

CURSO:	Engenharia de Software		
DISCIPLINA:	Introdução aos Jogos Eletrônicos	SEMESTRE/ANO:	01/2015
CARGA HORÁRIA:	60 horas	CRÉDITOS:	04
PROFESSOR:	Edson Alves da Costa Júnior	TURMA:	A

PLANO DE ENSINO

1 Objetivos da Disciplina

A disciplina **Introdução aos Jogos Eletrônicos** busca propiciar aos alunos conhecimentos na área de desenvolvimento de Jogos Eletrônicos. Esta área de estudo envolve desde *Game Design*, Programação, Arte (ou Criação), até questões relacionadas à comercialização. Entretanto, como esta é uma disciplina voltada para os alunos da Engenharia de Software da FGA, a maior parte do programa se concentrará nos fundamentos de programação para a criação de jogos de computadores.

2 Ementa do Programa

I. *Overview*

- i. História dos Jogos
- ii. Jogos e a Sociedade
- iii. Diversão
- iv. Game Design

II. *Linguagens e Arquitetura*

- i. Arquitetura de Jogos
- ii. SDL - *Simple DirectMedia Layer*

III. *Fundamentos de Programação Aplicados a Jogos*

- i. Vetores e Matrizes
- ii. Listas Encadeadas e Filas
- iii. Filas de Prioridade
- iv. Grafos e Máquinas de Estado Finitas

IV. *Tópicos Avançados*

- i. Detecção e Resolução de Colisão
- ii. *Pathfinding*

3 Horário das aulas e atendimento

AULAS: quartas e sextas, das 14:00 às 15:50 hrs.

ATENDIMENTO: segundas, das 14:00 às 16:00 hrs.

4 Metodologia

A metodologia consiste em aulas expositivas, com o auxílio do quadro branco e projetor digital. A fim de fortalecer a aprendizagem da disciplina, as aulas serão complementadas com exercícios e atividades, presenciais e extra-classe, principalmente na forma de tarefas atribuídas na plataforma GitHub. Os alunos receberão um *feedback* destas tarefas por parte do professor ou dos monitores, com sugestões de correções e melhorias, e comentários que os auxiliarão no entendimento e aprofundamento dos conceitos desenvolvidos nas aulas.

A fim de promover o trabalho em uma equipe multidisciplinar, conforme a prática adotada na indústria profissional, serão realizadas 16 aulas no *campus* Darcy Ribeiro, em conjunto com as turmas de **Introdução ao Desenvolvimento de Jogos**, do CIC, e **Design de Jogos**, do Desenho Industrial. Estes encontros estão marcados no Cronograma com o texto **Aula Conjunta**.

5 Critérios de Avaliação

Os alunos serão divididos em grupos de 3 (três) ou 4 (quatro) integrantes, para o desenvolvimento de um **jogo eletrônico** ao longo de todo o semestre letivo. Este desenvolvimento será direcionado através da criação de *issues* no GitHub¹ semanalmente pelo professor, relacionadas ao desenvolvimento do jogo.

5.1 Composição do grupo

Cada grupo funcionará como uma empresa de desenvolvimento de jogos fictícia, onde cada membro terá uma função principal, embora todos devam contribuir em todas as etapas e em todas as tarefas. As funções a serem preenchidas são a de **gerente**, **game designer** e **programador**, sendo que em uma equipe de quatro membros haverão 2 (dois) programadores. Poderão ainda participar da equipe um ou mais alunos oriundos dos cursos IDJ ou DJ, conforme o que for decidido na aula 05 (o limite de quatro alunos diz respeito aos alunos oriundos da FGA. A equipe pode extrapolar este limite se acrescidos novos alunos dos cursos do Darcy Ribeiro).

Cada *issue* terá maior ênfase nas características de uma destas três funções, e terá peso dobrado para o membro que assumir o respectivo papel.

5.2 Tarefas

Serão, no mínimo, 30 *issues* (no mínimo 10 para cada perfil). A *issue* será considerada completa quando for fechada no GitHub pelo professor ou pelos monitores. Uma *issue* parcialmente resolvida ou com erros em sua resolução podem vir a gerar novas *issues*, até que o problema central seja contornado. As *issues* só poderão ser resolvidas, **impreterivelmente**, até a aula onde ocorrerá a entrega da versão final do jogo.

Em relação à pontuação T das *issues*, a princípio todos os membros do time tem 40 pontos. Cada *issue* que ficar aberta ao final do prazo de desenvolvimento acarretará na penalidade de -1 ponto (-2 pontos caso a *issue*

¹<https://github.com>

esteja associada ao perfil do membro da equipe), com uma penalidade máxima de -10 pontos (-20 pontos, para o perfil do membro) para cada perfil.

Por exemplo, se a equipe deixar em aberto 3 *issues* de *Game Design*, 12 de gerência e 5 de programação, o programador da equipe ficará com pontual final T igual a

$$T = 40 + 3(-1) + 10(-1) + 5(-2) = \mathbf{17},$$

o gerente terá pontuação

$$T = 40 + 3(-1) + 10(-2) + 5(-1) = \mathbf{12},$$

e o *game designer*

$$T = 40 + 3(-2) + 10(-1) + 5(-1) = \mathbf{19}.$$

A pontuação de um aluno está sujeita à sua colaboração efetiva na resolução das *issues* atribuídas à sua equipe. Esta colaboração C será mensurada pelo número de *pull requests* aceitos em tarefas do seu perfil (R_P) ou nas demais tarefas (R_D) em relação ao número total de *issues* N , de acordo com a equação abaixo:

$$C = \frac{2R_P + R_D}{N} \times 100$$

Alunos com colaboração C inferior a 25 pontos terão nota T igual a zero.

5.3 Jogo

As *issues* nortearão o desenvolvimento de um jogo eletrônico ao longo do semestre, cujo tema será determinado pela própria equipe. jogo deve ser entregue e apresentado nas datas previstas no cronograma.

O jogo será avaliado segundo uma planilha a ser divulgada na data oportuna, e o resultado da avaliação dará a equipe uma pontuação J entre 0 e 60 pontos. Um jogo só poderá ser apresentado se tiver sido entregue na data prevista.

5.4 Menção Final

A **nota final** N_F será dada pela soma das pontuações das *issues* T e do jogo J , isto é,

$$N_F = T + J$$

A menção final será dada conforme tabela abaixo:

Menção	Descrição	Nota Final
SR	<i>Sem rendimento</i>	0
II	<i>Inferior</i>	de 1 a 29
MI	<i>Médio inferior</i>	de 30 a 49
MM	<i>Médio</i>	de 50 a 69
MS	<i>Médio superior</i>	de 70 a 89
SS	<i>Superior</i>	90 ou mais

5.5 Critérios de aprovação

Obterá **aprovação** no curso o aluno que cumprir as **duas** exigências abaixo:

1. Ter presença em 75% ou mais das aulas ministradas no *campus* do Gama
2. Obter menção final igual ou superior a MM

6 Cronograma

Semana	Aula	Data	Conteúdo
01	1	11/03	<i>Apresentação do curso. Ciclo de desenvolvimento de um jogo</i>
	2	13/03	Aula Conjunta: <i>Apresentação dos jogos anteriores</i>
02	3	18/03	Aula Conjunta: <i>Histórico dos jogos</i>
	4	20/03	Aula Conjunta: <i>Definição dos grupos e temas</i>
03	5	25/03	<i>Arquitetura do Jogo</i>
	6	27/03	Aula Conjunta: <i>Game Design e Diversão</i>
04	7	01/04	<i>Introdução à SDL</i>
	-	03/04	Feriado: Páscoa
05	8	08/04	<i>SDL: Manipulação de pixels e imagens</i>
	9	10/04	Aula Conjunta: <i>Game Audio</i>
06	10	15/04	<i>SDL: Dispositivos de entrada e eventos</i>
	11	17/04	Aula Conjunta: <i>Apresentação das propostas</i>
07	12	22/04	Aula Conjunta: <i>Apresentação das propostas</i>
	13	24/04	<i>SDL: Animações e Colisões</i>
08	14	29/04	<i>SDL: Textos e Fontes</i>
	-	01/05	Feriado: Dia do Trabalhador
09	15	06/05	<i>SDL: Áudio</i>
	16	08/05	Aula Conjunta: <i>GDD</i>
10	17	13/05	<i>SDL: Threads</i>
	18	15/05	<i>Tile Maps</i>
11	19	20/05	<i>Grafos. Máquina de Estados Finitos</i>
	20	22/05	Aula Conjunta: <i>Aula Reunião</i>
12	21	27/05	Aula Conjunta: <i>Bate papo com profissionais</i>
	22	29/05	Aula Conjunta: <i>MILESTONE 1 (apresentação: 30% do jogo pronto)</i>
13	23	03/06	Aula Conjunta: <i>MILESTONE 1 (apresentação: 30% do jogo pronto)</i>
	-	05/06	Feriado: Corpus Christie

Semana	Aula	Data	Conteúdo
14	24	10/06	<i>Scripts</i>
	25	12/06	<i>Scrolling e Parallax</i>
15	26	17/06	Aula Conjunta: <i>Aula Reunião</i>
	27	19/06	Aula Conjunta: <i>MILESTONE 2 (apresentação: 70% do jogo pronto)</i>
16	28	24/06	Aula Conjunta: <i>MILESTONE 2 (apresentação: 70% do jogo pronto)</i>
	29	26/06	<i>Saltos e Trajetórias</i>
17	30	01/07	<i>SDL: Rede</i>
	31	03/07	Aula Conjunta: <i>Bate papo com profissionais</i>
18	32	08/07	<i>Finalização dos jogos e entrega</i>
	33	10/07	Aula Conjunta: <i>MILESTONE 3 (apresentação: 100% do jogo pronto)</i>

7 Bibliografia

LIVROS TEXTOS

HALL, Jonh R. *Programming Linux Games*, No Starch Press, San Franscisco, 2001.

PENTON, Ron. *Data Structures for Game Programmers*, The Premier Press, 2004.

RABIN, Steve. *Introduction to Game Development*, Charles River Media, 2008.

LITERATURA COMPLEMENTAR

ERICSON, Christer. *Real-Time Collision Detection*, Morgan Kaufmann, 2005.

BUCKLAND, Mat. *Programming Game AI by Example*, Wordware Publishing, 2004.

PAZERA, Ernest. *Focus on SDL*, The Premier Press, 2003.

MILLINGTON, Ian. *Game Physics Engine Development*, Morgan Kaufmann, 2007.

BETHKE, Erik. *Game Development and Production*, Wordware Publishing, 2003.