

Windows Phone 7 - Tutoriel 4 : Éditeur de texte

Éditeur de texte avec 'Nouveau', 'Ouvrir' et 'Enregistrer' sur Windows Phone 7

par Peter Bull ([Blog](#)) Deepin Prayag (Traduction) ([Home](#))

Date de publication : 21/03/2012

Dernière mise à jour : 21/03/2012

Ce tutoriel fait partie d'une série de niveau débutant-intermédiaire pour apprendre Windows Phone 7 par le biais d'exemples pas à pas.

Langage : [C#](#)

Public visé : niveau *Intermédiaire*

Commentez ce tutoriel :

Traduction.....	3
Introduction.....	3
Éditeur de texte avec 'Nouveau', 'Ouvrir' et 'Enregistrer' sur Windows Phone 7.....	3
Étape 1.....	3
Étape 2.....	4
Étape 3.....	5
Étape 4.....	6
Étape 5.....	6
Étape 6.....	7
Étape 7.....	8
Étape 8.....	9
Étape 9.....	11
Étape 10.....	12
Étape 11.....	13
Étape 12.....	14
Étape 13.....	15
Étape 14.....	17
Étape 15.....	18
Étape 16.....	19
Étape 17.....	20
Étape 18.....	21
Étape 19.....	22
Étape 20.....	23
Étape 21.....	24
Étape 22.....	25
Conclusion.....	25
Liens.....	25
Remerciements.....	26

Traduction

Ce tutoriel est la traduction la plus fidèle possible du tutoriel original de **Peter Bull, Windows Phone 7 - Text Editing application with New, Open and Save on Windows Phone 7.**

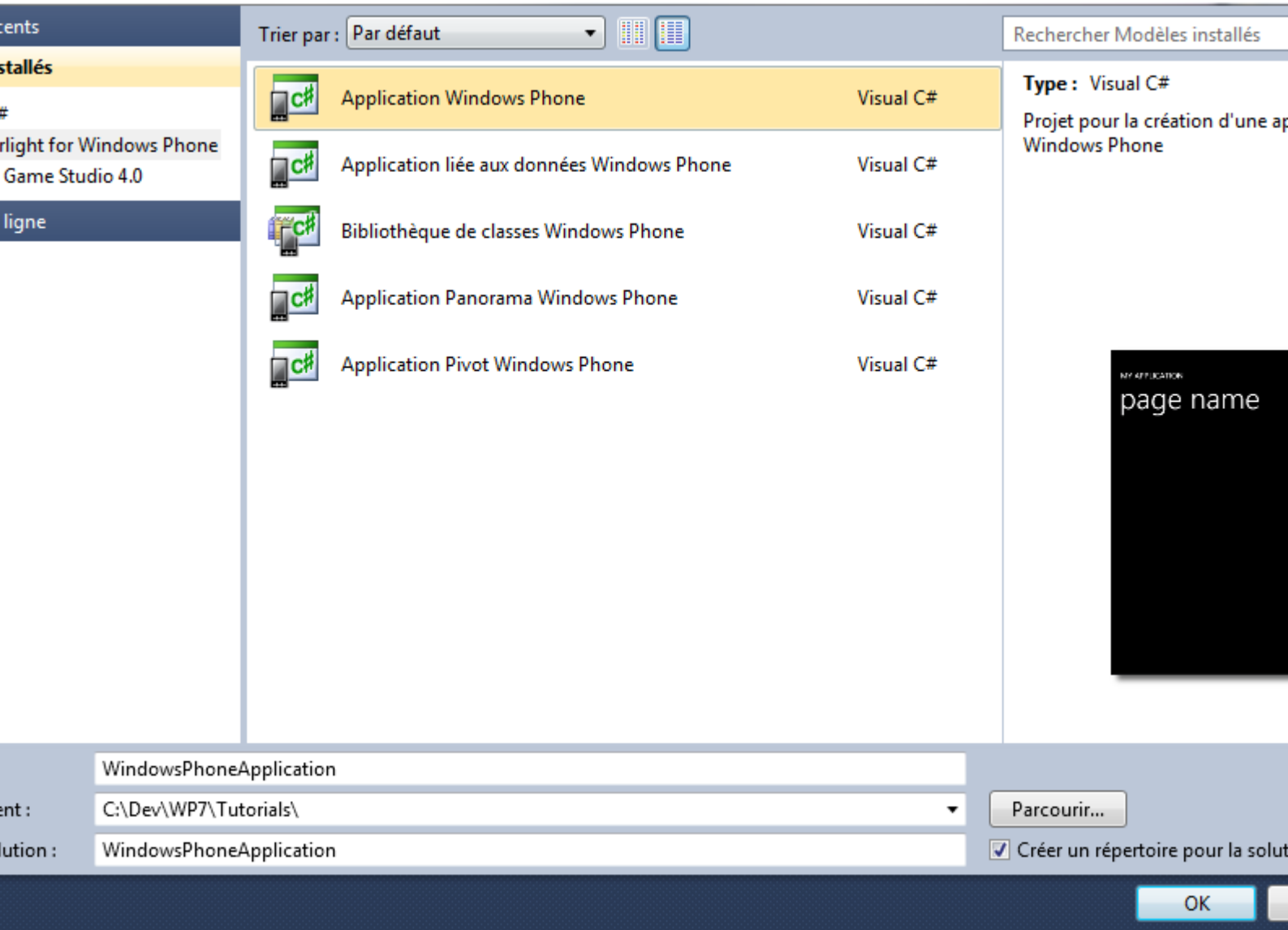
Introduction

L'une des tâches qu'on effectue le plus couramment est l'édition de texte, et en utilisant Silverlight avec Windows Phone 7 il est possible de créer un éditeur de texte basique avec les supports *Ouvrir* et *Enregistrer* pour le dispositif.

Éditeur de texte avec 'Nouveau', 'Ouvrir' et 'Enregistrer' sur Windows Phone 7

Étape 1

Démarrez Microsoft Visual Web Developer 2010 Express pour Windows Phone, puis sélectionnez *Fichier* puis *Nouveau Projet...* Sélectionnez *Visual C#* ensuite *Silverlight for Windows Phone*, puis *Application Windows Phone* dans les modèles installés, sélectionnez un emplacement si vous le souhaitez, puis entrez un nom pour le projet et appuyez sur OK :



Étape 2

Une page *Application Windows Phone* nommée *MainPage.xaml* devrait alors apparaître :



Étape 3

Faites un clic droit sur *App.xaml* dans l'*Explorateur de solutions* pour la *solution* et choisissez *Afficher le code*, puis dans la *vue de code* pour *App.xaml.cs* au-dessus de `public App()` tapez le code suivant :

```
private string _filename = "";
private object _content = null;

public string Filename
{
    get { return _filename; }
    set { _filename = value; }
}

public object Content
{
    get { return _content; }
    set { _content = value; }
}
```

```

namespace TextEditor
{
    public partial class App : Application
    {
        // Easy access to the root frame
        public PhoneApplicationFrame RootFrame { get; private set; }

        private string _filename = "";
        private object _content = null;

        public string Filename
        {
            get { return _filename; }
            set { _filename = value; }
        }

        public object Content
        {
            get { return _content; }
            set { _content = value; }
        }

        // Constructor
        public App()
    }
}

```

Étape 4

Retournez au *Concepteur de vues MainPage* en sélectionnant l'onglet *MainPage.xaml*. Ensuite dans le volet XAML au-dessus de la ligne `<Grid x:Name="LayoutRoot" Background="{StaticResource PhoneBackgroundBrush}">` tapez le code XAML suivant :

```

<phone:PhoneApplicationPage.ApplicationBar>
  <shell:ApplicationBar IsVisible="True" IsMenuEnabled="True">
    <shell:ApplicationBar.MenuItems>
      <shell:ApplicationBarMenuItem Text="new" Click="New_Click"/>
      <shell:ApplicationBarMenuItem Text="open" Click="Open_Click"/>
      <shell:ApplicationBarMenuItem Text="save" Click="Save_Click"/>
    </shell:ApplicationBar.MenuItems>
  </shell:ApplicationBar>
</phone:PhoneApplicationPage.ApplicationBar>

```

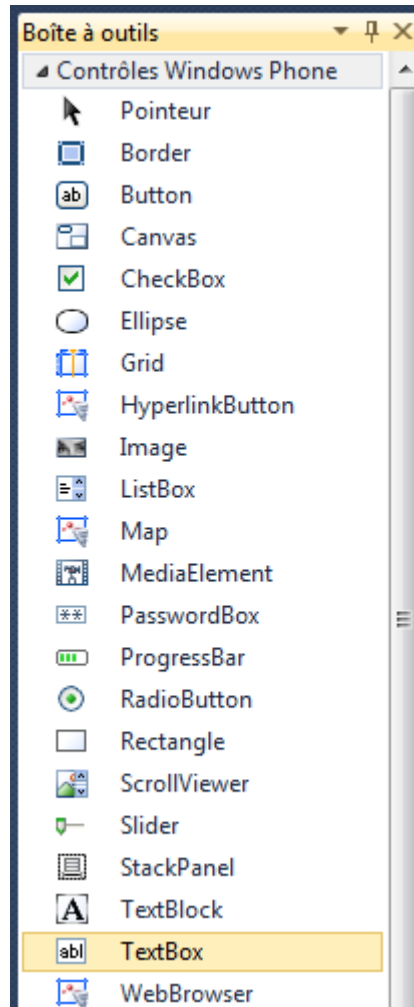
```

<phone:PhoneApplicationPage.ApplicationBar>
  <shell:ApplicationBar IsVisible="True" IsMenuEnabled="True">
    <shell:ApplicationBar.MenuItems>
      <shell:ApplicationBarMenuItem Text="new" Click="New_Click"/>
      <shell:ApplicationBarMenuItem Text="open" Click="Open_Click"/>
      <shell:ApplicationBarMenuItem Text="save" Click="Save_Click"/>
    </shell:ApplicationBar.MenuItems>
  </shell:ApplicationBar>
</phone:PhoneApplicationPage.ApplicationBar>

```

Étape 5

Puis dans la section *Contrôles Windows Phone* de la *Boîte à outils*, sélectionnez le contrôle *TextBox* :



Étape 6

Dessinez un *TextBox* sur la page, en dessous du titre de la page, et dans le volet XAML entre les lignes `<Grid x:Name="ContentGrid" Grid.Row="1">` et `</Grid>`, modifiez *TextBox1* comme ceci :

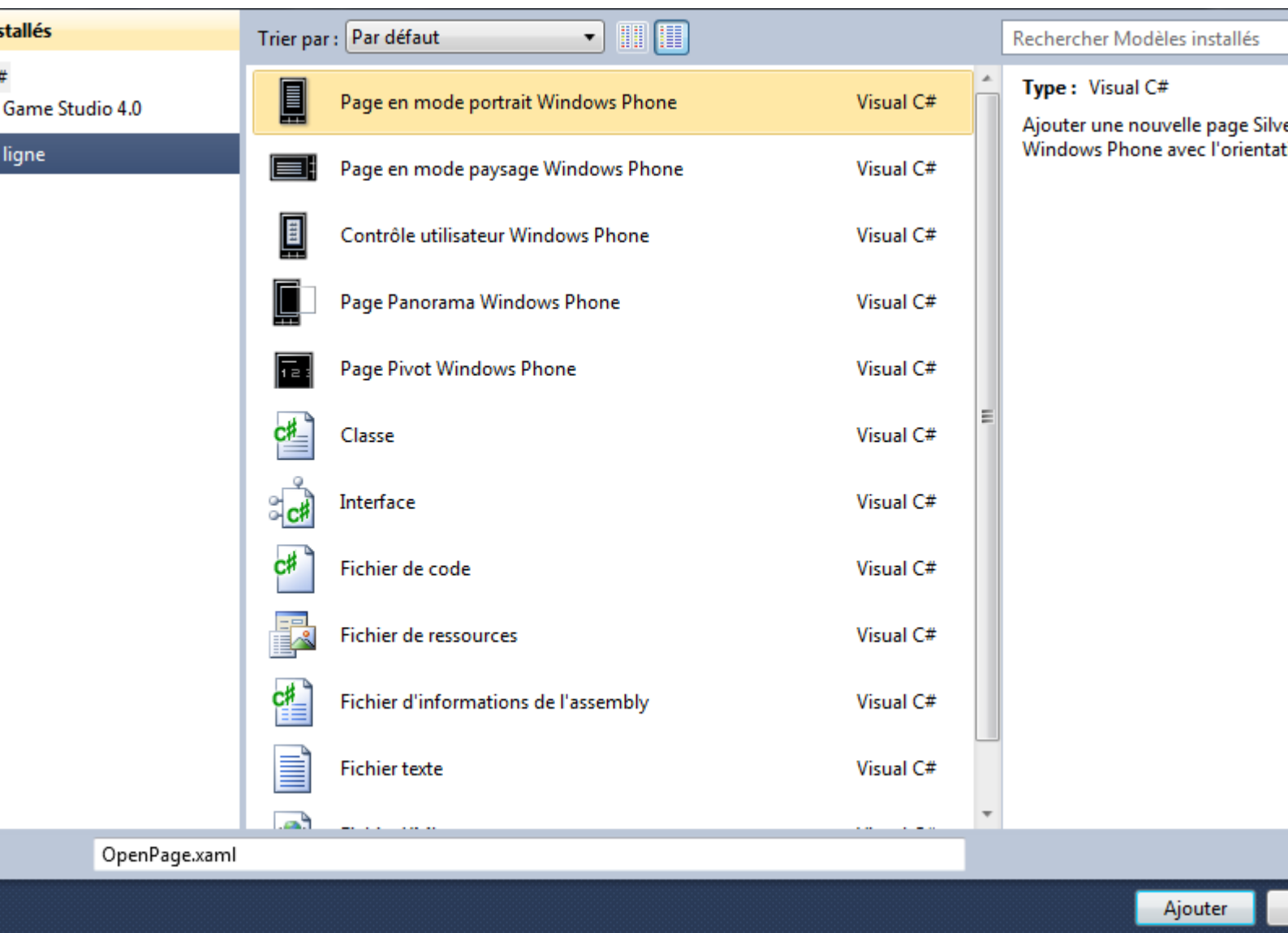
```
<TextBox AcceptsReturn="True" TextWrapping="Wrap" VerticalScrollBarVisibility="Auto" FontSize="24" Name="Editor"/>
```

Voir ci-dessous :



Étape 7

Sélectionnez *Projet* ensuite *Ajouter un nouvel élément...* Sélectionnez le modèle *Page en mode portrait Windows Phone*, ensuite modifiez le *Nom* à *OpenPage.xaml* :



Étape 8

Dans le *Concepteur de vues* pour *OpenPage.xaml*, dans le volet XAML entre les lignes `<Grid Grid.Row="1" x:Name="ContentGrid">` et `</Grid>` tapez le code XAML suivant :

```
<Grid x:Name="ContentMain">
  <Grid.RowDefinitions>
    <RowDefinition Height="*" />
    <RowDefinition Height="Auto" />
  </Grid.RowDefinitions>
  <!-- Content -->
  <ListBox Grid.Row="0" Margin="10" FontSize="48" Name="Files" />
  <Grid Grid.Row="1" x:Name="Buttons" >
    <Grid.ColumnDefinitions>
      <ColumnDefinition Width="160" />
      <ColumnDefinition Width="160" />
      <ColumnDefinition Width="160" />
    </Grid.ColumnDefinitions>
    <!-- Buttons -->
    <Button Grid.Column="0" Content="open" Click="Open_Click" />
    <Button Grid.Column="1" Content="delete" Click="Delete_Click" />
    <Button Grid.Column="2" Content="cancel" Click="Cancel_Click" />
  </Grid>
</Grid>
```

XAML :

```

<Grid x:Name="ContentGrid" Grid.Row="1">
  <Grid x:Name="ContentMain">
    <Grid.RowDefinitions>
      <RowDefinition Height="*" />
      <RowDefinition Height="Auto" />
    </Grid.RowDefinitions>
    <!-- Content -->
    <ListBox Grid.Row="0" Margin="10" FontSize="48" Name="Files" />
    <Grid Grid.Row="1" x:Name="Buttons" >
      <Grid.ColumnDefinitions>
        <ColumnDefinition Width="160" />
        <ColumnDefinition Width="160" />
        <ColumnDefinition Width="160" />
      </Grid.ColumnDefinitions>
      <!-- Buttons -->
      <Button Grid.Column="0" Content="open" Click="Open_Click" />
      <Button Grid.Column="1" Content="delete" Click="Delete_Click" />
      <Button Grid.Column="2" Content="cancel" Click="Cancel_Click" />
    </Grid>
  </Grid>
</Grid>

```

Design :



Étape 9

Faites un clic droit sur la page ou sur l'entrée pour *OpenPage.xaml* dans l'*Explorateur de solutions* et choisissez l'option *Afficher le code*. Dans la *vue de code* au-dessus de `namespace` `TextEditor` tapez le code suivant :

```
using System.IO.IsolatedStorage;
```

Toujours dans la *vue de code*, au-dessus de `public` `OpenPage ()` tapez la déclaration *Application* suivante :

```
public App app = (App)Application.Current;
```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Net;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Documents;
using System.Windows.Input;
using System.Windows.Media;
using System.Windows.Media.Animation;
using System.Windows.Shapes;
using Microsoft.Phone.Controls;
using System.IO.IsolatedStorage;

namespace TextEditor
{
    public partial class OpenPage : PhoneApplicationPage
    {
        public App app = (App)Application.Current;

        public OpenPage()
        {
            InitializeComponent();
        }
    }
}

```

Étape 10

Puisque nous en sommes à la *vue de code* pour *OpenPage.xaml.cs*, dans le *constructeur* `public OpenPage()` en dessous du `InitializeComponent();` ; tapez le code suivant :

```

ApplicationTitle.Text = "TEXT EDITOR";
PageTitle.Text = "open";
Loaded += (object sender, RoutedEventArgs e) =>
{
    Files.Items.Clear();
    using (IsolatedStorageFile storage = IsolatedStorageFile.GetUserStoreForApplication())
    {
        foreach (string filename in storage.GetFileNames("*.txt"))
        {
            Files.Items.Add(filename.ToLower());
        }
    }
};

```

```

public OpenPage()
{
    InitializeComponent();
    ApplicationTitle.Text = "TEXT EDITOR";
    PageTitle.Text = "open";
    Loaded += (object sender, RoutedEventArgs e) =>
    {
        Files.Items.Clear();
        using (IsolatedStorageFile storage = IsolatedStorageFile.GetUserStoreForApplication())
        {
            foreach (string filename in storage.GetFileNames("*.txt"))
            {
                Files.Items.Add(filename.ToLower());
            }
        }
    };
}
    
```

Étape 11

Puisque nous en sommes toujours à la *vue de code* pour *OpenPage.xaml.cs*, en dessous du « } » de la méthode `public OpenPage()` tapez les *gestionnaires d'évènements* suivants :

```

private void Open_Click(object sender, RoutedEventArgs e)
{
    if (Files.SelectedItem != null)
    {
        app.Filename = (string)Files.SelectedItem;
        using (IsolatedStorageFile storage = IsolatedStorageFile.GetUserStoreForApplication())
        {
            IsolatedStorageFileStream location = new IsolatedStorageFileStream(app.Filename,
                System.IO.FileMode.Open, storage);
            System.IO.StreamReader file = new System.IO.StreamReader(location);
            app.Content = file.ReadToEnd();
            file.Dispose();
            location.Dispose();
        }
        NavigationService.GoBack();
    }
}

private void Delete_Click(object sender, RoutedEventArgs e)
{
    if (Files.SelectedItem != null)
    {
        string _selected = (string)Files.SelectedItem;
        if (MessageBox.Show("Delete selected Document " + _selected + "?", "Text Editor",
            MessageBoxButton.OKCancel) == MessageBoxResult.OK)
        {
            using (IsolatedStorageFile storage = IsolatedStorageFile.GetUserStoreForApplication())
            {
                if (storage.FileExists(_selected))
                {
                    storage.DeleteFile(_selected);
                }
            }
            NavigationService.GoBack();
        }
    }
}

private void Cancel_Click(object sender, RoutedEventArgs e)
{
    NavigationService.GoBack();
}
    
```

```

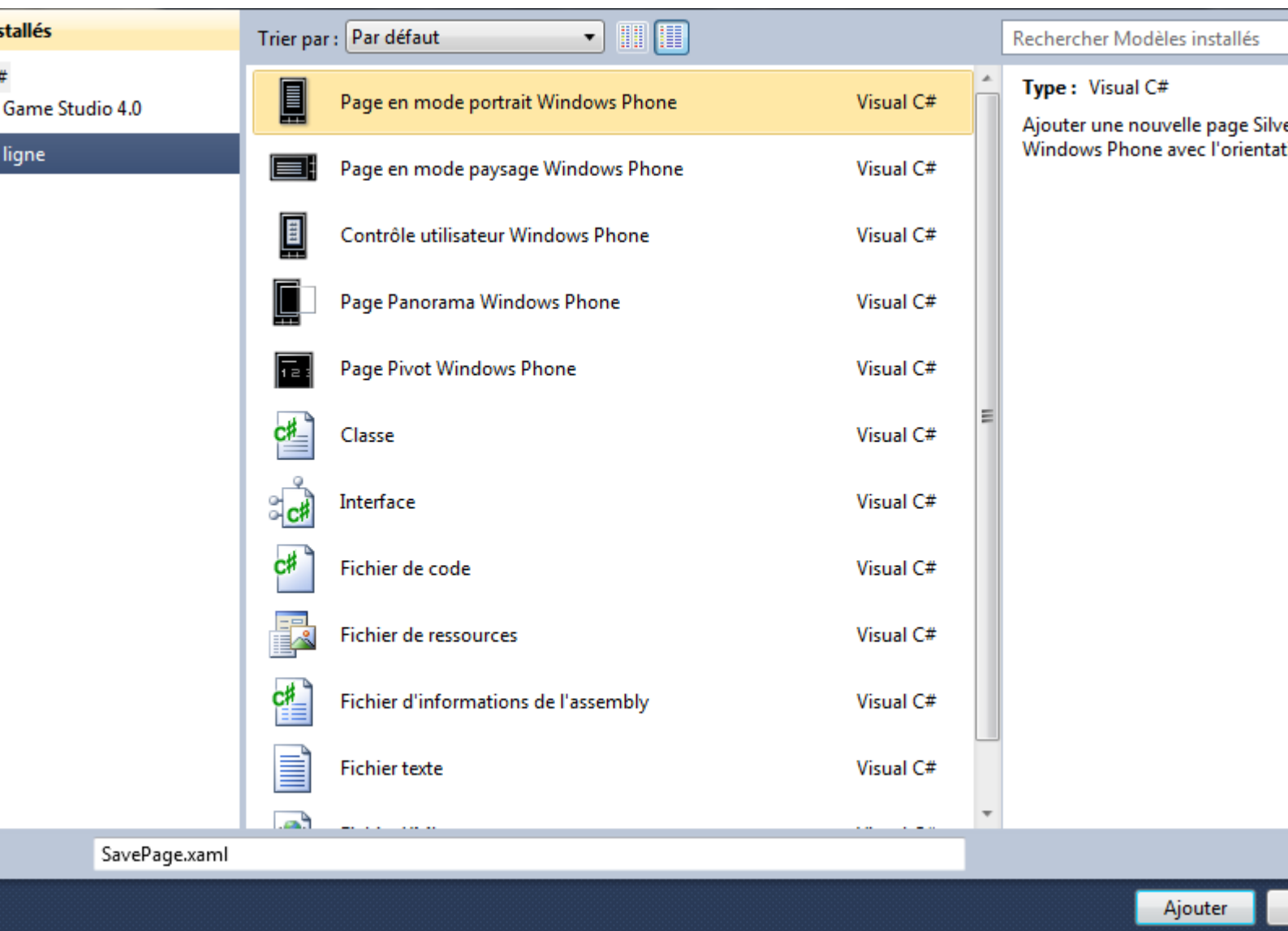
private void Open_Click(object sender, RoutedEventArgs e)
{
    if (Files.SelectedItem != null)
    {
        app.Filename = (string)Files.SelectedItem;
        using (IsolatedStorageFile storage = IsolatedStorageFile.GetUserStoreForApplication())
        {
            IsolatedStorageFileStream location = new IsolatedStorageFileStream(app.Filename,
                System.IO.FileMode.Open, storage);
            System.IO.StreamReader file = new System.IO.StreamReader(location);
            app.Content = file.ReadToEnd();
            file.Dispose();
            location.Dispose();
        }
        NavigationService.GoBack();
    }
}

private void Delete_Click(object sender, RoutedEventArgs e)
{
    if (Files.SelectedItem != null)
    {
        string _selected = (string)Files.SelectedItem;
        if (MessageBox.Show("Delete selected Document " + _selected + "?", "Text Editor",
            MessageBoxButton.OKCancel) == MessageBoxResult.OK)
        {
            using (IsolatedStorageFile storage = IsolatedStorageFile.GetUserStoreForApplication())
            {
                if (storage.FileExists(_selected))
                {
                    storage.DeleteFile(_selected);
                }
            }
            NavigationService.GoBack();
        }
    }
}

private void Cancel_Click(object sender, RoutedEventArgs e)
{
    NavigationService.GoBack();
}
    
```

Étape 12

Sélectionnez *Projet* ensuite *Ajouter un nouvel élément...* Sélectionnez le modèle *Page en mode portrait Windows Phone*, puis modifiez le *Nom* à *SavePage.xaml* :



Étape 13

Dans le *Concepteur de vues* pour *SavePage.xaml*, dans le volet XAML entre les lignes `<Grid Grid.Row="1" x:Name="ContentGrid">` et `</Grid>` tapez le code XAML suivant :

```
<Grid x:Name="ContentMain">
  <Grid.RowDefinitions>
    <RowDefinition Height="Auto"/>
    <RowDefinition Height="*/>
    <RowDefinition Height="Auto"/>
  </Grid.RowDefinitions>
  <!-- Content -->
  <TextBox Grid.Row="0" FontSize="24" Name="Filename">
    <TextBox.InputScope>
      <InputScope>
        <InputScopeName NameValue="FileName"/>
      </InputScope>
    </TextBox.InputScope>
  </TextBox>
  <Grid Grid.Row="2" x:Name="Buttons" >
    <Grid.ColumnDefinitions>
      <ColumnDefinition Width="160"/>
      <ColumnDefinition Width="160"/>
      <ColumnDefinition Width="160"/>
    </Grid.ColumnDefinitions>
  </Grid>
</Grid>
```

```

</Grid.ColumnDefinitions>
<!-- Buttons -->
<Button Grid.Column="0" Content="save" Click="Save_Click"/>
<Button Grid.Column="2" Content="cancel" Click="Cancel_Click"/>
</Grid>
</Grid>

```

```

<Grid x:Name="ContentGrid" Grid.Row="1">
  <Grid x:Name="ContentMain">
    <Grid.RowDefinitions>
      <RowDefinition Height="Auto"/>
      <RowDefinition Height="*/>
      <RowDefinition Height="Auto"/>
    </Grid.RowDefinitions>
    <!-- Content -->
    <TextBox Grid.Row="0" FontSize="24" Name="Filename">
      <TextBox.InputScope>
        <InputScope>
          <InputScopeName NameValue="FileName"/>
        </InputScope>
      </TextBox.InputScope>
    </TextBox>
    <Grid Grid.Row="2" x:Name="Buttons" >
      <Grid.ColumnDefinitions>
        <ColumnDefinition Width="160"/>
        <ColumnDefinition Width="160"/>
        <ColumnDefinition Width="160"/>
      </Grid.ColumnDefinitions>
      <!-- Buttons -->
      <Button Grid.Column="0" Content="save" Click="Save_Click"/>
      <Button Grid.Column="2" Content="cancel" Click="Cancel_Click"/>
    </Grid>
  </Grid>
</Grid>

```

Design :



Étape 14

Faites un clic droit sur la page ou sur l'entrée pour *SavePage.xaml* dans l'*Explorateur de solutions* et choisissez l'option *Afficher le code*. Dans la *vue de code* au-dessus de `namespace TextEditor` tapez le code suivant :

```
using System.IO.IsolatedStorage;
```

Toujours dans la *vue de code* au-dessus de `public SavePage ()` tapez la déclaration *Application* suivante :

```
public App app = (App)Application.Current;
```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Net;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Documents;
using System.Windows.Input;
using System.Windows.Media;
using System.Windows.Media.Animation;
using System.Windows.Shapes;
using Microsoft.Phone.Controls;
using System.IO.IsolatedStorage;

namespace TextEditor
{
    public partial class SavePage : PhoneApplicationPage
    {
        public App app = (App)Application.Current;

        public SavePage()
        {
            InitializeComponent();
        }
    }
}

```

Étape 15

Puisque nous en sommes à la *vue de code* pour *SavePage.xaml.cs*, dans le *constructeur* `public SavePage()` en dessous de `InitializeComponent();` tapez le code suivant :

```

ApplicationTitle.Text = "TEXT EDITOR";
PageTitle.Text = "save";
Loaded += (object sender, RoutedEventArgs e) =>
{
    if (app.Filename == "")
    {
        Filename.Text = "untitled.txt";
    }
    else
    {
        Filename.Text = app.Filename;
    }
};

```

```

public SavePage()
{
    InitializeComponent();
    ApplicationTitle.Text = "TEXT EDITOR";
    PageTitle.Text = "save";
    Loaded += (object sender, RoutedEventArgs e) =>
    {
        if (app.Filename == "")
        {
            Filename.Text = "untitled.txt";
        }
        else
        {
            Filename.Text = app.Filename;
        }
    };
}

```

Étape 16

Puisque nous en sommes toujours à la *vue de code* pour *SavePage.xaml.cs*, en dessous du « } » de la méthode `public SavePage()` tapez les *gestionnaires d'événements* suivants :

```

private void Save_Click(object sender, RoutedEventArgs e)
{
    if (Filename.Text != "")
    {
        try
        {
            app.Filename = Filename.Text.Trim().ToLower();
            using (IsolatedStorageFile storage = IsolatedStorageFile.GetUserStoreForApplication())
            {
                IsolatedStorageFileStream location = new IsolatedStorageFileStream(app.Filename,
                    System.IO.FileMode.Create, storage);
                System.IO.StreamWriter file = new System.IO.StreamWriter(location);
                file.Write(app.Content);
                file.Dispose();
                location.Dispose();
            }
            NavigationService.GoBack();
        }
        catch
        {
            // Ignore Errors
        }
    }
}

private void Cancel_Click(object sender, RoutedEventArgs e)
{
    NavigationService.GoBack();
}

```

```

private void Save_Click(object sender, RoutedEventArgs e)
{
    if (Filename.Text != "")
    {
        try
        {
            app.Filename = Filename.Text.Trim().ToLower();
            using (IsolatedStorageFile storage = IsolatedStorageFile.GetUserStoreForApplication())
            {
                IsolatedStorageFileStream location = new IsolatedStorageFileStream(app.Filename,
                    System.IO.FileMode.Create, storage);
                System.IO.StreamWriter file = new System.IO.StreamWriter(location);
                file.Write(app.Content);
                file.Dispose();
                location.Dispose();
            }
            NavigationService.GoBack();
        }
        catch
        {
            // Ignore Errors
        }
    }
}

private void Cancel_Click(object sender, RoutedEventArgs e)
{
    NavigationService.GoBack();
}

```

Étape 17

Faites un clic droit sur la page ou sur l'entrée pour *MainPage.xaml* dans l'*Explorateur de solutions* et choisissez l'option *Afficher le code*. Dans la *vue de code* au-dessus de `public MainPage()` tapez la déclaration *Application* suivante :

```
public App app = (App)Application.Current;
```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Net;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Documents;
using System.Windows.Input;
using System.Windows.Media;
using System.Windows.Media.Animation;
using System.Windows.Shapes;
using Microsoft.Phone.Controls;

namespace TextEditor
{
    public partial class MainPage : PhoneApplicationPage
    {
        public App app = (App)Application.Current;

        // Constructor
        public MainPage()
        {
            InitializeComponent();
        }
    }
}

```

Étape 18

Puisque nous en sommes à la *vue de code* pour *MainPage.xaml.cs*, dans le *constructeur* `public MainPage()` en dessous de `InitializeComponent();` tapez le code suivant :

```

ApplicationTitle.Text = "TEXT EDITOR";
PageTitle.Text = "untitled.txt";
Loaded += (object sender, RoutedEventArgs e) =>
{
    if (app.Content != null)
    {
        Editor.Text = (string)app.Content;
    }
    if (app.Filename != "")
    {
        PageTitle.Text = app.Filename;
    }
    else
    {
        PageTitle.Text = "untitled.txt";
    }
};

```

```

public MainPage()
{
    InitializeComponent();
    ApplicationTitle.Text = "TEXT EDITOR";
    PageTitle.Text = "untitled.txt";
    Loaded += (object sender, RoutedEventArgs e) =>
    {
        if (app.Content != null)
        {
            Editor.Text = (string)app.Content;
        }
        if (app.Filename != "")
        {
            PageTitle.Text = app.Filename;
        }
        else
        {
            PageTitle.Text = "untitled.txt";
        }
    }
};
}

```

Étape 19

Puisque nous en sommes toujours à la *vue de code* pour *MainPage.xaml.cs*, en dessous du « } » de la méthode `public MainPage()` tapez la méthode d'assistance (en anglais, *helper method*) et les *gestionnaires d'évènements* suivants :

```

private void New_Click(object sender, EventArgs e)
{
    if (MessageBox.Show("Start a new Document?", "Text Editor",
        MessageBoxButton.OKCancel) == MessageBoxResult.OK)
    {
        PageTitle.Text = "untitled.txt";
        Editor.Text = "";
        app.Filename = PageTitle.Text;
        app.Content = Editor.Text;
    }
}

private void Open_Click(object sender, EventArgs e)
{
    app.Content = Editor.Text;
    NavigationService.Navigate(new Uri("/OpenPage.xaml", UriKind.Relative));
}

private void Save_Click(object sender, EventArgs e)
{
    app.Content = Editor.Text;
    NavigationService.Navigate(new Uri("/SavePage.xaml", UriKind.Relative));
}

```

```

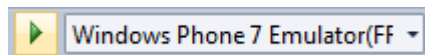
private void New_Click(object sender, EventArgs e)
{
    if (MessageBox.Show("Start a new Document?", "Text Editor",
        MessageBoxButton.OKCancel) == MessageBoxResult.OK)
    {
        PageTitle.Text = "untitled.txt";
        Editor.Text = "";
        app.Filename = PageTitle.Text;
        app.Content = Editor.Text;
    }
}

private void Open_Click(object sender, EventArgs e)
{
    app.Content = Editor.Text;
    NavigationService.Navigate(new Uri("/OpenPage.xaml", UriKind.Relative));
}

private void Save_Click(object sender, EventArgs e)
{
    app.Content = Editor.Text;
    NavigationService.Navigate(new Uri("/SavePage.xaml", UriKind.Relative));
}
    
```

Étape 20

Enregistrez le *Projet* maintenant que vous avez terminé l'application *Windows Phone Silverlight*. Sélectionnez l'option *Windows Phone Emulator*, ensuite sélectionnez *Debug* puis *Démarrer le débogage* ou cliquez sur *Démarrer le débogage* :

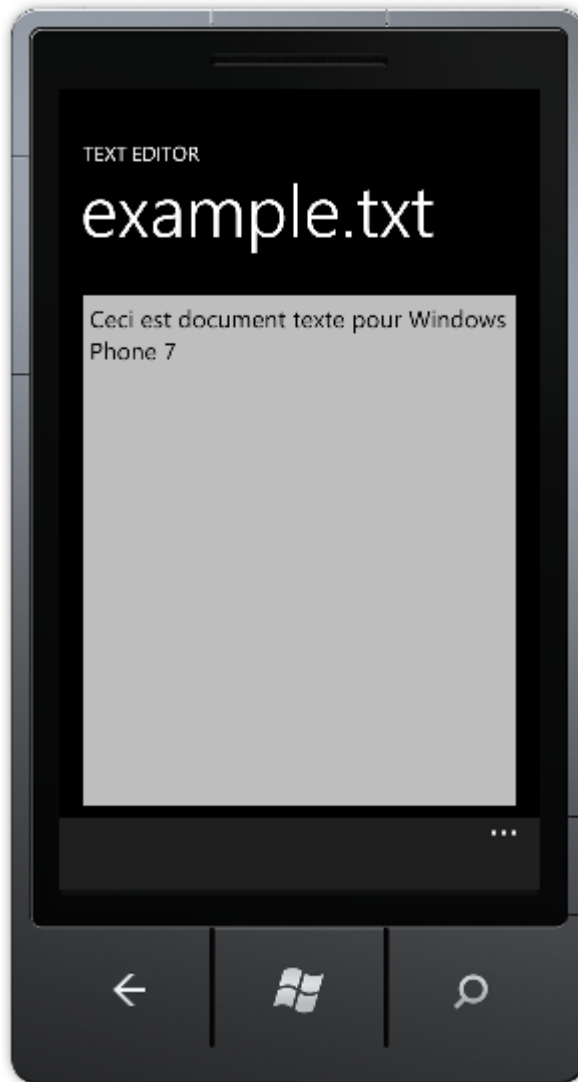


Une fois que vous l'aurez fait, ce qui suit apparaîtra dans l'*émulateur Windows Phone* après qu'il aura été chargé :



Étape 21

Si vous sélectionnez le *TextBox* - vous pouvez ensuite commencer à typer un nouveau document, puis si vous voulez l'enregistrer - cliquez sur le « ... » dans le menu. Choisissez *Save*, entrez un nom de fichier et cliquez sur *Save* pour enregistrer le document et retournez à l'écran principale. Vous pouvez ensuite utiliser *Open* pour l'ouvrir plus tard :



Étape 22

Vous pouvez ensuite *Arrêter l'application* en sélectionnant la fenêtre d'application *Visual Studio 2010* et cliquez sur le bouton *Arrêter le débogage* :



Conclusion

Ceci est une application d'édition de texte simple, avec la possibilité d'ouvrir et d'enregistrer des fichiers textes en utilisant une application Windows Phone 7. Vous pouvez essayer d'ajouter vos propres fonctionnalités telles que la possibilité de changer la taille de la police, etc. - personnalisez-la !

Liens

Télécharger le code source du tutoriel.

Remerciements

Je tiens ici à remercier **Peter Bull** de m'avoir autorisé à traduire son tutoriel.
Je remercie **xxx** pour sa relecture technique et ses propositions.
Je remercie également **yyy** pour sa relecture orthographique et ses propositions.