

Nou Esquelet

- L'esquema d'aquest esquelet és el que usarem per a l'examen ProvaLab
- Classe NouGLWidget:
 - Hereta de MyGLWidget
 - Pot reimplementar mètodes de MyGLWidget
 - Pot accedir als atributs i mètodes protected de MyGLWidget
- Aquest és un exemple, no exactament el de l'examen

Classe NouGLWidget

MyGLWidget.h

MyGLWidget.cpp

Classe MyGLWidget.

No s'ha de tocar!

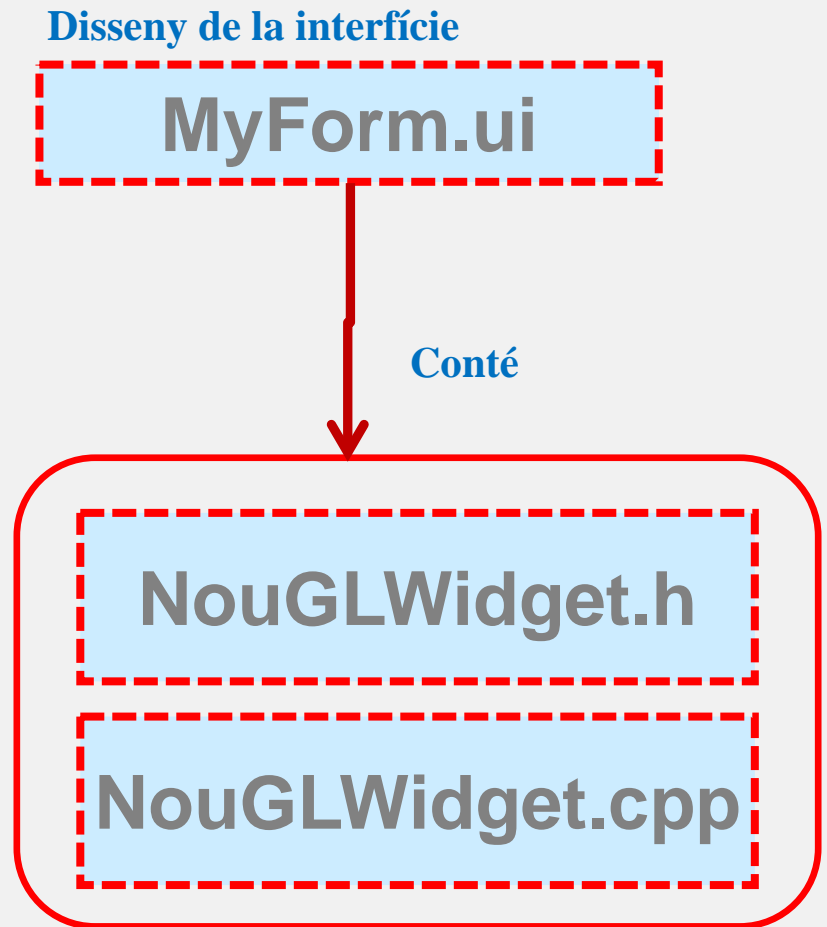
Classe que hereta de MyGLWidget
Implementarà els exercicis

NouGLWidget.h

NouGLWidget.cpp

Classe NouGLWidget

- La interfície de l'aplicació de Qt (MyForm.ui) conté un element de tipus NouGLWidget
- Quan Qt inicia l'aplicació crida als mètodes initializeGL() i resizeGL(...) de NouGLWidget
- initializeGL() està reimplementat en NouGLWidget i crida al mètode de la classe base
- resizeGL() no està reimplementat, s'executa el de MyGLWidget.



Nou Esquelet: MyGLWidget.h

.....

```
class MyGLWidget : public QOpenGLWidget, protected QOpenGLFunctions_3_3_Core
{
    Q_OBJECT
public:
    MyGLWidget (QWidget *parent=0);
    virtual ~MyGLWidget ();
protected:
    virtual void initializeGL (); // Inicialitzacions del contexte gràfic
    virtual void paintGL (); // Mètode de pintat
    virtual void resizeGL (int width, int height); // Es crida quan canvia dimensió finestra
    ..... // Tots els mètodes virtuals són susceptibles de ser reimplementats en la
            classe derivada

    QOpenGLShaderProgram *program; // Els atributs protected es veuen des de la
    glm::mat4 View, legoTG;          // classe derivada
private:
    ..... // Tot el que és privat no es veu des de la classe derivada
};
```

Nou Esquelet: NouGLWidget.h

```
#include "MyGLWidget.h"
```

```
Class NouGLWidget : public MyGLWidget
```

```
{
```

```
    Q_OBJECT
```

```
public:
```

```
    MyGLWidget (QWidget *parent=0);
```

```
    ~MyGLWidget ();
```

```
public slots:    // per si calen slots
```

```
signals:        // per si calen signals
```

```
protected:
```

```
    virtual void initializeGL ();        // mètodes que cal reimplementar en NouGLWidget
```

```
    virtual void viewTransform ();
```

```
    virtual void modelTransformLego ();
```

```
    virtual void keyPressEvent (QKeyEvent *event);
```

```
private:        // per si calen nous atributs o mètodes privats
```

```
};
```

Nou Esquelet:

NouGLWidget.cpp (1)

```
#include "NouGLWidget.h"
```

```
NouGLWidget::NouGLWidget (QWidget* parent) : MyGLWidget (parent) {  
    grabKeyboard();  
}  
NouGLWidget::~~NouGLWidget () {  
}
```

// Mètodes susceptibles de ser modificats

```
void NouGLWidget::initializeGL () {  
    MyGLWidget::initializeGL ();    // crida a InitializeGL de la classe MyGLWidget  
}  
void NouGLWidget::modelTransformLego () {  
    MyGLWidget::modelTransformLego ();  
}  
void NouGLWidget::viewTransform () {  
    MyGLWidget::viewTransform ();  
}
```

Nou Esquelet: NouGLWidget.cpp

```
void NouGLWidget::keyPressEvent (QKeyEvent *event)
{
    makeCurrent();
    switch (event->key()) {
        case Qt::Key_A: {
        }
        case Qt::Key_D: {
        }
        case Qt::Key_F: {
        }
        default: {          // si no és cap de les altres crida al mètode de MyGLWidget
            MyGLWidget::keyPressEvent(event);
            break;
        }
    }
    update();
}
```