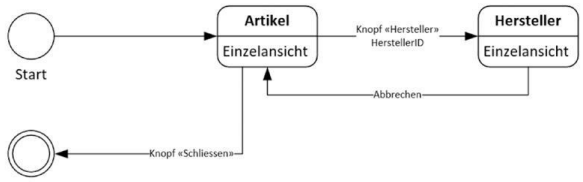
Modul 120 Spicker

**Ergonomie:** Aufgabenangemessenheit, Steuerbarkeit, Erwartungskonformität, Selbstbeschreibungsfähigkeit, Individualisierbarkeit, Lernförderlichkeit, Fehlertoleranz

| Präsentationsschicht / View | Benutzerinteraktion, Darstellen von Daten jeglicher Art, Eingaben verarbeiten | Benutzereingaben, Dateien (Bild, Text, ..), Steuerelemente |
|---|---|---|
| Applikationsschicht / Controller | Berechnungen, Prozesse abbilden, Daten abfragen, Mehrbenutzerbetrieb sicherstellen | Objekte |
| Persistenzschicht / Model | Speichern der Anwendungsdaten in «langlebiger» Form | Datensätze in Tabellen, Objekte in strukturierten Dateien (XML) |

**UML Zustandsübergangsdiagramm:**




**Klassen ändern:**

1. Rechtsklick auf Hauptklasse, 'Umbenennen', 'Ja'
2. In Klasse Attribute (Datenbankschicht) aufklappen, Rechtsklick und umbenennen
3. Property hinzufügen mit «prop» + Enter
4. Context.cs öffnen, ToTable(''Artikel'') ändern
5. DbSet ändern und in APIDemo.cs neue Attribute hinzufügen

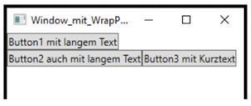
**XAML:** Layout

**Canvas:**

```
<Canvas>
    <Button Canvas.Left="10" Canvas.Top="10" Content="Button1"/>
    <Button Canvas.Right="20" Canvas.Top="20" Content="Button2"/>
</Canvas>
```


**WrapPanel:**

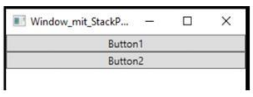
```
<WrapPanel>
    <Button Content="Button1 mit langem Text"/>
    <Button Content="Button2 auch mit langem Text"/>
    <Button Content="Button3 mit Kurztext"/>
</WrapPanel>
```




Kann mit *<WrapPanel Orientation='Vertical'> geändert werden*

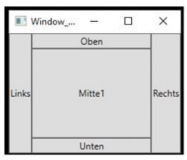
**StackPanel:**

```
<StackPanel>
    <Button Content="Button1"/>
    <Button Content="Button2"/>
</StackPanel>
```




**DockPanel:**

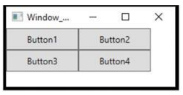
```
<DockPanel>
    <Button DockPanel.Dock="Left" Content="Links"/>
    <Button DockPanel.Dock="Right" Content="Rechts"/>
    <Button DockPanel.Dock="Top" Content="Oben"/>
    <Button DockPanel.Dock="Bottom" Content="Unten"/>
    <Button Content="Mitte1"/>
</DockPanel>
```



Modul 120 Spicker

**Grid:**

```
<Grid>
    <Grid.RowDefinitions>
        <RowDefinition Height="30"/>
        <RowDefinition Height="30"/>
    </Grid.RowDefinitions>
    <Grid.ColumnDefinitions>
        <ColumnDefinition Width="100"/>
        <ColumnDefinition Width="100"/>
    </Grid.ColumnDefinitions>
    <Button Grid.Column="0" Grid.Row="0" Content="Buttc
    <Button Grid.Column="1" Grid.Row="0" Content="Buttc
    <Button Grid.Column="0" Grid.Row="1" Content="Buttc
    <Button Grid.Column="1" Grid.Row="1" Content="Buttc
</Grid>
```



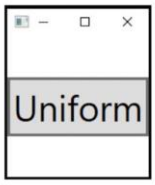

Entwurfsansicht:


RowDefinition etc kann als * mitgegeben werden -> Relativ (Beispiel: 2*, 3* -> 2 zu 3)

**ViewBox:**

```
<Grid>
<Viewbox Stretch="Fill">
<Button Content="Fill"/>
</Viewbox>
</Grid>
```


```
<Grid>
<Viewbox
Stretch="Uniform">
<Button
Content="Uniform"/>
</Viewbox>
</Grid>
```




```
<Grid>
<Viewbox Stretch="UniformToFill">
<Button Content="Uniform To Fill"/>
</Viewbox>
</Grid>
```


**Grössenangaben:**

| | |
|---|---|
| **px** | Pixel (Standard) Geräteunabhängige Einheit. «px» kann auch weggelassen werden. |
| **in** | Inches 1 Inch = 96 Pixel |
| **cm** | Zentimeter 1 cm = 96/2.54 Pixel |
| **pt** | Point 1 pt = 96/72 Pixel |

Mit * als Width wird sie automatisch auf das Maximum gescaled, mit Auto wird sie auf die maximale Grösse des Objekts basierend auf dem Inhalt gescaled.

**Margin:** Margin=''10,20,30,40''

**Schreibweisen:** Attribut-Schreibweise (<Button FontSize=''20''>Hello World</Button>), Eigenschaft-Element-Syntax (<Button><Button.FontSize>20</Button.FontSize>Hello World</Button>)

**Fokus:** TabIndex=''1'', TabIndex=''2''

**Standartaktionen:** IsDefault=''True'' -> Enter, IsCancel=''True'' -> Esc-Taste

**Code behind:** x:Name=''Name'' -> Name hinzufügen bei allen Elementen!

**Benutzersteuerelemente:** Rechtsklick auf Projektordner, Hinzufügen, Benutzersteuerelement -> <local:Window1 /> -> Je nach dem in <ScrollViewer>-Klasse einbauen!

**Navigation:**

```
1 Verweis | -Änderungen | -Änderungen, -Änderungen
private void Button_Click(object sender, RoutedEventArgs e)
{
    MainWindow mainWindow = new MainWindow();
    mainWindow.Show();
    this.Close();
}
```

**Klassendiagramm:** Klassendiagramm.cd öffnen und Klasse hineinziehen.