kembali saat gim dijalankan,

sehingga pemain dapat melanjutkan permainan dari progres terakhir tanpa kehilangan data.

1.9.1.5 Skenario Utama

Pemain hendak keluar dari gim setelah menyelesaikan beberapa *level* dalam gim CodeLogicCraft 2.0.

Prakondisi: Pemain telah menyelesaikan beberapa *level* dan memperoleh bintang sesuai pencapaian pada *level-level* tersebut.

Pascakondisi: Data progres permainan, termasuk *level* yang telah diselesaikan dan jumlah bintang yang diperoleh, berhasil disimpan ke dalam file berformat JSON di penyimpanan internal.

Langkah-langkah:

1. Pemain menyelesaikan satu atau lebih *level*;
2. Pemain memperoleh bintang sesuai pencapaian disetiap *level* yang telah diselesaikan;
3. Sistem secara otomatis menyimpan progres permainan, termasuk data *level* dan jumlah bintang, ke dalam file json;
4. Pemain dapat melanjutkan permainannya dari progress terakhir yang telah tersimpan.

1.9.1.6 Skenario Eksepsional 1

Pemain memilih untuk mengulang *level* yang sebelumnya sudah diselesaikan. Setelah menyelesaikan ulang *level* tersebut, sistem akan memperbarui jumlah bintang berdasarkan hasil terbaru.

Prakondisi: Pemain telah menyelesaikan beberapa *level* dan memiliki data progres sebelumnya.

Pascakondisi: Setelah pemain menyelesaikan ulang *level* yang telah dimainkan, sistem secara otomatis memperbarui jumlah bintang pada *level* tersebut di file JSON sesuai pencapaian terbaru.

Langkah-langkah:

1. Pemain telah menyelesaikan beberapa *level* dalam gim;
2. Pemain memilih untuk mengulang salah satu *level* yang telah diselesaikan sebelumnya;
3. Pemain memperoleh jumlah bintang yang berbeda dari pencapaian sebelumnya;
4. Sistem memperbarui data bintang pada *level* yang dimainkan ulang dan menyimpannya ke dalam file JSON;
5. Jumlah bintang yang ditampilkan pada *level* tersebut diperbarui sesuai dengan pencapaian terbaru pemain.

1.9.2 Fitur Mengubah Karakter

1.9.2.1 Deskripsi

Pemain dapat memilih dan mengubah karakter sesuai dengan preferensi masing-masing. Karakter yang dipilih akan digunakan sebagai representasi tokoh utama pemain di dalam permainan.

1.9.2.2 Trigger

Fitur ini digunakan saat pemain ingin mengubah karakter yang dipakai dalam gim. Pemain mengakses fitur tersebut melalui tombol khusus yang tersedia pada antarmuka beranda gim. Setelah tombol ditekan, sistem menampilkan pilihan karakter yang dapat dipilih dan diubah sesuai keinginan pemain. Dengan demikian, pemain dapat mempersonalisasi pengalaman bermain dengan memilih karakter favoritnya sebelum memulai atau melanjutkan permainan.

1.9.2.3 Input

Pemain memilih karakter sesuai dengan preferensi atau keinginannya melalui antarmuka pemilihan karakter yang tersedia dalam gim.

1.9.2.4 Output

Karakter yang telah dipilih akan ditampilkan dalam permainan dan digunakan sebagai tokoh pemain untuk melanjutkan permainan.

1.9.2.5 Skenario Utama

Pemain berada pada fitur beranda dalam gim dan ingin mengubah karakter sesuai dengan preferensinya.

Prakondisi: Pemain telah berada di halaman beranda gim dan ingin memilih karakter lain yang tersedia.

Pascakondisi: Karakter berhasil diganti sesuai pilihan pemain melalui navigasi tombol kanan atau kiri yang tersedia pada bagian samping antarmuka beranda.

Langkah-langkah:

1. Pemain berada pada tampilan beranda gim;
2. Pemain memutuskan untuk mengubah karakter yang akan digunakan;
3. Pemain menekan tombol navigasi kanan atau kiri yang terletak di bagian samping antarmuka beranda untuk memilih karakter;
4. Setelah karakter dipilih, karakter tersebut secara otomatis siap digunakan dalam permainan.

1.9.3 Fitur Main

1.9.3.1 Deskripsi

Fitur Main merupakan fitur utama yang digunakan oleh pemain untuk memulai permainan. Fitur ini dapat diakses melalui tombol “Mulai” yang tersedia pada halaman beranda gim. Ketika pemain baru pertama kali mengakses gim, tombol ini menjadi pintu masuk untuk memulai petualangan dari *level* awal. Setelah tombol

“Mulai” ditekan, sistem akan mengarahkan pemain ke tampilan *level* yang tersedia dan siap dimainkan. Fitur ini dirancang untuk memberikan pengalaman awal bermain yang mudah dan intuitif, baik bagi pemain baru maupun yang sudah pernah memainkan gim sebelumnya.

1.9.3.2 Trigger

Fitur ini dapat diakses oleh pemain ketika berada pada tampilan beranda gim dan ingin memulai permainan untuk pertama kalinya atau mengulang permainan dari awal.

1.9.3.3 Input

Pemain menekan tombol “Mulai” yang tersedia pada tampilan beranda gim untuk memulai permainan dari *level* pertama atau mengulang permainan dari awal.

1.9.3.4 Output

Setelah tombol “Mulai” ditekan, gim akan menampilkan tampilan pemilihan *level* atau langsung mengarahkan pemain ke *level* pertama untuk memulai permainan. Pemain kemudian dapat mulai berinteraksi dengan lingkungan gim sesuai alur permainan yang telah dirancang.

1.9.3.5 Skenario Utama

Ketika pemain pertama kali memainkan gim CodeLogicCraft 2.0, proses dimulai dengan menekan tombol “Mulai” yang tersedia di halaman beranda untuk memulai permainan.

Prakondisi: Pemain berada pada tampilan beranda dan belum pernah memainkan gim CodeLogicCraft 2.0 sebelumnya.

Pascakondisi: Setelah pemain menekan tombol “Mulai”, sistem akan menampilkan jendela **input** untuk memasukkan nama, yang nantinya akan digunakan sebagai nama pada sertifikat penyelesaian gim.

Langkah-langkah:

1. Pemain baru pertama kali menjalankan gim CodeLogicCraft 2.0;
2. Pemain berada di halaman beranda gim;
3. Pemain mengklik tombol “Mulai” untuk memulai permainan;
4. Sistem menampilkan jendela **input** untuk mengisi nama, yang nantinya akan digunakan pada sertifikat setelah pemain menyelesaikan semua *level*;
5. Pemain diarahkan ke halaman *level* permainan dan siap memulai tantangan.

1.9.3.6 Skenario Eksepsional 1

Pemain yang sebelumnya sudah pernah bermain dan menyelesaikan satu atau lebih *level* ingin memulai permainan dari awal kembali.

Prakondisi: Pemain berada di halaman beranda dan memilih untuk mengulang permainan dari awal.

Pascakondisi: Pemain telah menekan tombol “Mulai” untuk memulai ulang gim dari *level* pertama.

Langkah-langkah:

1. Pemain telah menyelesaikan satu atau lebih *level* dalam gim;
2. Pemain ingin memulai permainan dari awal Kembali;
3. Pemain berada di halaman beranda gim;
4. Pada halaman beranda terdapat tombol “Mulai” dan “Lanjutkan”;
5. Pemain mengklik tombol “Mulai”;
6. Sistem menampilkan jendela **input** untuk mengisi nama, yang akan digunakan pada sertifikat setelah pemain menyelesaikan seluruh *level*;
7. Pemain akan ditampilkan sebuah prolog yang memperkenalkan latar cerita dari gim CodeLogicCraft 2.0;
8. Pemain diarahkan ke halaman *level* permainan dan siap memulai tantangan baru.

1.9.4 Fitur Tutorial

1.9.4.1 Deskripsi

Fitur *tutorial* pada gim CodeLogicCraft 2.0 berfungsi sebagai panduan interaktif bagi pemain untuk memahami cara bermain secara bertahap. Berbeda dari *tutorial* biasa yang hanya menampilkan instruksi teks, gim ini menyertakan animasi pembuka yang menampilkan latar belakang cerita dari karakter utama. Animasi ini tidak hanya memberikan konteks cerita tetapi juga membantu menciptakan suasana yang menarik dan tidak membosankan. *Tutorial* disesuaikan dengan tingkat kesulitan permainan dan diberikan di setiap tahapan, seperti pengenalan pergerakan dasar, konsep perulangan, percabangan logika, dan penggunaan *Method*. Melalui fitur ini, pemain diperkenalkan pada konsep-konsep pemrograman secara bertahap dengan cara yang menyenangkan dan tematis.

1.9.4.2 Trigger

Fitur ini akan dijalankan sebelum pemain dapat memasuki *level* utama. Pemain harus menyelesaikan *tutorial* terlebih dahulu, karena tanpa melakukannya, akses ke *level* permainan berikutnya tidak akan tersedia.

1.9.4.3 Input

Pemain menyelesaikan *level tutorial* dan memperoleh bintang sebagai bentuk penilaian atas pencapaian dalam *tutorial* tersebut.

1.9.4.4 Output

Setelah pemain menyelesaikan *tutorial*, *level* berikutnya akan terbuka dan dapat diakses untuk dimainkan.

1.9.4.5 Skenario Utama

Pemain telah memasuki halaman *level* dan memilih tombol *level* bernama "Mulai" untuk memulai *tutorial*.

Prakondisi: Pemain belum dapat mengakses *Level 1* karena harus menyelesaikan *tutorial* terlebih dahulu dengan memasuki *level "Mulai"*.

Pascakondisi: Pemain memperoleh bintang sebagai bentuk apresiasi karena telah mengikuti *tutorial* dengan baik, dan permainan dapat dilanjutkan ke *level* berikutnya.

Langkah-langkah:

1. Pemain telah memasuki halaman *level*;
2. Pemain mengklik tombol *level "Mulai"* untuk memulai *tutorial*;
3. Pemain akan disajikan cerita berupa animasi yang sesuai dengan tema tingkat kesulitan tersebut;
4. Pemain menerima instruksi untuk menjalankan *tutorial* yang tersedia;
5. Pemain mengikuti *tutorial* sesuai dengan instruksi yang diberikan;
6. Pemain menyelesaikan *tutorial* dengan tuntas;
7. Pemain siap untuk melanjutkan ke *level* berikutnya.

1.9.5 Fitur Kunci *Level*

1.9.5.1 Deskripsi

Fitur kunci *level* berfungsi untuk mengunci *level* berikutnya apabila pemain belum menyelesaikan *level* sebelumnya. Fitur ini penting agar pemain tidak dapat melewati atau memainkan *level* yang belum terbuka secara sah dengan menyelesaikan tahapan sebelumnya.

1.9.5.2 Trigger

Pemain tidak dapat memulai *level* berikutnya sebelum menyelesaikan *level* sebelumnya.

1.9.5.3 Input

Pemain telah menyelesaikan *level* sebelumnya sebagai syarat untuk membuka *level* berikutnya.

1.9.5.4 Output

Pemain diberikan akses untuk membuka dan melanjutkan ke *level* berikutnya setelah berhasil menyelesaikan tantangan pada *level* sebelumnya.

1.9.5.5 Skenario Utama

Pemain ingin membuka *level* yang sebelumnya masih terkunci.

Prakondisi: *Level* yang ingin diakses masih dalam keadaan terkunci karena pemain belum menyelesaikan *level* sebelumnya.

Pascakondisi: Setelah pemain berhasil menyelesaikan *level* sebelumnya, *level* yang dituju akan terbuka dan dapat dimainkan.

Langkah-langkah:

1. Pemain berada pada halaman *level* permainan;
2. Pemain ingin memainkan *level* 3, namun belum menyelesaikan *level* 2;
3. Sistem mendeteksi bahwa *level* 3 masih terkunci;
4. Pemain diwajibkan untuk menyelesaikan *level* 2 terlebih dahulu;
5. Setelah *level* 2 berhasil diselesaikan, *level* 3 akan terbuka;
6. Pemain dapat melanjutkan dan menyelesaikan *level* 3.

1.9.6 Fitur Bintang

1.9.6.1 Deskripsi

Fitur bintang merupakan sistem penilaian pada setiap *level* dalam permainan. Setiap *level* memiliki nilai maksimal sebanyak tiga bintang. Jika pemain berhasil menyelesaikan *level* dengan baik dan berhasil mengumpulkan seluruh barang yang ditentukan oleh sistem, maka pemain akan memperoleh jumlah bintang yang sesuai dengan pencapaiannya.

1.9.6.2 Trigger

Fitur ini diaktifkan ketika pemain menyelesaikan sebuah *level* dengan mengumpulkan semua barang yang tersedia dalam *level* tersebut, sehingga memungkinkan pemain memperoleh jumlah bintang maksimal.

1.9.6.3 Input

Pemain ditantang untuk mengumpulkan seluruh item atau objek yang tersebar di dalam *level* guna memperoleh skor maksimal atau membuka fitur tambahan.

1.9.6.4 Output

Pemain memperoleh jumlah bintang maksimal sesuai dengan pencapaian dalam menyelesaikan *level*.

1.9.6.5 Skenario Utama

Pemain memainkan sebuah *level* dan berusaha mengumpulkan semua barang yang tersedia dengan tujuan memperoleh bintang maksimal.

Prakondisi: Pemain telah memulai permainan pada sebuah *level* dan bertekad menyelesaikannya dengan baik, termasuk mengumpulkan semua barang yang ada di *level* tersebut.

Pascakondisi: Setelah menyelesaikan *level* dan berhasil mengumpulkan seluruh barang, pemain memperoleh jumlah bintang maksimal sesuai penilaian sistem.

Langkah-langkah:

1. Pemain memulai permainan pada *level* tertentu;
2. Pemain mengumpulkan semua barang yang tersedia di dalam *level* tersebut;
3. Pemain mengarahkan karakter hingga mencapai garis akhir (finish);
4. Setelah menyelesaikan *level* dan mengumpulkan seluruh barang, pemain mendapatkan poin maksimal, yaitu tiga bintang.

1.9.6.6 Skenario Eksepsional 1

Pemain menyelesaikan sebuah *level* tanpa mengambil seluruh barang yang tersedia di dalam *level* tersebut. Akibatnya, pemain hanya memperoleh dua bintang karena tidak memenuhi semua kriteria penilaian.

Prakondisi: Pemain berhasil menyelesaikan *level*, namun tidak mengumpulkan semua barang yang diwajibkan.

Pascakondisi: Pemain memperoleh dua bintang sebagai bentuk apresiasi atas pencapaian yang belum sempurna, dan *level* tetap dianggap telah diselesaikan.

Langkah-langkah:

1. Pemain memainkan *level* yang tersedia;
2. Pemain hanya mengambil sebagian barang yang ada di dalam *level*, tidak seluruhnya;
3. Pemain berhasil menyelesaikan *level* hingga titik akhir;
4. Pemain memperoleh dua bintang sebagai hasil dari pencapaian tersebut.

1.9.6.7 Skenario Eksepsional 2

Pemain menyelesaikan *level* tersebut tanpa mengambil barang sama sekali, sehingga hanya mendapatkan satu bintang.

Prakondisi: Pemain menyelesaikan *level* tanpa mengambil satu pun barang.

Pascakondisi: Pemain memperoleh satu bintang sebagai hasil penyelesaian *level* tersebut.

Langkah-langkah:

1. Pemain memainkan *level* tersebut;
2. Pemain tidak mengambil satu pun barang yang ada di *level* tersebut;
3. Pemain menyelesaikan permainan dengan memposisikan karakternya di garis finish;
4. Pemain mendapatkan satu bintang sebagai hasil penyelesaian.

1.9.7 Fitur Tombol Maju

1.9.7.1 Deskripsi

Fitur Tombol Maju merupakan fitur yang digunakan untuk menggerakkan karakter lurus ke depan. Fitur ini sangat berguna dalam menyelesaikan tantangan pada setiap *level*, karena memungkinkan pemain untuk mengarahkan karakter menuju tujuan dengan tepat.

1.9.7.2 Trigger

Fitur ini dipicu ketika pemain ingin menggerakkan karakter secara lurus ke depan dalam permainan.

1.9.7.3 Input

Pemain menyeret dan meletakkan fitur Tombol Maju ke dalam area Kotak *Main* atau Kotak *Method* sebagai bagian dari penyusunan urutan perintah.

1.9.7.4 Output

Karakter akan menjalankan animasi berlari lurus ke depan, berpindah dari petak awal ke satu petak di depannya.

1.9.7.5 Skenario Utama

Pemain menggunakan fitur Tombol Maju dengan memasukkannya ke dalam Kotak *Main* (kotak eksekusi) untuk menggerakkan karakter dari satu petak ke petak di depannya.

Prakondisi: Pemain telah memasukkan perintah "Maju" ke dalam Kotak *Main*.

Pascakondisi: Karakter menjalankan animasi bergerak maju dari petak pertama ke petak di depannya.

Langkah-langkah:

1. Pemain memasuki *level* tertentu;
2. Pemain memasukkan perintah "Maju" ke dalam Kotak *Main* (kotak eksekusi);
3. Pemain menekan tombol "Mainkan" atau "*Play*";
4. Sistem menjalankan animasi karakter yang bergerak maju;
5. Karakter berpindah dari petak awal ke petak di depannya.

1.9.8 Fitur Tombol *Play*

1.9.8.1 Deskripsi

Fitur Tombol *Play* berfungsi untuk menjalankan semua perintah yang telah dimasukkan ke dalam Kotak *Main* (kotak eksekusi). Fitur ini memungkinkan karakter bergerak sesuai dengan urutan perintah yang telah diatur oleh pemain dalam Kotak *Main*.

1.9.8.2 Trigger

Pemain harus mengklik fitur Tombol *Play* untuk menjalankan karakter sesuai arah perintah yang telah dimasukkan.

1.9.8.3 Input

Pemain mengklik tombol *Play*.

1.9.8.4 Output

Karakter bergerak sesuai dengan urutan perintah yang telah dimasukkan di Kotak *Main*.

1.9.8.5 Skenario Utama

Pemain telah memasukkan perintah ke dalam Kotak *Main* dan siap menjalankan karakter menuju titik finish.

Prakondisi: Pemain telah menyusun perintah di Kotak *Main* dan mengklik tombol *Play*.

Pascakondisi: Karakter menjalankan animasi dan bergerak sesuai dengan perintah yang telah disusun di Kotak *Main*.

Langkah-langkah :

1. Pemain memasukkan perintah ke dalam Kotak *Main*;
2. Pemain memastikan bahwa perintah yang dimasukkan sudah benar untuk mencapai tujuan di *level* tersebut;
3. Pemain mengklik tombol *Play*;
4. Sistem menjalankan animasi karakter sesuai dengan perintah yang ada di Kotak *Main*.

1.9.9 Fitur Tombol Hadap Kanan atau Kiri

1.9.9.1 Deskripsi

Fitur Tombol Hadap berfungsi untuk mengatur arah hadap karakter dalam gim, yaitu menghadap ke kanan atau ke kiri sesuai dengan perintah atau keinginan pemain. Fitur ini penting agar karakter dapat bergerak dan berinteraksi dengan lingkungan permainan secara tepat, sesuai dengan arah tujuan yang diinginkan pemain.

Dengan fitur ini, pemain dapat menentukan secara presisi arah pergerakan karakter sebelum menjalankan aksi lain, sehingga meningkatkan kontrol dan strategi selama bermain. Fitur Hadap juga membantu dalam menyelesaikan tantangan yang membutuhkan perubahan arah secara cepat dan tepat, serta mendukung animasi karakter agar terlihat lebih natural dan realistis saat berputar atau berpindah arah.

1.9.9.2 Trigger

Pemain wajib memasukkan perintah hadap ke kanan atau kiri jika ingin mengubah arah karakter menuju tujuan tertentu.

1.9.9.3 Input

Pemain memasukkan perintah hadap kanan atau kiri ke dalam Kotak *Main* atau Kotak *Method*.

1.9.9.4 Output

Sistem menjalankan perintah untuk menghadapkan karakter ke kanan atau kiri sesuai dengan perintah yang ada di Kotak *Main*.

1.9.9.5 Skenario Utama

Pemain ingin mengubah arah karakter ke kanan atau ke kiri sesuai dengan tujuan yang diinginkan.

Prakondisi: Pemain ingin mengubah arah karakter.

Pascakondisi: Pemain memasukkan perintah hadap kanan atau kiri ke Kotak *Main*, dan sistem mengubah arah karakter sesuai perintah tersebut.

Langkah-langkah:

1. Pemain sedang memainkan *level* tertentu.
2. Pemain ingin mengubah arah hadap karakter ke kanan atau ke kiri.
3. Pemain memasukkan perintah hadap kanan atau kiri ke Kotak *Main* atau Kotak *method*.
4. Pemain mengklik tombol *Play*.
5. Sistem menjalankan perintah untuk menghadapkan karakter sesuai dengan arah yang telah dipilih di Kotak *Main*.

1.9.10 Fitur Tombol Ambil

1.9.10.1 Deskripsi

Fitur Tombol Ambil berfungsi untuk memungkinkan karakter dalam permainan mengambil barang yang berada di petak tertentu pada papan permainan. Saat karakter berpindah dan berada di petak yang mengandung barang, fitur ini akan mengaktifkan mekanisme pengambilan sehingga barang tersebut dapat dikumpulkan oleh pemain. Barang yang diambil nantinya dapat berfungsi sebagai poin tambahan, alat bantu, atau elemen penting untuk menyelesaikan *level*. Fitur ini menjadi salah satu aspek penting dalam permainan karena berpengaruh pada penilaian akhir serta kemajuan pemain dalam menyelesaikan tantangan pada setiap *level*.

1.9.10.2 Trigger

Pemain harus mengaktifkan fitur ini ketika ingin mengambil barang yang tersedia di *level* tersebut.

1.9.10.3 Input

Pemain memasukkan perintah ambil dengan memposisikan karakter pada petak yang mengandung barang.

1.9.10.4 Output

Karakter akan mengambil barang yang terdapat pada petak tersebut di *level* yang sedang dimainkan.

1.9.10.5 Skenario Utama

Pemain diharuskan untuk mengambil semua barang yang ada di *level* tersebut dengan cara memasukkan perintah yang tepat dan sesuai, sehingga saat karakter berada di petak yang terdapat barang, lalu karakter dapat mengambil barang tersebut.

Prakondisi: Pemain sudah menyusun perintah dengan benar dan menempatkan karakter pada petak yang terdapat barang di *level* tersebut.

Pascakondisi: Karakter berhasil mengambil barang sesuai dengan perintah yang diberikan dan posisi petak yang mengandung barang.

Langkah-langkah:

1. Pemain berada di dalam *level* tersebut.
2. *Level* tersebut memiliki beberapa barang yang harus diambil.
3. Pemain menyusun perintah secara berurutan, seperti maju, hadap kanan atau kiri, dan mengambil barang.
4. Pemain telah menyusun perintah dengan tepat di Kotak *Main*.
5. Pemain mengklik tombol *Play* untuk menjalankan perintah.
6. Sistem menjalankan animasi karakter sesuai dengan perintah yang sudah disusun di Kotak *Main*.

1.9.11 Fitur Tombol Ulang

1.9.11.1 Deskripsi

Fitur ini berfungsi untuk mengulang perintah yang dimasukkan ke dalam tombol ulang, sehingga memungkinkan perintah tersebut dijalankan berulang kali sesuai dengan jumlah pengulangan yang diinginkan oleh pemain. Fitur ini sangat berguna untuk mempermudah pemain dalam menjalankan perintah yang sama secara berulang tanpa harus memasukkan perintah secara manual setiap kali.

1.9.11.2 Trigger

Fitur tombol ulang dapat digunakan apabila terdapat perintah yang sama dan pemain ingin menjalankannya secara berulang menggunakan perulangan.

1.9.11.3 Input

Pemain memasukkan tombol ulang lalu memasukkan tombol lainnya yang ingin dijalankan secara berulang ke dalam tombol ulang.

1.9.11.4 Output

Sistem menjalankan perintah ulang beserta semua perintah yang terdapat di dalamnya sesuai dengan jumlah pengulangan yang telah ditentukan.

1.9.11.5 Skenario Utama

Pemain menemukan jalur yang mengandung perintah berulang, sehingga pemain menggunakan fitur tombol ulang dengan memasukkan perintah yang sama untuk disusun secara efisien.

Prakondisi: Pemain menemukan jalur yang mengandung perintah berulang dan memasukkan fitur tombol ulang dengan perintah yang sama untuk disusun secara efisien.

Pascakondisi: Sistem menjalankan karakter sesuai dengan perulangan yang telah diatur dan tepat sesuai dengan perintah di Kotak *Main*.

Langkah-langkah:

1. Pemain memasuki *level* dengan tingkat kesulitan yang memerlukan pengulangan perintah;
2. Pemain menyadari bahwa ada perintah yang dapat diulang karena perintah tersebut sama;
3. Pemain memasukkan perintah ulang ke dalam Kotak *Main*;
4. Pemain memasukkan perintah yang ingin diulang ke dalam fitur perintah ulang;
5. Pemain mengklik tombol *Play*;
6. Sistem menjalankan karakter sesuai dengan perintah yang ada di Kotak *Main*.

1.9.12 Fitur Tombol Cabang

1.9.12.1 Deskripsi

Fitur tombol cabang adalah fitur yang wajib digunakan oleh pemain ketika karakter melewati petak berwarna biru jika ingin mendapatkan bintang yang maksimal. Petak berwarna biru menandakan adanya persimpangan jalur atau percabangan yang mengharuskan pemain untuk menentukan arah yang harus diambil oleh karakter agar bisa melanjutkan permainan dengan benar. Dengan menggunakan tombol cabang, pemain dapat mengatur perintah pilihan yang akan dijalankan ketika karakter mencapai petak tersebut, sehingga karakter dapat mengikuti jalur yang tepat menuju tujuan akhir atau menyelesaikan *level* dengan sukses. Fitur ini penting untuk menghindari kesalahan arah dan memastikan jalannya permainan dapat berjalan sesuai strategi yang diinginkan pemain.

1.9.12.2 Trigger

Pemain wajib menggunakan fitur ini saat melewati petak berwarna biru agar dapat memilih jalur yang benar dan berkesempatan mendapatkan bintang maksimal.

1.9.12.3 Input

Pemain memasukkan perintah percabangan dan memasukkan satu tombol lain kedalam tombol percabangan tersebut.

1.9.12.4 Output

Sistem akan memeriksa bahwa perintah yang disusun sudah tepat, sehingga pemain berhak mendapatkan bintang maksimal.

1.9.12.5 Skenario Utama

Pemain ingin memperoleh bintang maksimal pada *level* dengan tingkat kesulitan percabangan. Pada *level* tersebut, terdapat petak berwarna biru yang menandakan adanya percabangan jalur. Sesuai dengan *tutorial* pada tingkat kesulitan ini, pemain harus memberikan perintah cabang pada petak biru agar dapat meraih bintang maksimal.

Prakondisi: Pemain berada pada *level* dengan tingkat kesulitan percabangan, berkeinginan untuk mendapatkan bintang maksimal, dan telah menyusun perintah dengan benar.

Pascakondisi: Sistem memeriksa apakah pemain menggunakan fitur percabangan dengan tepat, memastikan pemain mengambil semua barang yang tersedia di *level* tersebut, dan memberikan bintang maksimal sebagai hasil dari pelaksanaan perintah yang benar.

Langkah-langkah:

1. Pemain berada pada *level* dengan tingkat kesulitan percabangan;
2. Pemain berniat menyelesaikan *level* tersebut dengan hasil sempurna;
3. Pemain menyusun perintah percabangan dan memastikan seluruh barang diambil dengan tepat di Kotak *Main*;
4. Pemain mengklik tombol *Play*;
5. Sistem menjalankan animasi karakter sesuai dengan perintah yang telah disusun;
6. Sistem memberikan bintang maksimal berdasarkan keberhasilan pemain menyelesaikan *level* dengan benar.

1.9.12.6 Skenario Eksepsional 1

Pemain ingin menyelesaikan *level* dengan tingkat kesulitan percabangan tanpa fokus pada perolehan bintang maksimal. Pada *level* ini terdapat petak berwarna biru yang menandakan adanya percabangan jalur. Berdasarkan panduan yang tersedia, penggunaan fitur percabangan pada petak biru diperlukan untuk memperoleh bintang secara maksimal. Namun, pemain dapat memilih untuk pakai atau tidak fitur tersebut.

Prakondisi: Pemain berada pada *level* dengan tingkat kesulitan percabangan, ingin menyelesaikan *level* tanpa mempertimbangkan perolehan bintang maksimal, dan telah menyusun perintah tanpa menyertakan fitur percabangan.

Pascakondisi: Sistem mengevaluasi apakah pemain menggunakan fitur percabangan dengan benar, memverifikasi apakah seluruh barang telah diambil, dan memberikan bintang yang tidak maksimal sebagai konsekuensi dari perintah yang tidak sesuai dengan kondisi *level*.

Langkah-langkah:

1. Pemain berada pada *level* dengan tingkat kesulitan percabangan;
2. Pemain berniat menyelesaikan *level* tersebut tanpa fitur percabangan;
3. Pemain menyusun perintah tanpa fitur percabangan dan memastikan seluruh barang diambil dengan tepat di Kotak *Main*;
4. Pemain mengklik tombol *Play*;
5. Sistem menjalankan animasi karakter sesuai dengan perintah yang telah disusun;
6. Sistem memberikan bintang yang tidak maksimal karena perintah yang disusun pemain tidak sepenuhnya sesuai untuk menyelesaikan *level* dengan benar.

1.9.13 Fitur *Method*

1.9.13.1 Deskripsi

Fitur *Method* berfungsi sebagai wadah untuk menampung serangkaian perintah yang telah disusun oleh pemain. Ketika tombol *Method* dipanggil atau digunakan di Kotak *Main*, sistem akan menjalankan seluruh perintah yang ada di Kotak *Method* secara berurutan hingga selesai. Fitur ini sangat berguna untuk menyederhanakan perintah yang berulang atau kompleks agar lebih efisien dan mudah dikelola. Dengan menggunakan *Method*, pemain dapat menghemat ruang di Kotak *Main* serta mempermudah pengaturan logika pergerakan karakter dalam menyelesaikan *level* permainan.

1.9.13.2 Trigger

Fitur *Method* digunakan ketika pemain ingin meringkas atau menyederhanakan perintah yang panjang agar Kotak *Main* tidak penuh dan tetap mudah dibaca.

1.9.13.3 Input

Pemain memasukkan serangkaian perintah ke dalam Kotak *Method*, lalu memanggil atau menempatkan perintah *Method* tersebut di Kotak *Main*.

1.9.13.4 Output

Sistem akan menjalankan perintah yang ada di Kotak *Main* secara berurutan, termasuk perintah-perintah yang terdapat di dalam *Method*, sesuai urutan pemanggilan.

1.9.13.5 Skenario Utama

Pemain berada pada *level* dengan tingkat kesulitan *Method*. Dalam *level* ini, pemain diwajibkan menggunakan fitur *Method* karena keterbatasan ruang pada Kotak *Main*. Pemain harus menyusun perintah di Kotak *Method* dan memanggilnya di Kotak *Main* agar seluruh perintah dapat dijalankan dengan efisien.

Prakondisi: Pemain berada di *level* dengan tingkat kesulitan *Method*. Pemain telah menyusun perintah-perintah yang diperlukan di Kotak *Method* dan memanggil *Method* tersebut di Kotak *Main* sesuai urutan yang benar.

Pascakondisi: Sistem menjalankan seluruh perintah sesuai urutan yang ada di Kotak *Main*, termasuk perintah-perintah yang dipanggil melalui *Method*.

Langkah-langkah:

1. Pemain berada pada *level* dengan tingkat kesulitan *Method*;
2. Pemain diminta untuk menyusun beberapa perintah di Kotak *Method*;
3. Pemain memasukkan perintah *Method* dan perintah lainnya secara tepat di Kotak *Main*;
4. Pemain menekan tombol *Play* untuk menjalankan perintah;
5. Sistem menjalankan karakter sesuai dengan urutan perintah yang ada di Kotak *Main*, termasuk perintah yang dipanggil dari *Method*.

1.9.14 Fitur Ulang Permainan

1.9.14.1 Deskripsi

Fitur Ulang Permainan berfungsi sebagai tombol untuk mengulang *level* yang telah diselesaikan oleh pemain. Setelah pemain berhasil menyelesaikan suatu *level*, sistem akan menampilkan tampilan akhir berupa ucapan selamat, jumlah bintang yang diperoleh, serta beberapa tombol navigasi seperti Ulang dan Lanjut. Dengan menekan tombol Ulang, pemain dapat memutar ulang *level* terakhir yang dimainkan, baik untuk mencoba strategi baru, memperbaiki kesalahan, maupun meningkatkan perolehan bintang.

1.9.14.2 Trigger

Fitur Ulang Permainan dipicu ketika pemain ingin mengulang permainan pada *level* yang baru saja diselesaikan, baik untuk memperbaiki kesalahan, mencoba strategi berbeda, maupun meningkatkan hasil permainan.

1.9.14.3 Input

Pemain mengklik tombol Ulang yang muncul setelah menyelesaikan *level*.

1.9.14.4 Output

Sistem akan memuat ulang *level* yang baru saja dimainkan dan mengatur ulang kondisi permainan seperti semula, memungkinkan pemain untuk memainkannya kembali dari awal.

1.9.14.5 Skenario Utama

Pemain telah berhasil menyelesaikan sebuah *level*. Setelah itu, sistem menampilkan pesan ucapan selamat, jumlah bintang yang diperoleh, serta dua tombol yaitu Ulang dan Lanjut. Pemain memilih untuk mengulang permainan di *level* yang baru saja diselesaikan dengan menekan tombol Lanjut.

Prakondisi: Pemain telah menyelesaikan satu *level*, menerima pesan ucapan selamat beserta jumlah bintang, dan disajikan pilihan tombol Ulang dan Lanjut. Pemain kemudian memilih untuk menekan tombol Ulang.

Pascakondisi: Sistem memuat ulang *level* yang sama, dan pemain dapat memainkan ulang *level* tersebut dari awal.

Langkah-langkah:

1. Pemain berhasil menyelesaikan suatu *level* dalam permainan;
2. Sistem menampilkan pesan ucapan selamat, jumlah bintang yang diperoleh, serta dua tombol pilihan: Ulang dan Lanjut;
3. Pemain memilih tombol Ulang untuk mengulang *level*;
4. Sistem memulai ulang permainan pada *level* yang sama, dan pemain kembali memainkan *level* tersebut dari awal.

1.9.15 Fitur Lanjut Permainan

1.9.15.1 Deskripsi

Fitur Lanjut Permainan merupakan tombol Lanjut yang berfungsi untuk melanjutkan permainan ke *level* berikutnya setelah pemain berhasil menyelesaikan sebuah *level*. Setelah suatu *level* diselesaikan, sistem akan menampilkan tampilan akhir yang mencakup ucapan selamat, jumlah bintang yang diperoleh, serta tombol-tombol navigasi seperti Ulang dan Lanjut. Dengan menekan tombol Lanjut, pemain akan diarahkan untuk memainkan *level* selanjutnya dalam urutan permainan.

1.9.15.2 Trigger

Fitur Lanjut Permainan akan terpicu ketika pemain memilih untuk melanjutkan permainan ke *level* berikutnya setelah berhasil menyelesaikan *level* sebelumnya.

1.9.15.3 Input

Pemain mengklik tombol Lanjut yang muncul setelah menyelesaikan *level*.

1.9.15.4 Output

Sistem akan memuat *level* berikutnya dan menampilkan tampilan awal *level* tersebut, sehingga pemain dapat mulai memainkannya dari awal.

1.9.15.5 Skenario Utama

Pemain telah berhasil menyelesaikan sebuah *level*. Setelah itu, sistem menampilkan pesan ucapan selamat, jumlah bintang yang diperoleh, serta dua tombol yaitu Ulang dan Lanjut. Pemain memilih untuk melanjutkan ke *level* selanjutnya dengan menekan tombol Lanjut.

Prakondisi: Pemain telah menyelesaikan satu *level*, menerima pesan ucapan selamat beserta jumlah bintang, dan disajikan pilihan tombol Ulang dan Lanjut. Pemain kemudian memilih untuk menekan tombol Lanjut.

Pascakondisi: Sistem memuat *level* yang baru, dan pemain dapat langsung memainkan *level* tersebut dari awal.

Langkah-langkah:

1. Pemain berhasil menyelesaikan suatu *level* dalam permainan;
2. Sistem menampilkan pesan ucapan selamat, jumlah bintang yang diperoleh, serta dua tombol pilihan: Ulang dan Lanjut;
3. Pemain memilih tombol Lanjut untuk melanjutkan ke *level* selanjutnya;
4. Sistem memuat permainan di *level* yang baru.

1.9.16 Fitur Tombol Kembali

1.9.16.1 Deskripsi

Fitur Tombol Kembali berfungsi sebagai navigasi untuk memudahkan pemain kembali ke halaman sebelumnya dalam gim, seperti dari halaman *level* ke halaman utama atau dari pengaturan ke menu awal. Fitur ini biasanya ditampilkan dalam bentuk ikon panah dan ikon rumah yang mudah dikenali dan diakses. Dengan fitur ini, pemain dapat berpindah antarhalaman tanpa kehilangan progres atau harus memulai ulang dari awal. Kehadiran fitur ini memberikan pengalaman bermain yang lebih fleksibel dan nyaman, serta meminimalkan kebingungan saat menjelajahi antarmuka gim.

1.9.16.2 Trigger

Fitur ini dipicu ketika pemain ingin melakukan navigasi kembali ke halaman sebelumnya.

1.9.16.3 Input

Pemain mengklik tombol panah ke kiri atau ikon rumah yang tersedia di tampilan.

1.9.16.4 Output

Sistem akan menampilkan halaman sebelumnya sesuai dengan lokasi terakhir pemain berada.

1.9.16.5 Skenario Utama

Pemain ingin kembali ke halaman sebelumnya.

Prakondisi: Pemain berada di suatu halaman dan ingin kembali ke halaman sebelumnya; pemain kemudian mengklik tombol navigasi kembali.

Pascakondisi: Sistem menampilkan halaman sebelumnya, dan pemain berhasil kembali ke halaman tersebut.

Langkah-langkah:

1. Pemain berada di suatu halaman;
2. Pemain ingin kembali ke halaman sebelumnya;
3. Pemain mengklik tombol navigasi kembali;
4. Pemain berhasil kembali ke halaman sebelumnya.

1.9.17 Fitur Geser Halaman *Level*

1.9.17.1 Deskripsi

Fitur Geser Halaman *level* berfungsi untuk menggeser halaman pemilihan *level* berdasarkan tingkat kesulitannya. Terdapat beberapa tingkat kesulitan, yaitu: tingkat kesulitan Dasar-Pergerakan, Perulangan-Kode Berulang, Percabangan-Pilihan Jalan, dan Method-Panggil Bantuan. Pemain dapat menggunakan fitur ini untuk berpindah antar halaman *level* sesuai kategori tingkat kesulitan tersebut.

1.9.17.2 Trigger

Fitur Geser Halaman *level* dipicu ketika pemain ingin melihat atau berpindah ke halaman *level* dengan tingkat kesulitan yang berbeda.

1.9.17.3 Input

Pemain mengklik tombol panah kanan atau kiri pada halaman pemilihan *level*.

1.9.17.4 Output

Sistem menampilkan halaman *level* berikutnya atau sebelumnya sesuai arah tombol yang diklik, dengan tingkat kesulitan yang berbeda.

1.9.17.5 Skenario Utama

Pemain berada di halaman pemilihan *level* dan ingin memainkan *level* dengan tingkat kesulitan tertentu.

Prakondisi: Pemain sudah berada di halaman pemilihan *level*, Pemain berniat untuk mengganti tingkat kesulitan, dan Pemain mengklik tombol panah kanan atau kiri untuk menggeser halaman dan memilih tingkat kesulitan yang diinginkan.

Pascakondisi: Tingkat kesulitan pada halaman berpindah sesuai dengan arah geseran dan ditampilkan secara otomatis.

Langkah-langkah:

1. Pemain berada di halaman pemilihan *level*;
2. Pemain ingin memainkan *level* dengan tingkat kesulitan tertentu;
3. Pemain mengklik tombol kanan atau kiri untuk menggeser halaman dan memilih tingkat kesulitan yang diinginkan;
4. Tingkat kesulitan pada halaman berpindah sesuai dengan arah geseran dan ditampilkan secara otomatis.

1.9.18 Fitur Pengaturan Suara

1.9.18.1 Deskripsi

Fitur pengaturan suara ini berfungsi untuk memberikan kontrol kepada pemain dalam mengelola suara latar (*background*) yang ada di dalam gim. Dengan fitur ini, pemain dapat memilih untuk mengaktifkan suara latar agar suasana permainan menjadi lebih hidup dan menarik, atau menonaktifkannya jika menginginkan suasana yang lebih tenang atau sedang dalam kondisi yang tidak memungkinkan untuk mendengarkan suara. Pengaturan ini dapat diakses kapan saja selama

permainan berlangsung sehingga pemain dapat dengan mudah menyesuaikan pengalaman bermain sesuai preferensi mereka.

1.9.18.2 Trigger

Fitur pengaturan suara akan aktif ketika pemain ingin menghidupkan atau mematikan suara latar dalam permainan.

1.9.18.3 Input

Pemain mengklik tombol dengan ikon suara (lambang speaker) pada antarmuka permainan.

1.9.18.4 Output

Sistem akan mengaktifkan suara latar jika sebelumnya dalam keadaan mati, atau menonaktifkan suara latar jika sebelumnya dalam keadaan aktif.

1.9.18.5 Skenario Utama

Pemain ingin menonaktifkan suara latar karena dianggap mengganggu kenyamanan saat bermain.

Prakondisi: Pemain merasa terganggu oleh suara latar gim dan memutuskan untuk menonaktifkannya. Pemain mengklik tombol dengan ikon speaker yang terdapat di halaman beranda.

Pascakondisi: Suara latar gim berhasil dinonaktifkan, sehingga tidak terdengar lagi selama permainan berlangsung.

Langkah-langkah:

1. Pemain berada di halaman beranda gim;
2. Pemain merasa terganggu dengan suara latar yang menyala selama berada di halaman tersebut;
3. Pemain mengklik tombol dengan ikon speaker untuk menonaktifkan suara;

4. Suara latar berhasil dimatikan dan tidak terdengar lagi.

1.9.18.6 Skenario Eksepsional 1

Pemain ingin mengaktifkan kembali suara latar karena merasa permainan menjadi lebih menyenangkan dengan adanya suara tersebut.

Prakondisi: Pemain merasa permainan akan lebih menarik jika terdapat suara; Pemain memutuskan untuk mengaktifkan kembali suara; Pemain mengklik tombol dengan ikon speaker yang disilang di halaman beranda.

Pascakondisi: Suara latar gim berhasil diaktifkan kembali dan akan terdengar selama permainan berlangsung.

Langkah-langkah:

1. Pemain berada di halaman beranda gim;
2. Pemain merasa ingin mendengar suara kembali;
3. Pemain mengklik tombol dengan ikon speaker yang disilang untuk mengaktifkan suara;
4. Suara latar berhasil diaktifkan dan kembali terdengar selama permainan.

1.10 Kebutuhan Nonfungsional

1.10.1 Atribut Kualitas

Gim CodeLogicCraft 2.0 memiliki atribut kualitas sebagai berikut:

1. Keandalan (*Reliability*)

Gim CodeLogicCraft 2.0 didesain agar menyenangkan dan tidak membosankan. Gim ini membawa pengguna ke dalam dunia bertema "terdampar di pulau terpencil", di mana pemain harus mencari cara untuk bertahan hidup dan keluar dari pulau tersebut. Dalam perjalanan, pemain akan menghadapi berbagai rintangan berbasis logika pemrograman dan harus mengumpulkan barang penting untuk bertahan hidup, sesuai dengan kebutuhan karakter dalam gim. CodeLogicCraft 2.0 juga menghadirkan

alur cerita yang menarik melalui animasi dan efek suara yang menyertai setiap aktivitas karakter, sehingga menciptakan pengalaman bermain yang lebih realistis dan *imersif*, seolah-olah pemain benar-benar berada di dalam dunia gim tersebut.

2. Kegunaan (*Usability*)

Gim CodeLogicCraft 2.0 dirancang untuk digunakan oleh masyarakat umum sebagai media pembelajaran logika pemrograman. Banyak individu menghadapi kesulitan dalam mempelajari pemrograman karena kurang memahami prinsip dasar logika yang mendasarinya. Gim ini hadir untuk membantu mengatasi permasalahan tersebut melalui pendekatan yang interaktif dan menarik.

3. Pemeliharaan (*Maintainability*)

Gim CodeLogicCraft 2.0 tidak memerlukan pemeliharaan rutin karena dirancang untuk berjalan secara mandiri (*standalone*) tanpa ketergantungan pada sistem eksternal tambahan.

1.10.2 Kebutuhan Legal

Gim YARSI Students Simulation memiliki aspek legal seperti berikut:

1. Penggunaan Perangkat Lunak

Gim CodeLogicCraft 2.0 merupakan perangkat lunak yang dirancang untuk digunakan secara umum sebagai alat pengenalan dalam pembelajaran logika pemrograman.

2. Penggunaan Aset

Dalam pengembangan gim CodeLogicCraft 2.0, aset yang digunakan terdiri dari aset gratis dan aset berlisensi. Aset berlisensi tersebut adalah milik Tim Marbay dan tidak boleh didistribusikan tanpa izin.

1.10.3 Kebutuhan Pengembangan

Gim CodeLogicCraft 2.0 memiliki aspek pengembangan seperti berikut:

1. *Refactoring*

Gim CodeLogicCraft 2.0 memiliki skrip dan kode yang menjadi dasar sistem gim saat ini. Tim menganggap perlu dilakukan *refactoring* untuk meningkatkan performa gim. *Refactoring* meliputi modifikasi, penambahan, dan perubahan skrip agar sesuai dengan fitur dan tujuan dari Gim CodeLogicCraft versi 1.0.

2. *Debugging*

Pengujian gim CodeLogicCraft 2.0 dilakukan guna memastikan seluruh fitur dan sistem berfungsi dengan baik. Proses perbaikan kesalahan pada skrip atau kode merupakan bagian dari kegiatan *debugging* yang akan dilaksanakan.

3. Dokumentasi

Proses pengembangan dilakukan dengan mendokumentasikan setiap penambahan, perubahan, pengurangan, maupun perbaikan fitur dalam gim. Dokumentasi yang dihasilkan meliputi dokumen pengembangan, panduan pengguna, serta dokumen yang memuat skrip dan kode program.

4. *Splash Screen*

Fitur splash screen adalah fitur dasar yang disediakan oleh Unity, berfungsi sebagai tampilan pembuka yang menampilkan informasi mengenai pihak-pihak yang terlibat dalam pengembangan gim. Tim pengembang akan membuat splash screen yang memuat identitas dan informasi terkait pihak-pihak tersebut dalam gim CodeLogicCraft 2.0.