

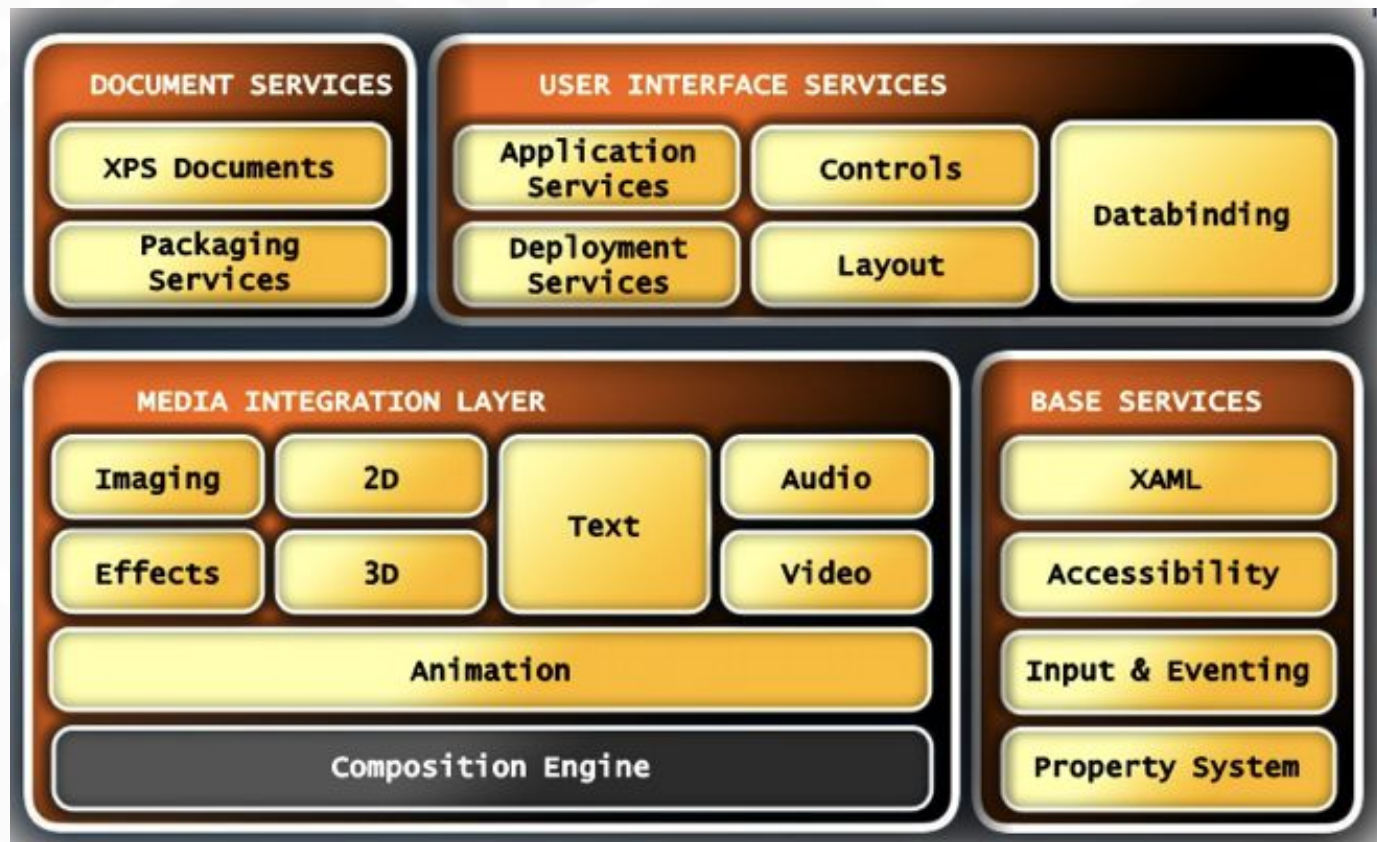
Tạo giao diện cơ bản

Nội dung chính

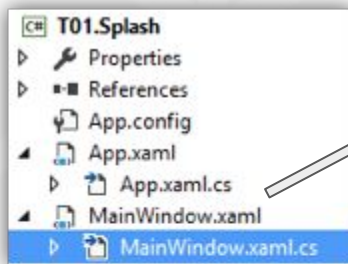
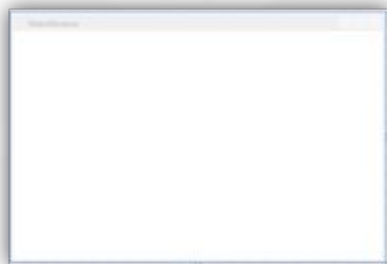
- ❑ WPF & XAML
- ❑ Window, MessageBox, Dialog
- ❑ Button
- ❑ Label
- ❑ TextBox / TextBlock
- ❑ Image
- ❑ CheckBox / RadioButton

WPF & XAML

Windows Presentation Foundation - Kiến trúc



Hello world



XAML code để tạo ra giao diện

Code C# xử lý nghiệp vụ

```
<Window x:Class="T01.Splash.MainWindow"
        xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
        xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
        xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
        xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
        xmlns:local="clr-namespace:T01.Splash"
        mc:Ignorable="d"
        Title="MainWindow" Height="350" Width="525">
    <Grid>

    </Grid>
</Window>
```

Đặc điểm

❑ Phân tách

- **Giao diện** (XAML)
- Xử lý **logic** - nghiệp vụ (C# - code behind)

❑ Tăng tốc phần cứng

❑ Đồ họa dựa trên vector: ko suy giảm chất lượng khi zoom

Window & Dialog

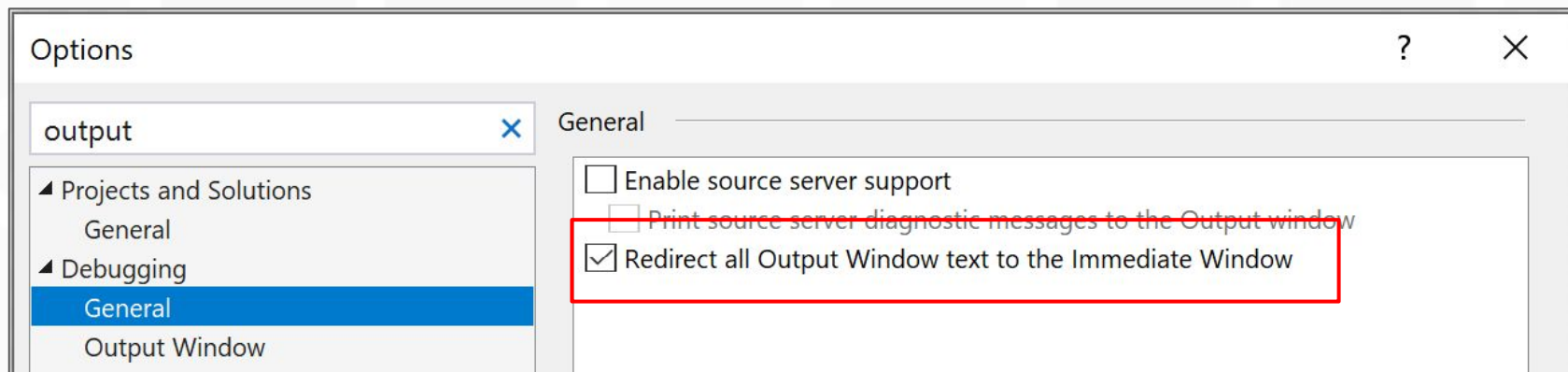
Một số thuộc tính hay dùng

- ❑ **Title**: Tiêu đề của cửa sổ
- ❑ **Icon**
- ❑ **WindowStartupLocation**: Vị trí bắt đầu
 - ❑ Canh giữa màn hình: **CenterScreen**
- ❑ **Width, Height**
- ❑ Để hiển thị Window khác, gọi hàm **Show()**

Thêm thông tin debug

❑ **Debug.WriteLine**

❑ Xuất ra màn hình **Intermediate** thay cho **Output**



Sự kiện hay gọi

Loaded

- Vừa khởi tạo màn hình xong
- Nạp / Khởi tạo dữ liệu ban đầu ở đây

Closing

- Giải phóng tài nguyên trước khi đóng cửa sổ

Closed

- Cửa sổ đã tắt

Chỉ định cửa sổ bắt đầu khi có nhiều cửa sổ

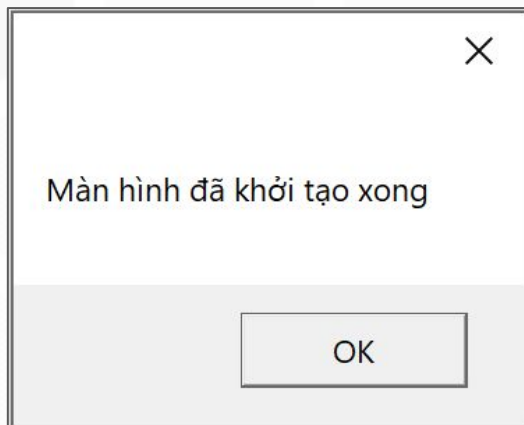
❏ App.xaml

```
<Application x:Class="BasicUIControl.App"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:local="clr-namespace:BasicUIControl"
    StartupUri="MainWindow.xaml">
    <Application.Resources>
```

MessageBox

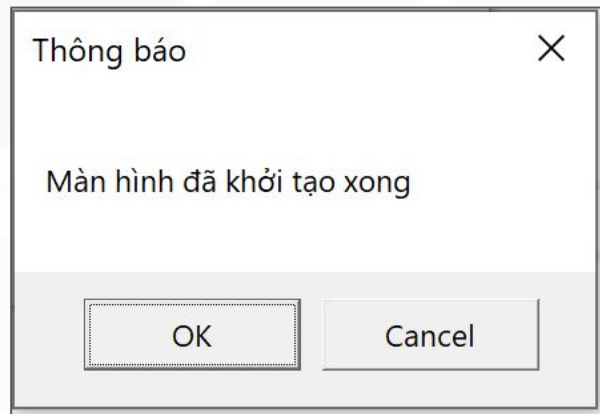
Chỉ nhằm hiển thị thông báo đơn thuần

```
MessageBox.Show("Màn hình đã khởi tạo xong");
```



Thêm tiêu đề và kiểm tra nút bấm

```
MessageBox.Show("Màn hình đã khởi tạo xong",  
"Thông báo", // Tiêu đề  
MessageBoxButton.OKCancel); // Các nút bấm sẽ xuất hiện
```



Làm sao biết vừa bấm nút nào?

```
MessageBoxResult choice = MessageBox.Show("Màn hình đã khởi tạo xong",  
    "Thông báo", // Tiêu đề  
    MessageBoxButton.OKCancel); // Các nút bấm sẽ xuất hiện
```

```
if (choice == MessageBoxResult.OK)
```

```
{
```

```
    Debug.WriteLine("Lựa chọn OK");
```

```
}
```

```
else if (choice == MessageBoxResult.Cancel)
```

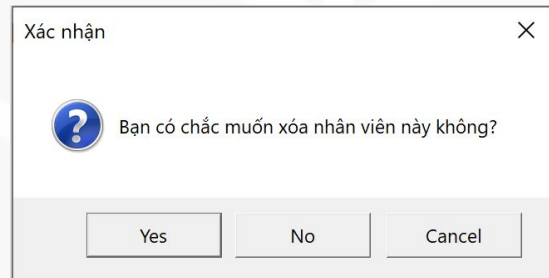
```
{
```

```
    Debug.WriteLine("Lựa chọn Cancel");
```

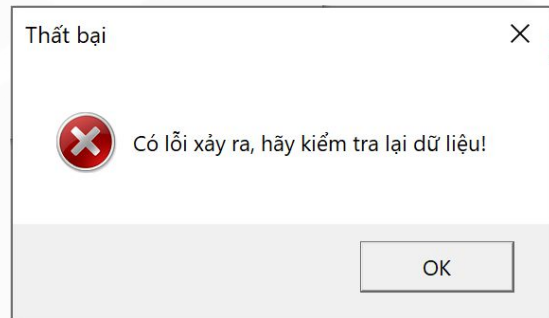
```
}
```

Thêm icon cho thông báo

```
MessageBoxResult choice =  
    MessageBox.Show("Bạn có chắc muốn xóa nhân viên này không?",  
        "Xác nhận",  
        MessageBoxButton.YesNoCancel,  
        MessageBoxImage.Question);
```



```
MessageBoxResult choice =  
    MessageBox.Show("Có lỗi xảy ra, hãy kiểm tra lại dữ liệu!",  
        "Thất bại",  
        MessageBoxButton.OK,  
        MessageBoxImage.Error);
```





Dialog

Hiển thị dạng dialog

- ❑ Bắt buộc phải tắt dialog mới tương tác được với
- ❑ Gọi với hàm **ShowDialog()**, thay vì **Show()**

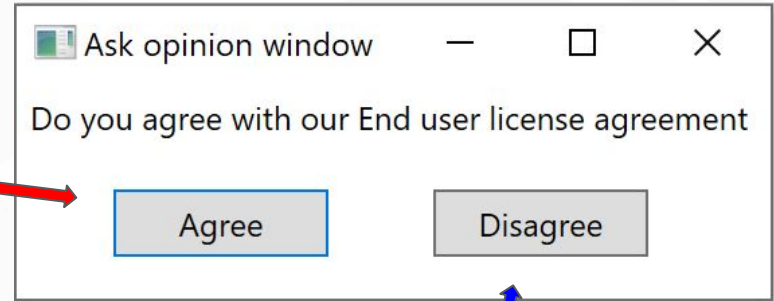
Làm sao để lấy kết quả của dialog?

Bước 3: Hiển thị dialog và xử lí kết quả dialog

```
var screen = new AskOpinionWindow();  
bool? result = screen.ShowDialog();  
  
if (result == true)  
{  
    Debug.WriteLine("The answer is agree.");  
}  
else if (result == false)  
{  
    Debug.WriteLine("The answer is disagree");  
}
```

Bước 1: Chuẩn bị các nút bấm quan trọng

Button có viền được nhấn mạnh, nhấn Enter để kích hoạt



```
<Button Content="Agree" Width="80" Height="25" Canvas.Left="36" Canvas.Top="40"
IsDefault="True" Click="agreeButton_Clicked"/>
<Button Content="Disagree" Width="80" Height="25" Canvas.Left="155" Canvas.Top="40"
IsCancel="True"/>
```

Nhấn Escape để kích hoạt, tự gán DialogResult = false

Bước 2: Viết code thêm cho nút Agree

```
private void agreeButton_Clicked(object sender, RoutedEventArgs e)
{
    DialogResult = true;
}
```

Lấy dữ liệu khi dialog thành công - Cách đơn giản

Sử dụng public attribute / property

1 reference

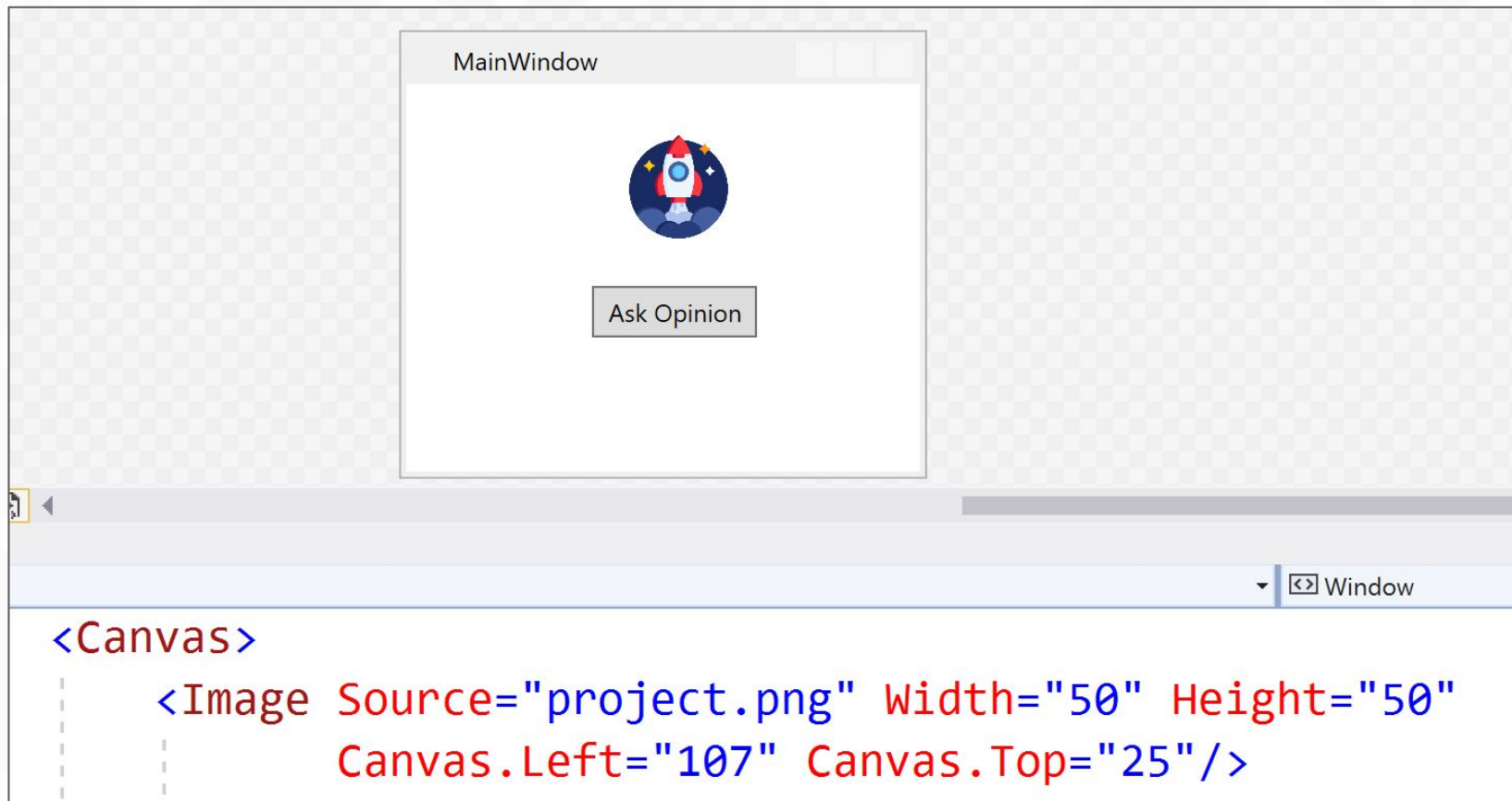
```
public string Opinion { get; set; } = "";
```

1 reference

```
private void agreeButton_Clicked(object sender, RoutedEventArgs e)
{
    // Gán kết quả vào property trước khi kết thúc dialog
    Opinion = "I have read, understood and agreed.";
    DialogResult = true;
}
```

Image

Thêm hình vào project và hiển thị



Gán hình bằng code

```
// URI: Universal resource identifier  
var uri = new Uri("question.png", UriKind.Relative);  
var bitmap = new BitmapImage(uri);  
myImage.Source = bitmap;
```



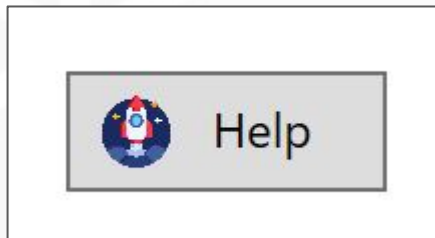
Button

Các thuộc tính & sự kiện hay sử dụng

Content

Click

Thêm hình vào nút bấm

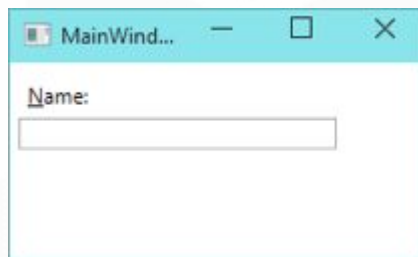


```
<Button Name="askOpinionButton" Width="80" Height="30"  
    Click="askOpinionButton_Click" Canvas.Left="90" Canvas.Top="98"  
    Padding="5"  
    >  
    <StackPanel Orientation="Horizontal" VerticalAlignment="Center" >  
        <Image Source="project.png" />  
        <TextBlock Margin="10,0">Help</TextBlock>  
    </StackPanel>  
</Button>
```

Label / TextBlock

Label

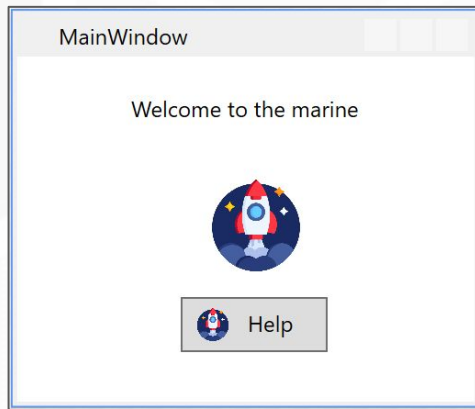
Target: control sẽ nhận focus khi bấm phím tắt



```
<WrapPanel Margin="5">  
  <Label Target="{Binding ElementName=txtFullName}">_Name:</Label>  
  <TextBox x:Name="txtFullName" Width="180"></TextBox>  
</WrapPanel>
```

TextBlock

Chỉ thuần túy hiển thị text



```
<TextBlock Text="Welcome to the marine" Canvas.Left="62" Canvas.Top="20"/>
```

Sự khác biệt giữa Label và TextBlock

- ❑ **TextBlock**: Kế thừa **FrameworkElement**.
- ❑ **Label**: kế thừa **ContentControl**, có thể:
 - Tùy biến control template (nhờ thuộc tính Template).
 - Hiển thị dữ liệu phức tạp ngoài chuỗi (thuộc tính Content)
 - Áp DataTemplate (nhờ thuộc tính ContentTemplate)
 - Label text khi disabled sẽ có màu xám
 - Label hỗ trợ access keys
 - Label nặng nề hơn TextBlock

TextBox

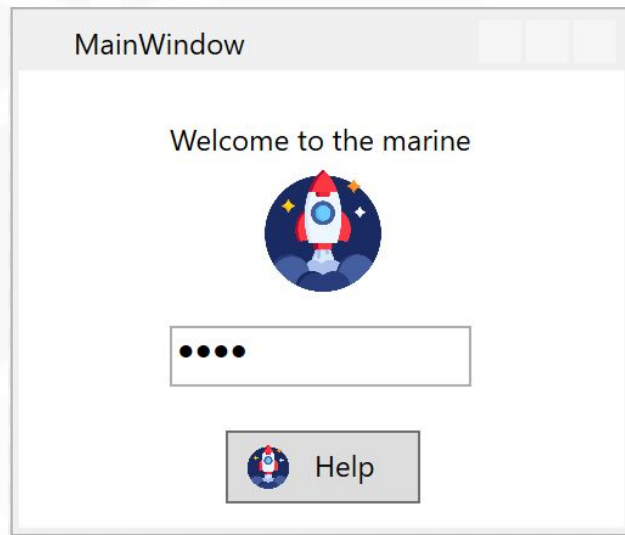
Textbox

- ❑ AcceptsReturn = True: edit nhiều dòng



```
<Label Target="{Binding ElementName=txtNote}">_Notes:</Label>
<TextBox x:Name="txtNote" Width="180" Height="70" AcceptsReturn="True"
VerticalScrollBarVisibility="Visible"
VerticalAlignment="Center" xml:space="preserve">Kim
Mộc
Thủy
Hỏa
Thổ</TextBox>
```

Hiển thị mật khẩu - PasswordBox



```
<PasswordBox Width="124" Height="25" Canvas.Left="62" Canvas.Top="105" Password="1234"/>
```

Bài tập áp dụng 01 - Màn hình đăng nhập

- ❑ Tạo ra màn hình đăng nhập với hai textbox
- ❑ Yêu cầu nhập mật khẩu và tên đăng nhập
- ❑ Kiểm tra hai textbox không được để trống
- ❑ Sử dụng phím tắt **Alt + U** để kích hoạt textbox **Username**
- ❑ Sử dụng phím tắt **Alt + P** để kích hoạt textbox **Password**
- ❑ Khi username = “admin” và mật khẩu = “qwe3@1” thì chuyển tới màn hình Dashboard (rỗng)

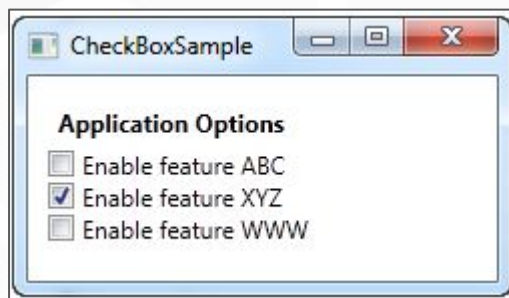
Bài tập áp dụng 02 - Màn hình đổi mật khẩu

- ❑ Tạo ra màn hình có 3 textbox
- ❑ Một textbox nhập mật khẩu hiện tại (== “qwe3@1”)
- ❑ Một textbox nhập mật khẩu mới
- ❑ Một textbox thứ ba nhập lại mật khẩu mới
- ❑ Kiểm tra 3 textbox phải có dữ liệu
- ❑ Kiểm tra textbox mật khẩu mới giống với ô nhập lại
- ❑ Kích hoạt textbox mật khẩu hiện tại với phím **Alt + P**
- ❑ Kích hoạt textbox mật khẩu mới với phím **Alt + N**
- ❑ Kích hoạt textbox nhập lại mật khẩu mới với phím **Alt + R**

CheckBox

CheckBox đơn giản


```
<StackPanel Margin="10">  
    <Label FontWeight="Bold">Application Options</Label>  
    <CheckBox>Enable feature ABC</CheckBox>  
    <CheckBox IsChecked="True">Enable feature XYZ</CheckBox>  
    <CheckBox>Enable feature WWW</CheckBox>  
</StackPanel>
```



CheckBox phức tạp

```
<StackPanel Margin="10">
  <Label FontWeight="Bold">Application Options</Label>
  <CheckBox>
    <TextBlock>
      Enable feature <Run Foreground="Green" FontWeight="Bold">Quick Save</Run>
    </TextBlock>
  </CheckBox>
  <CheckBox IsChecked="True">
    <WrapPanel>
      <TextBlock>
        Enable feature <Run FontWeight="Bold">Auto backup</Run>
      </TextBlock>
      <Image Source="check.png" Width="16" Height="16" Margin="5,0" />
    </WrapPanel>
  </CheckBox>
  <CheckBox>
    <TextBlock>
      Enable feature <Run Foreground="Blue" TextDecorations="Underline" FontWeight="Bold">Premium theme</Run>
    </TextBlock>
  </CheckBox>
</StackPanel>
```

Application Options

- ☐ Enable feature **Quick Save**
- ☒ Enable feature **Auto backup** 
- ☐ Enable feature Premium theme

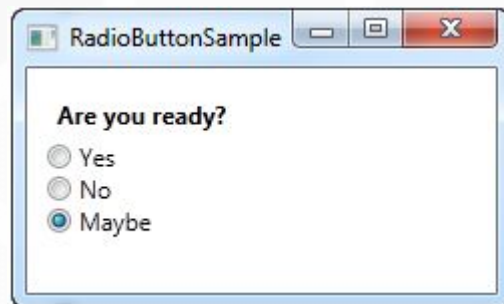
Sự kiện thay đổi trạng thái

❏ `CheckedChanged`

RadioButton

Radio Button cơ bản

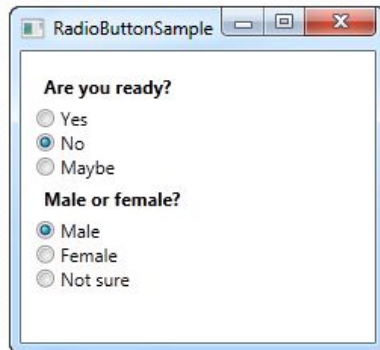
```
<StackPanel Margin="10">  
    <Label FontWeight="Bold">Are you ready?</Label>  
    <RadioButton>Yes</RadioButton>  
    <RadioButton>No</RadioButton>  
    <RadioButton IsChecked="True">Maybe</RadioButton>  
</StackPanel>
```



Gom nhóm RadioButton

```
<StackPanel Margin="10">
    <Label FontWeight="Bold">Are you ready?</Label>
    <RadioButton GroupName="ready">Yes</RadioButton>
    <RadioButton GroupName="ready">No</RadioButton>
    <RadioButton GroupName="ready" IsChecked="True">Maybe</RadioButton>

    <Label FontWeight="Bold">Male or female?</Label>
    <RadioButton GroupName="sex">Male</RadioButton>
    <RadioButton GroupName="sex">Female</RadioButton>
    <RadioButton GroupName="sex" IsChecked="True">Not sure</RadioButton>
</StackPanel>
```



Tùy biến Radio Button

```
<StackPanel Margin="10">
    <Label FontWeight="Bold">Are you ready?</Label>
    <RadioButton>
        <WrapPanel>
            <Image Source="check.png" Width="16" Height="16" Margin="0,0,5,0"
                RenderOptions.BitmapScalingMode="HighQuality"/>
            <TextBlock Text="Yes" Foreground="Green" />
        </WrapPanel>
    </RadioButton>
    <RadioButton Margin="0,5">
        <WrapPanel>
            <Image Source="delete.png" Width="16" Height="16" Margin="0,0,5,0"
                RenderOptions.BitmapScalingMode="HighQuality"/>
            <TextBlock Text="No" Foreground="Red" />
        </WrapPanel>
    </RadioButton>
    <RadioButton IsChecked="True">
        <WrapPanel>
            <Image Source="conversation.png" Width="16" Height="16" Margin="0,0,5,0"
                RenderOptions.BitmapScalingMode="HighQuality"/>
            <TextBlock Text="Maybe" Foreground="Gray" />
        </WrapPanel>
    </RadioButton>
</StackPanel>
```

Are you ready?

☐ ☒ Yes

☐ ☒ No

☒ ☐ Maybe

ProgressBar

ProgressBar

Các thuộc tính quan trọng

- ❑ **Minimum**: giá trị tối thiểu (mặc định 0)
- ❑ **Maximum**: giá trị tối đa (mặc định 100)
- ❑ **Value**: giá trị hiện tại (mặc định 0)

