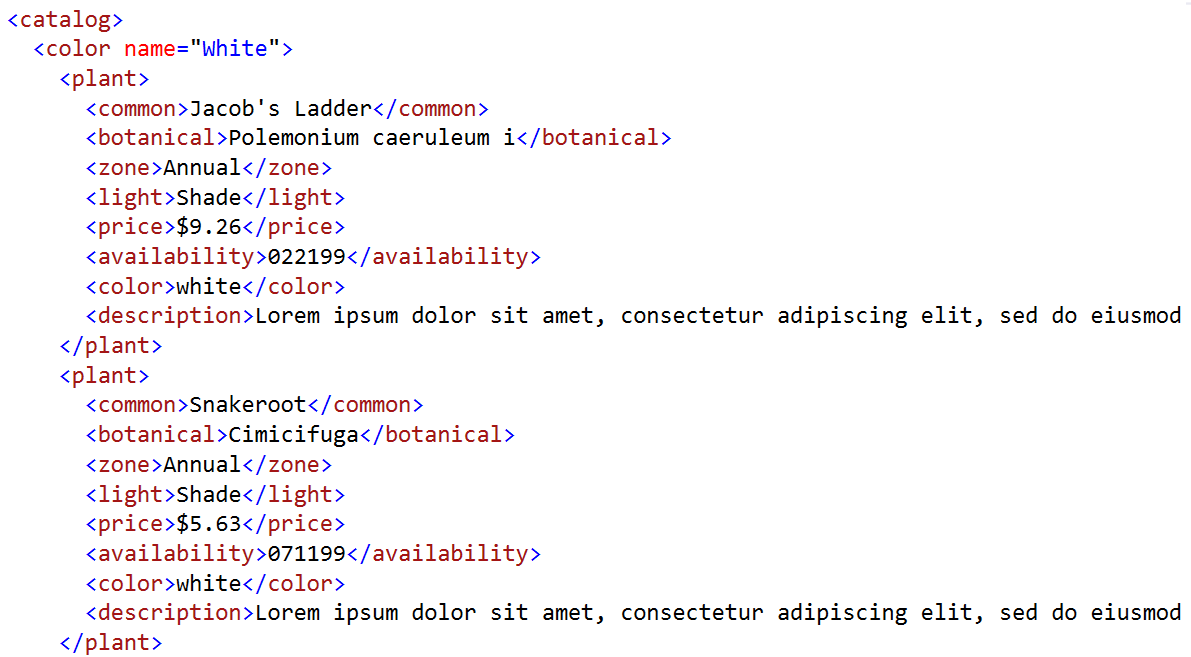
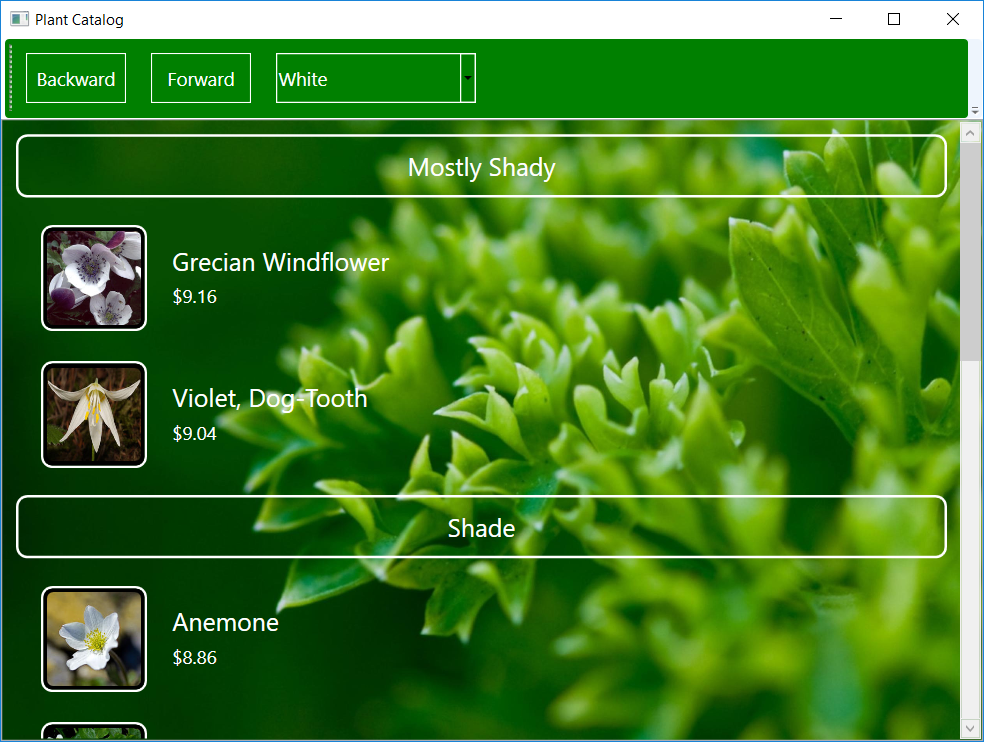
***Oefening Planten BIS***

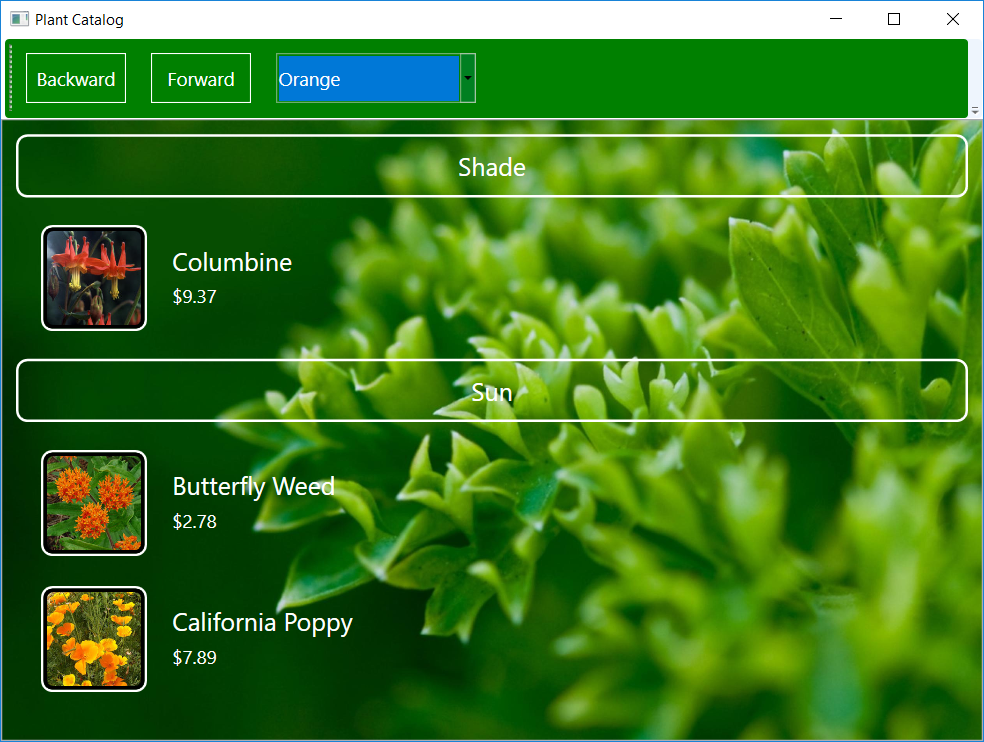
Maak een *WPF* toepassing waarmee je een plantencatalogus kan bekijken. De gegevens voor de catalogus bevinden zich in het xml-bestand *catalogus.xml*:



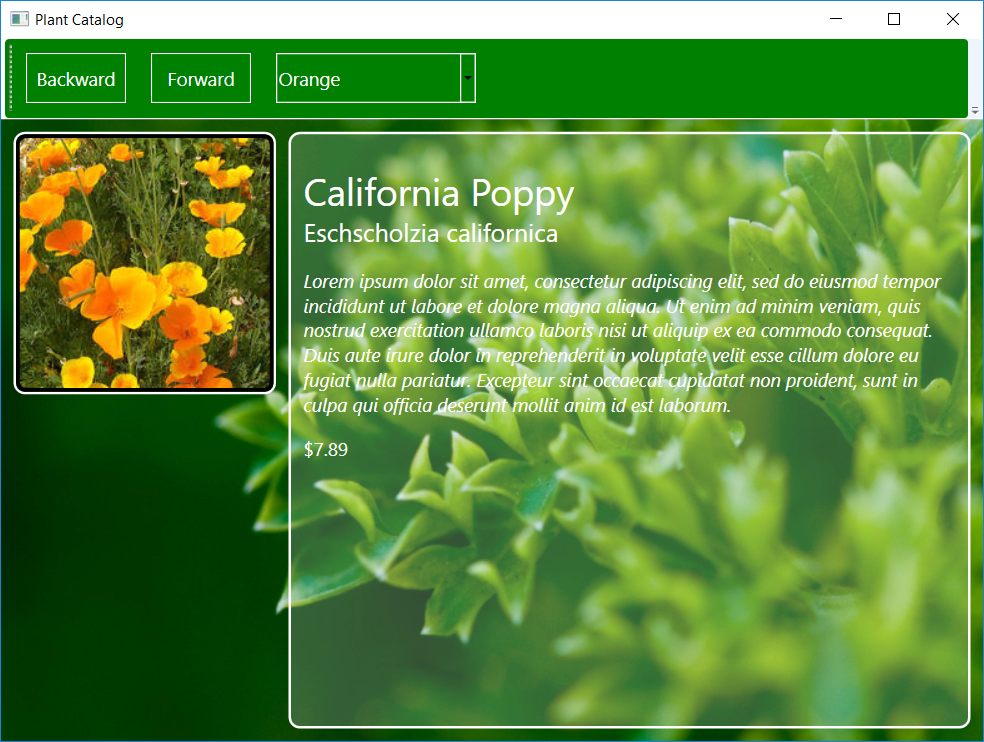
Het *main window* van de toepassing ziet er als volgt uit:



Met behulp van de *ComboBox* bovenaan kan je een andere bloemenkleur selecteren:



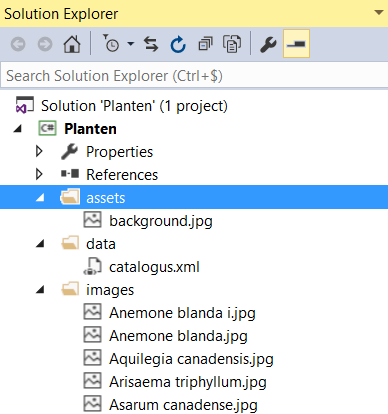
Als je dubbel klikt op een plant, dan krijg je de detailgegevens:



Met de knoppen *Backward* en *Forward* kan je bladeren doorheen de *history*.

***1. Project maken***

Maak een nieuw WPF project *Planten*. Voeg een map *assets* (met de achtergrond), een map *data* (met het xml-bestand) en een map *images* (met de afbeeldingen) toe aan je project:



In de Application Resources definiëren we volgende kleuren:

<SolidColorBrush x:Key="ToolBarKleur" Color="Green" />

<SolidColorBrush x:Key="RandKleur" Color="White" />

<SolidColorBrush x:Key="AchtergrondKleur" Color="Black" />

<SolidColorBrush x:Key="LetterKleur" Color="White" />

Gebruik straks in je toepassing de *Keys*. Heb je later liever andere kleuren in je toepassing, dan hoef je alleen in de resources de *Color values* aan te passen.

***2. MainWindow layout***

We ontwerpen nu het *MainWindow*. Plaats de breedte op 800 en de hoogte op 600. De titel is *Plant Catalog*. *MainWindow* doet dienst als container van de toepassing. Je kan dit doen door een *NavigationWindow* te creëren, maar het kan ook door *frames* te gebruiken. Hier doen we het met *frames*.

Vervang in *MainWindow* de root-*Grid* door een *DockPanel* om bovenaan een *Toolbar* te tonen en daaronder een *Frame*. De achtergrondkleur van de *Toolbar* is *ToolbarKleur*. Stop in de *Toolbar* twee *Buttons* en een *ComboBox*. Voor het *Frame* gebruik je onderstaande code:

<Frame Source="PageOverzicht.xaml" Name="frame" NavigationUIVisibility="Hidden" >

<Frame.Background>

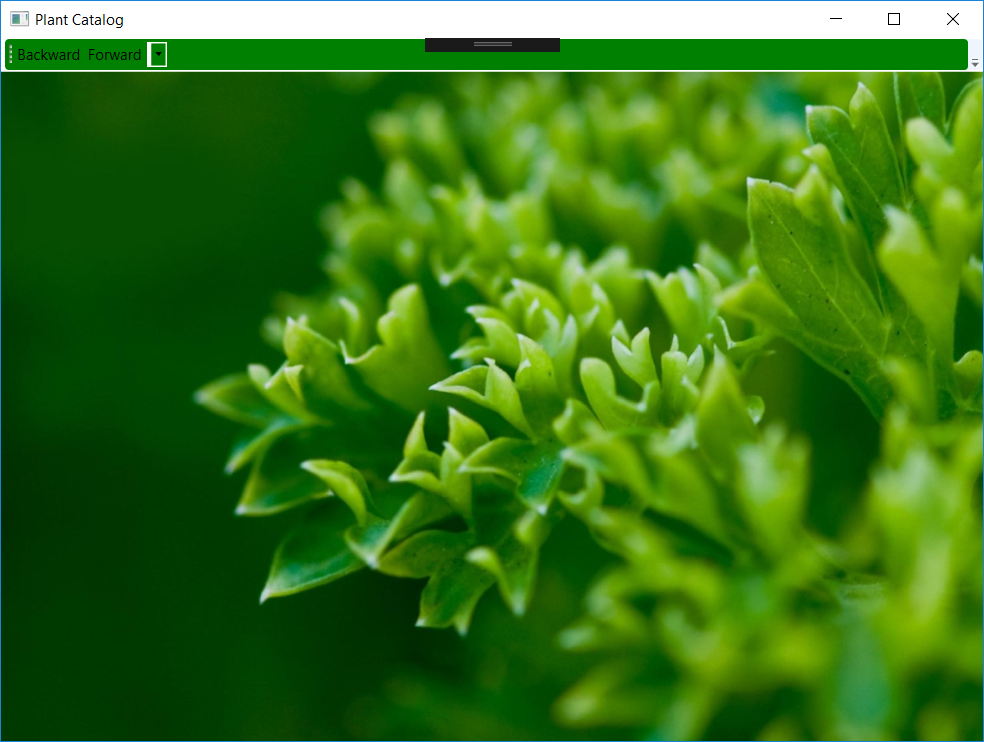
<ImageBrush ImageSource="assets/background.jpg" Stretch="UniformToFill"/>

</Frame.Background>

</Frame>

De eigenschap *NavigationUIVisibility* gebruiken we om de lelijke navigatiebalk van WPF te verbergen. We hebben nl. onze eigen *Toolbar*. De andere *Xaml*-code spreekt voor zich.

Maak nu de (voorlopig lege) pagina *PageOverzicht.xaml* aan. Bij uittesten krijg je nu alvast:

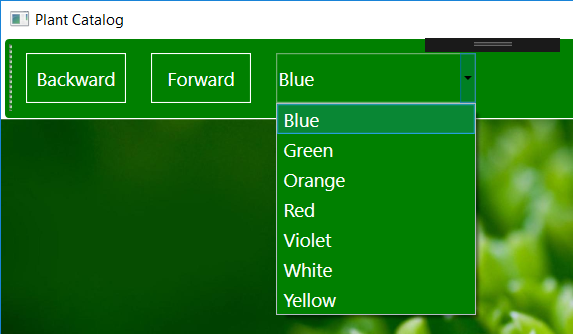


Definieer nu binnen *MainWindow* een stijl *ToolBarButton* en een stijl *ToolBarCombo* zodat je volgende knoppen/combobox in de *ToolBar* krijgt (haal de kleuren uit de *Application Resources*):



***3. Kleuren inlezen***

Vul de *ComboBox* nu op met de kleuren uit het xml-bestand (gebruik een *XmlDataProvider* en *DataBinding*). Om de kleuren in alfabetische volgorde te laten verschijnen, zoek je op hoe je *SortDescriptions* via code toevoegt aan een *XAML-ComboBox,* die gebruik maakt van *DataBinding*. Selecteer ook meteen de eerste kleur. Dit is het resultaat:

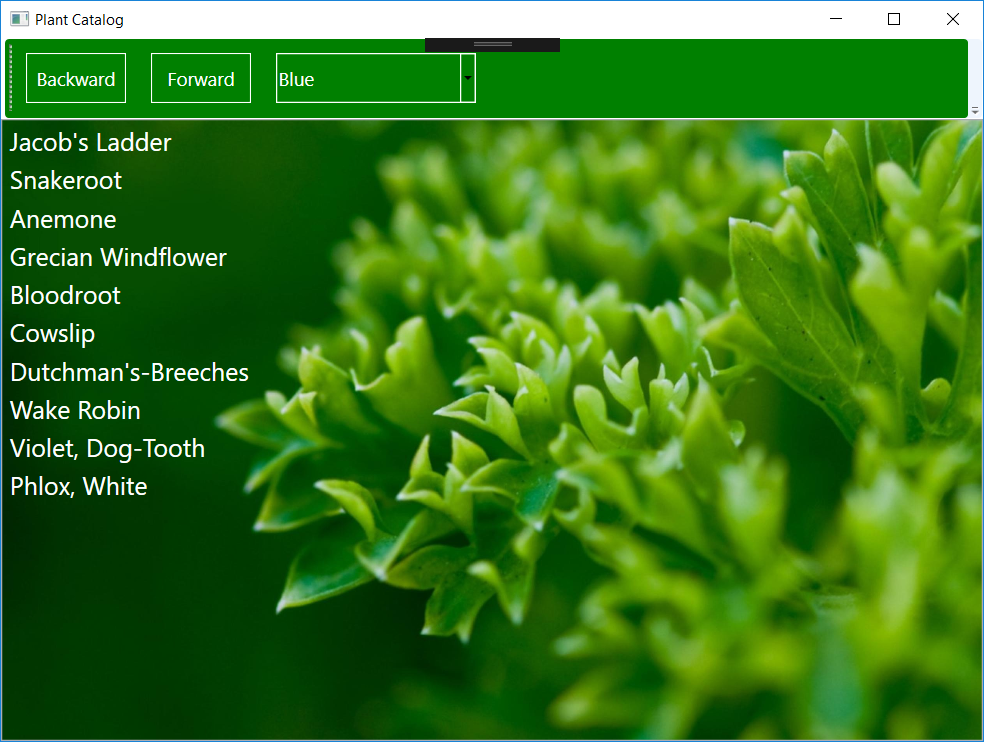


***4. Planten inlezen***

Plaats in de root-*Grid* van de pagina een *ListBox* met transparante achtergrond. Gebruik dan *DataBinding* om de namen van alle planten in die ListBox te tonen. Voorlopig tonen we gewoon de witte bloemen. Dit doe je door met de *DataContext* van de *ListBox* alleen de witte kleur te selecteren:

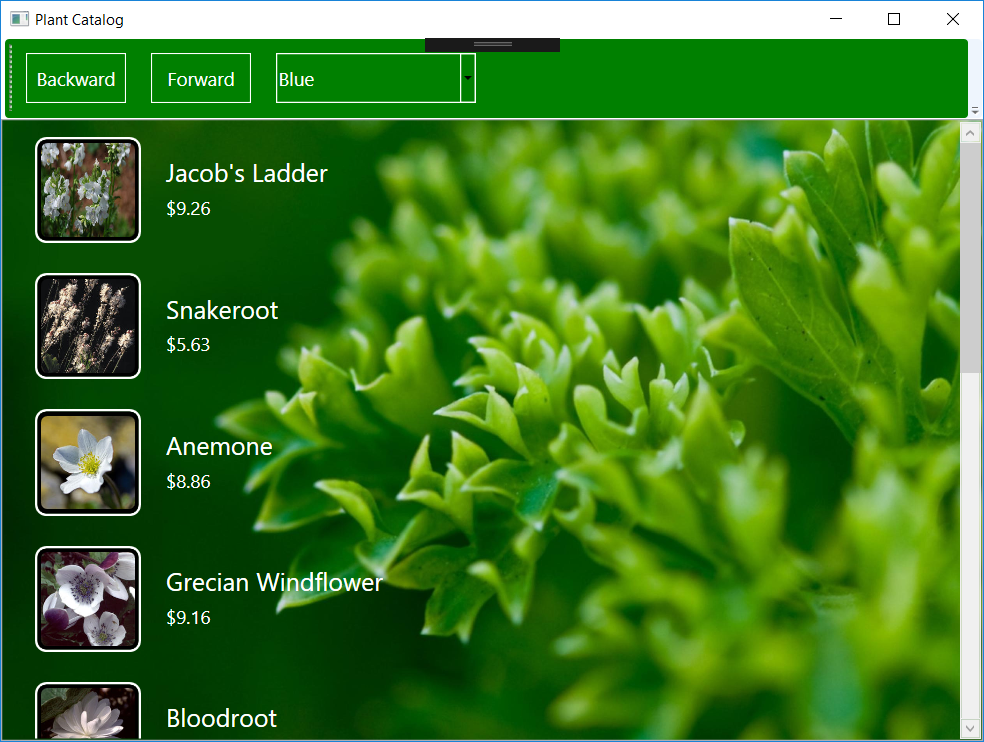
DataContext="{Binding Source={StaticResource xmlData}, XPath=color[@name\=\'White\']}"

Je krijgt:



***5. ListBoxPlanten Layout***

Pas de *DataTemplate* aan zodat de (witte) planten als volgt getoond worden:

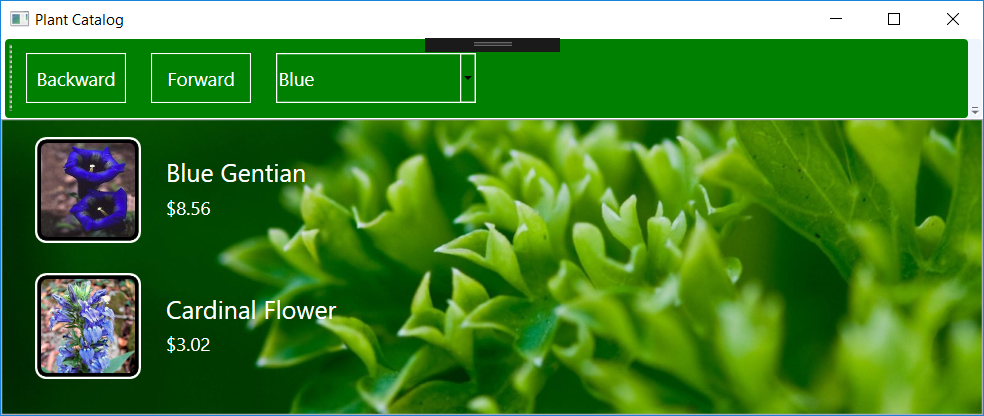


De witte rand rond de afbeelding is een *Border* (met afgeronde hoeken en opgevuld met de achtergrondkleur). Om ook de afbeelding met afgeronde hoeken te krijgen, gebruik je een Rectangle (met afgeronde hoeken), die je opvult met de foto van de bloem (*ImageBrush*):

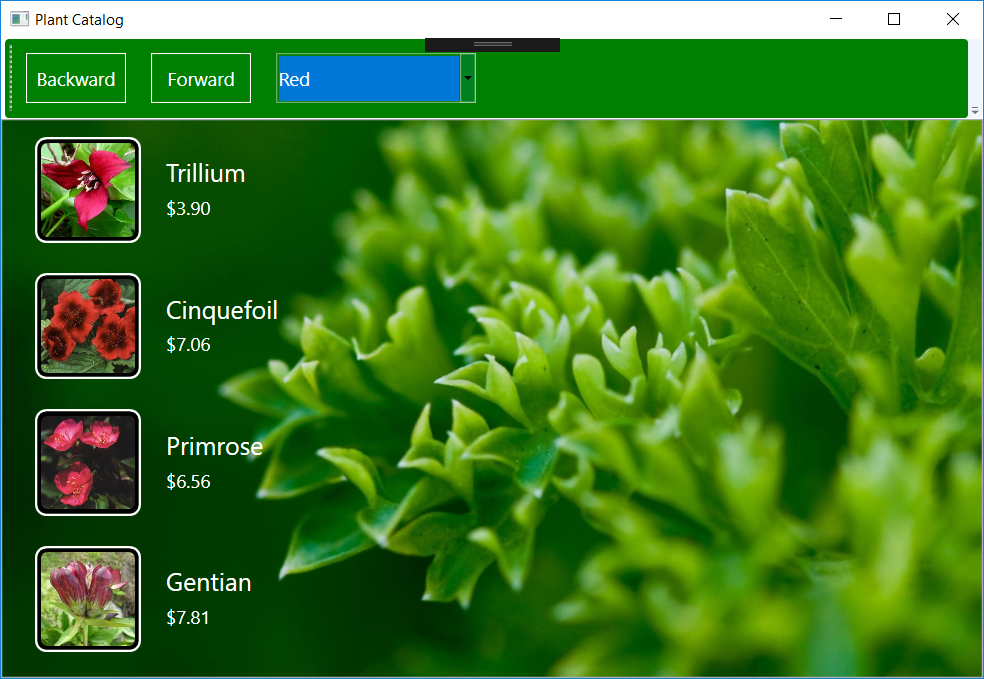


***6. Kleuren koppelen***

We proberen nu de juiste bloemen te tonen (afhankelijk van de gekozen kleur in de *ComboBox*). Doe eerst de hardgecodeerde *DataContext* weg. Programmeer dan *Page\_Loaded* van *PageOverizcht*. Stel via C#-code de *DataContext* gelijk aan het geselecteerde item uit de *ComboBox*. Bij opstarten krijg je nu alvast de blauwe bloemen:

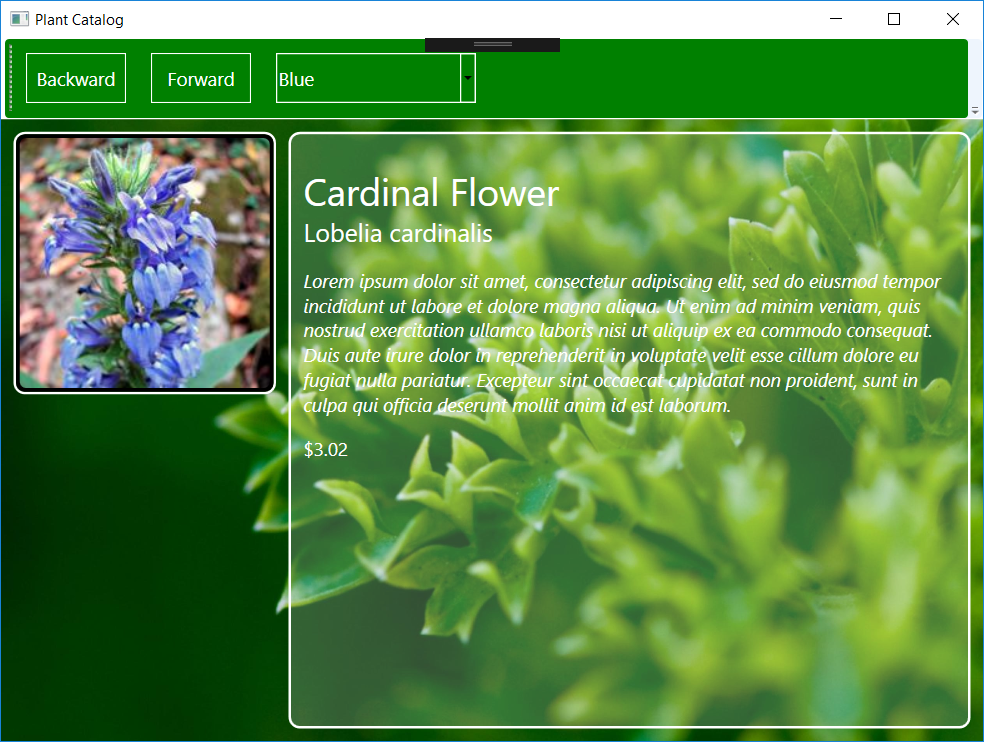


Programmeer dan *SelectionChanged* van de *ComboBox*. Koppel ook daar de *DataContext* van de *ListBox* met de *SelectedItem* van de *ComboBox*. Na selecteren van *Red*, krijg je:



***7. Details opvragen***

Zorg er nu voor dat je kan dubbel klikken op een plant. Je springt dan naar (een nieuwe pagina) *PageDetail.xaml* waarin je de detail-gegevens toont van de geselecteerde plant:

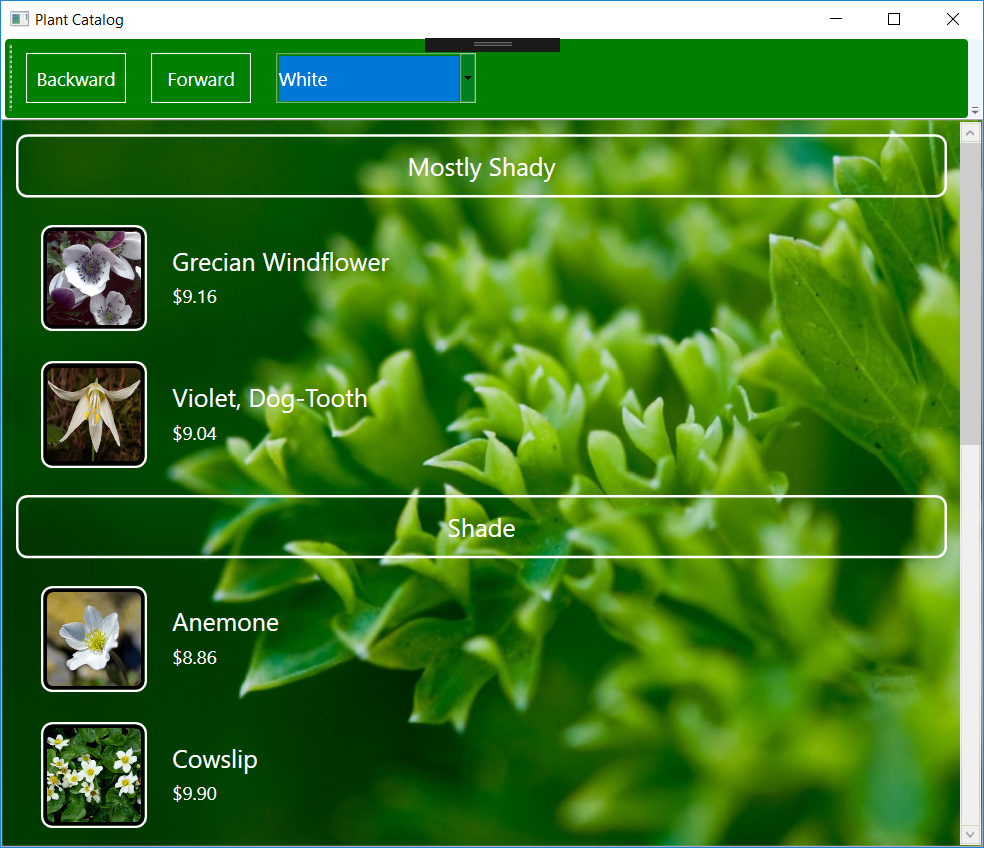


***8. Backward en Forward***

Programmeer nu deze twee knoppen zodat je verder en terug kan gaan in je *history* (als die er is). Controleer ook eens dat je geen foutmelding krijgt als je de kleur wijzigt op *PageDetail*.

***9. Groeperen op standplaats***

Zorg er nu voor dat de planten gegroepeerd worden op standplaats. Voor elke groep toon je een gepaste header. Binnen de groep worden de planten op alfabetisch op naam gesorteerd:



***10. Application Resources***

Probeer nu eens met andere kleuren in de Application Resources. Worden de kleuren overal mooi aangepast?