

# Cours 5 : Les interfaces graphiques

**Rabii EL GHORFI**

Module : Technique de programmation avancées

Département : Mathématiques, informatique et géomatique (MIG)

EHTP 2017-2018



# Principaux axes du cours

- Introduction au CLR / CLI Support
- Installer VS 2017 avec CLR / CLI Support
- Mettre en place un projet avec CLR / CLI Support

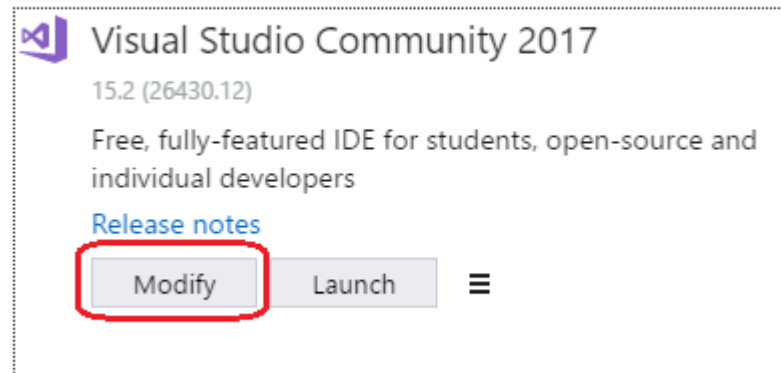
# CLR / CLI Support

- Common Language Runtime (**CLR**) est le nom choisi par Microsoft pour la machine virtuelle du framework **.NET**
- Common Language Infrastructure (**CLI**) définit l'environnement d'exécution des codes de programmes du framework **.NET**
- Plusieurs langages reposent sur cette technologie puissante du .NET tel que : C# et ASP
- Nous utiliserons CLR / CLI Support pour créer des interfaces graphiques

# Installation (1)

- Assurez vous d'avoir installer Visual Studio 2017 avec C++ et le package CLR / CLI Support

Installed



# Installation (2)

Modifying — Visual Studio Community 2017 — 15.2 (26430.12)

Workloads **Individual components** Language packs

- Dependency Validation
- Developer Analytics tools
- DGML editor
- Git for Windows
- GitHub extension for Visual Studio
- Help Viewer
- LINQ to SQL tools
- NuGet package manager
- PowerShell tools
- PreEmptive Protection - Dotfuscator
- Static analysis tools
- Text Template Transformation

Compilers, build tools, and runtimes

- C# and Visual Basic Roslyn compilers
- C++/CLI support
- Clang/CLZ (experimental)
- IncrediBuild
- MSBuild
- Python 2 32-bit (2.7.13)
- Python 2 64-bit (2.7.13)
- Python 3 32-bit (3.6.0)
- Python 3 64-bit (3.6.0)

Location  
C:\Program Files (x86)\Microsoft Visual Studio\2017\Community

Install size: --- MB

Please close all instances of Microsoft Visual Studio 2017 before proceeding with this operation.

**Summary**

- > Visual Studio core editor
- > .NET desktop development
- > Desktop development with C++
- > ASP.NET and web developm... \*
- ▼ Individual components
  - Included
    - .NET Framework 4.6.1 SDK
    - .NET Framework 4.6.1 targeting pack
    - C++/CLI support
    - Microsoft Visual Studio 2017 Installer Pr...

**Modify**

# Installation (3)

Modifying — Visual Studio Community 2017 — 15.2 (26430.12) ✕

**Workloads** Individual components Language packs

Windows (3)

- Universal Windows Platform development  
Create applications for the Universal Windows Platform with C#, VB, JavaScript, or optionally C++.
- Desktop development with C++  
Build classic Windows-based applications using the power of the Visual C++ toolset, ATL, and optional features like MFC and...
- .NET desktop development  
Build WPF, Windows Forms and console applications using the .NET Framework.

Web & Cloud (7)

- ASP.NET and web development  
Build web applications using ASP.NET, ASP.NET Core, HTML, JavaScript, and CSS.
- Azure development  
Azure SDK, tools, and projects for developing cloud apps and creating resources.
- Python development  
Editing, debugging, interactive development and source control for Python.
- Node.js development  
Build scalable network applications using Node.js, an asynchronous event-driven JavaScript runtime.
- Data storage and processing
- Data science and analytical applications

Location  
C:\Program Files (x86)\Microsoft Visual Studio\2017\Community

Install size: 0 KB

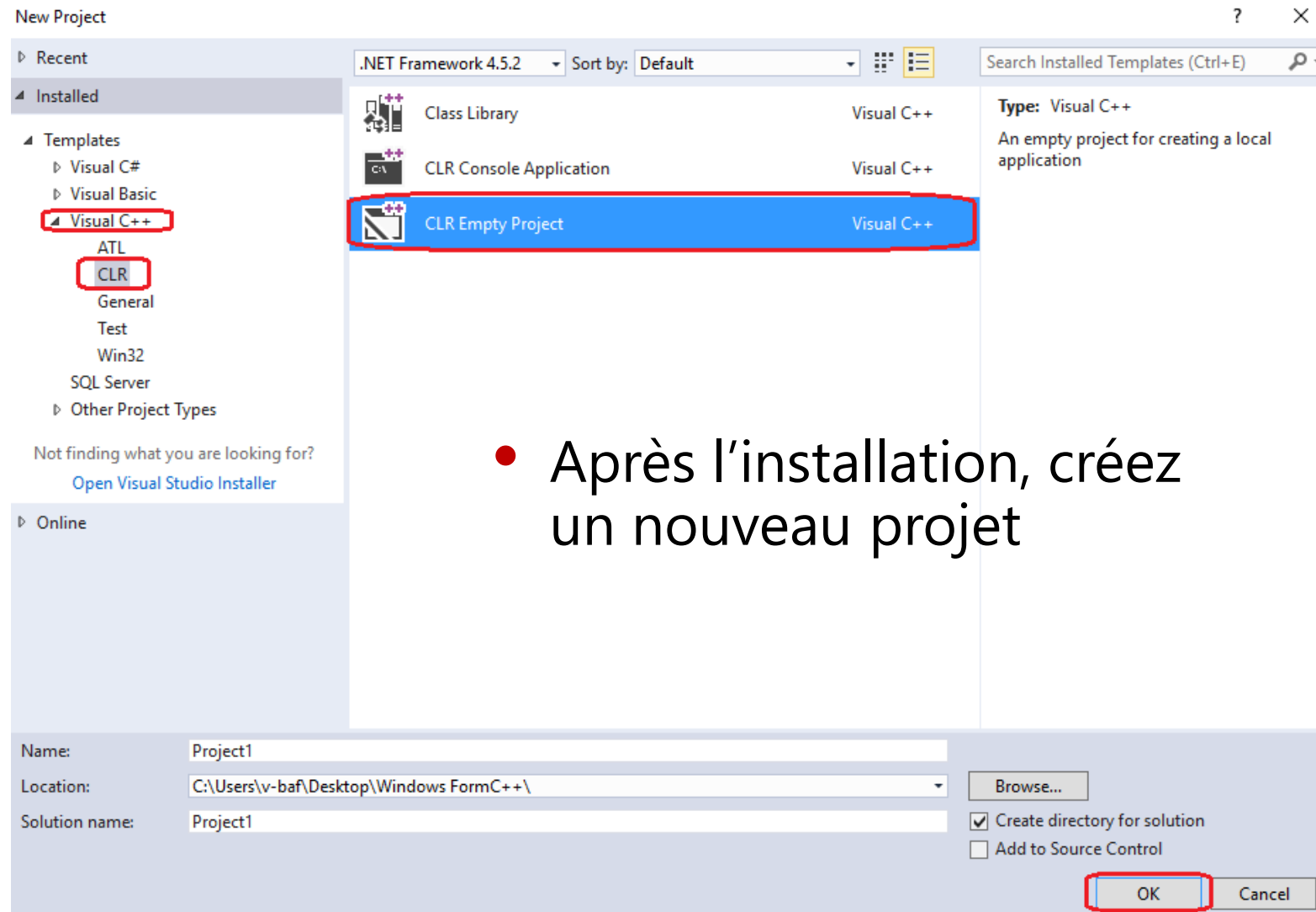
**Modify**

### Summary

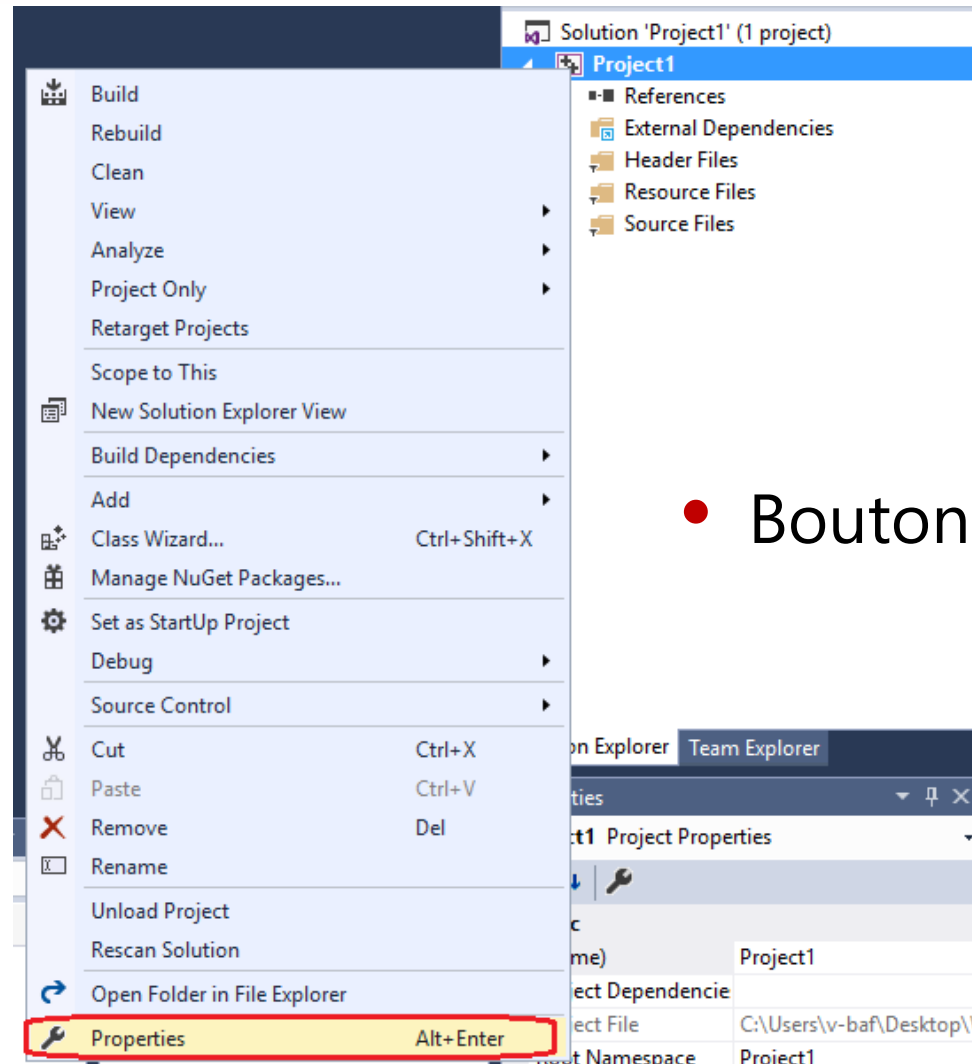
- > Visual Studio core editor
- > .NET desktop development
- > Desktop development with C++
- > ASP.NET and web developm... \*
- ∨ Individual components
  - Included
    - .NET Framework 4.6.1 SDK
    - .NET Framework 4.6.1 targeting pack
    - C++/CLI support
    - Microsoft Visual Studio 2017 Installer Pr...

**i** By continuing, you agree to the [license](#) for the Visual Studio edition you selected. We also offer the ability to download other software with Visual Studio. This software is licensed separately, as set out in the [3rd Party Notices](#) or in its accompanying license. By continuing, you also agree to those licenses.

# Mise en place d'un projet (1)



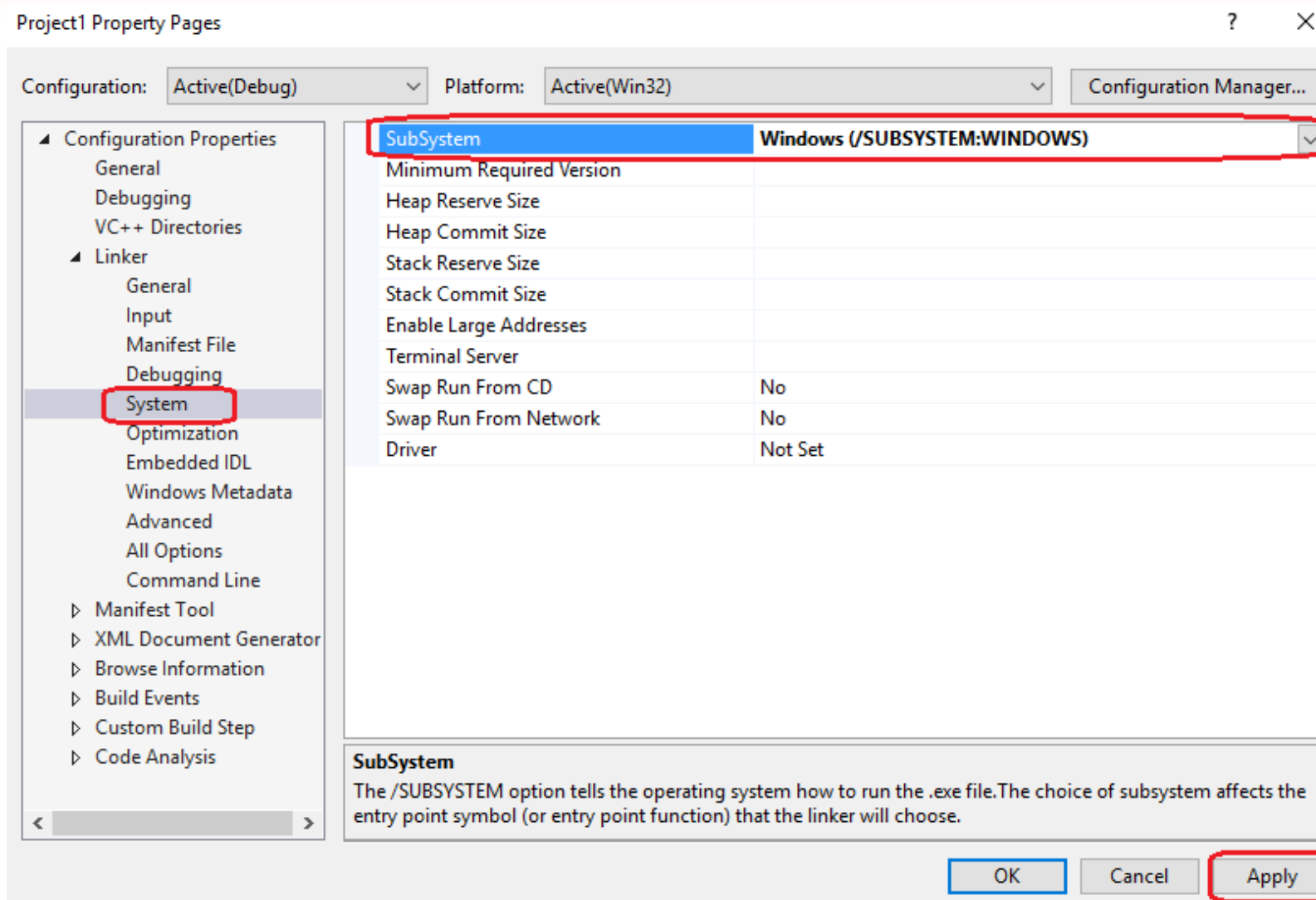
# Mise en place d'un projet (2)



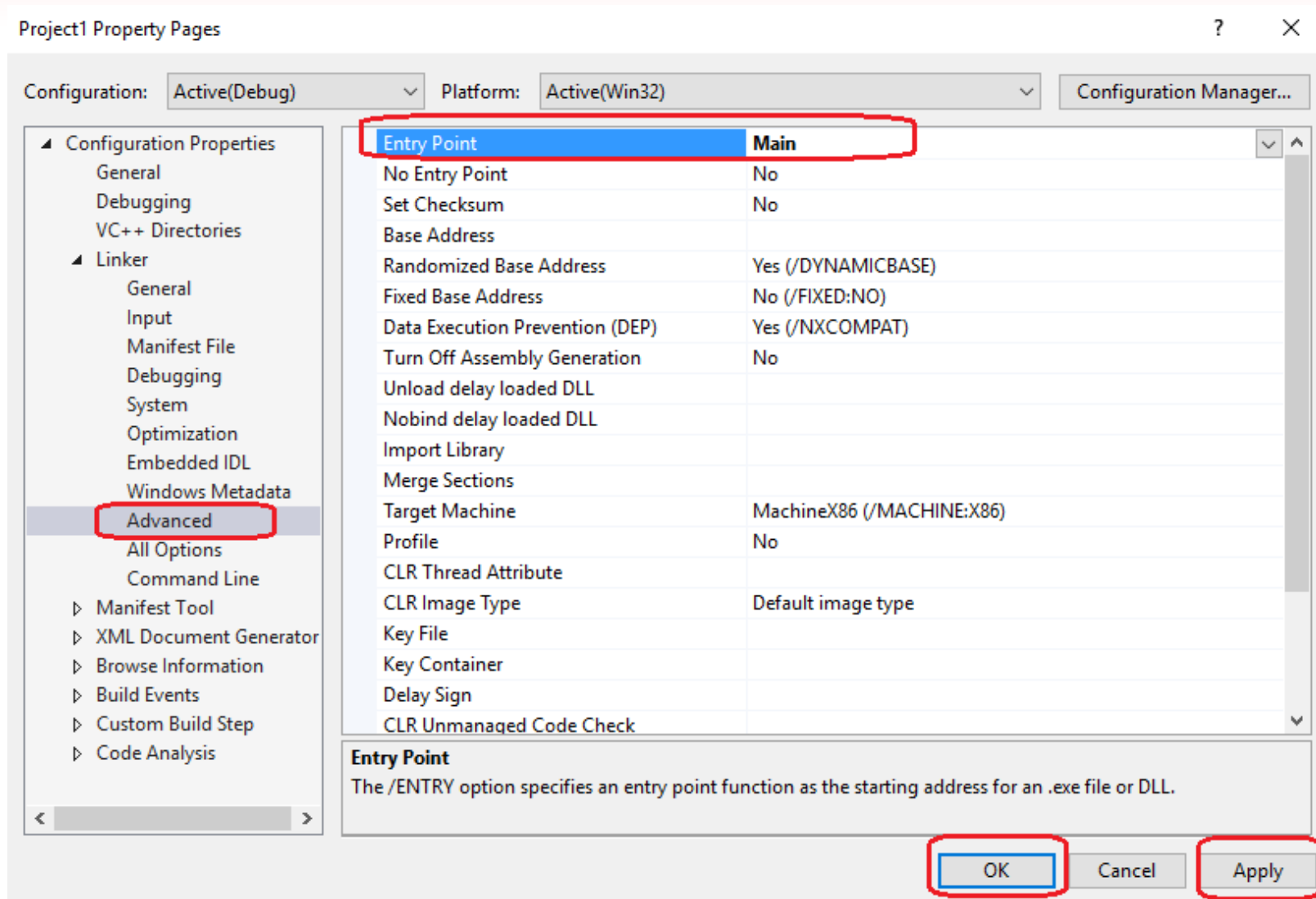
- Bouton droit sur Project1



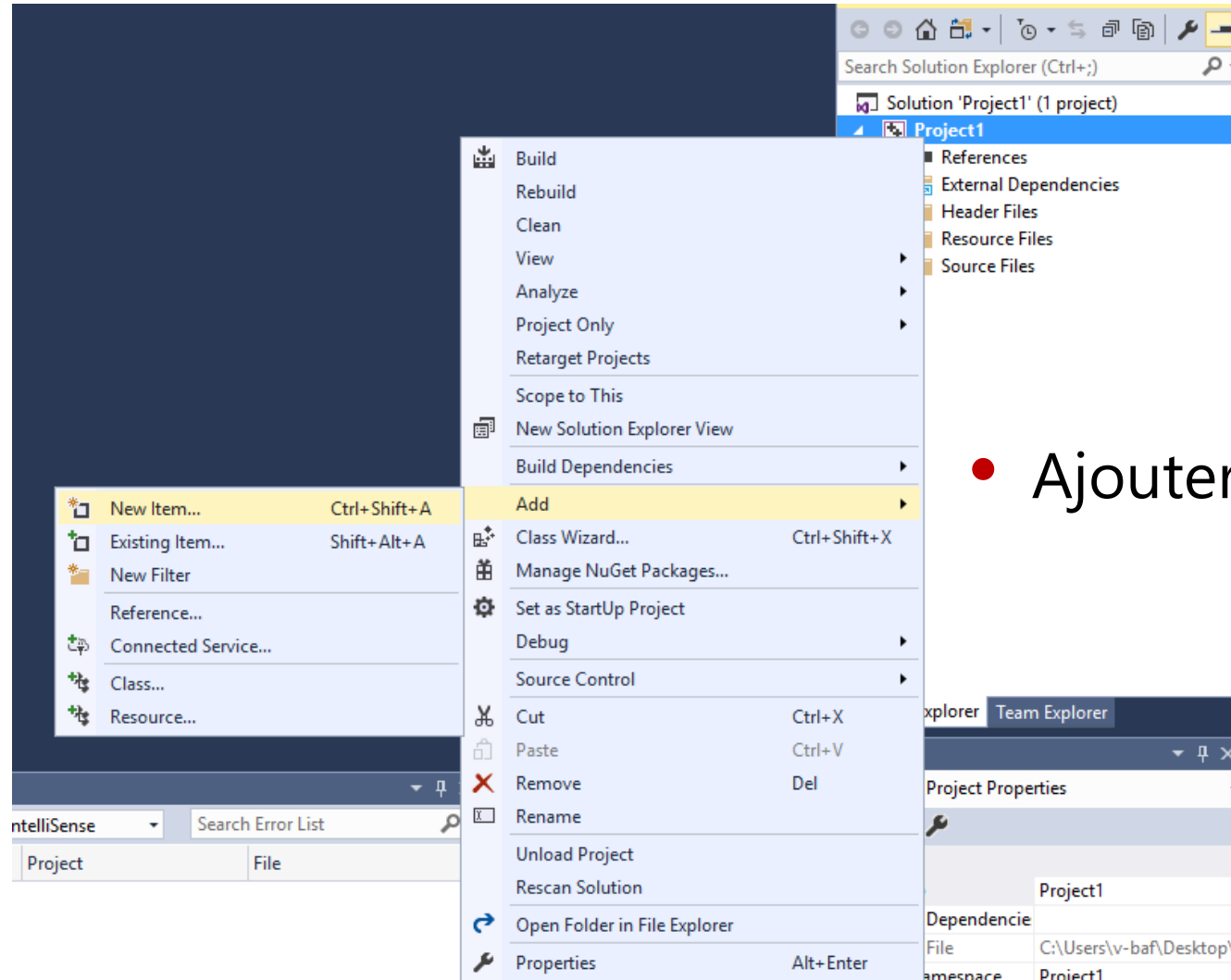
# Mise en place d'un projet (3)



# Mise en place d'un projet (4)

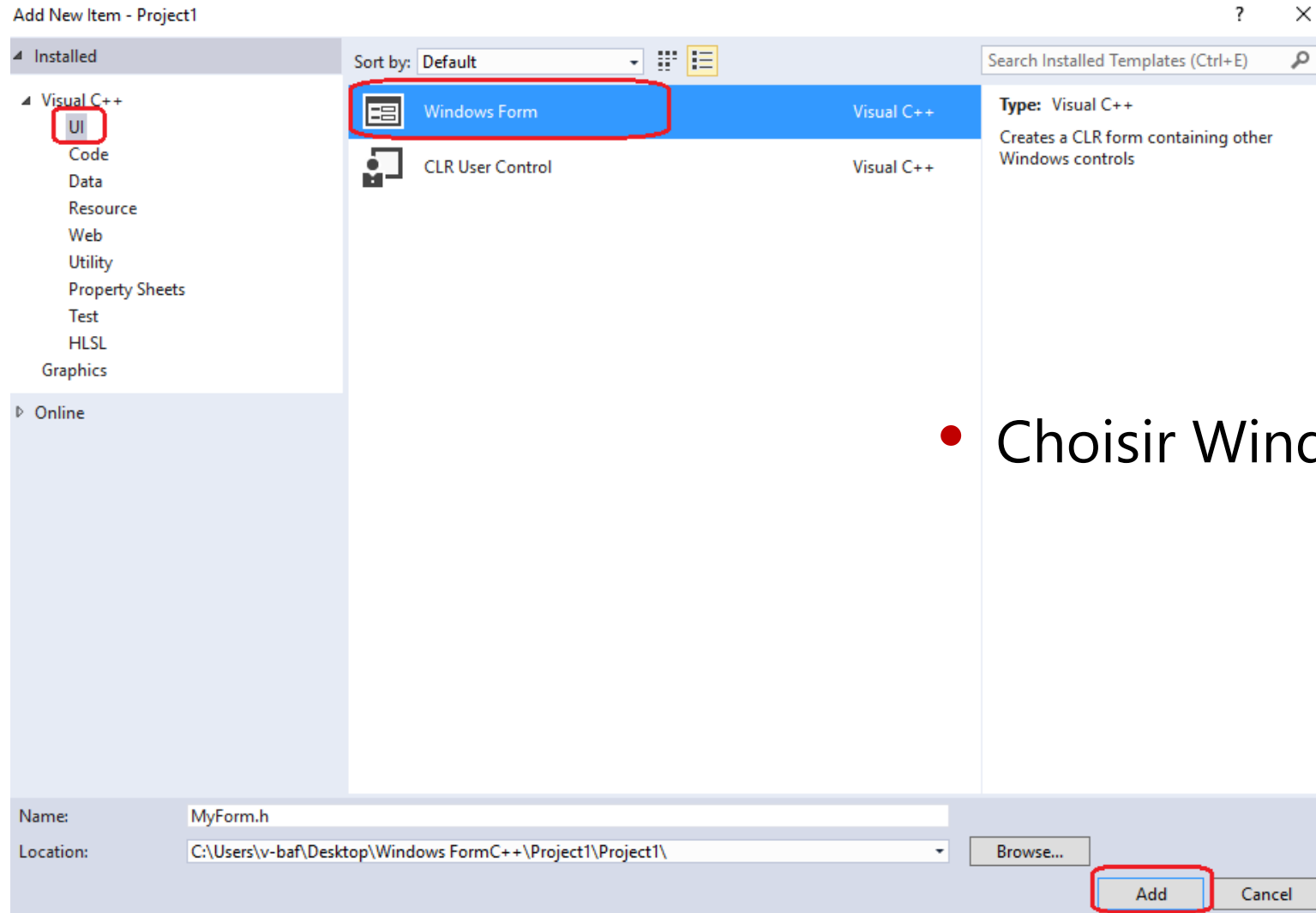


# Mise en place d'un projet (5)



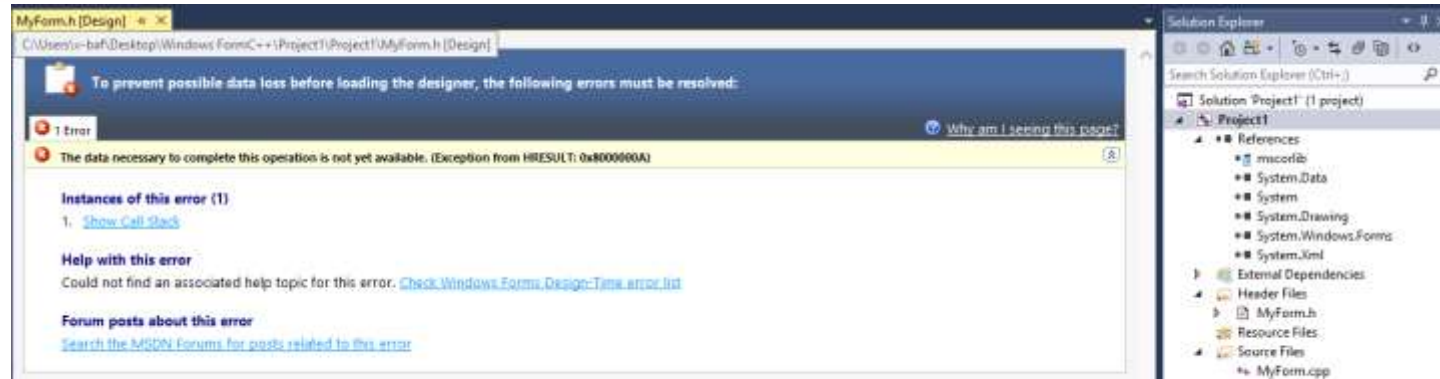
- Ajouter un nouvel élément

# Mise en place d'un projet (6)



- Choisir Windows Form

# Mise en place d'un projet (7)



- Fermez cette fenêtre d'erreur, et copiez le code suivant à MyForm.cpp:

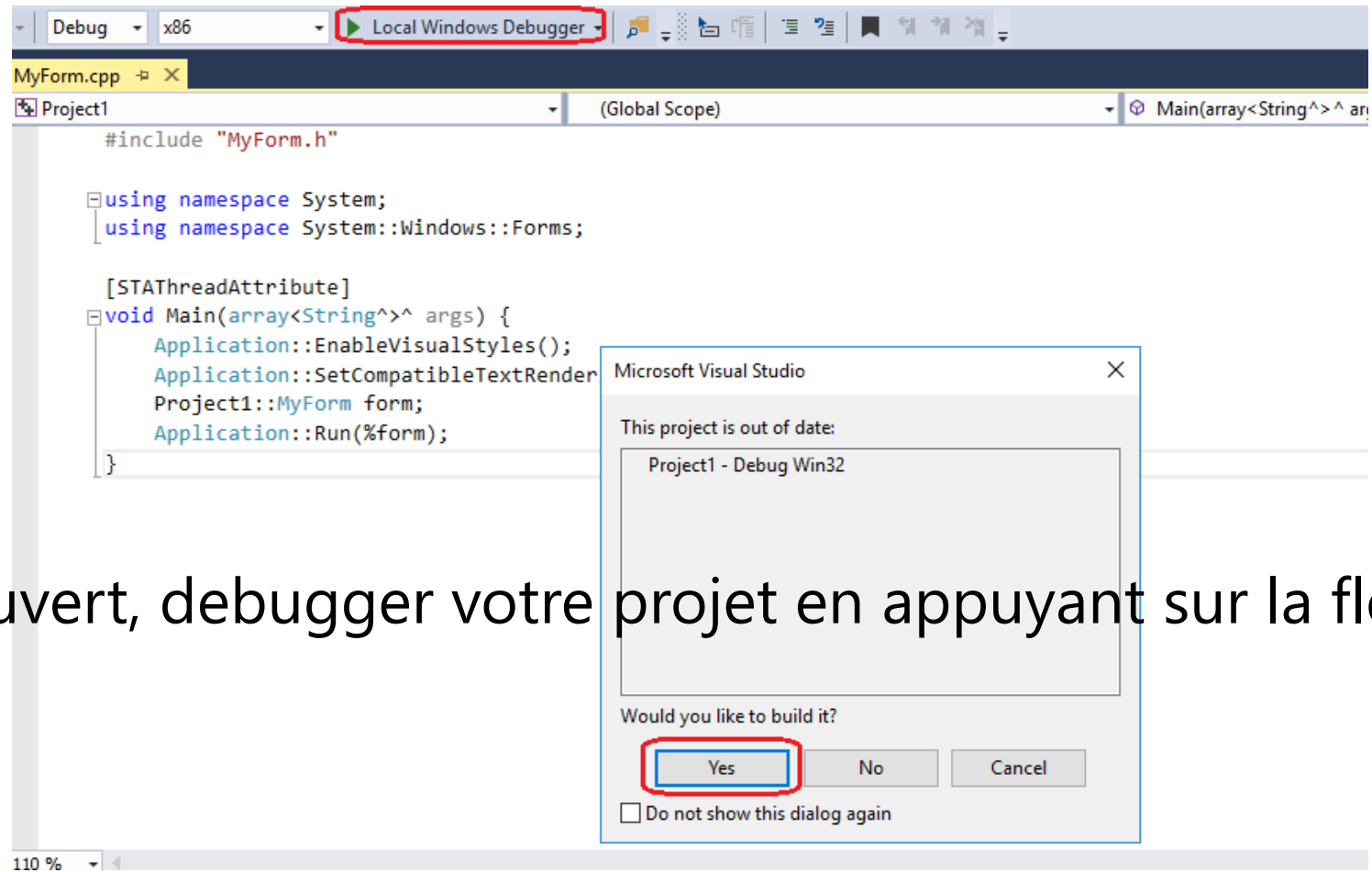
```
MyForm.cpp*  -  X
Project1      (Global Scope)
#include "MyForm.h"

using namespace System;
using namespace System::Windows::Forms;

[STAThreadAttribute]
void Main(array<String^>^ args) {
    Application::EnableVisualStyles();
    Application::SetCompatibleTextRenderingDefault(false);
    Project1::MyForm form;
    Application::Run(%form);
}
```

# Mise en place d'un projet (8)

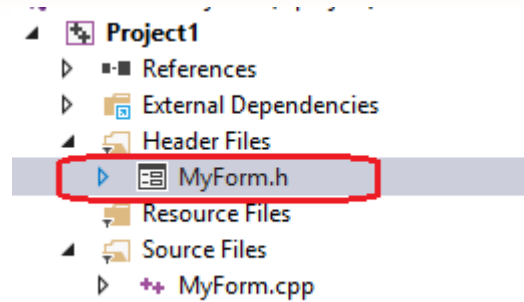
- Enregistrez, fermez et réouvrez VS 2017, puis votre projet1



- Une fois ouvert, debugger votre projet en appuyant sur la flèche verte

# Mise en place d'un projet (9)

- En cliquant sur MyForm.h



- Vous avez accès à la Toolbox : button, label, picturebox, etc ...

