

Tópicos Especiais em Engenharia de Software (Jogos II)

Apresentação da Disciplina

Edirlei Soares de Lima
<edirlei@iprj.uerj.br>



Objetivos da Disciplina

- Apresentar os fundamentos de **jogos eletrônicos** e as **técnicas de programação** para o desenvolvimento de jogos utilizando a **Unity 3D**.
- Ao final do curso, o aluno deverá:
 - Compreender os conceitos fundamentais para a elaboração de projetos de jogos eletrônicos;
 - Conhecer e saber aplicar as técnicas fundamentais para o desenvolvimento e implementação de jogos eletrônicos.

Jogos Eletrônicos II

Programação de Jogos 3D

+

Modelagem e Animação 3D

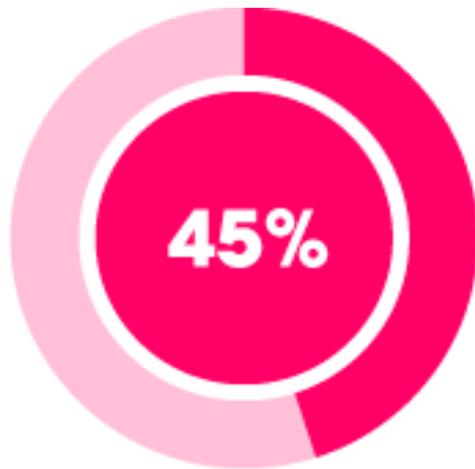


O que é a Unity3D?

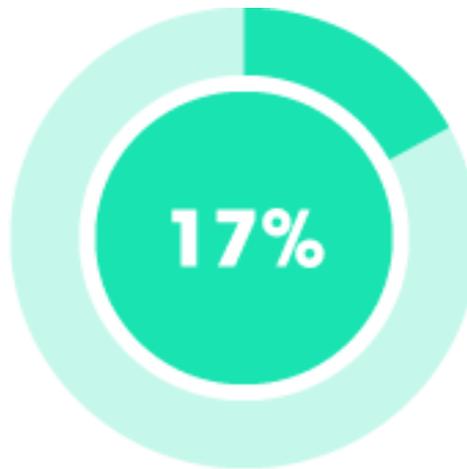
- **Game Engine**
- **Primeira versão lançada em 2005**
- **Plataformas:**
 - Windows, Linux, Mac
 - WEB (Web Player e Flash)
 - Celulares e Tablets (Android, iOS e Windows Phone)
 - Consoles (PS3, PS4, PSVita, XBOX 360, XBOX One, Wii)

O que é a Unity3D?

- Global game engine market share:



Unity



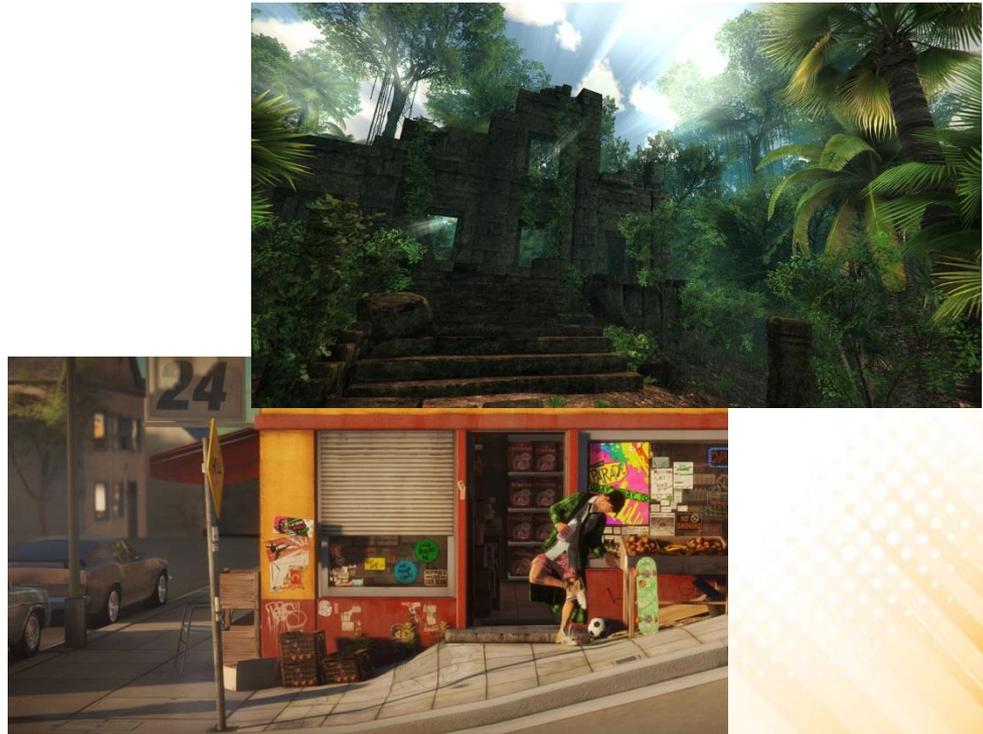
Closest competitor



Others

O que é a Unity3D?

- **Utilizada por pequenas e grandes empresas:**
 - Cartoon Network, Coca-Cola, Disney, Electronic Arts, LEGO, Microsoft, NASA, Nexon, Nickelodeon, Square, Ubisoft, Warner Bros...
- **DirectX e OpenGL**
- **Script:**
 - JavaScript (UnityScript)
 - C#
 - Boo
- **Asset Store**



Ementa da Disciplina

1. Unity3D: Interface e Conceitos Básicos

- Interface
 - Projeto
 - Objetos Primitivos
 - Importação de Assets
 - Standard Assets
 - Exportação
 - GameObjects
 - Components
 - Prefabs
 - Scenes
- 

Ementa da Disciplina

2. Unity3D: Graphics

- Iluminação
 - Sombras
 - Skybox
 - Materiais
 - Shaders
 - Câmeras
 - Editor de Terrenos
 - Editor de Árvores
 - Sistema de Partículas
 - Água
- 

Ementa da Disciplina

3. Unity3D: Scripting

- Introdução ao C#
 - Scripts como Components
 - Introdução a Orientação a Objetos
 - MonoBehaviour (Start, Update, FixedUpdate, ...)
 - Estruturas Condicionais
 - Estruturas de Repetição
 - Vetores, Matrizes, Listas e Dicionários
 - Herança, Polimorfismo, Encapsulamento, Composição
 - Scripts para o Editor da Unity
- 

Ementa da Disciplina

4. Unity3D: Física

- Colliders
 - Triggers
 - Rigidbodies
 - Physics Forces
 - Physics Torque
 - Physics Materials
 - Physics Joints
 - Raycasting
- 

Ementa da Disciplina

5. Unity3D: Animação

- Ferramentas de Animação 3D
 - Sistema de Animação
 - Importação de Animação
 - Criação e Edição de Animações
 - Animator Component
 - Animator Controller
 - Animator Scripting
 - Animator Sub-state Machine hierarchies
 - Humanoid Avatars
- 

Ementa da Disciplina

6. Unity3D: User Interface

- UI Canvas
 - UI Button
 - UI Image
 - UI Text
 - UI Events and Event Triggers
 - UI Slider
 - UI Transitions
 - UI Scroll Rect
 - UI Scrollbar
 - UI Mask
- 

Ementa da Disciplina

7. Unity3D: Áudio

- Audio 3D
 - Audio Listeners e Sources
 - Audio Mixer e Audio Mixer Groups
 - Audio Effects
- 

Ementa da Disciplina

8. Unity3D: Inteligência Artificial

- Conceitos e estado da arte
- Waypoints e pathfinding
- Máquinas de estados finitos

Ementa da Disciplina

9. Unity3D: Multiplayer and Networking

- Conceitos de jogos em rede
 - NetworkManager
 - High Level API
- 

Esquema de Aulas

- **Aulas teóricas e praticas:**
 - **Aulas teóricas** sobre conceitos de jogos, Unity 3D, tecnologias, programação e técnicas de desenvolvimento;
 - **Aulas praticas** para a elaboração de projeto final e resolução de exercícios;
- 

Critério de Avaliação

- **Avaliação Prática:**
 - Trabalho e exercícios desenvolvidos em grupo ou individualmente;
 - Apresentação em aula;
 - **Avaliação Teórica:**
 - Prova teórica envolvendo o conteúdo teórico e prático apresentado durante as aulas;
- 

Critério de Avaliação

- **Trabalho:**

- Um único jogo desenvolvido durante todo o semestre
- Três avaliações parciais do projeto (T1, T2 e T3)
- Uma avaliação final do projeto (T4)
- $MT = (T1 + T2 + T3 + T4) / 4$ (com PESO 5.0 na MP)

- **Prova:**

- PT: prova teórica individual envolvendo todo o conteúdo do curso (com PESO 5.0 na MP)

- **$MP = (MT + PT)/2$**

Critério de Avaliação

- Se a frequência nas aulas for $< 75\%$ o aluno será REPROVADO POR FALTA;
- Se a frequência nas aulas for $\geq 75\%$, então:
 - Se $MP \geq 7.0$, o aluno será aprovado e $MF = MP$;
 - Se $MP < 4.0$, o aluno será reprovado e $MF = MP$;
 - Se $MP < 7.0$ e $MP \geq 4.0$, o aluno irá para o exame final e então:
 - $MF = (MP + PF)/2$;
 - Se $MF < 5.0$ o aluno será reprovado;
 - Se $MF \geq 5.0$ o aluno será aprovado;

Avaliação

- **Trabalho desenvolvido em equipe:**
 - 3 ou 4 alunos por equipe;
 - Os grupos serão definidos nas próximas aulas;
- **Avaliação individual baseada nos trabalhos desenvolvido e na apresentação:**
 - Desenvolvimento;
 - Participação;
 - Apresentação;
 - Questionamentos;
 - Notas individuais;

Avaliação do Projeto

- **1º Avaliação Parcial:**
 - Projeto do jogo: ideia, objetivos, história, personagens, mecânica do jogo (gameplay), universo, inimigos, interface
 - Data: 26/09
- **2º Avaliação Parcial:**
 - Implementação da mecânica do jogo (gameplay)
 - Data: 31/10

Avaliação do Projeto

- **3º Avaliação Parcial:**
 - Implementação do universo/fases do jogo
 - Implementação dos personagens/inimigos
 - Data: 28/11

- **Avaliação Final:**
 - Implementação completa do jogo
 - Data: 12/12

Festival de Jogos Independentes

- Os melhores jogos poderão ser submetidos ao Festival de Jogos Independentes do SBGames ~~2016~~ 2017?
 - Possibilidade dos autores participarem do evento
- Exposição na feira
- Categorias:
 - Principal (PC, Web, Mobile)
 - Serious games
 - Advergame
 - Outras Plataformas



Pré-Requisito

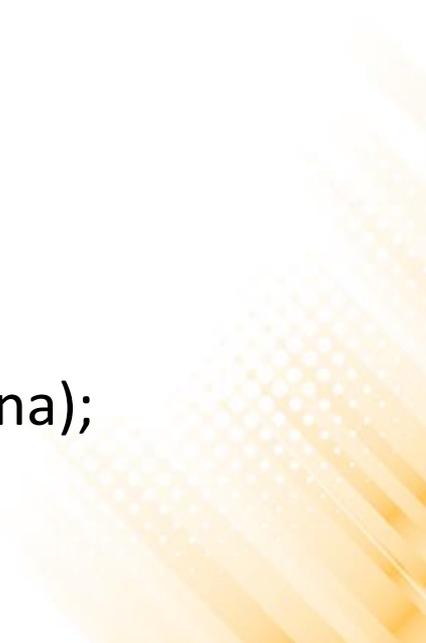
- Introdução à Programação
 - Tópicos Especiais em Linguagens de Programação (Jogos I)
- 

Controle de Turma

- **Presença obrigatória!!!**
 - Chamada em qualquer momento da aula;
 - Alunos com menos de 75% de presença serão reprovados automaticamente e não poderão fazer prova final (independente da nota);
 - Não será permitido trocar de turma ou frequentar as aulas de outra turma;
 - Não será aberta nenhuma exceção!

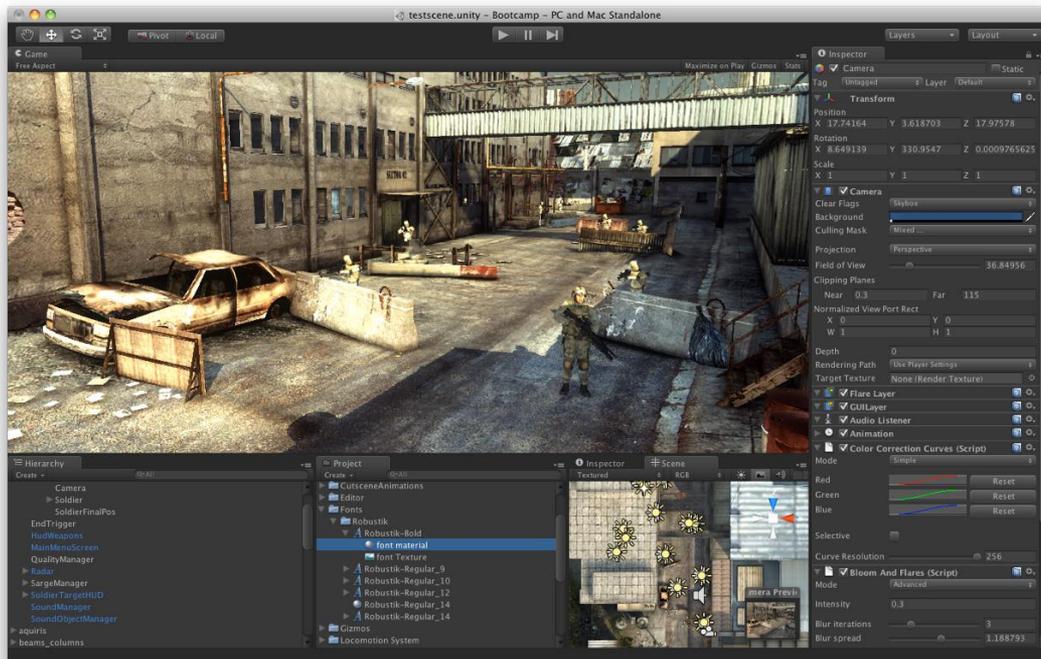
- **Projetos desenvolvidos em equipe:**
 - Todos os membros da equipe devem participar ativamente do desenvolvimento dos projetos;
 - Não participar do desenvolvimento dos projetos resultará em redução individual da nota;

Aulas Praticas

- Nas aulas usar apenas os softwares da disciplina;
 - Se for possível, trazer notebook pessoal para as aulas;
 - Se for usar os computadores do laboratório, salve os trabalhos no pendrive com regularidade;
 - Verificar regularmente o site da disciplina;
 - Prática extraclasse constante (mínimo de 4h/semana);
- 

Software

- O software que será utilizado nesta disciplina é:
 - **Unity 3D** - <http://unity3d.com/>

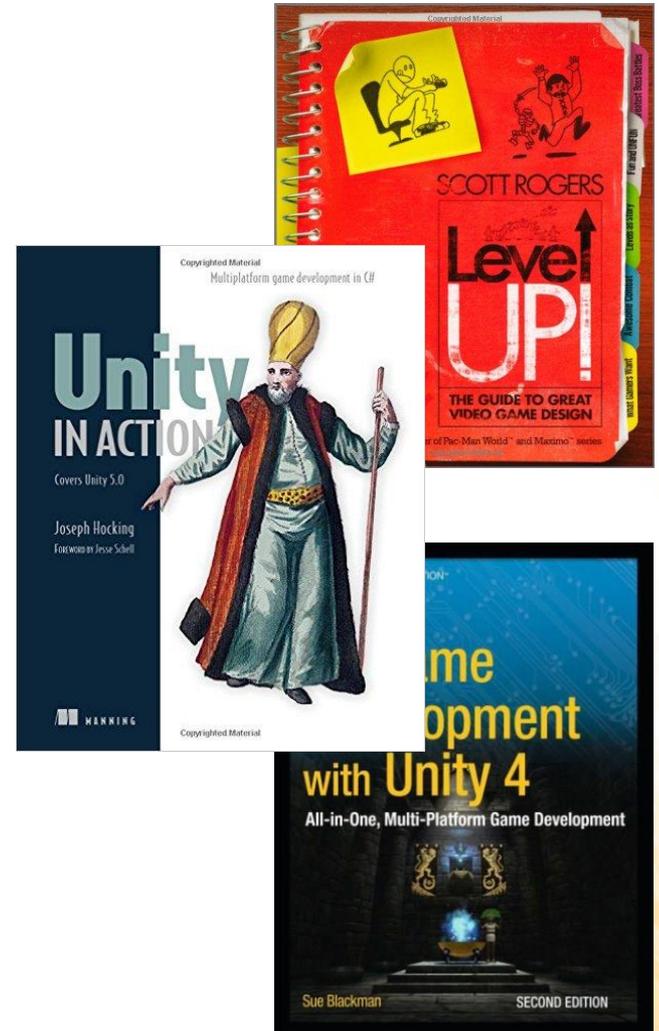


Material das Aulas

- **Página do Curso:**
 - www.inf.puc-rio.br/~elima/jogos3d/
- **Contato:**
 - edirlei@iprj.uerj.br

Bibliografia Principal

- Rogers, S. **Level Up!: The Guide to Great Video Game Design**; Wiley, 2010.
- Hocking, J. **Unity in Action: Multiplatform Game Development in C# with Unity 5**, Manning Publications, 2015.
- Blackman, S. **Beginning 3D Game Development with Unity 4**, Apress, 2013.



Bibliografia Complementar

- Smith, M., Queiroz, C. **Unity 5.x Cookbook**, Packt Publishing, 2015.
- Fullerton, T., Swain, C., Hoffman, S. **Game Design Workshop: Designing, Prototyping and Playtesting Games**, CMP Books, 2004.
- Rollings, A., Morris, D. **Game Architecture and Design**, The Coriolis Group, 2000.
- Millington, I., Funge, J. **Artificial Intelligence for Games**, Morgan Kaufmann, 2009.