

UNIVERSIDADE DE SÃO PAULO
INSTITUTO DE MATEMÁTICA E ESTATÍSTICA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

**Desenvolvimento de um Programa
de Construção, Simulação e
Geração de Montanhas-Russas
Virtuais Voltado para Jogos Digitais**

César Gasparini Fernandes

PROPOSTA APRESENTADA À DISCIPLINA

MAC0499 — TRABALHO DE
FORMATURA SUPERVISIONADO

Supervisor: Prof. Dr. Marcel P. Jackowski

São Paulo
16 de Abril de 2021

1. Introdução

As montanhas-russas são as principais atrações dos parques de diversão. Jogos de gerenciamento e simulação de parques de diversão, como RollerCoaster Tycoon[®] 2 e Planet Coaster[®], permitem que o jogador construa e simule sua própria montanha-russa virtual. Tais jogos disponibilizam uma série de objetos e ferramentas para o usuário decorar o terreno ao redor de suas montanhas-russas para que ele possa personalizá-la ao máximo possível.

Entretanto, nenhum jogo famoso de construção e simulação de parques de diversão apresenta a possibilidade de gerar automaticamente uma montanha-russa dados alguns parâmetros de controle; apenas disponibilizam montanhas-russas pré-prontas para o jogador montar o seu parque. Tal ferramenta melhoraria a experiência do jogador e encurtaria o tempo entre a construção e a simulação, aumentando o divertimento.

Portanto, este projeto surge do desafio de desenvolver um gerador e simulador de montanhas-russas virtuais voltados a jogos digitais. Assim como os construtores e simuladores de montanhas-russas dos principais jogos de parques de diversão, este trabalho não tem o escopo de modelar ou simular montanhas-russas reais. No entanto, a geração automática de trechos ao longo do trajeto, respeitando os limites físicos, e grau de diversão prescritos, podem até sugerir futuros modelos de montanhas-russas reais.

2. Objetivo

Desenvolver uma aplicação que permita ao usuário construir ou gerar proceduralmente sua própria montanha-russa virtual além de simular um carro que tráfegará ao longo do trajeto dadas as forças físicas atuantes. Adicionalmente, o software também permitirá a edição do relevo do solo e demais decorações do parque, como árvores e rochas.

3. Metodologia

O programa será desenvolvido utilizando a *engine* Unity[®] por apresentar o pipeline de renderização pronto e por ser comumente empregado em desenvolvimento de jogos no mercado. Porém, para o desenvolvimento das partes principais do projeto, que são o construtor, o simulador e o gerador, será utilizado apenas o básico de suas bibliotecas, como os objetos “Vector3”, “Matrix4x4”, “Mesh”, entre outros. Já para a interface gráfica, a biblioteca de interface gráfica do Unity[®] será utilizada massivamente, dado que, para ajustar

as posições, animações e eventos dos elementos da interface (como botões e painéis), é quase imprescindível utilizar a biblioteca.

Para o desenvolvimento do construtor de montanhas-russas, será modelado um corte transversal de tipos de trilhos de montanhas-russas e, em seguida, esses modelos 2D serão extrusados ao longo de Curvas de Bézier cúbicas definidas pelo jogador através da interface gráfica, o qual poderá ajustar a elevação, a rotação, a inclinação e o tamanho do segmento de trilho que está construindo. Estas curvas serão “ligadas” de maneira que seus pontos de controle tornem as ligações de cada segmento de classe infinita, tornando a curva inteira contínua em todas as suas derivadas. Também serão desenvolvidas algumas ferramentas como a de autocompletar a montanha-russa, a de chegar se os trilhos estão se intersectando. Por fim, serão disponibilizados trechos pré-prontos de trilhos ao usuário para facilitar a construção.

Para o desenvolvimento do simulador dos vagões, que é a simulação dos vagões, serão calculadas as diferentes forças aplicadas ao vagão de acordo com o tipo de trilho que ele está. Por exemplo, em um trilho do tipo “normal”, serão aplicadas a força da gravidade e do atrito ao vagão, já em um trilho do tipo “alavanca”, será aplicada, também, a força da alavanca. A simulação do carro no trilho será realizada utilizando o Método de Runge-Kutta de quarto grau para aproximar sua velocidade entre dois momentos curtos distintos (quadros). Também serão pré-computados mapas de calor da velocidade, da força-g e da altura dos vagões ao longo do trilho a fim de auxiliar o jogador na construção de sua montanha-russa.

O gerador de montanhas-russas gerará os trilhos proceduralmente, se baseando em um *L-System* estocástico paramétrico. A cada trilho, ele analisará as propriedades atuais, como altura e velocidade atuais, e sorteará um trecho possível, como *looping* ou queda, segundo as probabilidades impostas pela gramática. Inicialmente, o gerador deverá seguir as seguintes regras para gerar a montanha-russa:

1. Os trilhos não podem se intersectar;
2. O percurso deve ser circular, ou seja, a montanha-russa deve “fechar”;
3. O carro deve completar o percurso.

A imposição destas três regras citadas acima, e o fato de não se ter um espaço discreto no problema impossibilita a utilização dos algoritmos comumente utilizados em geração procedural, logo, deverá ser desenvolvida um algoritmo especial para esta condição.

Para o desenvolvimento da ferramenta de edição de terreno, será gerada uma malha para o terreno passível de deformações pelo usuário para poder criar aclives e declives. Também será desenvolvida uma ferramenta que possibilite ao jogador colocar alguns objetos, como árvores e rochas, ao longo do terreno. Esses objetos serão modelos de terceiros (preferencialmente de domínio público).

A interface gráfica será prototipada utilizando o programa diagrams.net[®] e, em seguida, será desenvolvida utilizando a biblioteca de interface gráfica do Unity[®]. Na interface, estarão disponíveis informações estruturais do segmento de trilho atual, como elevação, rotação, inclinação e tamanho, informações físicas dos vagões no trilho, como velocidade e força-g, elementos que auxiliem o usuário na construção, como setas para rotacionar o trilho e botões que permitam que o jogador possa acessar os recursos do programa, como salvar, carregar, construir e gerar uma montanha-russa. O protótipo da interface gráfica do construtor manual da montanha-russa pode ser vista na figura 1.

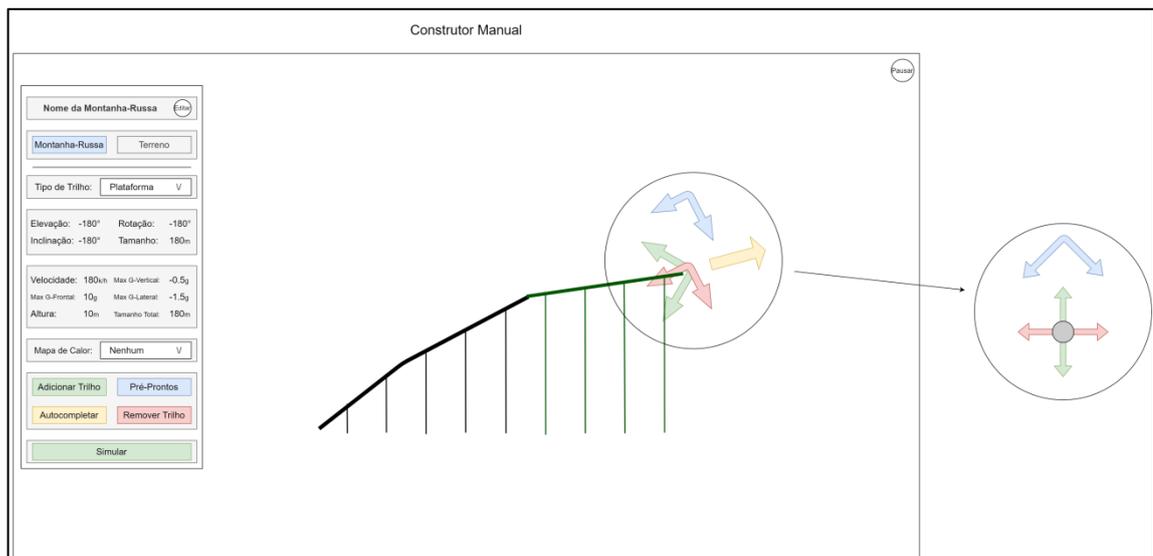


Figura 1: Protótipo da interface gráfica do construtor manual da montanha-russa.

4. Cronograma

O cronograma do desenvolvimento do projeto foi dividido nas seguintes etapas:

1. Revisão Bibliográfica;
2. Cálculos algébricos necessários para desenvolver o construtor e o simulador;
3. Modelagem de cortes transversais de trilhos;
4. Desenvolvimento da base construtor;
5. Desenvolvimento da base do simulador;

6. Desenvolvimento da interface gráfica;
7. Pesquisas de experiência do usuário;
8. Finalização do construtor;
9. Finalização do simulador;
10. Desenvolvimento do gerador;
11. Desenvolvimento da ferramenta de edição do terreno;
12. Adição de áudio na interface;
13. Aprimoramento da interface e do jogo;
14. Escrita da monografia.

As etapas enumeradas acima tem expectativa de serem realizadas nos períodos descritos pela tabela 1.

Etapa	Abr.	Mai.	Jun.	Jul.	Ago.	Set.	Out.	Nov.	Dez.
1 – Revisão Bibliográfica	X	X	X	X	X	X	X	X	X
2 - Cálculos	X	X		X					
3 - Modelagem	X	X							
4 – Base do Construtor		X	X						
5 – Base do Simulador		X							
6 - Interface			X						
7 – Pesquisa de Experiência			X				X		
8 – Finalização do Construtor				X					
9 - Finalização do Simulador				X					
10 - Gerador				X	X				
11 - Terreno						X			
12 - Áudio							X		
13 - Aprimoramentos							X		
14 - Monografia							X	X	X

Tabela 1: Cronograma das etapas do desenvolvimento deste projeto no ano de 2021.