



MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA

PLANO DE DISCIPLINA			
IDENTIFICAÇÃO			
CAMPUS: Esperança - PB			
CURSO: Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas			
DISCIPLINA: Jogos Digitais		CÓDIGO DA DISCIPLINA: 65.2	
PRÉ-REQUISITO: Programação Orientada a Objetos e Introdução a Redes de Computadores			
UNIDADE CURRICULAR: Obrigatória [] Optativa [X] Eletiva []		SEMESTRE/ANO: 6º	
CARGA HORÁRIA			
TEÓRICA: 33h	PRÁTICA: 34h	EaD¹: 0h	EXTENSÃO: 0h
CARGA HORÁRIA SEMANAL: 4h/a			
CARGA HORÁRIA TOTAL: 67h			
DOCENTE RESPONSÁVEL: Alvaro Magnum Barbosa Neto			

EMENTA

Desenvolver conhecimentos técnico-científicos para criar, projetar, testar e implantar diferentes tipos de jogos digitais em plataformas computacionais. Os alunos serão capazes de gerenciar projetos de jogos digitais 2D e 3D; além de preparar roteiros, cenas, simulações, personagens e mecanismos para o jogo, avaliar, selecionar e utilizar tecnologias e ferramentas de desenvolvimento para adaptá-los às diversas plataformas utilizadas (videogames, dispositivos móveis, computadores, redes sociais, etc.).

OBJETIVOS DA DISCIPLINA/COMPONENTE CURRICULAR

Objetivo Geral:

Capacitar os alunos a conceber, projetar, desenvolver, testar e implantar jogos digitais, utilizando conhecimentos técnico-científicos e ferramentas adequadas para diversas plataformas computacionais, com foco na aplicação prática do ciclo de vida de desenvolvimento de jogos 2D e 3D.

Objetivos Específicos:

1. Compreender e aplicar os conceitos fundamentais do design de jogos digitais (Game Design), incluindo mecânicas, dinâmicas, estéticas, narrativa e experiência do jogador.
2. Elaborar elementos essenciais de um jogo digital, como roteiros/storyboards, design de níveis/cenas, definição de regras, criação conceitual de personagens e desenvolvimento de mecânicas de jogo.
3. Identificar, avaliar e utilizar ferramentas e tecnologias de desenvolvimento (como game engines, linguagens de programação, APIs gráficas e de áudio) apropriadas para a criação de jogos digitais 2D e 3D.
4. Aplicar princípios de gerenciamento de projetos específicos para o desenvolvimento de jogos digitais, abrangendo planejamento, escopo, prototipagem e acompanhamento de tarefas.
5. Realizar testes funcionais e de jogabilidade (playtesting) para identificar e corrigir problemas, refinar a experiência do usuário e validar o conceito do jogo.
6. Compreender e aplicar os processos necessários para adaptar e implantar jogos digitais em diferentes plataformas computacionais, como computadores (PC/Mac), dispositivos móveis (Android/iOS) e web.

CONTEÚDO PROGRAMATICO

Etapa 1: Fundamentos e Concepção

- Introdução aos Jogos Digitais: Visão geral da história, gêneros e mercado.
- Princípios Fundamentais de Game Design: Mecânicas, Dinâmicas, Estéticas (MDA), Experiência do Jogador e Elementos de Jogos.
- Concepção e Documentação: Geração de ideias, Game Design Document (GDD) e introdução à narrativa/roteiro.
- Visão Geral de Ferramentas: Apresentação das principais Game Engines e softwares de apoio.
- Introdução ao Gerenciamento de Projetos de Jogos: Pipeline básico e metodologias ágeis adaptadas.

Etapa 2: Desenvolvimento e Implementação

- Uso Prático de Game Engines: Estrutura, interface, cenas, objetos e componentes.
- Programação de Jogos e Lógica: Scripting (Ex: C# ou Blueprints), input, física e implementação de mecânicas centrais.
- Criação e Integração de Assets: Pipeline de importação e uso de recursos 2D/3D, animação e áudio.
- Construção de Níveis e Interfaces (UI/UX): Design de fases/cenários e criação de menus/HUDs.
- Implementação de Gameplay: Colocando em prática as mecânicas e sistemas do jogo (personagens, interações).

Etapa 3: Finalização, Testes e Aplicação

- Polimento e Otimização: Melhoria da experiência (Game Feel), efeitos visuais/sonoros e otimização de performance.
- Testes (Playtesting) e Garantia de Qualidade (QA): Metodologias de teste, coleta de feedback e correção de bugs.
- Gerenciamento de Projeto (Fase Final): Controle de versão (Git), refinamento de escopo e finalização.
- Adaptação para Plataformas e Build: Considerações sobre diferentes plataformas (PC, Mobile, Web) e geração da versão executável.
- Aplicação Prática e Distribuição: Desenvolvimento de projeto final/protótipo e introdução aos modelos de distribuição/monetização.

METODOLOGIA DE ENSINO

1. Aulas Expositivas Curtas e Contextualizadas (20% do tempo):

- Início de cada aula com uma breve explicação teórica sobre o tema do dia.
- Utilização de exemplos reais para contextualizar os conceitos.

2. Aprendizado Baseado em Projetos (Project-Based Learning - PBL):

- Divisão da turma em pequenos grupos para desenvolver projetos ao longo do semestre.
- Cada projeto deve abordar conceitos específicos da disciplina, como a implementação de uma interface gráfica ou a criação de uma API back-end.

3. Atividades Práticas Guiadas (40% do tempo):

- Proposta de exercícios práticos durante a aula, com acompanhamento do professor.
- Desenvolvimento de funcionalidades específicas, como um componente de interface ou uma integração front-end/back-end.

4. Dinâmica de Aprendizado Colaborativo:

- Incentivo ao trabalho em dupla ou em grupo para solucionar problemas e compartilhar conhecimentos.
- Momentos de "peer review", onde os alunos avaliam o trabalho uns dos outros.

5. Acompanhamento do Projeto Final (40% do tempo):

- Alocação de tempo em cada aula para que os grupos avancem no projeto final.
- Sessões de feedback individualizadas com o professor para discutir progresso e desafios.

6. Demonstrações e Discussões:

- No final de cada aula ou etapa, os alunos apresentam o que desenvolveram para a turma.
- Discussões coletivas para troca de ideias e solução de problemas comuns.

7. Uso de Ferramentas Modernas:

- Aplicação de ferramentas com o Unity Game Engine
- Incentivo ao uso de ferramentas de IA.

8. Avaliação Contínua e Formativa:

- Acompanhamento contínuo por meio das entregas parciais do projeto.
- Feedback detalhado para cada etapa do aprendizado.

RECURSOS DIDÁTICOS

- [x] Quadro
- [x] Projetor
- [x] Vídeos/DVDs
- [x] Periódicos/Livros/Revistas/Links

- Equipamento de Som
- Laboratório
- Softwares²
- Outros³

CRITÉRIOS DE AVALIAÇÃO

Serão executados 3 projetos por equipes de até 3 alunos. Cada projeto contemplará a temática da etapa correspondente. O trabalho será em equipe e a nota individual. A média do aluno será igual a média das notas dos projetos, variando de 0 a 100. As regras para aprovação, reprovação e final seguirão de acordo com o regimento institucional.

ATIVIDADE DE EXTENSÃO

BIBLIOGRAFIA

Bibliografia Básica:

1. CHANDLER, Heather Maxwell. Manual de produção de jogos digitais. Traduzido por Aldir José Coelho Corrêa da Silva; revisado por João Ricardo Bittencourt. 2. ed. Porto Alegre: Bookman, 2012. 478 p. ISBN 9788540701830.
2. ARRUDA, Eucídio Pimenta. Fundamentos para o desenvolvimento de jogos digitais. Porto Alegre: Bookman, 2014. 102 p. (Tekne). ISBN 9788582601433.
3. ROGERS, Scott. Level up: um guia para o design de grandes jogos. São Paulo: Blucher, 2013. 494 p. ISBN 9788521207009.

Bibliografia Complementar:

1. KINSLEY, Harrison; MCGUGAN, Will. Introdução ao desenvolvimento de jogos em Python com Pygame. São Paulo: Novatec, 2015. 360 p. ISBN 9788575224526.
2. MENARD, Michelle; WAGSTAFF, Bryan. Game development with Unity. 2. ed. Boston: Cengage Learning, 2015. xx, 444 p. ISBN 9781305110540; 1305110544.
3. FEIJÓ, Bruno; CLUA, Esteban; CORRÊA DA SILVA, Flávio S. Introdução à ciência da computação com jogos: aprendendo a programar com entretenimento. Rio de Janeiro: Elsevier, 2010. 263 p. (Série Editora Campus/SBC). ISBN 9788535234190.
4. PORTELLA, Ricardo. Unity para principiantes. S.I.: Bookman, 2016. 55 p. ISBN 1534768467; 9781534768468.
5. SANCHES, Murilo Henrique Barbosa. Jogos digitais, gamificação e autoria de jogos na educação. São Paulo: Senac, 2021. 176 p.

Bibliografia Suplementar:

1. REHMAN, Muhammad Hamza. Comparison between React and Angular JavaScript Frameworks. 2024.
2. PIASTOU, Mikita. Comprehensive Performance and Scalability Assessment of Front-End Frameworks: React, Angular, and Vue. js. World Journal of Advanced Engineering Technology and Sciences, v. 9, n. 2, p. 366-376, 2024.
3. ANASTASIA, Terzi; STAMATIA, Bibi. Managing Security Vulnerabilities Introduced by Dependencies in React. JS JavaScript Framework. In: 2024 IEEE International Conference on Software Analysis, Evolution and Reengineering-Companion (SANER-C). IEEE, 2024. p. 126-133.

OBSERVAÇÕES

1 Para a oferta de disciplinas na modalidade à distância, integral ou parcial, desde que não ultrapassem os limites definidos em legislação.

2 Definido de acordo com as necessidades e especificidades da turma.

3 Definido de acordo com as necessidades e especificidades da turma.

Documento assinado eletronicamente por:

■ Alvaro Magnum Barbosa Neto, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 01/05/2025 15:33:45.

Este documento foi emitido pelo SUAP em 01/05/2025. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifpb.edu.br/autenticar-documento/> e forneça os dados abaixo:

Código 707921

Verificador: ed9c838fb6

Código de Autenticação:



Rodovia PB 121, S/N, Centro, ESPERANÇA / PB, CEP 58135-000

<http://ifpb.edu.br> -