



C# Essentials

WPF Layouts

Sander De Puydt

**DE HOGESCHOOL
MET HET NETWERK**

Hogeschool PXL – Elfde-Liniestraat 24 – B-3500 Hasselt
www.pxl.be - www.pxl.be/facebook

Korte inhoud

- XAML vs Code
- Layouts
 - Grid
 - StackPanel
 - WrapPanel
 - DockPanel
 - Canvas
- Besturingselementen
- Samenvatting



XAML vs Code

- InitializeComponent();
 - code genereren en uitvoeren van het design
- XAML Voorbeeld

```
<StackPanel>  
    <TextBlock Margin="10">  
        Deze les gaan we zien hoe we WPF Layouts gebruiken  
    </TextBlock>  
    <Button Margin="20" HorizontalAlignment="Right">  
        OK  
    </Button>  
</StackPanel>
```



XAML vs Code

- Code Voorbeeld

```
// Create the StackPanel
StackPanel stackPanel = new StackPanel();
this.Content = stackPanel;

// Create the TextBlock
TextBlock textBlock = new TextBlock();
textBlock.Margin = new Thickness(10);
textBlock.Text = "Deze les gaan we zien hoe we WPF Layouts gebruiken";
stackPanel.Children.Add(textBlock);

// Create the Button
Button button = new Button();
button.Margin = new Thickness(20);
button.Content = "OK";
button.HorizontalAlignment = HorizontalAlignment.Right;
stackPanel.Children.Add(button);
```



XAML vs Code

- this = het venster (MainWindow)
- **Children** = de elementen binnen een element
- new TextBlock() = het aanmaken van object van het type TextBlock
- XAML is makkelijker om te lezen



Layouts

- Alle Layouts zijn Panels
- Alle Panels hebben de volgende properties
 - Children
 - Background
 - Height
 - Margin
 - Name
 - Orientation
 - Width



Layouts

- Grid
- StackPanel
- WrapPanel
- DockPanel
- Canvas



Layouts

- Grid Properties
 - ColumnDefinitions
 - RowDefinitions
 - Height en Width:
 - Vaste waarde
 - Auto: Neemt ruimte in nodig voor de inhoud
 - (*): Neemt de overige ruimte in
 - Grid.Row en Grid.Column voor het plaatsen van elementen
 - Grid.ColumnSpan en Grid.RowSpan



Layouts

- Grid voorbeeld (deel 1):

```
<Grid Background = "AliceBlue">
  <Grid.ColumnDefinitions>
    <ColumnDefinition Width = "Auto" />
    <ColumnDefinition />
    <ColumnDefinition />
  </Grid.ColumnDefinitions>

  <Grid.RowDefinitions>
    <RowDefinition Height = "*" />
    <RowDefinition Height = "*" />
  </Grid.RowDefinitions>
```

...



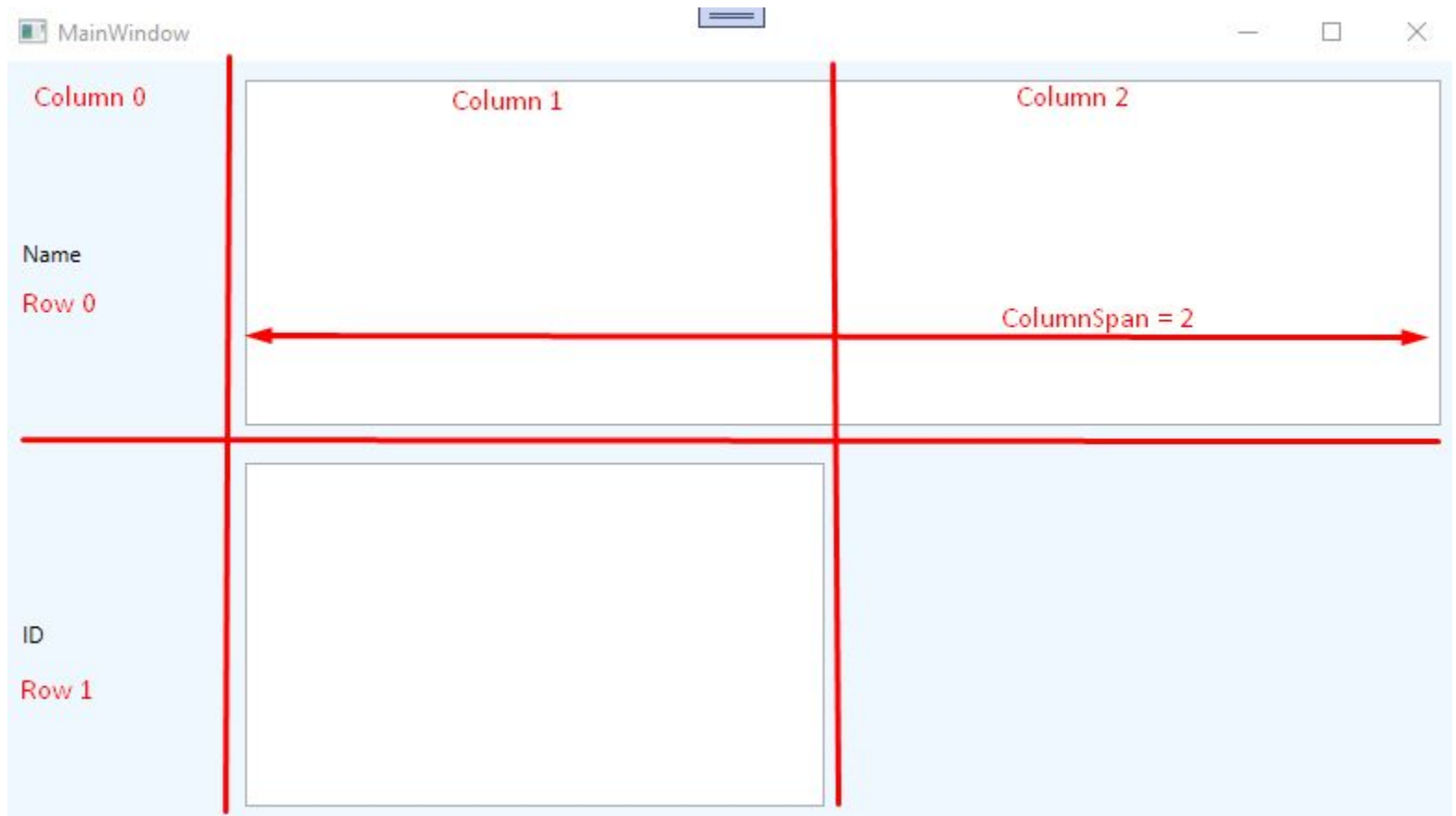
Layouts

- Grid voorbeeld (deel 2):

```
...  
    <TextBlock Grid.Row="0" Grid.Column="0" Text="Name" Margin="10"  
        HorizontalAlignment="Left" VerticalAlignment="Center"  
        Width="100" />  
    <TextBox Grid.Row="0" Grid.Column="1" Grid.ColumnSpan="2"  
        Margin="10" />  
    <TextBlock Grid.Row="1" Grid.Column="0" Text="ID" Margin="10"  
        HorizontalAlignment="Left" VerticalAlignment="Center"  
        Width="100" />  
    <TextBox Grid.Row="1" Grid.Column="1" Margin="10" />  
</Grid>
```

Layouts

- Grid:



Layouts

- StackPanel Properties
 - Elementen in één lijn
 - horizontaal
 - verticaal
 - **Orientation**



Layouts

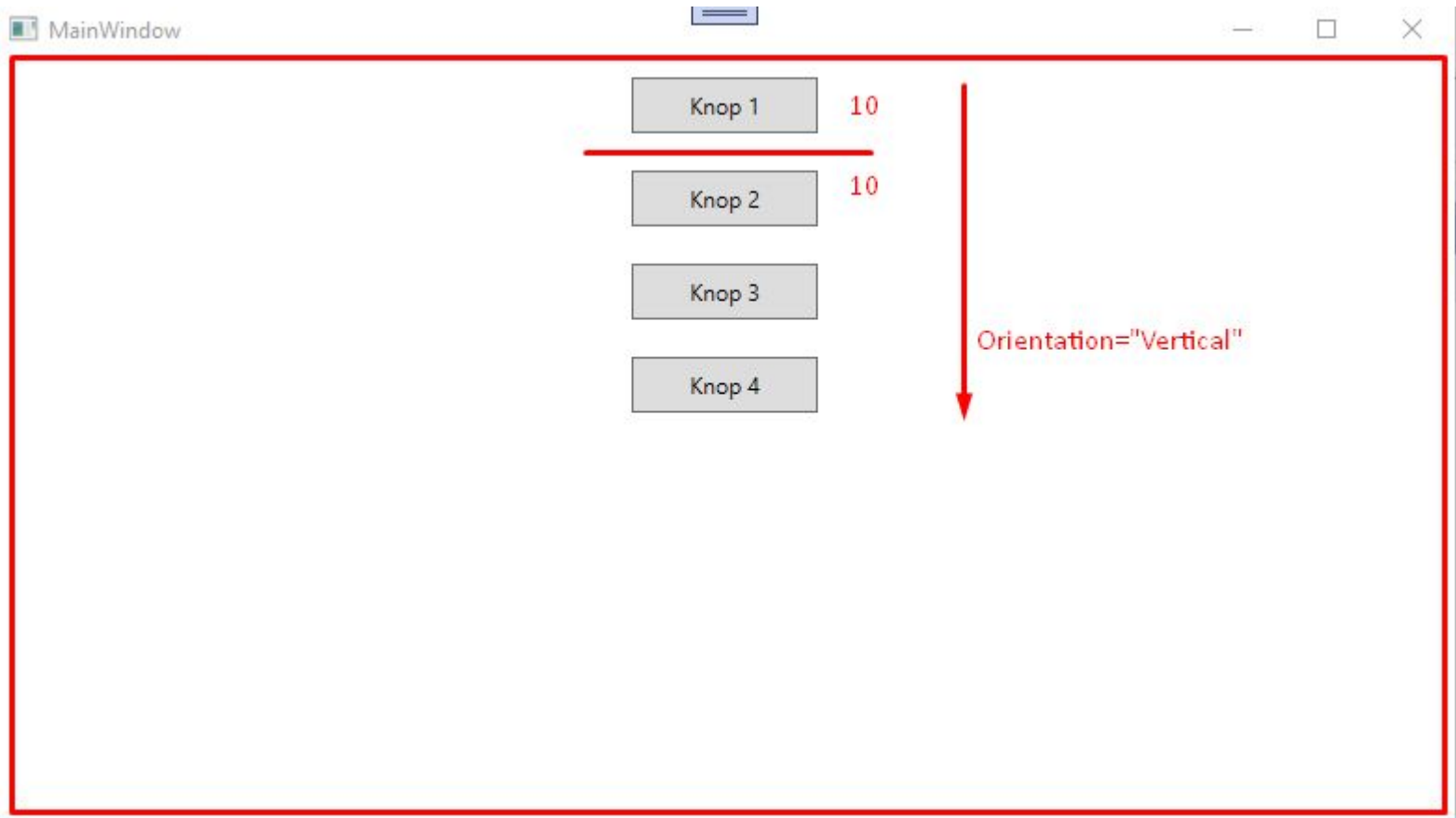
- StackPanel voorbeeld:

```
<StackPanel Orientation="Vertical">  
  <Button Height="30" Width="100" Margin="10">Knop 1</Button>  
  <Button Height="30" Width="100" Margin="10">Knop 2</Button>  
  <Button Height="30" Width="100" Margin="10">Knop 3</Button>  
  <Button Height="30" Width="100" Margin="10">Knop 4</Button>  
</StackPanel>
```



Layouts

- StackPanel:



Layouts

- WrapPanel Properties
 - gelijkaardig aan StackPanel, maar duwt elementen die niet meer passen naar de volgende kolom/rij
 - **Orientation**



Layouts

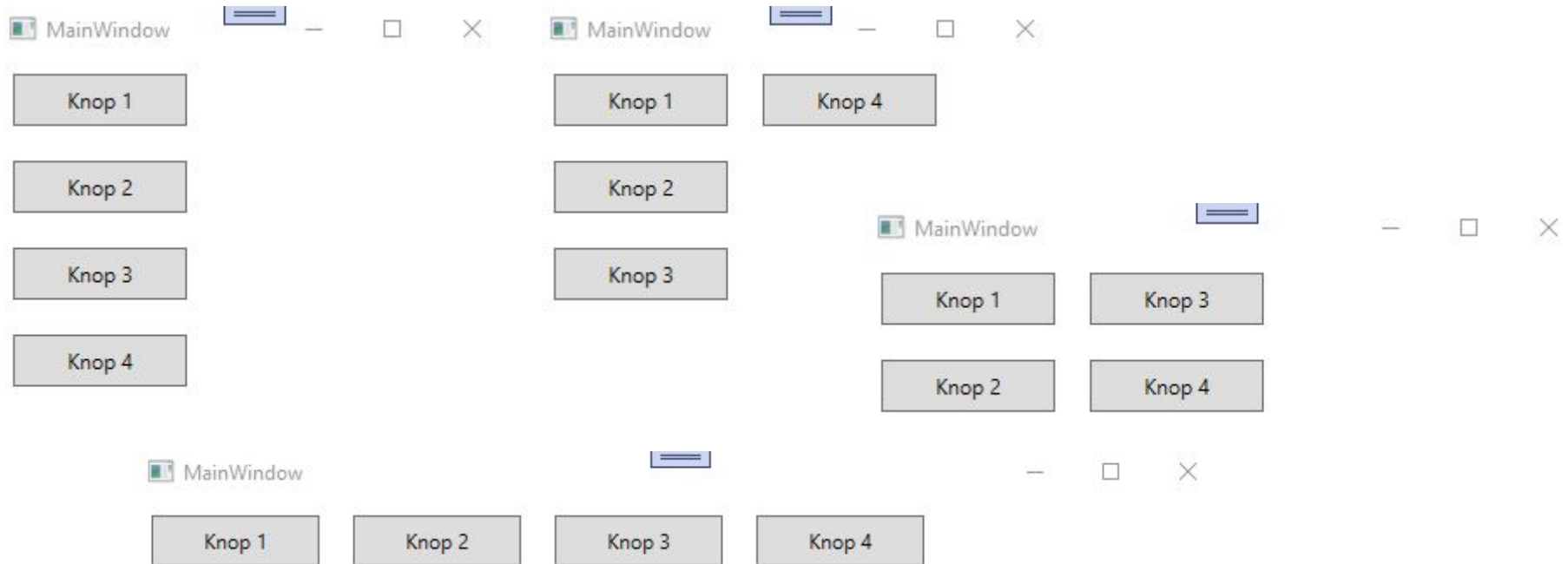
- WrapPanel voorbeeld:

```
<WrapPanel Orientation="Vertical">  
  <Button Height="30" Width="100" Margin="10">Knop 1</Button>  
  <Button Height="30" Width="100" Margin="10">Knop 2</Button>  
  <Button Height="30" Width="100" Margin="10">Knop 3</Button>  
  <Button Height="30" Width="100" Margin="10">Knop 4</Button>  
</WrapPanel>
```



Layouts

- WrapPanel past zich automatisch aan



Layouts

- DockPanel Properties
 - LastChildFill
- Children kan je “Docken” via DockPanel.Dock
 - Left
 - Top
 - Right
 - Bottom



Layouts

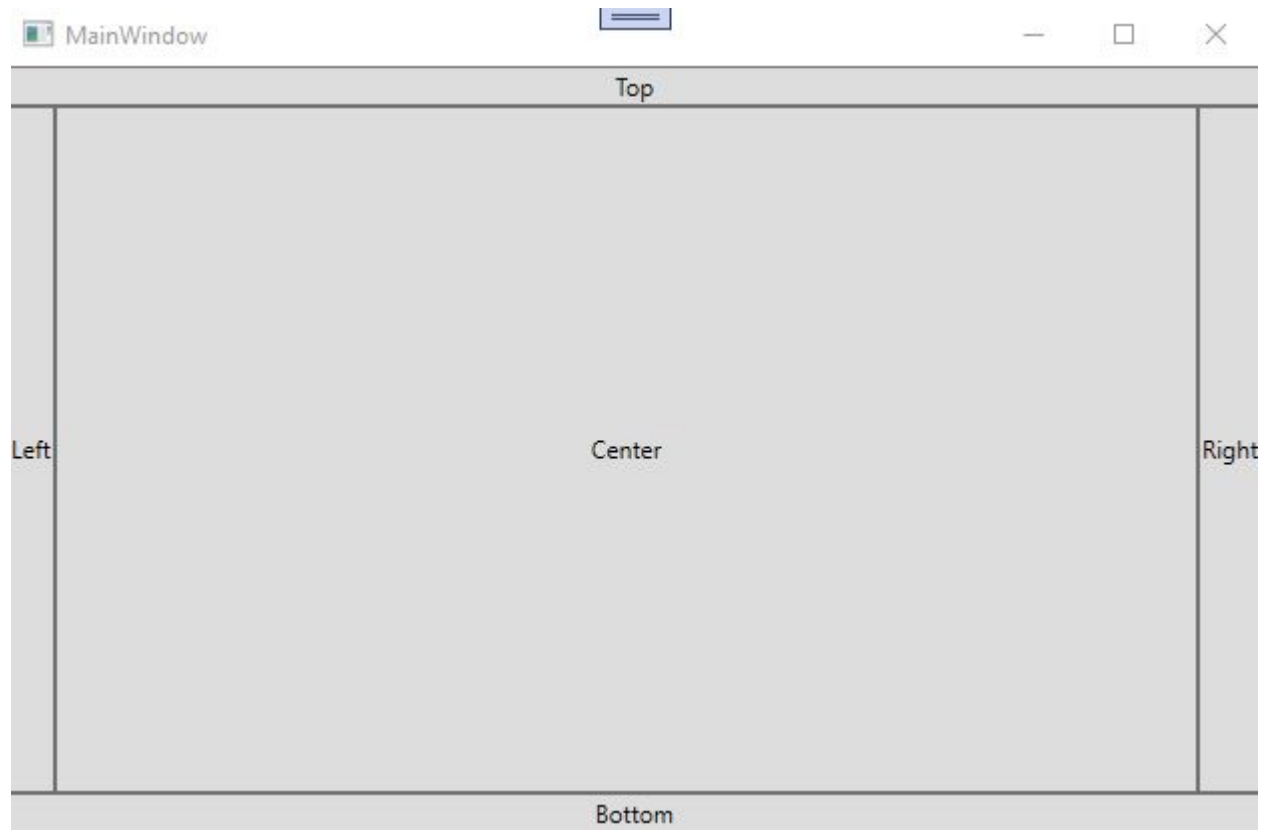
- DockPanel Voorbeeld:

```
<DockPanel LastChildFill = "True">  
  <Button Content = "Top" DockPanel.Dock = "Top" />  
  <Button Content = "Bottom" DockPanel.Dock = "Bottom" />  
  <Button Content = "Left" DockPanel.Dock = "Left" />  
  <Button Content = "Right" DockPanel.Dock = "Right" />  
  <Button Content = "Center" />  
</DockPanel>
```



Layouts

- DockPanel:



Layouts

- Canvas Properties
 - Gebruikt coördinaten om child elements te positioneren
 - Typisch gebruikt voor 2D graphic elements



Layouts

- Canvas Voorbeeld:

```
<Canvas Width = "580" Height = "360" >
  <Ellipse Canvas.Left = "30" Canvas.Top = "30"
    Fill = "Gray" Width = "200" Height = "120" />
  <Ellipse Canvas.Right = "30" Canvas.Top = "30"
    Fill = "Aqua" Width = "200" Height = "120" />
  <Button Canvas.Left = "30" Canvas.Bottom = "30"
    Content="Ik ben een knop" Width = "100" Height = "50" />
  <TextBox Canvas.Right = "30" Canvas.Bottom = "30"
    Text = "Hier kan je tekst typen" Width = "200"
      Height = "120" />
</Canvas>
```

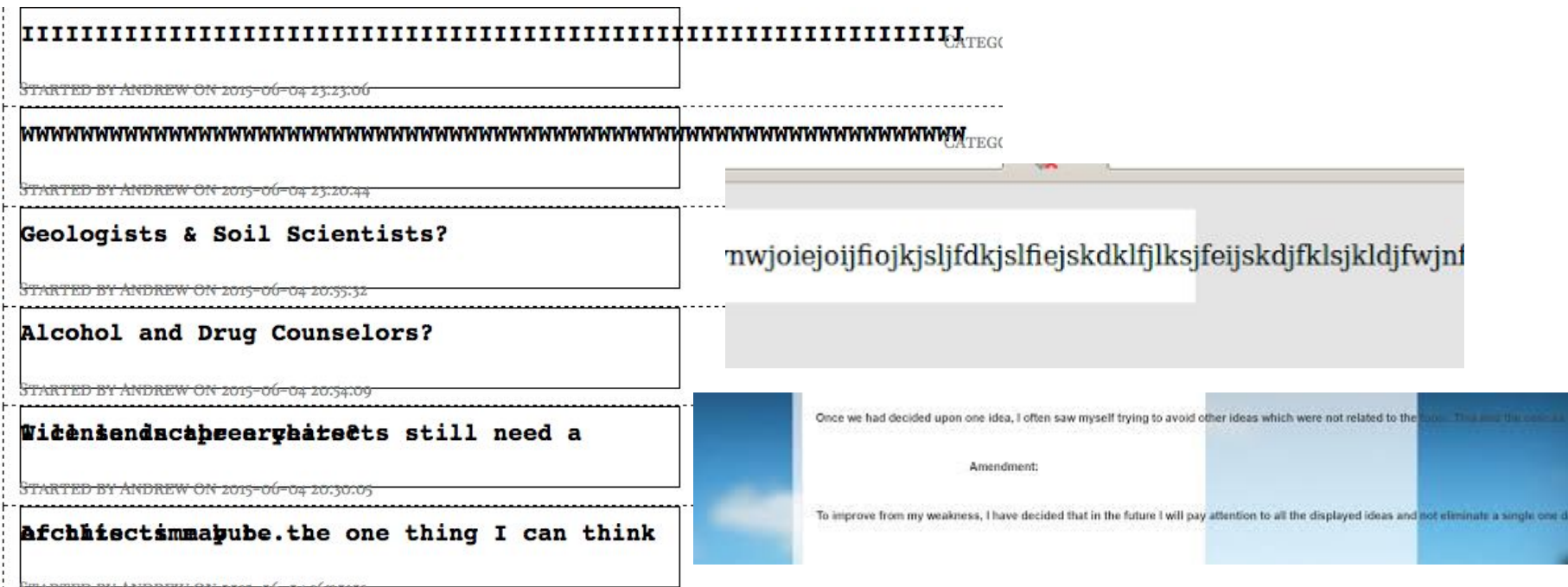
Layouts

- Canvas Voorbeeld:



Besturingselementen

- Hoe werken met lange teksten?
 - TextBlock
 - TextBox



Besturingselementen

- TextBlock
 - lange strings
 - linebreak
 - TextWrapping
 - TextTrimming
 - inline formatting
 - cursief
 - onderlijnen
 - vetgedrukt



Besturingselementen

- TextBlock: lange strings

```
<TextBlock Margin="10" Foreground="Red">
```

Dit is de eerste manier om tekst op te breken in meerdere lijnen met een LineBreak

```
</TextBlock>
```

```
<TextBlock Margin="10" TextTrimming="CharacterEllipsis" Foreground="Green">
```

De tweede methode om lange tekst te verwerken in een TextBlock is door de tekst af te kappen wanneer die te groot is.

```
</TextBlock>
```

```
<TextBlock Margin="10" TextWrapping="Wrap" Foreground="Blue">
```

Met TextWrapping kan men de tekst automatisch laten verder gaan op de volgende lijn net zoals bij het WrapPanel.

```
</TextBlock>
```

Besturingselementen

- TextBlock: lange strings



Besturingselementen

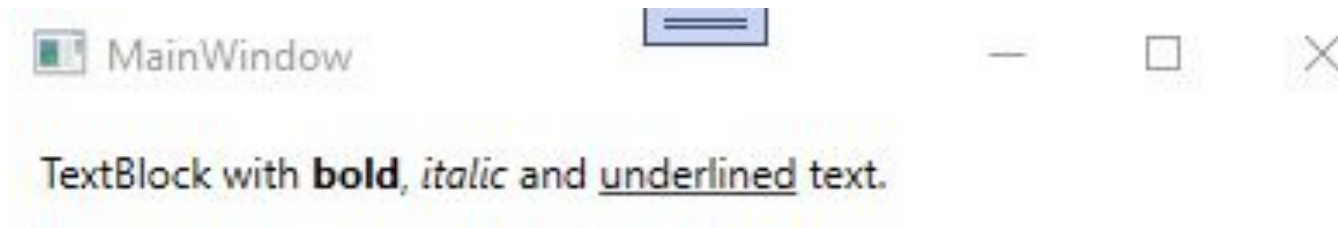
- TextBlock: inline formatting

```
<TextBlock Margin="10" TextWrapping="Wrap">  
    TextBlock with  
        <Bold>bold</Bold>,  
        <Italic>italic</Italic> and  
        <Underline>underlined</Underline>  
    text.  
</TextBlock>
```



Besturingselementen

- TextBlock: inline formatting



Besturingselementen

- TextBox
 - Tekst van meerdere regels
 - TextWrapping
 - AcceptsReturn



Besturingselementen

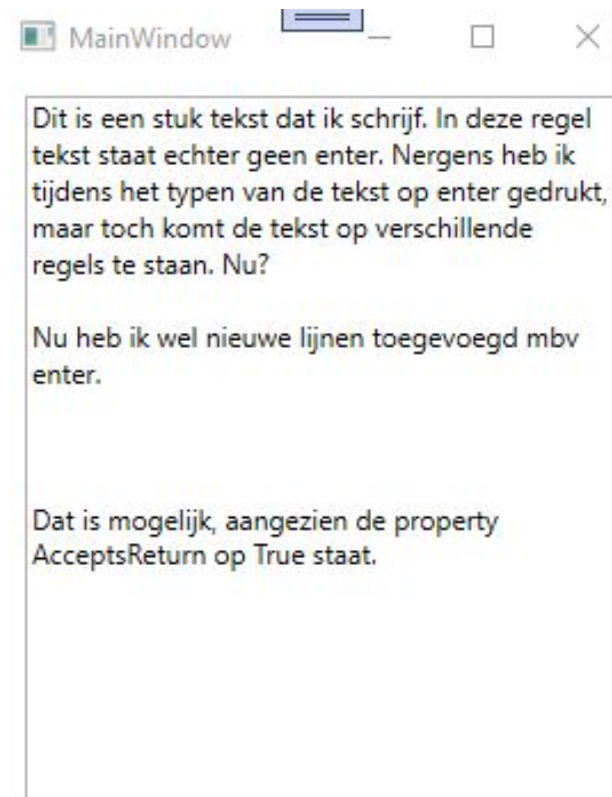
- TextBox: Voorbeeld

```
<Grid Margin="10">  
    <TextBox AcceptsReturn="True" TextWrapping="Wrap" />  
</Grid>
```



Besturingselementen

- TextBox



Samenvatting

- Kies een layout op basis van:
 - posities van child elements
 - grote en doel van child elements
- Lange tekst in een besturingselement behandelen

