

C# Mobile

Les 2

- Layouts



**DE HOGESCHOOL
MET HET NETWERK**

Elfde-Liniestraat 24, 3500 Hasselt

www.pxl.be

Doelstellingen

- De junior-collega kan verschillende layouts implementeren
- De junior-collega kan op een pagina meerdere layouts combineren



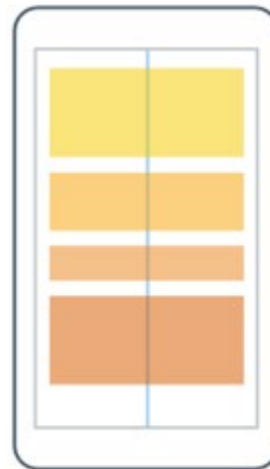
Layouts

Layouts in .NET MAUI

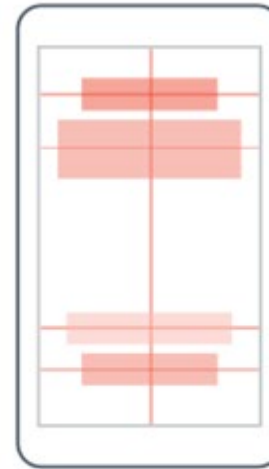
Layouts gedragen zich als **containers** om UI elementen te schikken en groeperen

De belangrijkste zijn:

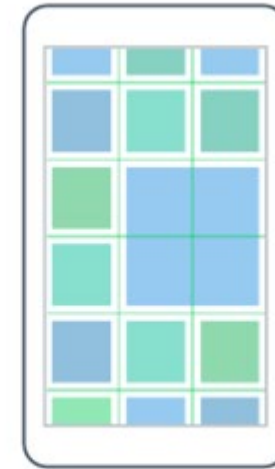
- **StackLayout**
- **Grid**
- **AbsoluteLayout**
- **FlexLayout**



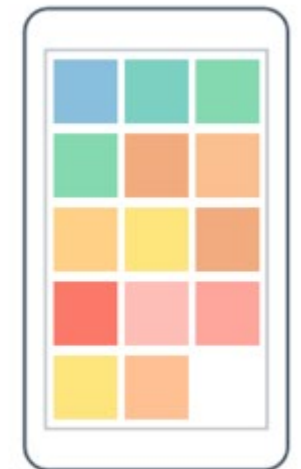
StackLayout



AbsoluteLayout



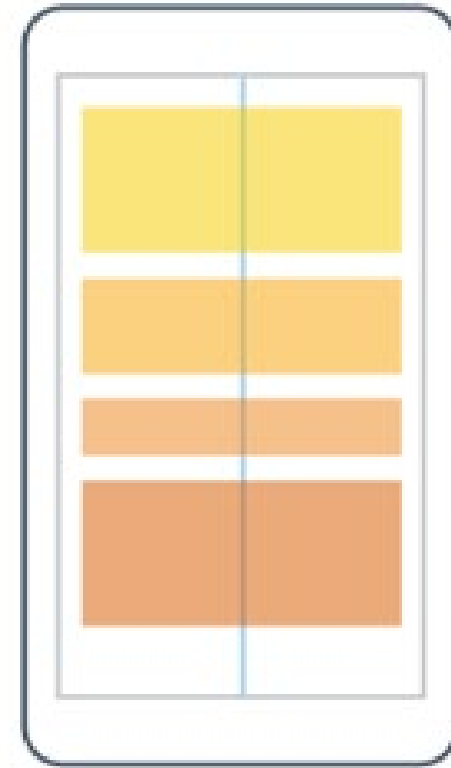
Grid



FlexLayout

Layout: StackLayout

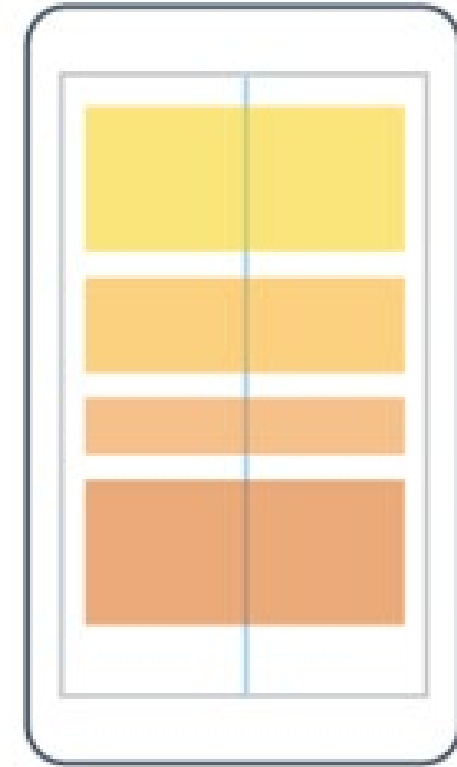
- Heel vaak gebruikt
 - Plaatst de elementen automatisch in 1 dimensie
 - Instellen met *Orientation* property
 - Horizontaal of verticaal in volgorde van toevoegen
 - Plaats tussen de elementen is gedefinieerd door *HorizontalOptions* en *VerticalOptions*
 - Standaard wordt alle ruimte benut!
- Twee varianten: `HorizontalStackLayout` en `VerticalStackLayout`



StackLayout

Layout: StackLayout

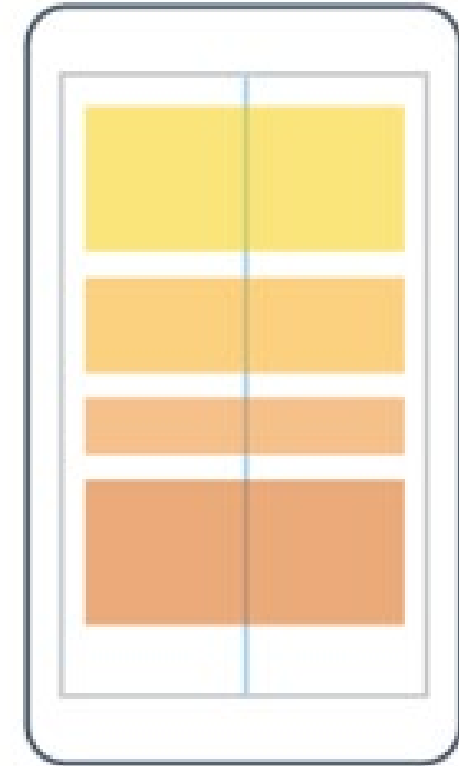
- VerticalOptions/HorizontalOptions
 - Hoe moeten de children gepositioneerd of uitgerekend worden
- Spacing
 - Plaats gereserveerd tussen de *child elements*
 - Gedefinieerd op het StackLayout-element
- Padding
 - Witruimte binnen de elementen
- Margin
 - Witruimte buiten de elementen



StackLayout

Layout: StackLayout

- WidthRequest/HeightRequest
 - “Verzoek” voor een specifieke width/height
 - Uitgedrukt in device units:
 - iOS: 64 units in één centimeter
 - Android: 64 units in één centimeter
 - UWP: 96 units in één centimeter
- MinimumWidthRequest/MinimumHeightRequest



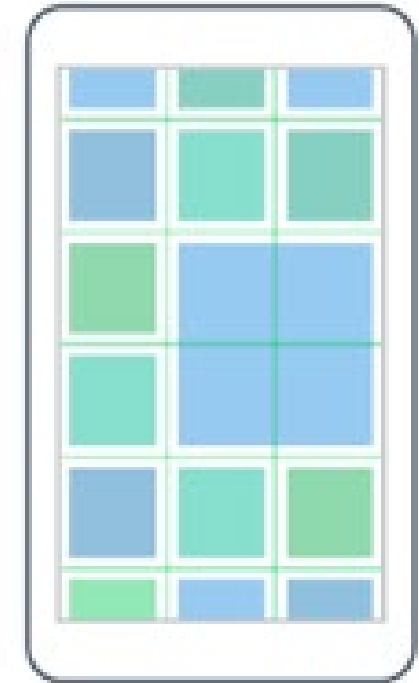
StackLayout

Layout: Labo

- Open *Labo – Layouts* in de Blackboard cursus
- Accepteer de assignment
- Open je repository
- Maak een clone van je repository
- Open het README.md bestand
- Voltooi Deel 1

Layout: Grid

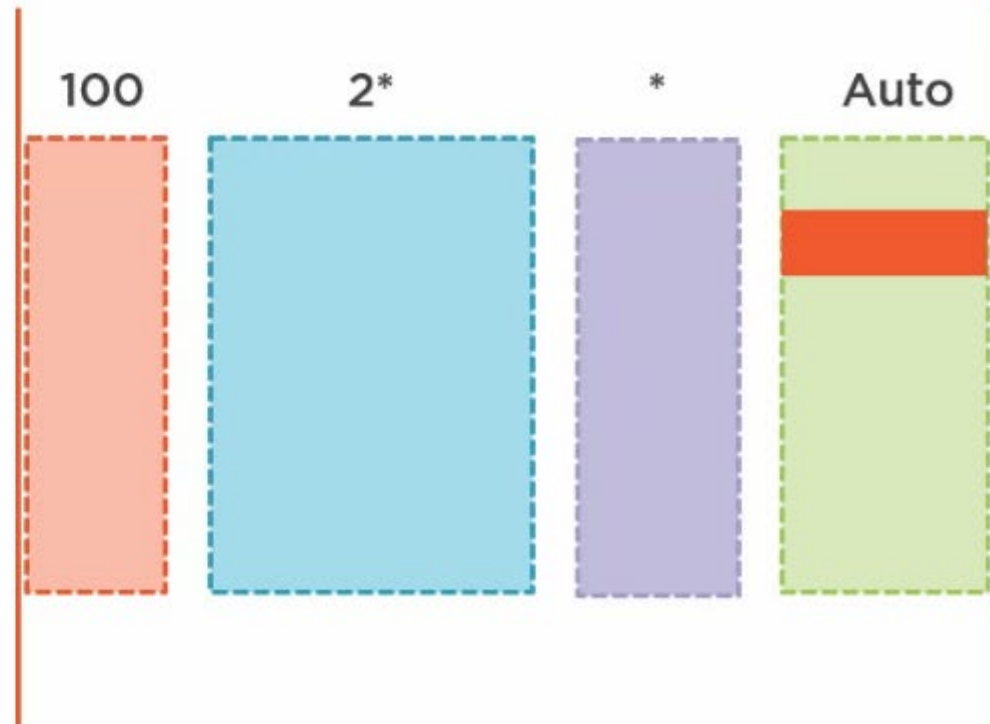
- Layout die gebruik maakt van rijen en kolommen
- Flexibeler dan de StackLayout
- Drie manieren om groottes aan te geven voor de rijen en kolommen:
 - Absoluut
 - Auto
 - Ster-notatie



Grid

Layout: Grid

```
<Grid>  
  <Grid.ColumnDefinitions>  
    <ColumnDefinition Width="100">  
    </ColumnDefinition>  
    <ColumnDefinition Width="2*">  
    </ColumnDefinition>  
    <ColumnDefinition Width="*">  
    </ColumnDefinition>  
    <ColumnDefinition Height="Auto">  
    </ColumnDefinition>  
  </Grid.ColumnDefinitions>  
</Grid>
```



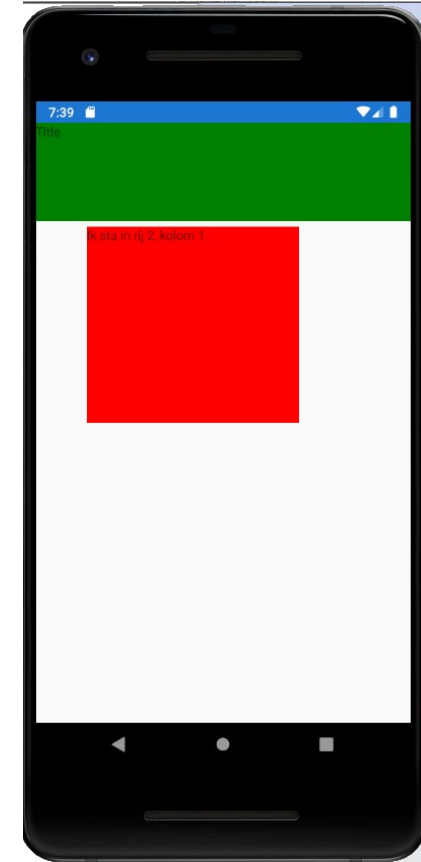
Layout: Grid

- Bij het definiëren van een Grid moeten de rijen en kolommen aangegeven worden:
 - RowDefinitions
 - ColumnDefinitions
- Dit noteren we met de Property Element Syntax:
 - `<Grid.RowDefinitions>`
 `<RowDefinition Height="50"></RowDefinition>`
 `...`
- Elementen plaatsen we na de Definitions en via Attached Properties geven we ze een plaats
- Elementen kunnen ook meerdere kolommen/rijen overspannen (Grid.RowSpan)

Layout: Grid

```
<Grid>
  <Grid.RowDefinitions>
    <RowDefinition Height="1*"></RowDefinition>
    <RowDefinition Height="2*"></RowDefinition>
    <RowDefinition Height="3*"></RowDefinition>
  </Grid.RowDefinitions>
  <Grid.ColumnDefinitions>
    <ColumnDefinition Width="50"></ColumnDefinition>
    <ColumnDefinition Width="2*"></ColumnDefinition>
    <ColumnDefinition Width="1*"></ColumnDefinition>
  </Grid.ColumnDefinitions>

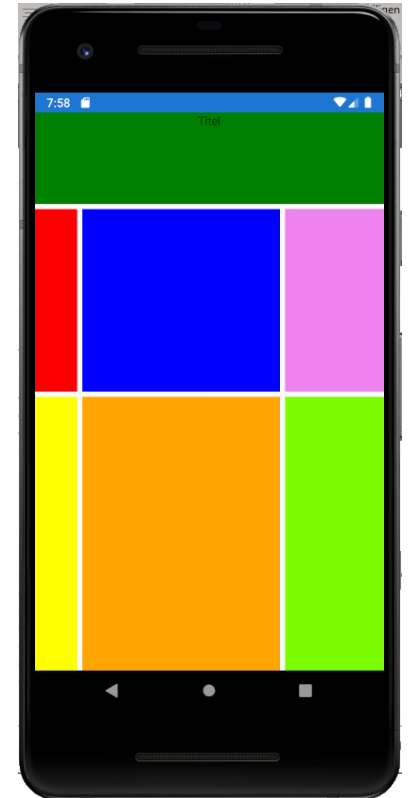
  <Label Grid.Row="0" Grid.ColumnSpan="3" BackgroundColor="■"Green">Title</Label>
  <Label Grid.Row="1" Grid.Column="1" BackgroundColor="■"red">Ik sta in rij 2, kolom 1</Label>
</Grid>
```



Layout: Grid

```
<Grid>
  <Grid.RowDefinitions>
    <RowDefinition Height="1*"></RowDefinition>
    <RowDefinition Height="2*"></RowDefinition>
    <RowDefinition Height="3*"></RowDefinition>
  </Grid.RowDefinitions>
  <Grid.ColumnDefinitions>
    <ColumnDefinition Width="50"></ColumnDefinition>
    <ColumnDefinition Width="2*"></ColumnDefinition>
    <ColumnDefinition Width="1*"></ColumnDefinition>
  </Grid.ColumnDefinitions>

  <Label Grid.Row="0" Grid.ColumnSpan="3" BackgroundColor="■"Green" HorizontalTextAlignment="Center">Titel</Label>
  <Label Grid.Row="1" Grid.Column="0" BackgroundColor="■"Red"></Label>
  <Label Grid.Row="1" Grid.Column="1" BackgroundColor="■"Blue"></Label>
  <Label Grid.Row="1" Grid.Column="2" BackgroundColor="■"Violet"></Label>
  <Label Grid.Row="2" Grid.Column="0" BackgroundColor="■"Yellow"></Label>
  <Label Grid.Row="2" Grid.Column="1" BackgroundColor="■"Orange"></Label>
  <Label Grid.Row="2" Grid.Column="2" BackgroundColor="■"LawnGreen"></Label>
</Grid>
```



Layout: Grid

- Verkorte schrijfwijze van kolommen en rijen:

```
<Grid>
  <Grid.RowDefinitions>
    <RowDefinition Height="100" />
    <RowDefinition Height="Auto" />
    <RowDefinition Height="1*" />
    <RowDefinition Height="2*" />
  </Grid.RowDefinitions>
  ...
</Grid>
```



```
<Grid RowDefinitions="100, Auto, 1*, 2*">
  ...
</Grid>
```

```
<Grid>
  <Grid.RowDefinitions>
    <RowDefinition />
    <RowDefinition />
    <RowDefinition />
  </Grid.RowDefinitions>
  <Grid.ColumnDefinitions>
    <ColumnDefinition />
    <ColumnDefinition />
  </Grid.ColumnDefinitions>
  ...
</Grid>
```



```
<Grid RowDefinitions="*, *, *" ColumnDefinitions="*, *">
  ...
</Grid>
```


Layout: Labo

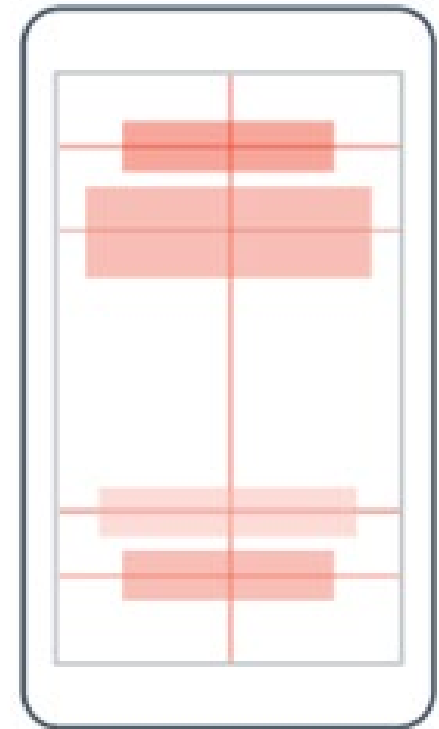
Labo – Layouts
- Voltooi Deel 2

Layout: Grid

- Je kan een Grid ook nesten
- Zie oefening

Layout: `AbsoluteLayout`

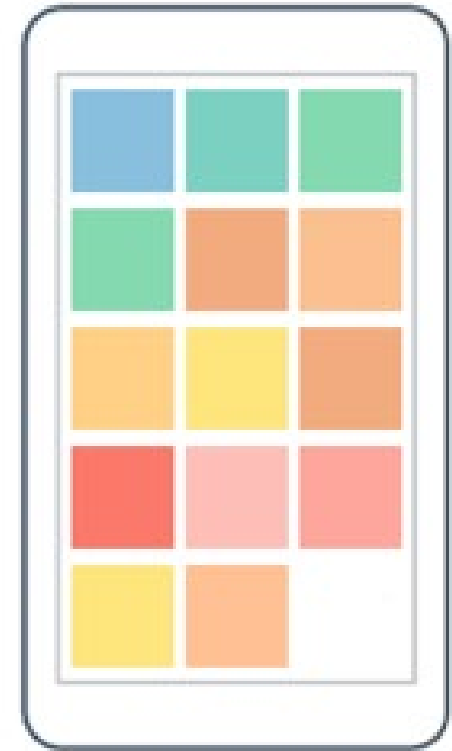
- Unmanaged container: elk element moet specifiek een plaats krijgen binnen de layout
- Omslachtig en moeilijker op verschillende resoluties



`AbsoluteLayout`

Layout: FlexLayout

- Child elementen komen verticaal en horizontaal in een stack
- Gebaseerd op de CSS Flexbox
- Zie <https://docs.microsoft.com/en-us/dotnet/maui/user-interface/layouts/flexlayout>



FlexLayout