**BAB 1**

**PENDAHULUAN**

Bab ini menjelaskan latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, ruang lingkup penelitian, dan sistematika penulisan.

**1.1 Latar Belakang**

Industri *game* video telah berkembang sejak tahun 1970-an. Pada masa itu, biaya pengembangan dan kapasitas komputer masih rendah, sehingga *game* dapat dibuat oleh seorang *programmer*.

Memasuki era modern, industri *game* telah berkembang pesat, dari produksi berbiaya rendah dengan ukuran tim pengembang yang kecil, hingga produksi berbiaya jutaan *dollar* dengan tim yang beranggotakan ratusan orang. Proyek *game* berukuran besar melibatkan banyak orang dengan berbagai peran dan spesifikasi pekerjaan, termasuk para *programmer*. Sebuah *game* AAA, atau *game* yang diproduksi dengan biaya dan promosi yang sangat tinggi, biasanya melibatkan banyak sekali *programmer*.

Beban kerja pengembang *game* sering kali ditentukan oleh genre dari *game* yang sedang dibuat. Salah satu *genre* yang paling berat untuk dikembangkan adalah genre *role-playing game*, melihat karakteristik dari genre tersebut yang biasanya memiliki banyak sekali aspek dan fitur yang berbeda.

Banyaknya *code* yang dikerjakan dan *programmer* yang berkontribusi sering kali menyebabkan *project hierarchy* tidak beraturan. Berbagai masalah sering dihadapi ketika berursan dengan *source code* yang besar, contohnya adalah ketika membuat sebuah perubahan pada *code* yang sudah ada. Mengubah *source code* yang hanya terdiri dari ratusan baris dan beberapa *file* mungkin bukan perkara sulit, tetapi ketika *code* yang dihadapi berjumlah puluhan hingga ratusan *file*, dengan tiap *file* memiliki hingga ribuan baris *code*, tentu akan menjadi sebuah pekerjaan yang rumit dan melelahkan. Banyak risiko yang dapat terjadi ketika mengubah *code*, salah satunya adalah kemungkinan bahwa terdapat *method* atau *class* yang gagal berfungsi karena perubahan yang dilakukan. Masalah akan menjadi semakin sulit untuk diatasi ketika *method* atau *class* tersebut berada dalam *file* yang berbeda dengan *file* yang diubah.

Dari kesulitan-kesulitan tersebut, dibutuhkan sebuah sistem yang mampu mengurangi beban kerja *programmer* dengan membuat struktur *code* yang lebih terorganisasi, sehingga perubahan pada *code* dapat dilakukan dengan mudah dan lancar. *Software product line*, atau SPL, muncul sebagai sebuah paradigma *development* yang didesain untuk mengoptimalkan waktu pemasaran, biaya, produktivitas, kualitas, dan aspek *development* lainnya. Penerapan SPL dalam pembuatan *game* dapat membantu memudahkan pengerjaan dengan menekankan struktur *code* yang baik, sehingga dapat mengurangi waktu produksi dan biaya yang disebabkan oleh proses pengubahan *code*.

**1.2 Rumusan Masalah**

Terdapat 2 rumusan masalah yang dapat dijawab oleh penelitian ini, yaitu:

1. Bagaimana contoh implementasi SPL untuk pengembangan game?

Alt: Bagaimana SPL dapat diimplementasikan untuk pengembangan game?

1. Apakah bentuk implementasi tersebut lebih baik dibandingkan dengan paradigma pengembangan yang asli?

**1.3 Tujuan Penelitian**

Berdasarkan pertanyaan-pertanyaan rumusan masalah, terdapat beberapa tujuan yang diharapkan dapat diraih oleh penelitian ini, yaitu:

1. Untuk mengetahui salah satu bentuk/contoh metode implementasi SPL dalam pengembangan game.
2. Untuk menganalisa kelebihan dan kekurangan metode yang diteliti.

**1.4 Manfaat Penelitian**

Penelitian ini diharapkan dapat menjadi acuan bagi penelitian-penelitian yang akan datang berikutnya baik dalam bidang SPL maupun pengembangan game, serta membantu menerapkan sebuah standar dalam proses pengembangan *game*.

**1.5 Ruang Lingkup Penelitian**

Lingkup penelitian: Domain game

Lingkup eksperimen: penambahan sebuah entity baru ke dalam game ,membuat monster baru dalam game DCSS

**1.6 Metodologi Penelitian**

Terdapat beberapa tahapan penelitian yang dilakukan dalam skripsi ini, yaitu:

1. Tinjauan Literatur

Dalam tahapan ini, penelitian-penelitian terkait dengan penelitian yang dilakukan dalam skripsi ini dipelajari dan dianalisis untuk dijadikan sebagai acuan dalam eksperimen.

1. Analisis Studi Kasus

Untuk mendalami lebih lanjut mengenai *game* studi kasus serta fitur/task yang akan diuji, dilakukan analisis terhadap berbagai aspek dari *game* studi kasus.

1. Merumuskan ide dan langkah eksperimen

Setelah melakukan tinjauan literatur dan analisis studi kasus, dirumuskan ide dan langkah-langkah yang akan dilakukan untuk eksperimen yang akan dilakukan.

1. Melakukan eksperimen

Dalam tahap ini, dilakukan eksperimen untuk menganalisa metodologi yang diteliti

1. Analisis hasil eksperimen

Dari eksperimen yang dilakukan, hasil yang didapat dicatat dan dianalisis mengenai kesesuaiannya dengan rumusan masalah, tinjauan literatur, serta analisis studi kasus.

1. Menarik kesimpulan

Dalam tahapan terakhir, yang dilakukan adalah menilai hasil dari eksperimen dan membuat kesimpulan sesuai dengan rumusan masalah.

**1.7 Outline**

Skripsi ini disusun menjadi lima bab dengan sistematika penulisan sebagai berikut:

* Bab 1 Pendahuluan

Bab ini menjelaskan latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, ruang lingkup penelitian, metodologi penelitian dan sistematika penulisan.

* Bab 2 Tinjauan Literatur

Bab ini berisi tinjauan literatur yang digunakan sebagai acuan dari penelitian.

* Bab 3 Studi Kasus

Bab ini mendeskripsikan *game* serta fitur/task yang diacu sebagai studi kasus untuk di-*refactor* dalam penelitian.

* Bab 4 Rancangan Eksperimen

Bab ini menjelaskan rincian eksperimen yang dilakukan, beserta penjelasan dasar-dasar dari langkah yang dilakukan.

* Bab 5 Eksperimen dan Simulasi

Bab ini memaparkan proses eksperimen yang dilakukan serta mencatat hasil yang didapat.

* Bab 6 Analisis

Bab ini berisi analisis terkait hasil eksperimen yang didapat dari bab sebelumnya.

* Bab 7 Kesimpulan

Bab ini berisi kesimpulan yang dapat ditarik dari hasil eksperimen dan analisis dalam penelitian, serta saran untuk penelitian selanjutnya.