

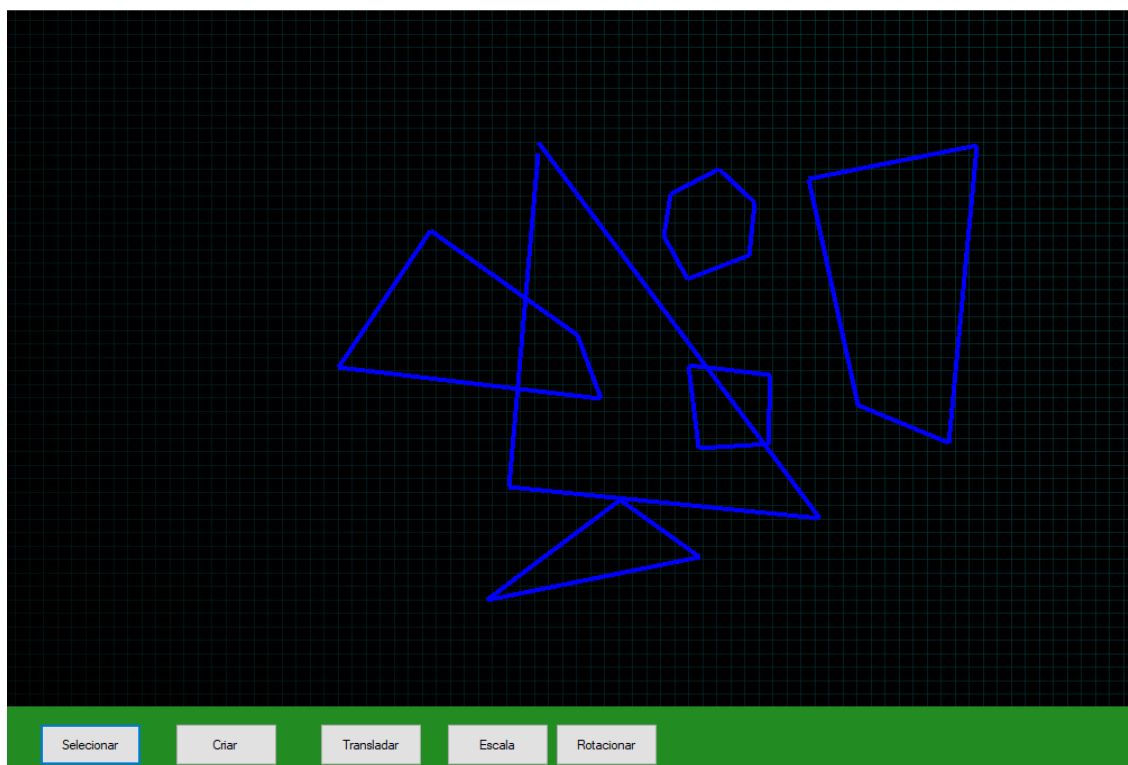
Palavra do professor:

O curso de Jogos Digitais preenche a demanda em muitas áreas, como Design de personagens e Designer de fase ou Designer de ambientação, Designer de sistema ou Designer de mecânica de jogo – elabora e equilibra as regras do jogo – entre outras. Mas há uma área que de fato está intrínseca a todas as essas oportunidades de emprego, que é a Computação Gráfica. A programação de elementos da Computação Gráfica costuma ser de fato a implementação estrutural dos Jogos Digitais, e costuma ter muitas linhas de código envolvidas.

Portanto, no curso de P.P.I. de Jogos Digitais, este Trabalho Avaliativo, terá como base a construção de um **Editor de Polígonos 2D**, nele será implementado: Transformações Geométricas(Translação, Rotação e Escala), Seleção de Polígonos com o mouse e Elementos de Interface Gráfica. Alguns dessas funcionalidades fazem como, por exemplo uma espaçonave se mover pelo cenário, trazendo oportunidades a você, estudante de Jogos Digitais, a se sair bem futuramente na construção de jogos 2D como Asteroids, jogo frequentemente pedido em entrevistas de emprego na área.

Descrevendo o trabalho:

O Trabalho consistirá em construir um **Editor de polígonos 2D** na tecnologia Windows Forms utilizando a linguagem C# tendo como alvo o Sistema Operacional Windows. Esse editor genérico será descrito a seguir.



*Figura 1: tela inicial do software, com botões de Selecionar, Criar, Transladar, Escala e Rotacionar.*

O Software tem como requisitos funcionais, isto é funcionalidades:

---

**[RF – 01] Ser totalmente orientado a objetos**

O software deverá implementar os conceitos aprendidos em sala de aula, clássicos de orientação a objetos, como: Classes, Objetos, Herança, Classe Abstrata, Classe Concreta, Interfaces, Encapsulamento e Polimorfismo.

*Prioridade: Altíssima*

---

---

**[RF – 02] Utilizar apenas no desenho de polígono, desenho de ponto e linha**

No desenvolvimento do software não poderá ser usado funções prontas das transformações geométricas nem classes de desenho típicas de biblioteca prontas como 2Point, 3Point, Vector2D, etc... pois o aluno DEVERÁ implementar as classes que irá utilizar.

O uso de coleções de dados nativas como System.Collections.Generic é altamente recomendado e se faz necessário encapsular as instâncias das coleções de System.Collections.Generic nas classes, segundo o requisito RF-01.

*Prioridade: Altíssima.*

---

---

**[RF – 03] Criar um polígono**

A criação do polígono deverá ser feita única e exclusivamente através do mouse. A criação de polígonos deverá permitir ao usuário cancelar a criação do polígono através de ou elementos da interface(botões, campos, etc) ou eventos do mouse ou até mesmo eventos no teclado.

*Prioridade: Altíssima.*

---

---

**[RF – 04] Selecionar um polígono**

A seleção deverá ser feita única e exclusivamente através do mouse. O software deverá mostrar ao usuário quando um polígono foi ou não selecionado.

*Prioridade: Alta.*

---

---

**[RF – 05] Transladar um polígono**

A translação do Objeto consiste em movimentar o polígono pela tela, e deverá ser feita única e exclusivamente através do mouse. A translação deverá ser suavizada de forma que não ocorram movimentos abruptos, que são péssimos para a usabilidade do Software.

*Prioridade: Alta.*

---

---

**[RF – 06] Escalar um polígono**

A Escala do Objeto consiste em mudar o tamanho do polígono segundo uma constante inteira, e deverá ser feita única e exclusivamente através do mouse. A Escala deverá ser suavizada de forma que não ocorram movimentos abruptos, que são péssimos para a usabilidade do Software.

*Prioridade: Alta.*

---

---

**[RF – 07] Rotacionar um polígono**

A Rotação do Objeto consiste em girar o polígono ao redor do seu centro, e pode ser feita através do mouse ou do teclado. A Rotação deverá ser suavizada de forma que não ocorram movimentos abruptos, que são péssimos para a usabilidade do Software.

*Prioridade: Média.*

---

---

**[RF – 08] Excluir um polígono**

A Exclusão de um polígono consiste removê-lo da cena, e pode ser feita através do mouse ou do teclado.

*Prioridade: Média.*

---

---

**[RF – 09] Salvar a cena em Disco.**

O software deverá implementar o salvamento da cena em disco, de forma que o usuário possa salvar o seu trabalho no sistema de arquivos para reabrir a cena depois, para continuar o seu trabalho.

*Prioridade: Baixa.*

---

O Software tem como requisitos não funcionais, isto é, características dele:

---

**[RNF – 01] Não travar ou fechar durante a apresentação.**

Durante a apresentação para o professor o software não deverá travar ou fechar por quaisquer motivos.

---

---

**[RNF – 02] Ter interface amigável.**

Durante a apresentação o software deverá ter um mínimo de usabilidade de forma que o usuário consiga usá-lo de maneira amigável.

---

Avaliando o trabalho: A avaliação será organizada da seguinte forma:

---

**Parte 1: Apresentação ao professor:**

A apresentação ao professor será segundo os seguintes critérios:

- A não implementação dos requisitos de prioridade altíssima tem caráter eliminatório, uma vez que esses requisitos não só serão implementados pelos alunos como também o professor irá implementar no data show em aulas práticas.
- Organização da pontuação:

Requisito 01, 02, 03	40 pontos
Requisito 04, 05, 06	40 pontos
Requisito 07	15 pontos
Requisito 08	5 pontos
Requisito 09	5 pontos (EXTRA)

- A implementação de todos os requisitos faz com que não seja necessário o aluno apresentar o software para os seus colegas no seminário, tornando opcional a parte 2.
- 

---

**Parte 2: Seminário:**

A parte dois consiste numa apresentação de vinte minutos por grupo de um software para a turma toda. Na apresentação os alunos deverão expor à turma:

- Como o software funciona.
- As dificuldades enfrentadas na implementação.
- Argumentar a não implementação de um ou outro requisito.

Na segunda parte as justificativas serão analisadas e conforme a situação, haverá revisão na nota da primeira parte, para mais ou para menos.

---