

## Trabalho de Laboratório – Curso EI

### Objetivos:

- Introdução ao JavaFX 2 - Painéis e controlos de interface. Ligação bidirecional (*Binding Bidirectional*) de objetos. Utilização de *Canvas* e *GraphicsContext*.

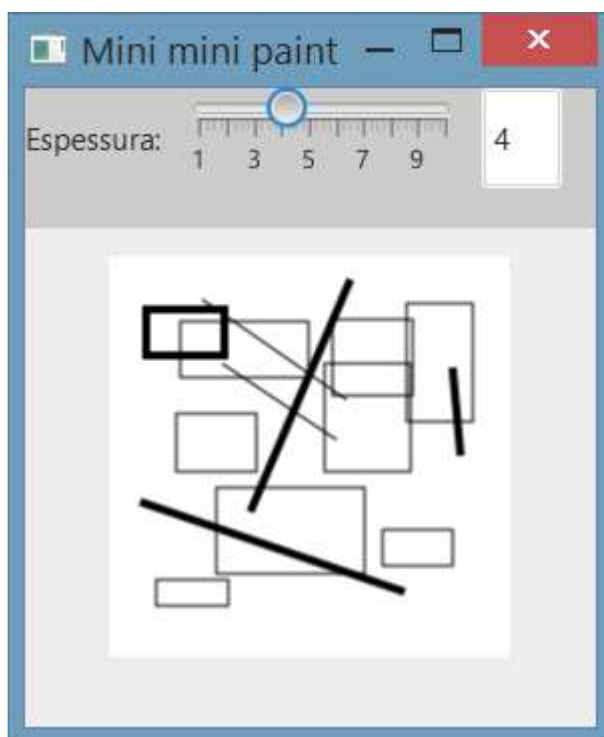
### Programas:

- Pretende-se desenvolver uma pequena aplicação para desenho de linhas e retângulos.

### Regras de implementação:

- Criar uma aplicação JavaFX com interface gráfica.
- Implementar o código necessário e testar no fim de cada nível.
- Use as convenções de codificação adotadas para a linguagem Java.
- As variáveis relativas aos componentes gráficos devem conter nomes sugestivos.
- Na classe do programa não ponha nenhum atributo nem crie nenhum método para além do *main*.
- Para comentários deve utilizar o Javadoc (ver documento disponível no Moodle).

### Aspeto pretendido para a interface gráfica:



## Trabalho de Laboratório – Curso EI

### Implementação:

#### Nível 1:

- Crie um novo projeto JavaFX Application.
- Crie uma classe **Desenhador** que estende **BorderPane**. Esta classe representa a estrutura principal da aplicação, na qual serão criados os restantes elementos, e deverá ser utilizada na criação da cena. Exemplo:

```
Desenhador root = new Desenhador();
Scene scene = new Scene(root, LARGURA, ALTURA);
```
- Crie agora uma classe **AreaDesenho** estendendo a classe **Canvas**. Esta classe irá representar a área de desenho da aplicação.
- No construtor do **Desenhador** crie uma **AreaDesenho** e coloque-a na zona central do próprio **Desenhador**.
- “Pinte” a respetiva área com uma cor distinta para confirmar que esta foi corretamente criada. Para o efeito, obtenha o **GraphicsContext2D** da área, aceda ao **PixelWriter** e utilize o método **setColor**.

#### Nível 2:

- Na classe **Desenhador**, adicione o código necessário para capturar os eventos de **OnMousePressed** e **OnMouseReleased** referentes à **AreaDesenho**, apresentando na consola as coordenadas em que o clique ocorreu (utilize os métodos **getX** e **getY** do **MouseEvent**).
- Modifique o evento **OnMousePressed** para que guarde em atributos da classe **AreaDesenho** a posição do clique (por exemplo: **x0** e **y0**).
- Modifique o evento **OnMouseReleased** para desenhar uma linha a partir do ponto onde o botão do rato foi premido (**x0**, **y0**) até ao ponto onde o botão do rato foi largado. Para esta funcionalidade, utilize os métodos **beginPath**, **moveTo**, **lineTo**, **closePath** e **stroke** do **GraphicsContext**. Para mais informações sobre o funcionamento destes métodos, poderá aceder à documentação em:  
<http://docs.oracle.com/javafx/2/api/javafx/scene/canvas/GraphicsContext.html>

#### Nível 3:

- Pretende-se agora criar a barra de configuração no topo, onde é possível definir a espessura de linha. Para o efeito, crie no **Desenhador** uma classe **ConfiguradorEspessura** que deriva da classe **HBox**, a qual deverá conter um **Slider** e uma **TextBox**, ambos para armazenar o valor da espessura da linha a desenhar.
- Configure os controlos para que os valores da espessura estejam sempre compreendidos no intervalo entre 1 e 10.
- Faça o *bind* bidirecional entre a propriedade de texto da **TextBox** e a propriedade de valor do **Slider**, para que os controlos fiquem sincronizados.

#### Nível 4:

- Na classe **ConfiguradorEspessura**, implemente um método para obter a espessura escolhida (a partir do **Slider**).
- Modifique a funcionalidade de desenhar linha para que seja utilizada a espessura selecionada pelo utilizador. Utilize o método **setLineWidth** do **GraphicsContext** para definir a espessura da linha antes de a desenhar.

## Trabalho de Laboratório – Curso EI

---

### Nível 5:

- Modifique a aplicação para que o processo de desenho crie linhas com o botão esquerdo do rato e retângulos com o botão direito.

**Notas:** Para os identificadores siga as convenções adotadas normalmente, em particular:

- 1) A notação *camelCase* para o nome das variáveis locais e identificadores de atributos e métodos
- 2) A notação *PascalCase* para os nomes das classes
- 3) Não utilize o símbolo ‘\_’, nem abreviaturas nos identificadores