



TP WPF

Les principes de base

Lemettre Arnaud

Version 1.0

10 Pages

31/08/2010

EPITA-MTI2014-WPF-TP-base

```
<Button Content="Prof-MTI">  
  <Button.Triggers>  
    <EventTrigger RoutedEvent="Button.Click">  
      <EventTrigger.Actions>  
        <BeginStoryboard Storyboard="{StaticResource StudentLearning}"/>  
      </EventTrigger.Actions>  
    </EventTrigger>  
  </Button.Triggers>  
</Button>
```

Propriétés du document

Auteur	Lemettre Arnaud
Version	1.0
Nombre de pages	10
Références	EPITA-MTI2014-WPF-TP-base

Historique du document

Date de vision	Version	Auteur	Changements
05/07/2010	0.1	Lemettre	création

Site de référence

Description	url
Site MTI	
Blog MTI	

Sommaire

Introduction	4
Partie 1 : Création de l'interface.....	5
Renseignements	5
Travail à Faire.....	6
Partie 2 : Ajout des images	7
Renseignements	7
Travail à faire	7
Partie 3 : Utilisation des styles.....	8
Renseignements	8
Travail à faire	8
Partie 4 : Utilisation des shaders	9
Renseignements :	9
Travail à faire :	9
Modalité de rendu	10

Introduction

Le but de ce TP est de se familiariser avec la technologie WPF de Microsoft.

Durant ce TP, les notions suivantes seront abordées :

- Réalisation d'une interface simple
- Les bases du binding
- Utilisation des styles

Ce travail est à faire individuellement, tout code similaire sur deux personnes sera considéré comme un travail non rendu et non négociable, il est recommandé de lire le sujet jusqu'à la fin.

Toute mention de login_l fait référence à votre login EPITA.

Ce TP ne demande pas plus de 4h de travail.

Bonne chance ;)

Partie 1 : Création de l'interface

Renseignements

Dans cette partie, vous allez devoir réaliser une interface similaire à l'image ci dessous.

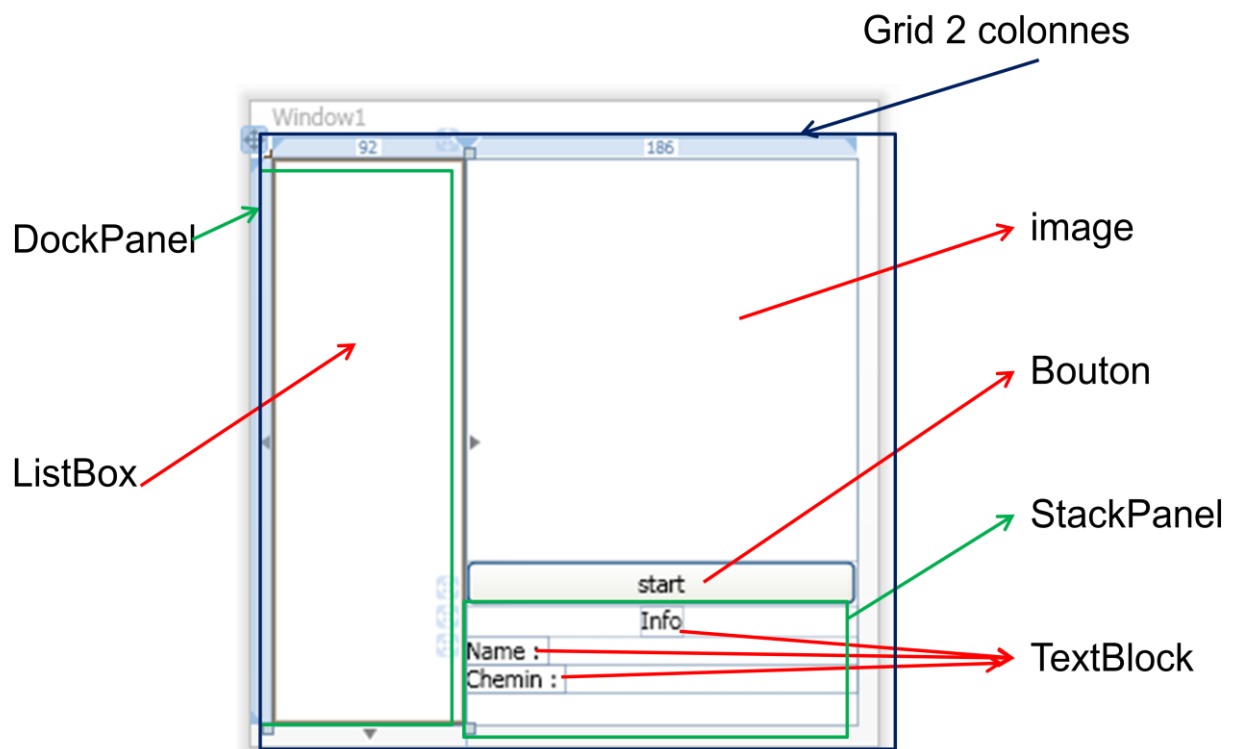


Figure 1 : Interface à réaliser

Cette interface est composée comme suit :

- D'une grille
 - o D'un dockpanel
 - D'une listbox
 - o D'un stackpanel
 - D'une image
 - D'un bouton
 - D'un stackpanel
 - D'un textblock
 - D'un stackpanel
 - o D'un textblock
 - o D'un textbox
 - D'un stackpanel
 - o D'un textblock
 - o D'un textbox

La grille est composée de deux colonnes permettant de pouvoir positionner plus facilement les différents éléments de l'interface.

Travail à Faire

Vous devez créer une solution visual studio, avec un projet de type WPF/C#. Vous pouvez créer cette solution avec le framework 4.5.

Le nom de la solution devra se nommer login_ITP1

Le nom du projet devra être WPFTp1

Pour l'ensemble du sujet du TP, il n'y aura qu'un seul projet à rendre.

Partie 2 : Ajout des images

Renseignements

Dans le but d'avoir des images à afficher dans notre interface, nous allons devoir ajouter un peu de code dans notre classe de code associée à notre interface. Nous allons donc créer une classe qui contiendra les informations que nous devons stocker et qui seront utilisées pour le reste de l'application.

Travail à faire

Créer une nouvelle classe contenant les informations suivantes, avec les getters / setters associés :

```
public class PictureInfo
{
    #region variables
    private string _path;
    private string _name;
    #endregion

    [getter / setter]
}
```

Figure 2 : Informations sur la classe

Pour charger nos images, nous allons inclure notre traitement à la construction de notre interface donc dans le constructeur de la classe. Pour cela vous allez ajouter le code suivant, dans votre classe, qui parcourt un répertoire et ajoute l'ensemble des images *.jpg à notre collection :

```
private void FillList()
{
    List<PictureInfo> list = new List<PictureInfo>();
    foreach (string item in Directory.GetFiles("Pictures/", "*.jpg"))
    {
        FileInfo info = new FileInfo(item);
        PictureInfo picture = new PictureInfo() { Name = info.Name, Path =
info.DirectoryName };
        list.Add(picture);
    }
    /*FIX ME : code pour relier une liste à l'interface*/
}
```

Il ne vous reste plus qu'à binder votre liste d'images sur la listbox de votre interface => il faudra utiliser son datacontext.

A ce stade il vous faut encore créer votre DataTemplate afin que votre listbox puisse afficher les informations. Pour ce DataTemplate on peut imaginer :

- Le nom du fichier
- Une miniature de l'image associée

Lorsque ce template sera créé, il ne vous restera qu'à l'associer à votre listbox.

N'oubliez pas de binder également les éléments restant de l'interface concernant les textbox qui devront afficher les éléments du chemin et du nom de fichier. De la même manière, vous utiliserez les bindings basés sur l'élément listbox et sa propriété current item.

Partie 3 : Utilisation des styles

Renseignements

Une des fonctionnalités les plus intéressantes de WPF est le fait de pouvoir customiser les éléments de l'interface. Pour cela on passe par l'utilisation de style.

Travail à faire

Dans cette partie, vous allez créer un style sur les boutons qui se déclenchera au survol de ceux-ci. Lorsque la souris sortira du bouton ils reprendront leur aspect initial.

Le style à appliquer sera de cette forme :

- Le background du bouton sera de couleur bleue
- Le texte dans le bouton devra être de couleur blanche

Partie 4 : Utilisation des shaders

Renseignements :

Afin de pouvoir utiliser les shaders, vous allez devoir utiliser la bibliothèque fournie pour la réalisation de ce TP. Pour l'utiliser vous devez l'ajouter en tant que référence sur votre projet.

Travail à faire :

Lors du clic sur le bouton de votre interface, ceci aura pour effet de modifier l'image en cours d'affichage par un des shaders que vous avez décidé d'utiliser.

Modalité de rendu

Les fichiers seront à rendre dans une tarball ayant pour nom :

login_.zip

Cette tarball devra comprendre à la racine:

- Un dossier contenant la solution Visual Studio qui devra compiler :

Nom : login_ITP1

Une fois décompressée nous devrions avoir :

/Login_ITP1

*.sln

/WPFTp1

Le tout à envoyer sur l'adresse mti.rendu.dotnet@gmail.com avec les balises suivantes :

[MTI2014][WPF][login_I][TP1]