

Zeus ViewControls Ver. 1.0

Manual

Copyright © 2023 Χρήστος Μουρατίδης

Πίνακας περιεχομένων

ZEUS VIEWCONTROLS VERSION 1.0	1
ΟΙ ΟΡΙΣΜΟΙ ΤΩΝ ΚΛΑΣΕΩΝ	2
ΔΙΑΝΟΜΗ	3
ΕΠΙΚΟΙΝΩΝΙΑ	4
CARDBOX	5
ΙΔΙΟΤΗΤΕΣ	12
<i>CornerRadius</i>	13
<i>Styles και Templates</i>	14
<i>Parts και States</i>	15
<i>To default Style και ControlTemplate</i>	16

Zeus ViewControls version 1.0



Κλάσεις: CardBox

Inherits: System.Windows.Controls.CardBox

Namespace: Zeus.WPF.Controls.ViewControls

Assembly: ZeusViewControls (in ZeusViewControls.dll)

Dependencies: -

Περιγραφή

Η βιβλιοθήκη **Zeus ViewControls** περιλαμβάνει, προς το παρόν, το **CardBox** control που είναι ένα προσαρμοσμένο **ListBox** για την εμφάνιση των δεδομένων σε μορφή κάρτας. Οι κάρτες εμφανίζονται οριζόντια και προς τα κάτω. Επιπλέον, μπορούμε να ορίσουμε ένα στρογγυλεμένο περίγραμμα (**CornerRadius**) για όλο το control.

Στην ενότητα **Style** και **Templates** μπορείτε να δείτε πώς έχει καθοριστεί το γραφικό μέρος του **CardBox**, στο οποίο μπορείτε να βασιστείτε για να κάνετε τις δικές σας τροποποιήσεις. Η κύρια αλλαγή αφορά στο ότι εσωτερικά το control χρησιμοποιεί ένα **WrapPanel** (αντί του τυπικού **VirtualizingStackPanel**) για τα items.

Παρακάτω, παρατίθενται τα **ViewControls**.

	<u>CardBox</u>	Ένα προσαρμοσμένο ListBox που εμφανίζει τα δεδομένα σε κάρτες.
--	--------------------------------	---

Οι ορισμοί των κλάσεων

Οι κλάσεις έχουν οριστεί ως εξής:

- Για την κλάση **CardBox** :

Σύνταξη:

VB:

```
<TemplateVisualState(Name:="Normal", GroupName:="CommonStates"),  
TemplateVisualState(Name:="MouseOver", GroupName:="CommonStates"),  
TemplateVisualState(Name:="Disabled", GroupName:="CommonStates"),  
TemplateVisualState(Name:="Focused", GroupName:="FocusStates"),  
TemplateVisualState(Name:="Unfocused", GroupName:="FocusStates")>  
Public Class CardBox  
    Inherits ListBox
```

XAML Object Element Usage:

Εισαγωγή namespace:

```
xmlns:zeus="clr-namespace:Zeus.WPF.Controls.ViewControls;assembly=ZeusViewControls"
```

Χρήση:

```
<zeus:CardBox ... />
```

Διανομή

Κατά τη διανομή, στο φάκελο της εφαρμογής σας πρέπει να αντιγράψετε το **assembly αρχείο ZeusViewControls.dll**.

Επικοινωνία

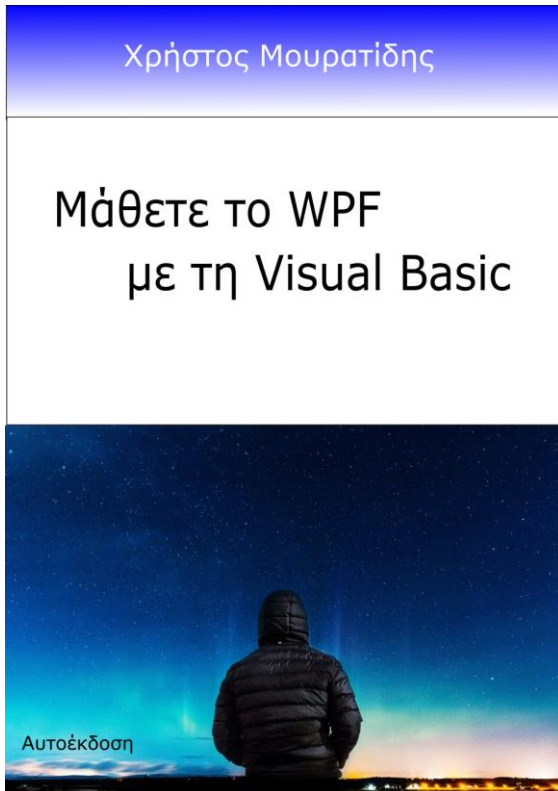
Για οποιαδήποτε πληροφορία ή διευκρίνιση παρακαλώ επικοινωνήστε στο :

mouratx@yahoo.com ή mouratx@hotmail.com



Χρήστος Μουρατίδης,
Πειραιάς, Μάρτιος 2023

Υ.Γ. Μπορείτε να επικοινωνήσετε μαζί μου για να προμηθευτείτε το **βιβλίο** μου
"Μάθετε το WPF με τη Visual Basic" (1.333 σελίδες, Αυτοέκδοση 2018).

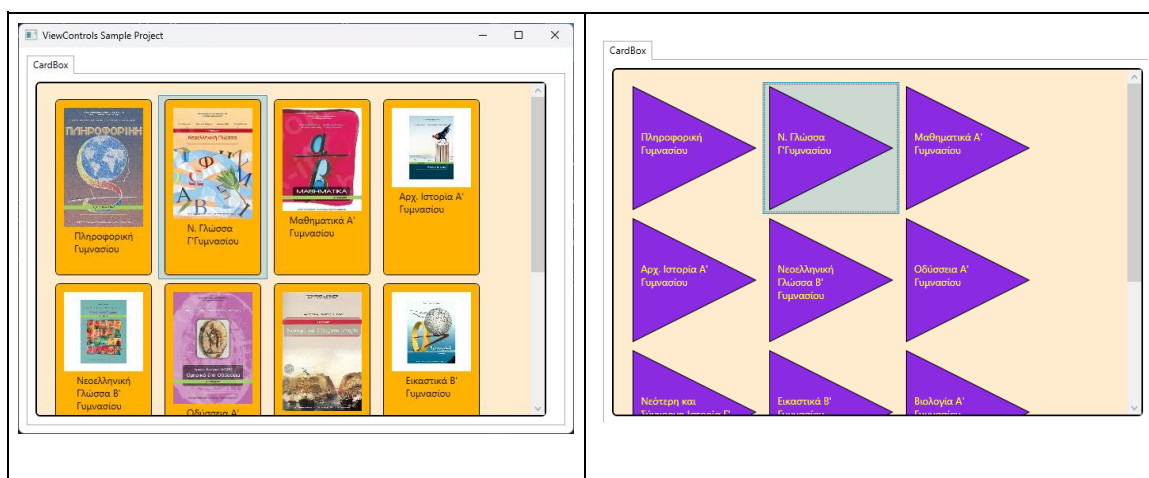




CardBox

Ένα προσαρμοσμένο **ListBox control** που εμφανίζει τα δεδομένα ενός αντικειμένου σε μορφή κάρτας. Μπορούμε να ορίσουμε και στρογγυλεμένο περίγραμμα με την ιδιότητα **CornerRadius**.

Η μορφή της κάρτας προσδιορίζεται με ένα **DataTemplate** και τίθεται στην ιδιότητα **ItemTemplate**.



Σύνταξη:

VB:

```
<TemplateVisualState(Name:="Normal", GroupName:="CommonStates"),
TemplateVisualState(Name:="MouseOver", GroupName:="CommonStates"),
TemplateVisualState(Name:="Disabled", GroupName:="CommonStates"),
TemplateVisualState(Name:="Focused", GroupName:="FocusStates"),
TemplateVisualState(Name:="Unfocused", GroupName:="FocusStates")>
Public Class CardBox
    Inherits ListBox
```

XAML Object Element Usage:

Εισαγωγή namespace:

```
xmlns:zeus="clr-namespace:Zeus.WPF.Controls.ViewControls;assembly=ZeusViewControls"
```

Χρήση:

```
<zeus:CardBox ... />
```

Παράδειγμα:

Στο παράδειγμα που ακολουθεί, τοποθετούμε ένα CardBox στο κύριο παράθυρο της εφαρμογής (κι εντός ενός TabItem). Τα δεδομένα των καρτών αφορούν κάποια βιβλία. Συνεπώς, ως ItemSource έχουμε μία λίστα από Book objects με ιδιότητες Id, Title, Writer, Publisher και ImageFront. Στον υποφάκελο Images της εφαρμογής έχουμε τοποθετήσει τα αρχεία εικόνων για τα εμπροσθόφυλλα των βιβλίων.

Το σημαντικό είναι ότι στα **Window.Resources** έχουμε δημιουργήσει ένα **DataTemplate resource** με όνομα **bookCard**. Εδώ, καθορίζουμε τη μορφή της κάρτας και πώς συνδέονται τμήματά της (συγκεκριμένα elements) με τις ιδιότητες του book object μέσω Binding εκφράσεων. Στον υποφάκελο Images της εφαρμογής υπάρχουν, επίσης, τα αρχεία εικόνων.

Το CardBox γεμίζει με δεδομένα στο συμβάν Loaded του MainWindow.

XAML:

```
<Window x:Class="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:z="clr-namespace:Zeus.WPF.Controls.ViewControls;assembly=ZeusViewControls"
  xmlns:p="clr-namespace:ViewControlsTestProject"
  mc:Ignorable="d"
  Title="MainWindow" Height="450" Width="800" WindowStartupLocation="CenterScreen"
  Loaded="Window_Loaded">

  <Window.Resources>

    <!-- Η κάρτα με τα δεδομένα. Εδώ διαμορφώνουμε τη μορφή της (data template). Καλό
    είναι να ορίσουμε σταθερό πλάτος-ύψος.-->
    <DataTemplate x:Key="bookCard" >

      <Border BorderBrush="Black" BorderThickness="1" CornerRadius="5"
        Padding="5" Background="#FFB300" Margin="3">

        <StackPanel Width="100" Height="200" Margin="5">
          <Image Source="{Binding ImageFront}"/>
          <TextBlock Text="{Binding Title}" TextWrapping="Wrap"
            HorizontalAlignment="Center" FontSize="12"/>
        </StackPanel>

      </Border>

    </DataTemplate>

  </Window.Resources>

  <Grid Margin="10">

    <TabControl Name="tabMain" Padding="10">

      <TabItem Header="CardBox" >
```



```
        <z:CardBox Name="crdBooks" Background="BlanchedAlmond" Width="645"
                Padding="15"
                BorderBrush="Black" BorderThickness="2" CornerRadius="5"
                ItemTemplate="{StaticResource bookCard}"
                HorizontalAlignment="Left"
                SelectionChanged="crdBooks_SelectionChanged" />
    </TabItem>
</TabControl>
</Grid>
</Window>
```

VB:

```
Class MainWindow
```

```
Private books As New List(Of Book)
```

```
#Region "Event handlers"
```

```
Private Sub Window_Loaded(sender As Object, e As RoutedEventArgs)
```

```
    If LoadBooksData() Then
```

```
        crdBooks.ItemsSource = books
```

```
    Else
```

```
        MessageBox.Show("Πρόβλημα στην φόρτωση των δεδομένων")
```

```
    End If
```

```
End Sub
```

```
'Όταν επιλεγεί κάποιο item.
```

```
Private Sub crdBooks_SelectionChanged(sender As Object, _
                                     e As SelectionChangedEventArgs)
```

```
    Dim crdBox As CardBox = TryCast(sender, CardBox)
```

```
    If crdBox IsNot Nothing Then
```

```
        If crdBox.SelectedIndex > -1 Then
```

```
            'Να εμφανιστεί ο τίτλος του βιβλίου που επιλέχθηκε.
            MessageBox.Show(books(crdBox.SelectedIndex).Title)
```

```
        End If
```

```
    End If
```

```
End Sub
```

```
#End Region
```

```
#Region "Private methods"
```

```
'Φόρτωση των δεδομένων.
```

```
Private Function LoadBooksData() As Boolean
```

```
Try
```

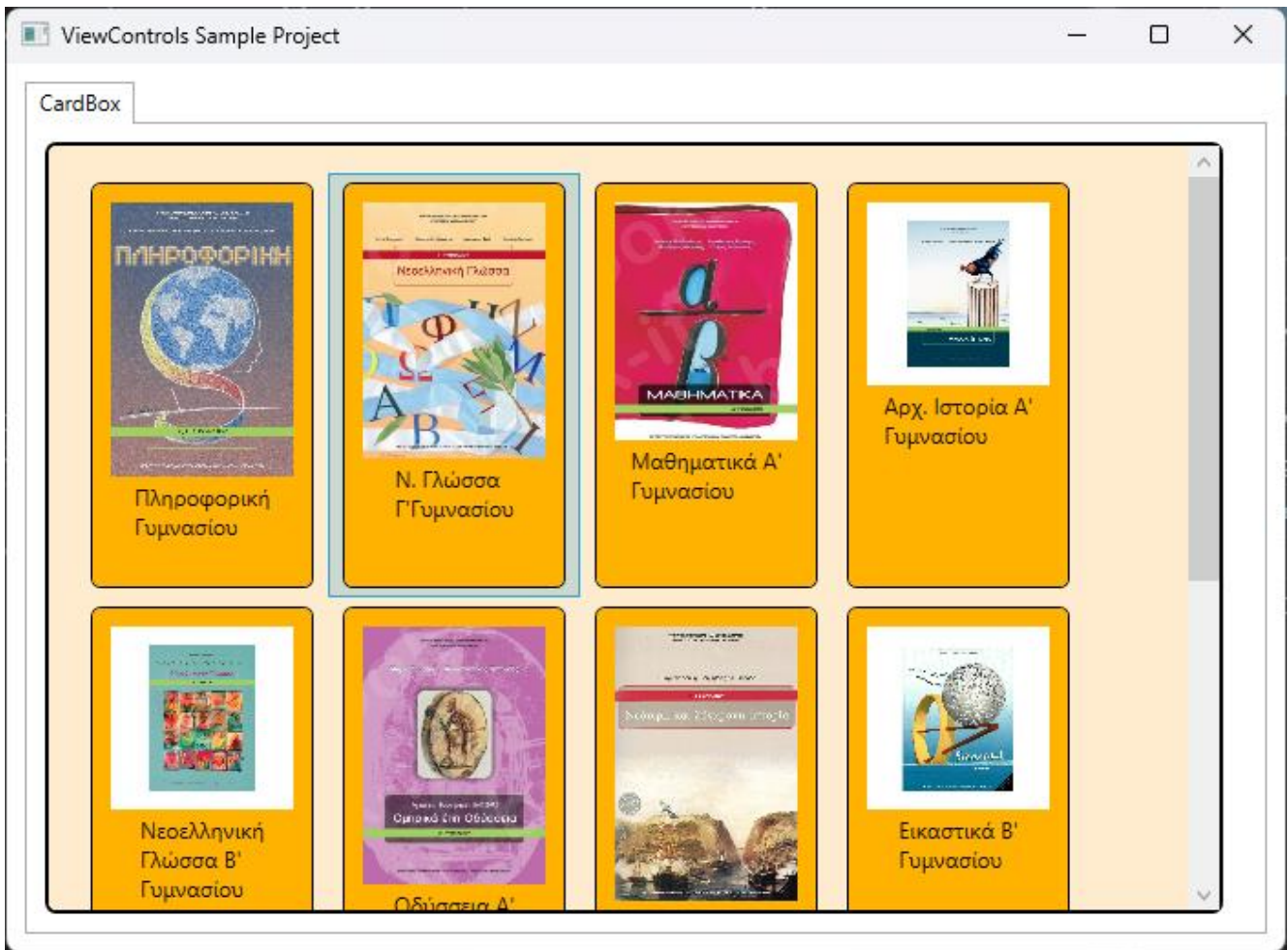
```
books.Clear()
```

```
With books
```

```
.Add(New Book With {.ID = 1, .Title = "Πληροφορική Γυμνασίου", _  
    .Publisher = "Διόφαντος", _  
    .Writer = "Αράπογλου, Μαβόγλου κλπ", _  
    .ImageFront = New BitmapImage(New _  
        Uri("Images/Book01.jpg", UriKind.Relative))})  
  
.Add(New Book With {.ID = 2, .Title = "N. Γλώσσα Γ' Γυμνασίου", _  
    .Publisher = "Διόφαντος", _  
    .Writer = "Κατσαρού, Μαγγανά κλπ", _  
    .ImageFront = New BitmapImage(New _  
        Uri("Images/Book02.jpg", UriKind.Relative))})  
  
.Add(New Book With {.ID = 3, .Title = "Μαθηματικά Α' Γυμνασίου", _  
    .Publisher = "Διόφαντος", _  
    .Writer = "Βανδουλάκης, Καλλιγιάς κλπ", _  
    .ImageFront = New BitmapImage(New _  
        Uri("Images/Book03.jpg", UriKind.Relative))})  
  
.Add(New Book With {.ID = 4, .Title = "Αρχ. Ιστορία Α' Γυμνασίου", _  
    .Publisher = "Διόφαντος", _  
    .Writer = "Κατσουράκος, Αλευρά κλπ", _  
    .ImageFront = New BitmapImage(New _  
        Uri("Images/Book04.jpg", UriKind.Relative))})  
  
.Add(New Book With {.ID = 5, .Title = "Νεοελληνική Γλώσσα Β' Γυμνασίου", _  
    .Publisher = "Διόφαντος", _  
    .Writer = "Γαβριηλίδου, Εμμανουηλίδης κλπ", _  
    .ImageFront = New BitmapImage(New _  
        Uri("Images/Book05.jpg", UriKind.Relative))})  
  
.Add(New Book With {.ID = 6, .Title = "Οδύσσεια Α' Γυμνασίου", _  
    .Publisher = "Διόφαντος", _  
    .Writer = "Σαμαρά, Τοπούζης", _  
    .ImageFront = New BitmapImage(New _  
        Uri("Images/Book06.jpg", UriKind.Relative))})  
  
.Add(New Book With {.ID = 7, .Title = "Νεότερη και Σύγχρονη Ιστορία Γ'  
        Γυμνασίου", _  
    .Publisher = "Διόφαντος", _  
    .Writer = "Λούβη, Ξιφαράς", _  
    .ImageFront = New BitmapImage(New _  
        Uri("Images/Book07.jpg", UriKind.Relative))})  
  
.Add(New Book With {.ID = 8, .Title = "Εικαστικά Β' Γυμνασίου", _  
    .Publisher = "Διόφαντος", _  
    .Writer = "Αντωνόπουλος, Δουκάκη", _  
    .ImageFront = New BitmapImage( New _  
        Uri("Images/Book08.jpg", UriKind.Relative))})
```

```
.Add(New Book With {.ID = 9, .Title = "Βιολογία Α' Γυμνασίου", _  
    .Publisher = "Διόφαντος", _  
    .Writer = "Μαυρικάκη, Γκούβρα κλπ", _  
    .ImageFront = New BitmapImage(New _  
        Uri("Images/Book09.jpg", UriKind.Relative))})  
  
.Add(New Book With {.ID = 10, .Title = "Αγγλικά Α' Γυμνασίου", _  
    .Publisher = "Διόφαντος", _  
    .Writer = "Καραγιάννη, Κούη κλπ", _  
    .ImageFront = New BitmapImage(New _  
        Uri("Images/Book10.jpg", UriKind.Relative))})  
  
End With  
  
Return True  
  
Catch  
    Return False  
  
End Try  
  
End Function  
  
#End Region  
  
End Class
```

Το αποτέλεσμα φαίνεται παρακάτω:



Δείτε με ένα διαφορετικό DataTemplate:

```
<!-- Μία δεύτερη μορφή (data template).-->
<DataTemplate x:Key="bookCard">
    <Canvas Width="150" Height="150" Margin="3">
        <Path Width="150" Height="150" Stretch="Fill"
            Stroke="Black" Fill="BlueViolet"
            Data="F1 M 319.344,237.333L 287.328,218.849L 287.328,255.818L
                319.344,237.333 Z" />
        <TextBlock Canvas.Left="10" Canvas.Top ="50" Width="110"
            Text="{Binding Title}"
            TextWrapping="Wrap" HorizontalAlignment="Center"
            VerticalAlignment="Center" Margin="0,3,0,0"
            Foreground="Yellow" FontSize="12"/>
    </Canvas>
</DataTemplate>
```

Το αποτέλεσμα φαίνεται παρακάτω:

CardBox

Πληροφορική Γυμνασίου	Ν. Γλώσσα Γ'Γυμνασίου	Μαθηματικά Α' Γυμνασίου
Αρχ. Ιστορία Α' Γυμνασίου	Νεοελληνική Γλώσσα Β' Γυμνασίου	Οδύσσεια Α' Γυμνασίου
Νεότερη και Σύγχρονη Ιστορία Β' Γυμνασίου	Εικαστικά Β' Γυμνασίου	Βιολογία Α' Γυμνασίου

Ιδιότητες

Όνομα	Περιγραφή
CornerRadius	Καθορίζει τη στρογγυλότητα των άκρων , τύπου ConrgerRadius .

CornerRadius

Καθορίζει τη **στρογγυλότητα των άκρων**, τύπου **CornerRadius**.

Σύνταξη:

VB:

```
Public Property CornerRadius As CornerRadius
```

Τύπος: **System.Windows.CornerRadius**

Προσδιορίζουμε μία τιμή **CornerRadius** που θα καθορίζει πόσο στρογγυλεμένες θα είναι οι άκρες του **CardBox**. Η default τιμή είναι "1" (ομοιόμορφα σε όλες τις γωνίες).

Dependency Property Information:

Identifier field: **CornerRadiusProperty**

Παράδειγμα:

Στο επόμενο παράδειγμα, καθορίζουμε μία αρκετά στρογγυλή τιμή στο περίγραμμα του **CardBox**:

XAML:

```
<zeus:CardBox Name="crdBooks" ... CornerRadius="20" />
```

VB:

```
Imports Zeus.WPF.Controls.ViewControls
```

```
crdBooks.CnerRadius = New CornerRadius(20)
```

Styles και Templates

- Parts και States
- To default Style και ControlTemplate

Parts και States

Το default `ControlTemplate` περιλαμβάνει κάποια **part names** και **visual states**. Μπορείτε να τροποποιήσετε το default `ControlTemplate` ώστε να δώσετε στο control μία μοναδική εμφάνιση.

Δεν έχουν οριστεί **part names** στο **CardBox**.

Ο παρακάτω πίνακας εμφανίζει τα **visual states** του **CardBox**:

VisualState	VisualStateGroup	Περιγραφή
Normal	CommonStates	Το default state.
MouseOver	CommonStates	Το state όταν ο δείκτης του ποντικιού είναι πάνω από το control.
Disabled	CommonStates	Το state όταν το control είναι disabled.
Focused	FocusStates	Το state όταν το control έχει το focus.
Unfocused	FocusStates	Το state όταν το control δεν έχει το focus.

Το default `ControlTemplate` έχει καθορισμένο μόνο το `Disabled state`. Μπορείτε να δημιουργήσετε ένα custom `ControlTemplate` για να το αλλάξετε ή/και για να καθορίσετε τα υπόλοιπα. Επιπλέον, χρησιμοποιούνται και `ControlTemplate Triggers`.

To default Style και ControlTemplate

Ο XAML κώδικας για το **default Style** και **ControlTemplate** φαίνεται παρακάτω. Μπορείτε να βασιστείτε σε αυτόν για να δημιουργήσετε μία μικρή ή μεγάλη παραλλαγή του δικού σας custom Style και ControlTemplate:

```
xmlns:z="clr-namespace:Zeus.WPF.Controls.ViewControls"

<!--===== -->
<!-- Default Style for CardBox -->
<!--===== -->

<Style TargetType="{x:Type z:CardBox}" >

    <Setter Property="ItemsControl.ItemsPanel">

        <Setter.Value>
            <ItemsPanelTemplate>
                <WrapPanel IsItemsHost="True" />
            </ItemsPanelTemplate>
        </Setter.Value>

    </Setter>

    <Setter Property="Template">
        <Setter.Value>

            <ControlTemplate TargetType="{x:Type z:CardBox}">

                <!-- Root element -->
                <Border Name="border"
                    Background="{TemplateBinding Background}"
                    BorderBrush="{TemplateBinding BorderBrush}"
                    BorderThickness="{TemplateBinding BorderThickness}"
                    CornerRadius="{TemplateBinding CornerRadius}" />

                <!-- Visual States -->
                <VisualStateManager.VisualStateGroups >

                    <VisualStateGroup Name="CommonStates">

                        <VisualStateGroup.Transitions >
                            <VisualTransition
                                GeneratedDuration="0:0:0.2">
                            </VisualTransition>
                        </VisualStateGroup.Transitions>

                        <VisualState Name="Normal"/>
                        <VisualState Name="MouseOver" />
                        <VisualState Name="Disabled">
                            <Storyboard >
                                <ColorAnimation
                                    Storyboard.TargetName="border"
                                    Storyboard.TargetProperty="Foreground"
                                    To="LightGray" />
                            </Storyboard>
                        </VisualState>
                    </VisualStateGroup>
                </VisualStateManager.VisualStateGroups >
            </ControlTemplate>
        </Setter.Value>
    </Setter>
</Style>
```

```
        </VisualState>
    </VisualStateGroup>
    <VisualStateGroup Name="FocusStates">
        <VisualStateGroup.Transitions >
            <VisualTransition
                GeneratedDuration="0:0:0.2">
            </VisualTransition>
        </VisualStateGroup.Transitions>
        <VisualState Name="Focused" />
        <VisualState Name="Unfocused"/>
    </VisualStateGroup>
</VisualStateManager.VisualStateGroups>
<!-- Content -->
<ScrollViewer Name="cards"
    IsEnabled="{TemplateBinding IsEnabled}"
    <ItemsPresenter Name="cardItems"
        Margin="{TemplateBinding Padding}" />
</ScrollViewer>
</Border>
<!-- ControlTemplate Triggers-->
<ControlTemplate.Triggers>
    <Trigger Property="IsEnabled" Value="False">
        <Setter Property="Background"
            TargetName="border" Value="Transparent"/>
        <Setter Property="BorderBrush"
            TargetName="border" Value="LightGray"/>
        <Setter Property="Opacity"
            TargetName="cards" Value="0.5" />
    </Trigger>
</ControlTemplate.Triggers>
</ControlTemplate>
</Setter.Value>
</Setter>
</Style>
```

Τέλος Manual