

# Cai Cai, Passarinho

**Edwino Alberto Lopes Stein<sup>1</sup>, Rafael Sá Menezes<sup>1</sup>, Rodrigo dos Santos Tavares<sup>1</sup>,  
Luciano Ferreira Silva<sup>1</sup>**

<sup>1</sup>Departamento de Ciência da Computação – Centro de Ciência e Tecnologia  
Universidade Federal de Roraima (UFRR)  
Boa Vista – RR – Brazil

{edwino.stein, rafael.sa, rodrigo.tavares}@ufrr.br, fsluciano.ufrr@gmail.com

## 1. Introdução

A computação gráfica é uma das principais áreas da computação, tendo sido muito desenvolvida nas últimas décadas. Com a evolução dela, a área de desenvolvimento de jogos ganhou força, sendo diretamente ligada a evolução da computação gráfica.

Este trabalho irá descrever sobre o jogo *Cai Cai, Passarinho*, desde seu roteiro, mecânicas, artes, até seu desenvolvimento e técnicas utilizadas.

## 2. Sobre o Jogo

O jogo faz uma crítica ao desmatamento, mostrando como uma espécie em extinção pode sobreviver até um certo ponto, mas sempre será extinta, sendo a única maneira de realmente acabar com isso acabar com a fonte do problema (nesse caso, o desmatamento).

### 2.1. Roteiro

O jogo gira em torno do Passarinho que está fugindo da Máquina Engolidora de Árvores.

### 2.2. Mecânicas do Jogo

O jogador deverá mover o Passarinho para os lados, utilizando o acelerômetro, para que ele possa subir de galho em galho e fugir da Máquina Engolidora de Árvores. O objetivo é sobreviver, chegando às alturas mais elevadas. O jogo acaba quando o Passarinho (jogador) não for rápido o suficiente para fugir da Máquina Engolidora de Árvores. Ao acabar, será mostrada a pontuação do jogador e seu recorde atual. Conforme o jogador atinge certos níveis de altura, o nível de dificuldade aumenta, deixando o algoritmo de gerar plataformas com mais peso para plataformas menores e mais distantes.

## 3. Arte Gráfica

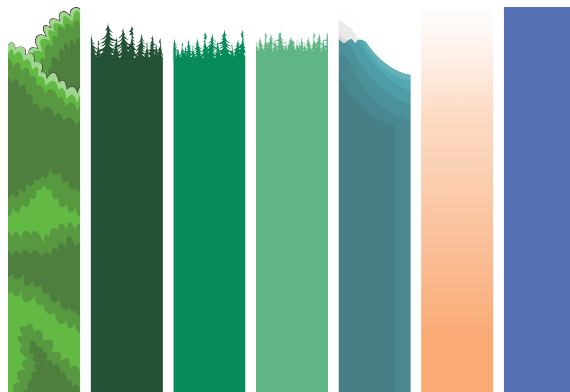
### 3.1. Cenário

O cenário escolhido foi de uma floresta, de dia e sem chuva. A figura 1 mostra os sprites utilizados, os quais representam diversos fundos se movimentando em velocidades diferentes, criando assim um efeito de profundidade.

A figura 2 mostra o sprite do tronco da árvore, o qual fica se replicando conforme o jogador atinge novas alturas.

A figura 3 mostra o sprite do galho, que sai do tronco e se alinha automaticamente a alguma plataforma.

A figura 4 mostra o sprite da plataforma, que pode ser dividida em três tamanhos: pequeno (1 unidade), médio (2 unidades) e grande (3 unidades).



**Figura 1. Background**



**Figura 2. Tronco da árvore**

### **3.2. Personagens**

O personagem principal do jogo é ilustrado pela figura 5. A figura 6 exibe o sprite da Máquina Engolidora de Árvores. A figura 7 mostra o sprite da serra da máquina.

## **4. Desenvolvimento**

### **4.1. Ferramentas de Desenvolvimento**

As ferramentas utilizadas durante o desenvolvimento foram:

**Unity:** Para desenvolvimento do jogo.

**Corel Draw:** Para criação dos sprites.

### **4.2. Pontuação**

O jogo utiliza um sistema de pontuação que conta os pontos baseado na altura máxima atingida. Ao final da partida, o jogo exibe para o jogador a maior pontuação já obtida e sua pontuação atual.

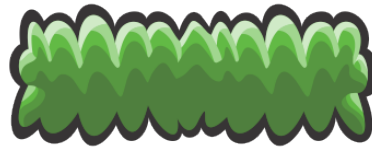
### **4.3. Técnicas Utilizadas**

#### **4.3.1. Gerador de plataformas**

Para geração das plataformas, foi utilizado um algoritmo próprio que funciona da seguinte forma:



**Figura 3. Galho da árvore**



**Figura 4. Plataforma**

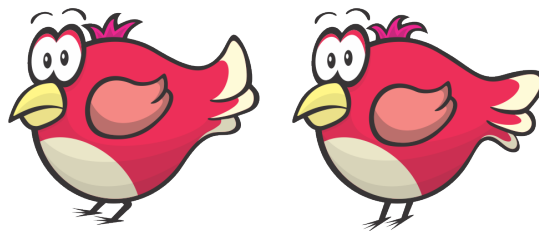
1. Separa o cenário em 14 colunas de tamanhos iguais;
2. Define a regra para criação de cada linha de plataforma;
3. Ao iniciar o jogo, são geradas algumas plataformas próximas ao jogador, de forma que garanta que ele morra no início do jogo.
4. Ao começar a subir, é inicializado o algoritmo de geração.
5. A cada linha, as plataformas são geradas baseadas no nível atual (que define possibilidade de geração, distância e tamanho).
6. Ao gerar as plataformas, verifica se o jogador consegue alcançar as próximas plataformas (baseando-se na distância do pulo do Passarinho), senão gera uma nova plataforma acessível.

#### **4.4. Algoritmos Clássicos**

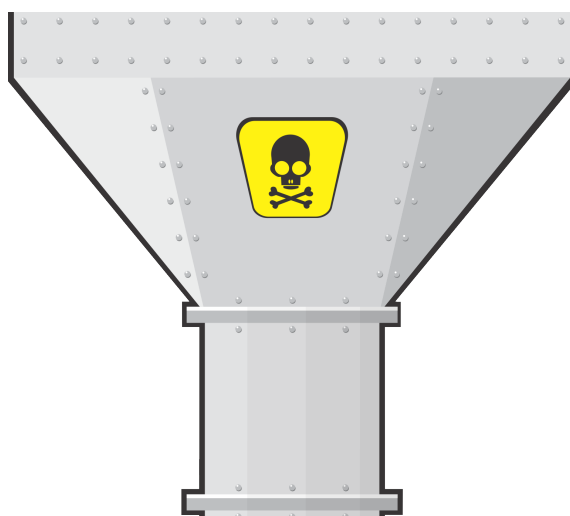
Para a exibição do cenário, a câmera utilizou algoritmos de recorte. Durante a criação das sprites, foi utilizado Curvas de Bézier para deixa-las mais suaves. Durante o *gameplay* são utilizadas transformações nos movimentos dos personagens.

### **5. Conclusão**

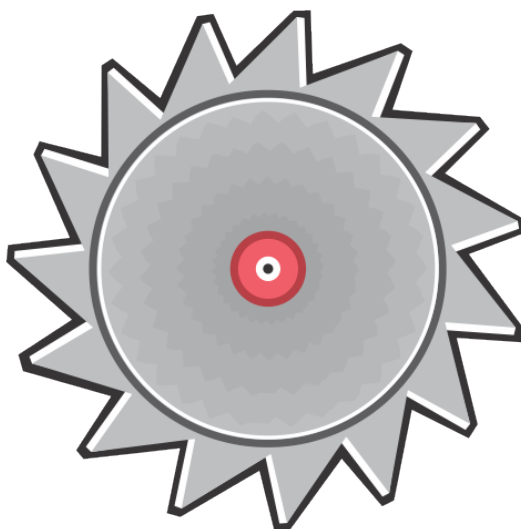
Neste trabalho foi apresentado o jogo *Cai, Cai Passarinho*, um *infinity run* onde tenta se subir o maior número possíveis de plataformas sem encostar na máquina. O jogo foi desenvolvido para Android utilizando Unity e Corel Draw e tenta passar uma crítica sobre o desmatamento.



**Figura 5. Passarinho**



**Figura 6. Máquina Engolidora de Árvores**



**Figura 7. Serra**