**UNIVERSIDADE DE SÃO PAULO**

**Instituto de Ciências Matemáticas e de Computação**

Utilizando Desenvolvimento de Jogos Eletrônicos para o Ensino de Computação

***Gustavo Ferreira Ceccon***

***[Nome do Aluno]***



Utilizando Desenvolvimento de Jogos Eletrônicos para o Ensino de Computação

***Gustavo Ferreira Ceccon***

##### Orientador: Claudio Fabiano Motta Toledo

|  |
| --- |
| Monografia de conclusão de curso apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP - para obtenção do título de Bacharel em Ciências de Computação. |
| Área de Concentração: Ensino, Desenvolvimento de Jogos, Ciências de Computação |

**USP – São Carlos**

Novembro **de 2015**

*A good teacher can inspire hope, ignite the imagination,*

*and instill a love of learning..*

Resumo

O projeto tem como objetivo incentivar professores de diversas disciplinas a usar jogos eletrônicos como forma a engajar o aluno em matérias de computação. Essas diretrizes são voltada para áreas como Introdução à Ciência da Computação I e II (SCC221 e SCC201), Geometria Analítica (SMA300), Algoritmos e Estruturas de Dados (SCC202), Programação Orientada a Objetos (SCC204), Programação Concorrente (SSC143), Computação Gráfica (SSC250) e Multimídia (SCC261). O desenvolvimento de jogos abrange muitas áreas e pode ser explorado para envolver o aluno em uma aplicação real e num contexto interessante. A monografia enumera atividades e projetos de desenvolvimento de jogos para cativar o aluno e dar uma visão de como aplicar o conteúdo aprendido. Um estudo foi feito sobre projetos similares de engajamento de alunos de Ciências de Computação em outras universidades.

Sumário

[Lista de Figuras v](#_Toc499132194)

[Lista de Tabelas vi](#_Toc499132195)

[Lista de Gráficos vii](#_Toc499132196)

[CAPÍTULO 1: INTRODUÇÃO 1](#_Toc499132197)

[1.1. Contextualização e Motivação 1](#_Toc499132198)

[1.2. Objetivos 1](#_Toc499132199)

[1.3. Organização da Monografia 1](#_Toc499132200)

[CAPÍTULO 2: REVISÃO BIBLIOGRÁFICA 2](#_Toc499132201)

[2.1. Considerações Iniciais 2](#_Toc499132202)

[2.2. Conceitos e Técnicas Relevantes 2](#_Toc499132203)

[2.3. Trabalhos Relacionados 2](#_Toc499132204)

[2.4. Considerações Finais 2](#_Toc499132205)

[CAPÍTULO 3: DESENVOLVIMENTO DO TRABALHO 2](#_Toc499132206)

[3.1. Considerações Iniciais 3](#_Toc499132207)

[3.2. Descrição do Problema 3](#_Toc499132208)

[3.3. Descrição das Atividades Realizadas 3](#_Toc499132209)

[3.4. Resultados Obtidos 3](#_Toc499132210)

[3.5. Dificuldades, Limitações e Trabalhos Futuros 4](#_Toc499132211)

[3.6. Considerações Finais 4](#_Toc499132212)

[CAPÍTULO 4: CONCLUSÃO 5](#_Toc499132213)

[4.1. Contribuições 5](#_Toc499132214)

[4.2. Considerações sobre o Curso de Graduação 5](#_Toc499132215)

[REFERÊNCIAS 6](#_Toc499132216)

[APÊNDICE A – Título deste apêndice 7](#_Toc499132217)

[ANEXO A – Título deste anexo 8](#_Toc499132218)

# Lista de Figuras

Nenhuma entrada de índice de ilustrações foi encontrada.

# Lista de Tabelas

Nenhuma entrada de índice de ilustrações foi encontrada.

# Lista de Gráficos

Nenhuma entrada de índice de ilustrações foi encontrada.

# CAPÍTULO 1: INTRODUÇÃO

## Contextualização e Motivação

Ensinar computação não é uma tarefa fácil, de acordo com (Beaubouef & Mason, 2005) existem 7 motivos principais para que 30% a 40% dos estudantes de Ciências de Computação larguem o curso no primeiro ou no segundo ano. Outro artigo (Biggers, Brauer, & Yilmaz, 2008) faz um estudo sobre o porquê dos estudantes largarem o curso no Geórgia Instituto de Tecnologia em Atlanta.

Um artigo (Bennedsen & Caspersen, 2007) faz uma pesquisa de diversas faculdades para ver a porcentagem de reprovações em disciplinas introdutórias de programação. Apesar de ter um baixa quantidade de resposta dos formulários por parte das universidades (12.7%), ele verifica o alto índice de reprovações dessas matérias (33%). Além disso, o mesmo estudo avalia o que é dado nessas disciplinas, por exemplo paradigma de programação, o que indicou que quase metade (49%) dos institutos usam orientação a objeto como introdutória, apesar que todos os paradigmas tem índices similares.

O mercado de jogos eletrônicos tem uma alta demanda e tem uma alta renda, de aproximadamente US$108,9 bilhões (McDonald, 2017). Enquanto isso, o Brasil está em 13º lugar em mercado de jogos, com 66,3 milhões de jogadores e US$1,3 bilhão gasto em jogo (Newzoo, 2017).

Além disso, os estudantes estão interessados em desenvolvimento de jogos, como indica a Figura 1 feita sobre os inscritos para o processo seletivo do grupo de desenvolvimento de jogos do ICMC-USP, o FoG (Fellowship of the Game). Não apenas alunos de Bacharelado em Ciências de Computação (BCC), mas alunos de Engenharia de Computação (EC), Bacharelado em Sistemas de Informação (BSI) etc. também tem interesse na área.

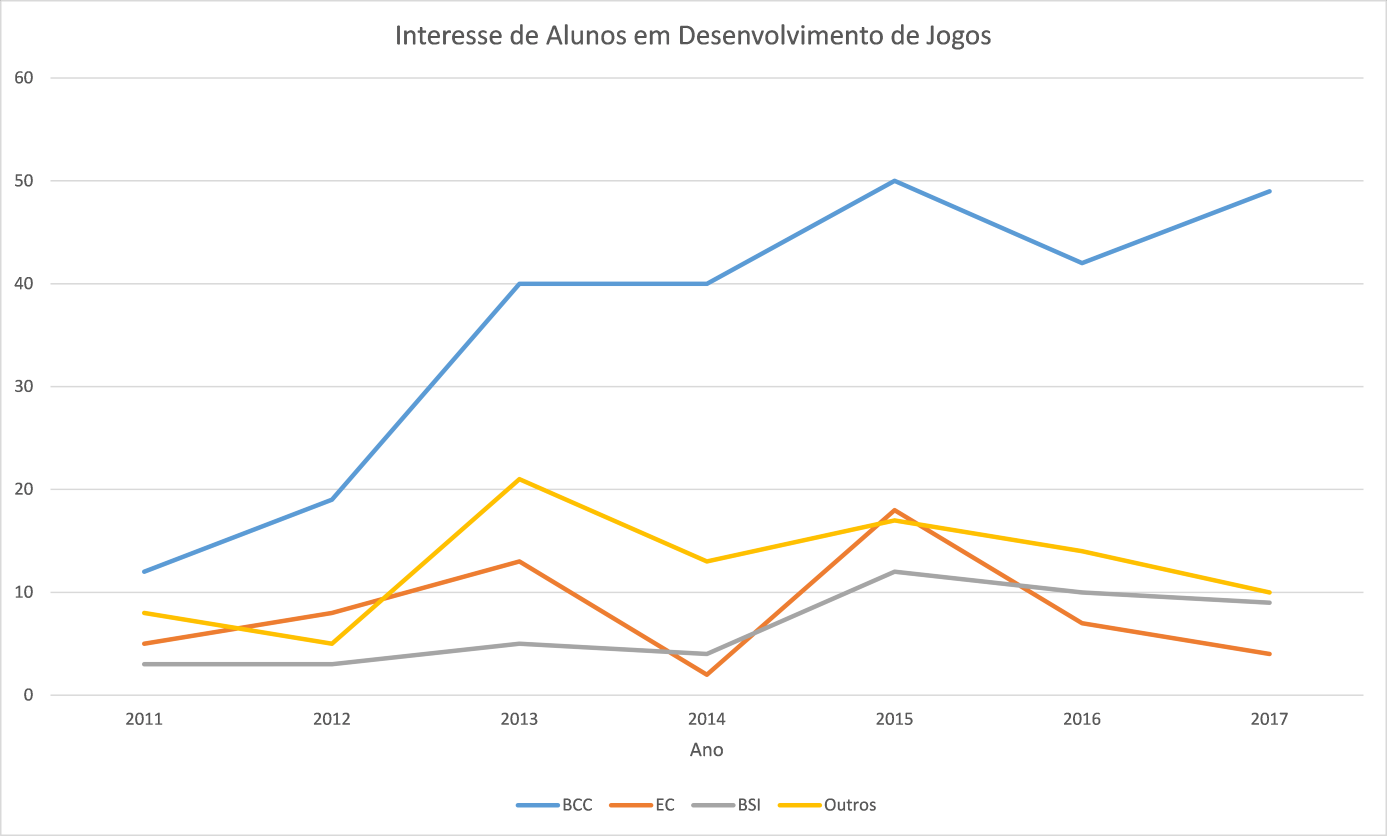


Figura 1. Interesse de Alunos em Desenvolvimento de Jogos

## 1.2. Objetivos

Alguns dos motivos (Beaubouef & Mason, 2005) podem ser lidados usando desenvolvimento de jogos:

* *Poor Math Skills and Problem Solving Abilities*
* *Poorly Designed CS1 Lab Courses*
* *Lack of Practice / Feedback*

A escolha entre programação orientada a objeto no início ou no meio da graduação parece ser um debate sem uma resolução certa. Há pontos positivos e negativos em ambos lados e não será enfoque desse trabalho. Jogos eletrônicos podem ser feitos em ambos paradigmas e, dependendo do gênero, um pode servir melhor que o outro.

Jogos envolvem muita matemática (principalmente em 3D com computação gráfica) e problemas de decisão. Quando apresentado de uma forma concisa, problemas matemáticos são fáceis de implementar e ajudam a visualizar melhor o que é aprendido em matérias como Geometria Analítica. Uma visão tridimensional de vetores, superfícies, volume etc. pode desenvolver a compreensão matemática. Quando não for possível desenvolver algo, seja por limitação de tempo ou dificuldade, uma ferramenta auxiliar pode ser usada.

Um ponto positivo de jogos no aprendizado é que eles são incrementais, ou seja, é possível desenvolver um jogo e conseguir um *feedback* das mudanças e alterações durante o processo. Isso é importante porque é possível dar laboratórios de desenvolvimento onde o professor consegue ir adicionando conteúdo e explicando algoritmos durante a aula. Além disso o aluno pode desenvolver em casa as atividades passadas em aula, incrementando aos poucos e adicionar o próprio conteúdo.

Ter um conteúdo pessoal a mais num projeto, um *flavor,* é importante para um aluno que gosta de jogos. Uma atividade pode ter uma avaliação mínima e adicionais podem fazer parte de pontos extra, estimulando o interesse do aluno.

## 1.3. Organização da Monografia

TODO: Capítulos do livro, exemplos e projetos

# CAPÍTULO 2: REVISÃO BIBLIOGRÁFICA

## 2.1. Considerações Iniciais

Jogos eletrônicos envolvem muito mais que programação apenas, envolve matemática, física, artistas, marketing etc. Reunir todos esses tópicos em apenas um trabalho é impossível, por isso essa monografia cobre apenas algumas partes de todo esse escopo, mesmo no escopo de programação.

Jogos ultimamente não são feitos do zero, eles não almejam apenas uma plataforma e com certeza levam tempo, por exemplo o jogo Antichamber[[1]](#footnote-1), feito por Alexander Bruce, levou aproximadamente 7 anos de desenvolvimento[[2]](#footnote-2). Escolhas de projeto altamente influenciam no tempo de desenvolvimento, custo e riscos.

Porém jogos não necessariamente são voltados para entretenimento, existem os chamados “jogos sérios”, voltados para educação, áreas da saúde etc. Existem diversos estudos (Michael & Chen, 2005 ) e (Ritterfeld, Cody, & Vorderer, 2009) sobre jogos sérios.

Além de livros sobre jogos sérios, existe uma vasta coleção bibliográfica sobre gêneros (tipos) de jogos (Apperley, 2006) e sobre como desenvolver jogos (game design), como (Rogers, 2014).

TODO: explicar engine e jogo, falar de projetos em cima dos dois, livros de desenvolvimento voltados as áreas, descrever rapidamente o que foi usado

TODO: ver se precisar criar novos capitulos

## 2.2. Conceitos e Técnicas Relevantes

## 2.2.1. Jogos Eletrônicos

## 2.2.1. Motores de Jogos

## 2.2.1. Gêneros de Jogos

TODO: jogo e engine, partes de uma engine, gênero de jogos, desenvolvimento paradigmas, complexidades e ferramentas

## 2.3. Trabalhos Relacionados

TODO: separar por tipo, citar os relevantes

# CAPÍTULO 3: DESENVOLVIMENTO DO TRABALHO

## 3.1. Considerações Iniciais

TODO: como icc1 e disciplinas são dadas, falhas de projetos, apresentação, aulas, avaliação, usar ferramentas, criar algo visual

TODO: criar novos capitulos

## 3.2. Descrição do Problema

TODO: falar da situação atual das aulas

## 3.3. Descrição das Atividades Realizadas

TODO: criação da disciplina, proposta de modelo de graduação

## 3.5. Dificuldades, Limitações e Trabalhos Futuros

TODO: limitação de créditos, criação de novas disciplinas, reorganização do curso, melhor didática dos professores e nova metodologia

# CAPÍTULO 4: CONCLUSÃO

TODO: falar do tamanho da área, oportunidades, inspiração, procura, cursos de jogos e relação com computação

TODO: criar novos capitulos

# Referências

Barnes, T., Powell, E., Chaffin, A., & Lipford, H. (Fevereiro de 2008). Game2Learn: Improving the motivation of CS1 students. *GDCSE*.

Barnes, T., Powell, E., Chaffin, A., Godwin, A., & Richter, H. (Junho de 2007). Game2Learn: Building CS1 Learning Games for Retention. *ITiCSE*.

Bayliss, J., & Strout, S. (Março de 2006). Games as a “Flavor” of CS1. *SIGCSE*.

Beaubouef, T., & Mason, J. (Junho de 2005). Why the High Attrition Rate for Computer Science Students: Some Thoughts and Observations. *SIGCSE*.

Becker, K. (Dezembro de 2001). Teaching with Games: The Minesweeper and Asteroids Experience. *Consortium for Computing in Small Colleges*.

Bennedsen, J., & Caspersen, M. (Junho de 2007). Failure Rates in Introductory Programming. *SIGCSE*.

Biggers, M., Brauer, A., & Yilmaz, T. (Março de 2008). Student perceptions of computer science: a retention study comparing graduating seniors with cs leavers. *SIGCSE*.

Chaffin, A., Doran, K., Hicks, D., & Barnes, T. (Agosto de 2009). Experimental Evaluation of Teaching Recursion in a Video Game. *Sandbox*.

Coleman, R., Krembs, M., Labouseur, A., & Weir, J. (Fevereiro de 2005). Game Design & Programming Concentration Within the Computer Science Curriculum. *SIGCSE*.

Hernandez, C. C., Silva, L., Segura, R. A., Schimiguel, J., Ledón, M. F., Bezerra, L. N., & Silveira, I. F. (Abril de 2010). Teaching Programming Principles through a Game Engine. *CLEI Eletronic Journal*.

Jones, R. (Janeiro de 2000). Design and implementation of computer games: A capstone course for undergraduate computer. *SIGCSE*.

Kelleher, C., Pausch, R., & Kiesler, S. (Abril de 2007). Storytelling Alice Motivates Middle School Girls to Learn Computer Programming. *CHI*.

Leutenegger, S., & Edgington, J. (Março de 2007). A Games First Approach to Teaching Introductory Programming. *SIGCSE*.

Muratet, M., Torguet, P., Viallet, F., & Jessel, J.-P. (Agosto de 2010). Experimental Feedback on Prog&Play: A Serious Game. *Computer Graphics Forum*.

Takanashi, F. (06 de Abril de 2009). *Folha de São Paulo*. Acesso em Novembro de 2017, disponível em Folha UOL: http://www1.folha.uol.com.br/educacao/2009/04/546576-matematica-e-ciencias-da-computacao-tem-alta-taxa-de-abandono.shtml

Walker, R. (13 de Fevereiro de 2011). *Computer Science Vs. Game Development (or Which Degree Should I Get?)*. Acesso em Novembro de 2017, disponível em Gamasutra: https://www.gamasutra.com/blogs/RobertWalker/20110213/88888/Computer\_Science\_Vs\_Game\_Development\_or\_Which\_Degree\_Should\_I\_Get.php

Zhang, F., Kaufman, D., & Fraser, S. (Agosto de 2014). Using Video Games In Computer Science Education. *European Scientific Journal*.

1. <http://www.antichamber-game.com/> [↑](#footnote-ref-1)
2. <https://www.youtube.com/watch?v=wOlcB-JxkFw> [↑](#footnote-ref-2)