Microsoft

developers
build what's next now

tech days 2012

tech 2012 days

# Windows Phone applicaties omzetten naar Windows 8

Fons Sonnemans
Trainer
Reflection IT

developers
build what's next now

Microsoft

# Fons Sonnemans

- Freelance Software Development Consultant
  - Visual C#, Visual Basic, JavaScript
  - Windows Forms, ASP.NET WebForms & MVC, Silverlight, Windows Phone, Windows 8
  - SQL Server, Oracle
  - Trainer, Coach, Advisor, Architect, Designer, Developer
- www.reflectionit.nl

developers
build what's next now

Microsoft

# Windows Phone Apps

http://www.windowsphone.com/en-us/search?q=reflection+it&wa=wsignin1.0

Marketplace search results

## Marketplace search results

reflection it

Search all of Windows Phone | Search the web with Bing

Animal Sounds
Ratings: 171
Free

Tattoo Tester
Ratings: 204
$1.99

Sudoku
Ratings: 268
$1.49

Photo Frames
Ratings: 86
$1.29

Sudoku Free
Ratings: 35
Free

Bloq
Ratings: 42
Free

Countdown
Ratings: 54
$0.99

Covers
Ratings: 36
$1.49

Blueprint
Ratings: 20
Free

Hot or Not Scanner
Ratings: 3
$0.99

Let It Snow
Ratings: 1
$0.99

TechDays NL
Ratings: 1
Free

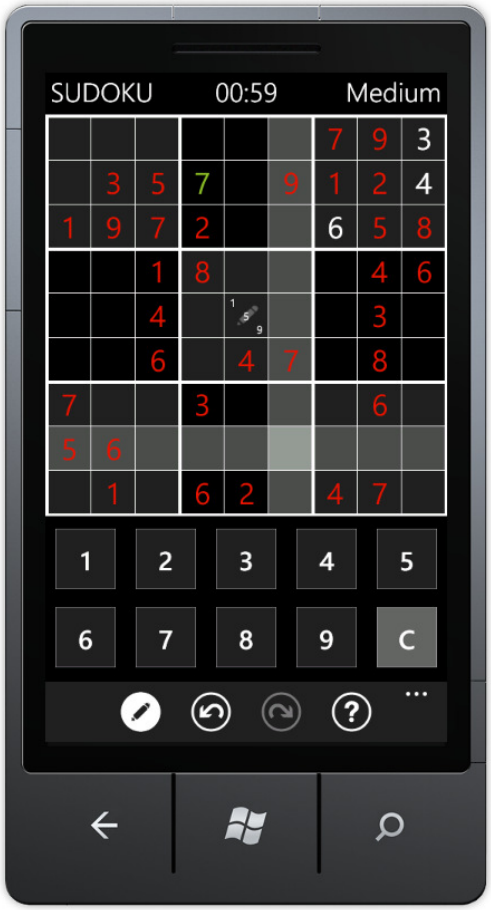
Week Schedule
Ratings: 1
$0.99

Valentine Stickers
No ratings
$1.49

developers
build what's next now

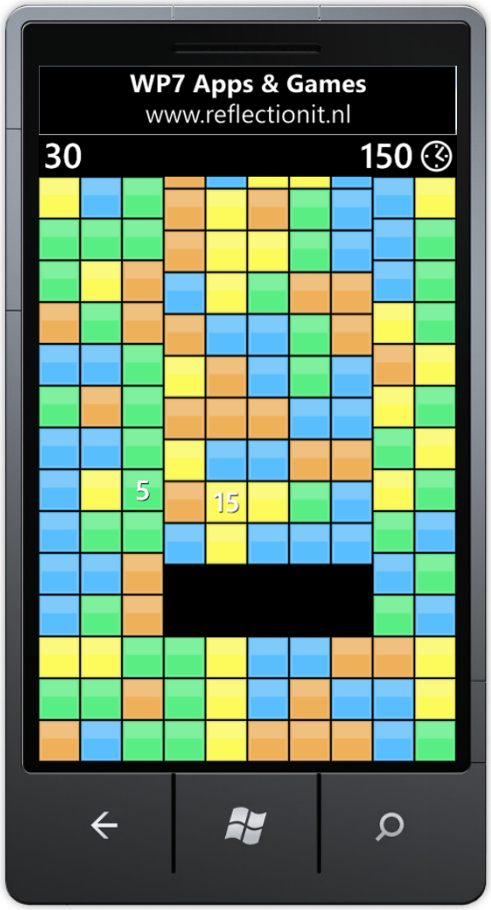
Microsoft

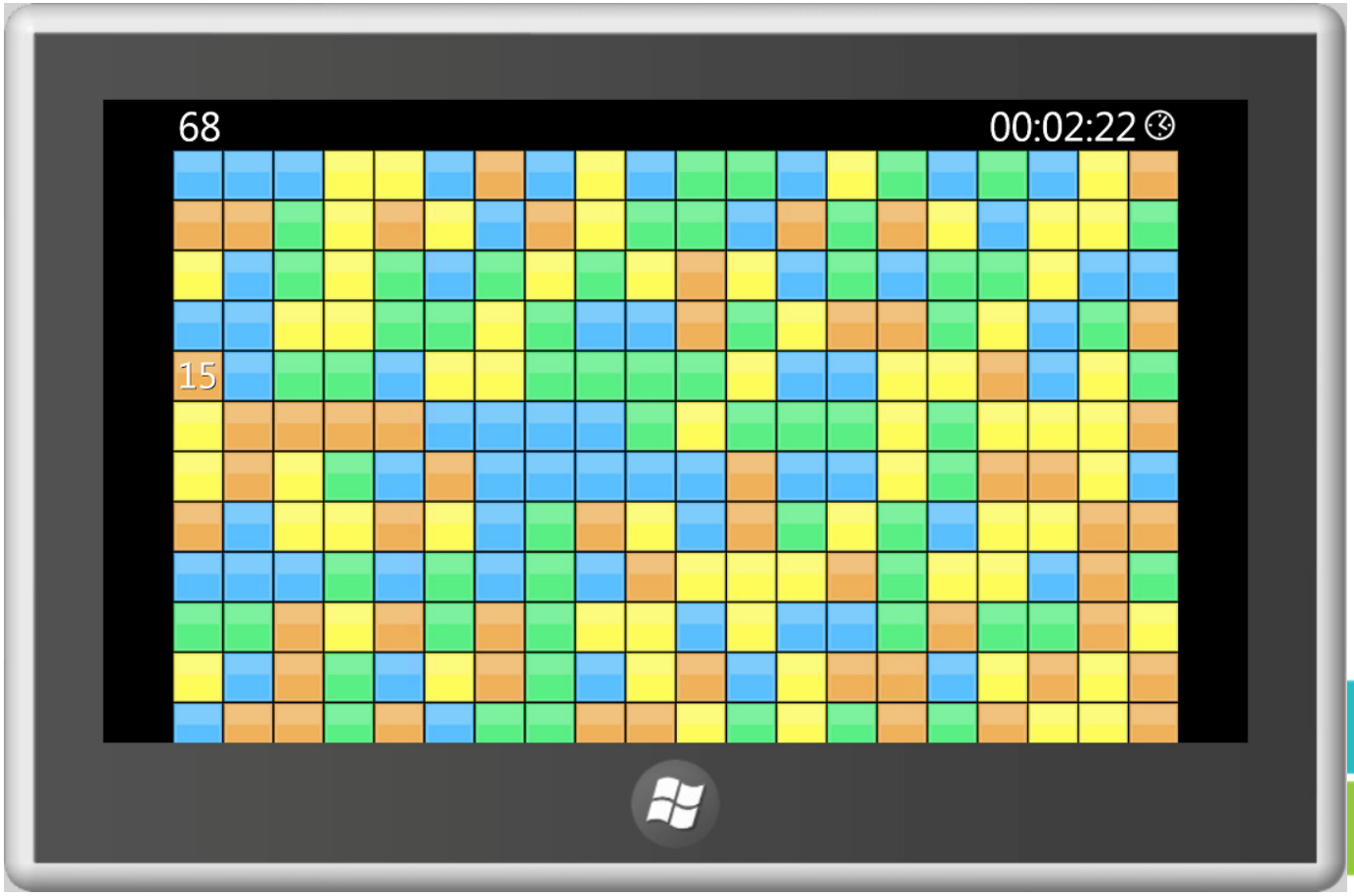
# Sudoku


SUDOKU
00:59
Medium
1
2
3
4
5
6
7
8
9
C

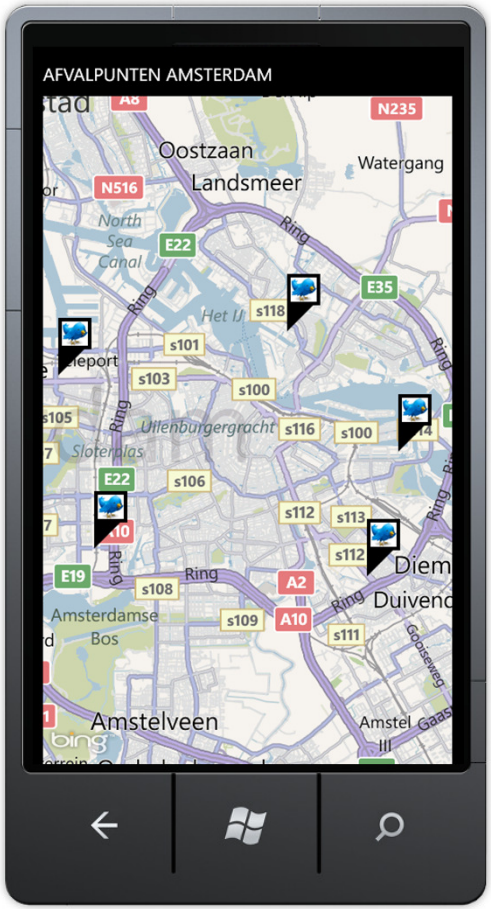

Notes
Undo
Redo
Validate
Hint
SUDOKU
easy
1
2
3
4
5
6
7
8
9
C

# Bloq



# Amsterdam Afvalpunten





developers
build what's next now

Microsoft

# Topics

- MVVM Design Pattern
- Windows Phone Flashcard Game
- Windows 8 Flashcard Game
- Tips & Tricks

developers
build what's next now

Microsoft

# MVVM Design Pattern

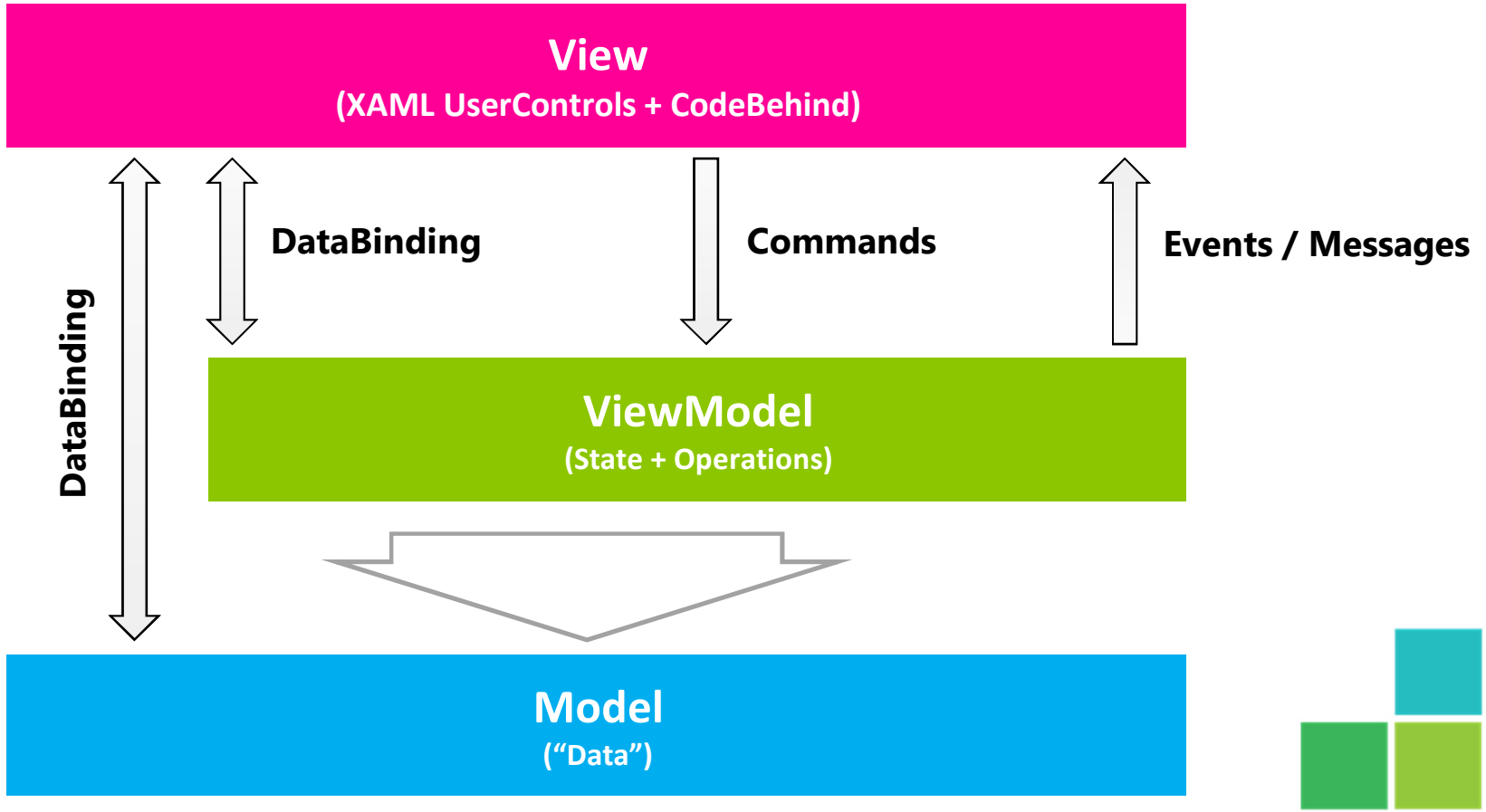
Microsoft

# Model View ViewModel

- Motivations:
  - Reduces complexity with Model to UI integration
  - Separation of concerns
  - Clear Designer-Developer  separation
  - Makes code more Unit testable

- Approach:
  - Split the  UI architecture into Model, View and View-Model
  - Model: Represents the data
  - View : UI defined declaratively in XAML
  - View Model: Specialization of the Model that View uses for data binding

# Model View ViewModel




developers
build what's next now

Microsoft

Sample application

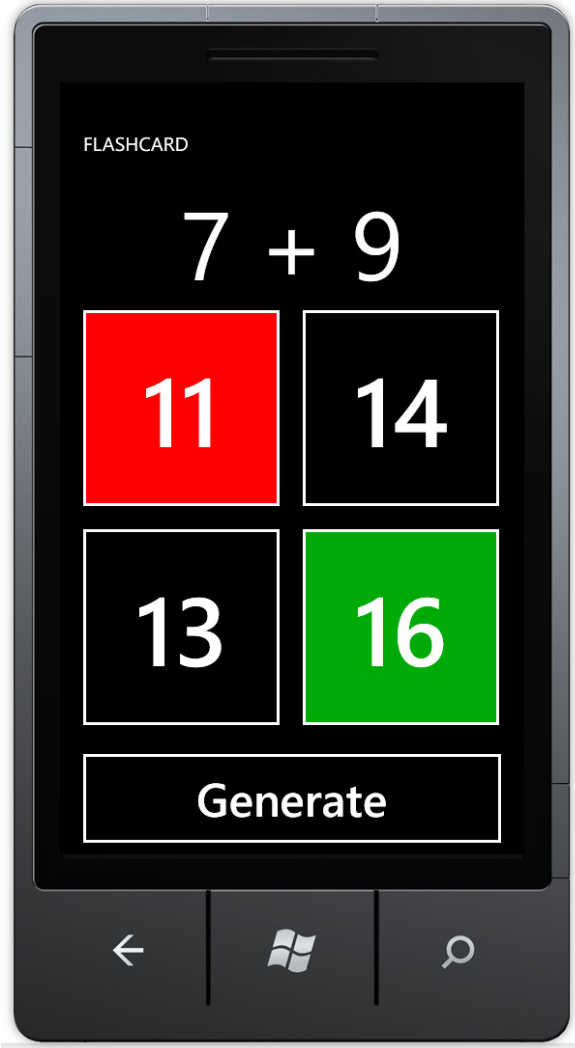
# Windows Phone Flashcard Game

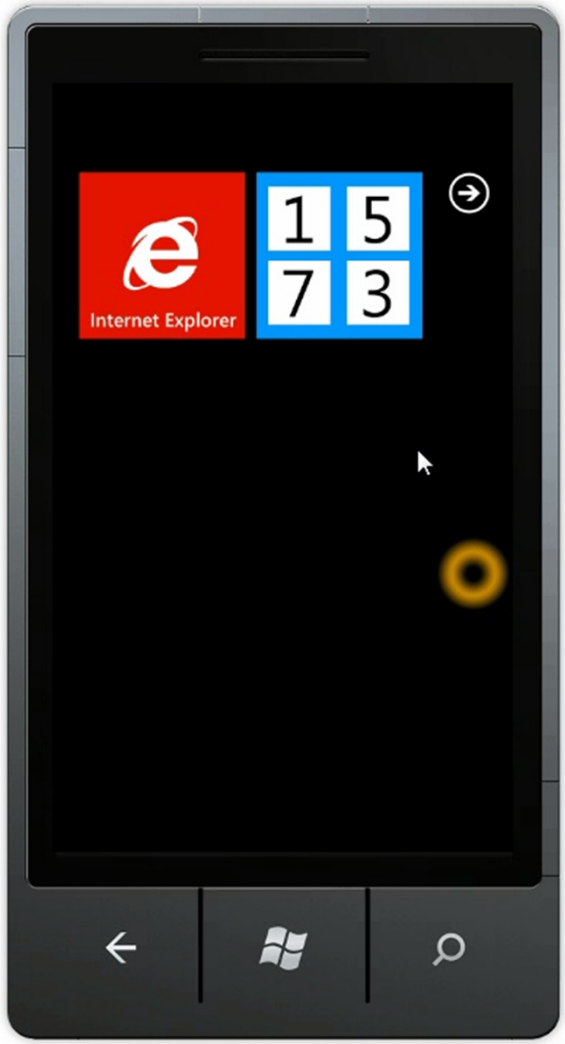
Microsoft

tech 2012 days

# Video

## Flashcard game



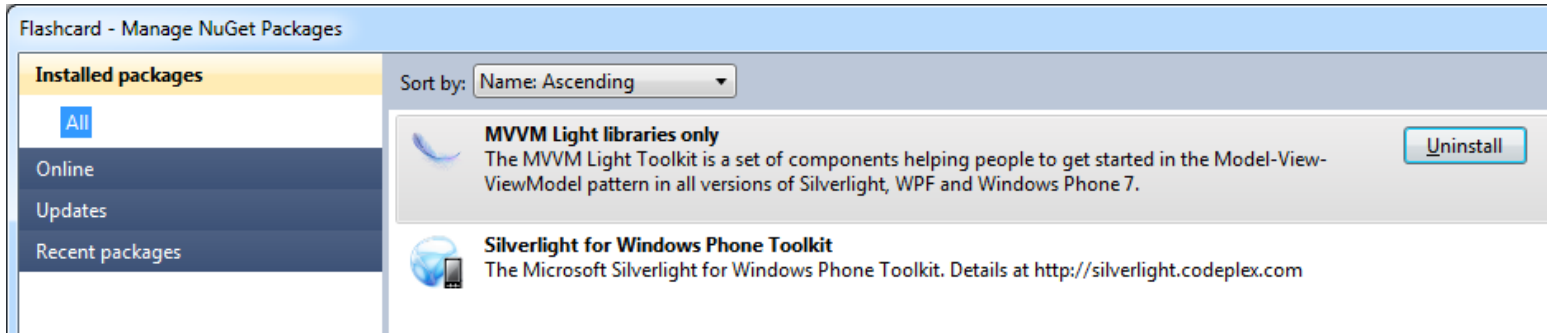



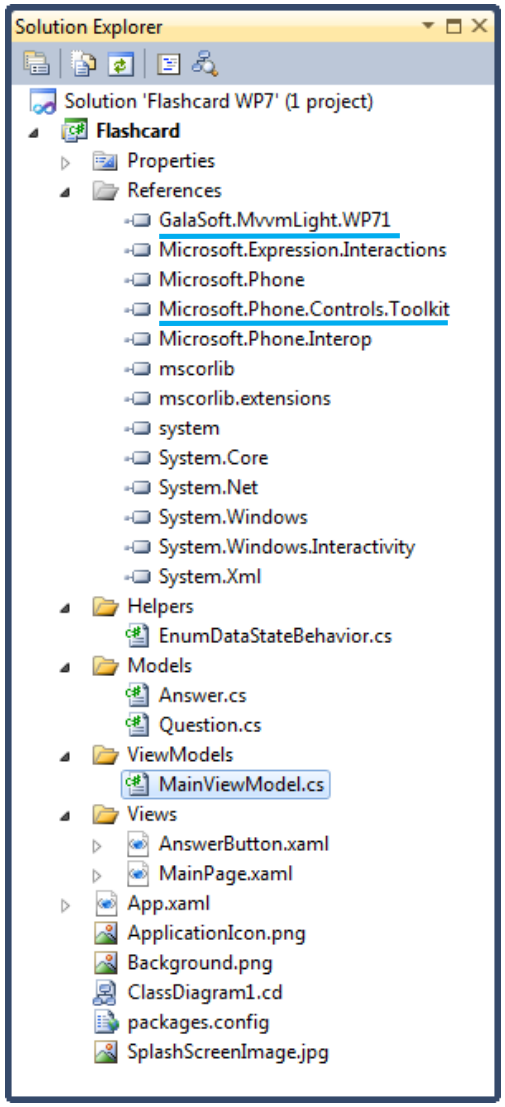

developers
build what's next now

Microsoft

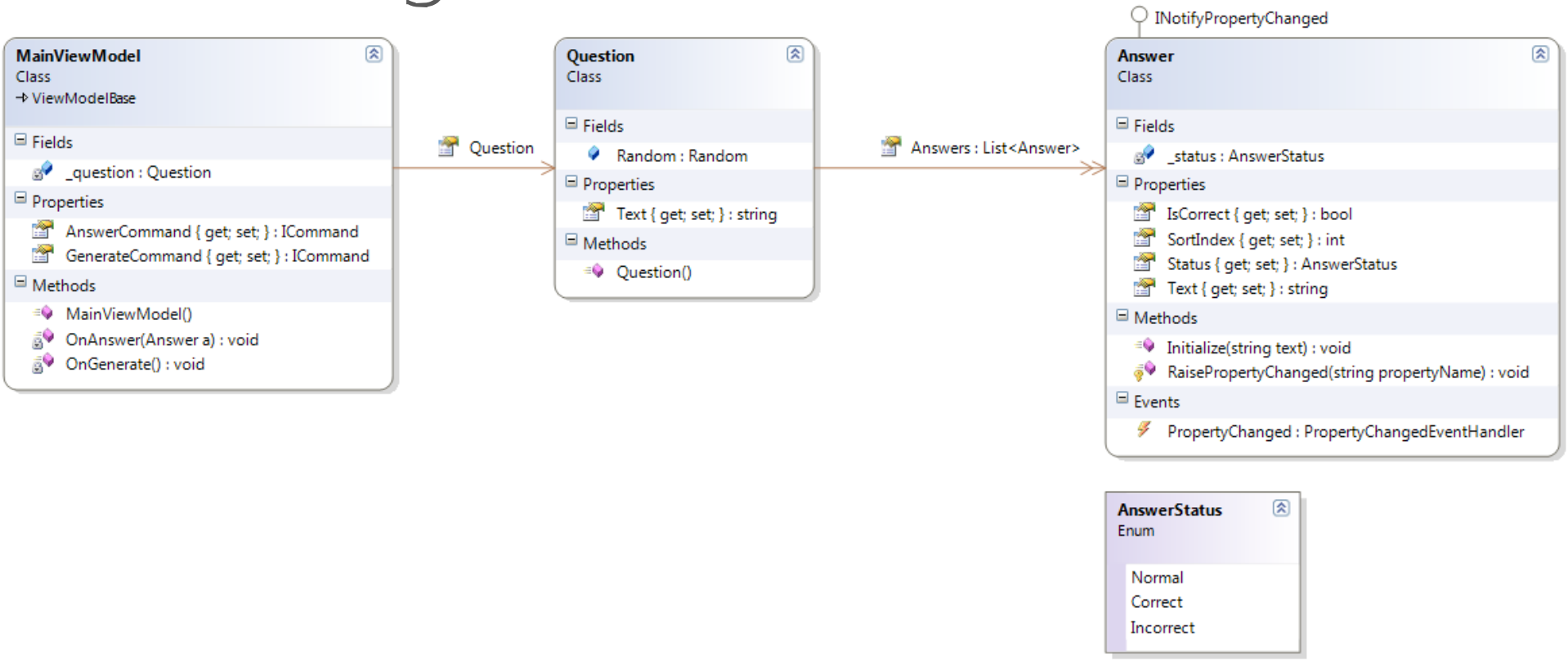
# NuGet Packages

- MVVM Light
  - ViewModelBase
  - RelayCommand
- Silverlight for Windows Phone Toolkit
  - WrapPanel



# Class Diagram




developers
build what's next now

Microsoft

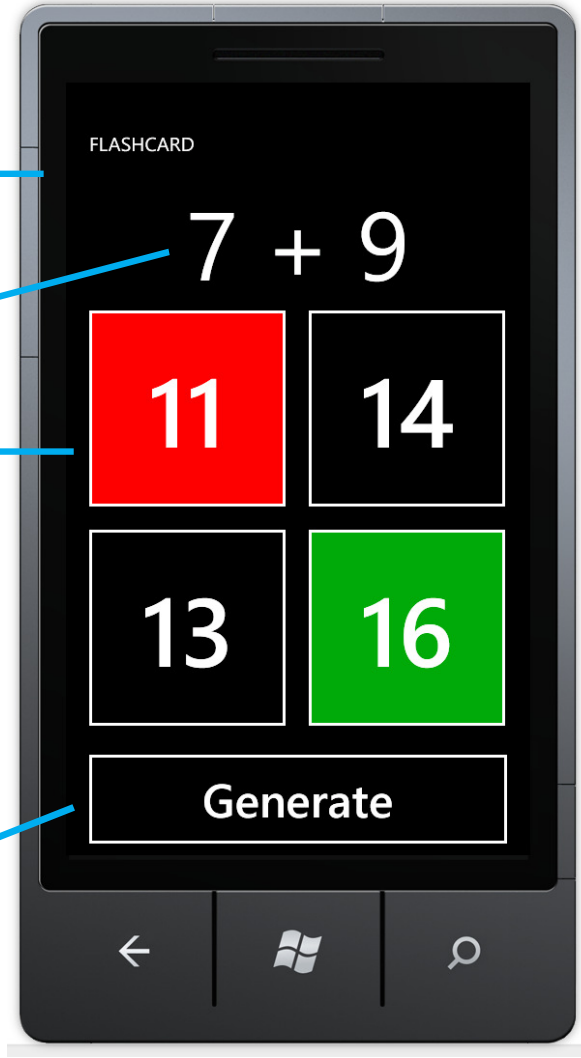
# MainViewModel

```
public class MainViewModel : ViewModelBase {

    private Question _question = new Question();
    public ICommand GenerateCommand { get; set; }
    public ICommand AnswerCommand { get; set; }

    public MainViewModel() {
        this.GenerateCommand = new RelayCommand(() => this.Question = new Question());
        this.AnswerCommand = new RelayCommand<Answer>(OnAnswer);
    }

    public Question Question {
        get { return _question; }
        set {
            if (_question != value) {
                _question = value;
                RaisePropertyChanged("Question");
            }
        }
    }

    private void OnAnswer(Answer a) {
        a.Status = a.IsCorrect ? AnswerStatus.Correct : AnswerStatus.Incorrect;
    }

}
```

Microsoft

# MainPage.xaml

```xaml
<Grid x:Name="ContentPanel“
      DataContext="{StaticResource MainViewModelDataSource}">

    <Grid.RowDefinitions>...</Grid.RowDefinitions>
    <TextBlock HorizontalAlignment="Center"
               VerticalAlignment="Center"
               FontSize="{StaticResource FontSize}"
               Text="{Binding Question.Text}" />
    <ItemsControl Grid.Row="1"
                  ItemsSource="{Binding Question.Answers}">
        <ItemsControl.ItemTemplate>
            <DataTemplate>
                <local:AnswerButton />
            </DataTemplate>
        </ItemsControl.ItemTemplate>
        <ItemsControl.ItemsPanel>
            <ItemsPanelTemplate>
                <toolkit:WrapPanel />
            </ItemsPanelTemplate>
        </ItemsControl.ItemsPanel>
    </ItemsControl>
    <Button Content="Generate"
            Grid.Row="2"
            FontSize="48"
            Command="{Binding GenerateCommand}" />
</Grid>
```




Microsoft

# AnswerButton.xaml

```
<UserControl xmlns=http://schemas.microsoft.com/winfx/2006/xaml/presentation"
             ...
             x:Class="Flashcard.AnswerButton"
             Width="227"
             Height="227">

    <i:Interaction.Behaviors>
        <ReflectionIT_Phone_Behaviors:EnumDataStateBehavior Binding="{Binding Status}" />
    </i:Interaction.Behaviors>

    <Grid x:Name="LayoutRoot"
          Background="Transparent">
        <VisualStateManager.VisualStateGroups>
            ...
        </VisualStateManager.VisualStateGroups>

        <Button x:Name="button"
                FontSize="{StaticResource FontSize}"
                Content="{Binding Text}"
                Command="{Binding AnswerCommand, Source={StaticResource MainViewModelDataSource}}"
                CommandParameter="{Binding}"
                Background="#FD000000">

        </Button>
    </Grid>
</UserControl>
```


Flashcard WP7.sln - Microsoft Expression Blend 4

File Edit View Object Project Tools Window Help

Projects Assets States

Base

StatusStates

Default transition 0,2 s

Normal

Correct

Incorrect

AnswerButton.xaml MainPage.xaml

Nothing selected

Correct state recording is on

68,28%

Objects and Timeline

Correct

[UserControl]

[UserControl]

[EnumDataStateBehavior]

LayouRoot

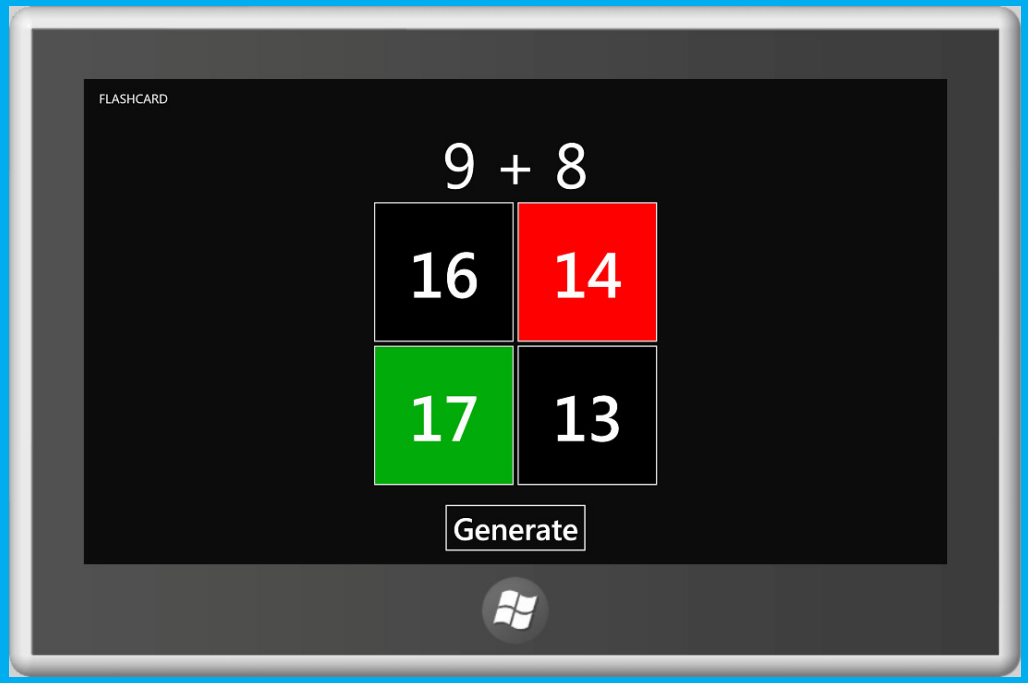
```
18          <VisualStateGroup x:Name="StatusStates">
19              <VisualStateGroup.Transitions>
20                  <VisualTransition GeneratedDuration="0:0:0.2" />
21              </VisualStateGroup.Transitions>
22              <VisualState x:Name="Normal" />
23              <VisualState x:Name="Correct">
24                  <Storyboard>
25                      <ColorAnimation Duration="0"
26                                      To="#FF00AB0A"
27                                      Storyboard.TargetProperty="(Control.Background).(SolidColorBrush.Color)"
28                                      Storyboard.TargetName="button" />
29                  </Storyboard>
30              </VisualState>
31              <VisualState x:Name="Incorrect">
32                  <Storyboard>
33                      <ColorAnimation Duration="0"
34                                      To="Red"
35                                      Storyboard.TargetProperty="(Control.Background).(SolidColorBrush.Color)"
36                                      Storyboard.TargetName="button" />
37                  </Storyboard>
38              </VisualState>
```

# EnumDataStateBehavior

```
public class EnumDataStateBehavior : Behavior<Control> {

    private object StateValue { get; set; }

    protected override void OnAttached() {
        base.OnAttached();
        this.AssociatedObject.Loaded += new RoutedEventHandler(AssociatedObject_Loaded);
    }

    private void AssociatedObject_Loaded(object sender, RoutedEventArgs e) {
        UpdateVisualState();
    }

    private void UpdateVisualState() {
        if (this.AssociatedObject != null && StateValue != null) {
            VisualStateManager.GoToState(this.AssociatedObject, StateValue.ToString(), this.UseTransitions);
        }
    }

    private static void OnBindingPropertyChanged(DependencyObject d, DependencyPropertyChangedEventArgs e) {
        var source = d as EnumDataStateBehavior;
        if (source != null) {
            source.StateValue = e.NewValue;
            source.UpdateVisualState();
        }
    }
```

Windows 8 Developer Preview

# Windows 8 Flashcard Game




Microsoft

# Steps 1

- Create ‘Windows Metro styled - Application’ Project
- Reference MVVM Light V4 for Windows 8
  - http://mvvmlight.codeplex.com/releases/view/74325
- Copy Model & ViewModel classes
- Fix the code problems

# Problem 1 - Namespaces

Flashcard Win8 - MainViewModel.cs

MainViewModel.cs

Flashcard.ViewModels.MainViewModel | GenerateCommand

```
using System;

//using System.Windows.Input;
using Windows.UI.Xaml.Input;

using Flashcard.Models;

using GalaSoft.MvvmLight;
using GalaSoft.MvvmLight.Command;

namespace Flashcard.ViewModels {

    public class MainViewModel : ViewModelBase {

        private Question _question = new Question();
        public ICommand GenerateCommand { get; set; }
        public ICommand AnswerCommand { get; set; }
```

100 %

Flashcard Win8 - Answer.cs

Answer.cs

Flashcard.Models.Answer | IsCorrect

```
using System;

using Windows.UI.Xaml.Data;
//using System.ComponentModel;

namespace Flashcard.Models {

    public class Answer : INotifyPropertyChanged {

        public bool IsCorrect { get; set; }
        public int SortIndex { get; private set; }
```

100 %

developers
build what's next now

Microsoft

# Problem 2 – Code Changes

- List<T>.Sort(Comparison T) is missing
- Solution 1:
  - Create Comparison<T> and Sort<T>(Comparison) extension method
- Solution 2:
  - Rewrite code: Linq OrderBy().ToList()

Flashcard Win8 - Question.cs*

Question.cs*

Flashcard.Models.Question | Question()

```
            // Random order
            //Answers.Sort((a, b) => a.SortIndex.CompareTo(b.SortIndex));
            Answers = Answers.OrderBy(a => a.SortIndex).ToList();
        }
```

100 %

developers
build what's next now

Microsoft

# Steps 2

- Create Views folder
  - Move MainPage.xaml
  - Add AnswerButton.xaml (UserControl)
- Copy App.xaml Resources
- Copy XAML (LayoutRoot) into the UserControls
- Fix the XAML problems

# Problem 3 – App.xaml

- system:Double -> x:Double

```
<Application xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
             xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
             xmlns:vm="using:Flashcard.ViewModels"
             x:Class="Flashcard.App">
    <Application.Resources>
        <!--<vm:MainViewModel x:Key="MainViewModelDataSource" />-->
        <x:Double x:Key="FontSize">100</x:Double>
    </Application.Resources>
</Application>
```

- MainViewModel causes compiler error
  - Register in code behind

```
protected override void OnLaunched(LaunchActivatedEventArgs args) {
    App.Current.Resources["MainViewModelDataSource"] = new MainViewModel();
    Window.Current.Content = new Views.MainPage();
    Window.Current.Activate();
}
```

# Problem 4 – MainPage.xaml

- Change `<toolkit:WrapPanel />` into `<WrapGrid />`
- XML Namespaces
  - Change `clr-namespace:` into `using:`

```
xmlns:local="using:Flashcard.Views"
```

- phoneTextNormalStyle
  - Add ThemeResources.xaml from WP7 SDK

```
<ResourceDictionary Source="Resources/ThemeResources.xaml" />
```

    - Change Segoe **WP** to Segoe **UI**

# Problem 5 – AnswerButton.xaml

- Button: Horizontal & Vertical Stretch
- Behavior are not (yet) supported
  - Replace EnumDataStateBehavior with an Attached Property

Flashcard Win8 - AnswerButton.xaml

AnswerButton.xaml

```
<UserControl x:Class="Flashcard.Views.AnswerButton"
             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:helpers="using:ReflectionIT.Windows8.Helpers"
             helpers:GoToStateServices.Binding="{Binding Status}"
             helpers:GoToStateServices.BindingType="Flashcard.Models.AnswerStatus"
             mc:Ignorable="d"
             Height="227"
             Width="227">

    <!--<i:Interaction.Behaviors>
        <ReflectionIT_Phone_Behaviors:EnumDataStateBehavior Binding="{Binding Status}" />
    </i:Interaction.Behaviors>-->

    <Grid x:Name="LayoutRoot"
          Background="Transparent">
```

100 %

XAML Design

develop
build wha

Microsoft

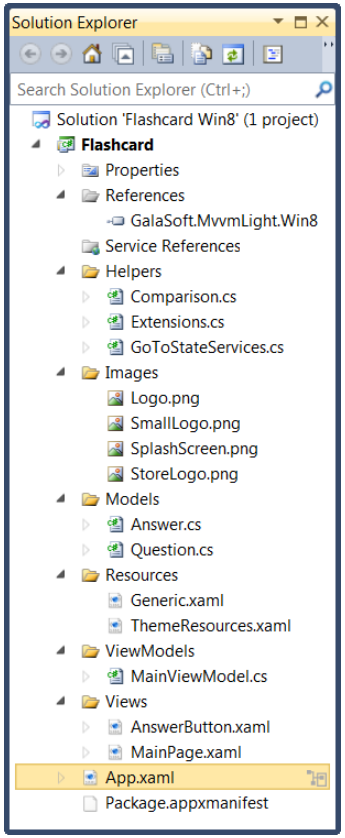
# GoToStateServices

```
public class GoToStateServices {

    #region Binding Attached Property

    /// <summary>
    /// Identifies the Binding attachted property. This enables animation, styling, binding, etc...
    /// </summary>
    public static readonly DependencyProperty BindingProperty =
        DependencyProperty.RegisterAttached("Binding",
                                            "Object",
                                            typeof(GoToStateServices).FullName,
                                            new PropertyMetadata(null, OnBindingChanged));

    /// <summary>
    /// Binding changed handler.
    /// </summary>
    /// <param name="d">FrameworkElement that changed its Binding attached property.</param>
    /// <param name="e">DependencyPropertyChangedEventArgs with the new and old value.</param>
    private static void OnBindingChanged(DependencyObject d, DependencyPropertyChangedEventArgs e) {
        var source = d as Control;
        if (source != null) {
            var value = e.NewValue;
            value = Enum.ToObject(Type.GetType(GoToStateServices.GetBindingType(source).ToString()), value);
            VisualStateManager.GoToState(source, value.ToString(), true);
        }
    }
```

# Problem 6 – Button MouseOver state

- Copy Generic.xaml from Sample app

```
<Application xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
             xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
             xmlns:vm="using:Flashcard.ViewModels"
             x:Class="Flashcard.App">
    <Application.Resources>

        <ResourceDictionary>
            <ResourceDictionary.MergedDictionaries>
                <ResourceDictionary Source="Resources/ThemeResources.xaml" />

<!--http://code.msdn.microsoft.com/windowsapps/Light-and-Dark-Theme-Xaml-eca02f2b -->
                <ResourceDictionary Source="Resources/Generic.xaml" />
            </ResourceDictionary.MergedDictionaries>

            <!--<vm:MainViewModel x:Key="MainViewModelDataSource" />-->
            <x:Double x:Key="FontSize">100</x:Double>

        </ResourceDictionary>

    </Application.Resources>

</Application>
```




Microsoft

# Problem 6 – Button MouseOver state

- Remove StoryBoard from MouseOver state

Flashcard Win8 - Generic.xaml

Generic.xaml

```
<!-- Default style for Windows.UI.Xaml.Controls.Button -->
<Style TargetType="Button">
    <Setter Property="Background" Value="{StaticResource ButtonBackgroundBrush}"/>
    <Setter Property="MinWidth" Value="94"/>
    <Setter Property="MinHeight" Value="36"/>
    <Setter Property="Foreground" Value="{StaticResource ControlForegroundBrush}"/>
    <Setter Property="FontFamily" Value="{StaticResource ContentFontFamily}"/>
    <Setter Property="FontWeight" Value="SemiBold"/>
    <Setter Property="FontSize" Value="{StaticResource ContentFontSize}"/>
    <Setter Property="Padding" Value="12,4,12,5"/>
    <Setter Property="HorizontalAlignment" Value="Left"/>
    <Setter Property="VerticalAlignment" Value="Center"/>
    <Setter Property="BorderThickness" Value="{StaticResource ControlBorderThickness}"/>
    <Setter Property="BorderBrush" Value="{StaticResource ControlNormalStrokeBrush}"/>
    <Setter Property="Template">
        <Setter.Value>
            <ControlTemplate TargetType="Button">
                <Grid>
                    <VisualStateManager.VisualStateGroups>
                        <VisualStateGroup x:Name="CommonStates">
                            <VisualState x:Name="Normal"/>
                            <VisualState x:Name="MouseOver">
                                <!--<Storyboard>
                                    <DoubleAnimation Duration="0" To="1" Storyboard.TargetProperty
                                    <DoubleAnimation Duration="0" To="0" Storyboard.TargetProperty
                                </Storyboard>-->
                            </VisualState>
                            <VisualState x:Name="Pressed">
```

100 %

XAML | Design

# Tips & Tricks

Microsoft

# Tips & Tricks

- ObservableCollection
- Animation Library
- More code changes
- Tombstoning
- Dependency & Attached Properties
- Artwork
- Localization
- Screen Layout
- Bugs

# ObservableCollection

- DataBinding BUG caused by incorrect INotifyPropertyChanged

- Workarounds:
  - http://blogs.msdn.com/b/avip/archive/2011/09/18/windows-8-development-tidbits-observablecollection-doesn-t-work.aspx
  - http://code.msdn.microsoft.com/windowsapps/Data-Binding-7b1d67b5
  - http://www.scottlogic.co.uk/blog/colin/2011/10/using-observablecollection-with-winrt-via-a-little-shim/
  - http://blogs.u2u.be/diederik/post/2012/01/03/Hello-ObservableVector-goodbye-ObservableCollection.aspx

# Animation Library

- Alternative to StoryBoards and VisualStates

```
<WrapGrid >
    <WrapGrid.ChildrenTransitions>
        <TransitionCollection>
            <EntranceThemeTransition FromHorizontalOffset="100" />
        </TransitionCollection>
    </WrapGrid.ChildrenTransitions>
</WrapGrid>
```

# More code changes

- Thickness constructor parameters  vs ThicknessHelper
- HttpUtility.UrlEncode() vs Uri.EscapeDataString()
- MessageBox.Show() vs MessageDialog.ShowAsync()

- Documented code changes
  - http://msdn.microsoft.com/en-us/library/windows/apps/br230302(v=vs.85).aspx

# MessageBox vs MessageDialog

- WP7 MessageBox.Show()

```
private void Button1_Click(object sender, RoutedEventArgs e) {
    var result = MessageBox.Show("This is MessageBox", "Title", MessageBoxButton.OKCancel);
    if (result == MessageBoxResult.OK) {
        // do something
    }
    Button1.Content = result.ToString();
}
```

- Windows8 MessageDialog.ShowAsync()

```
private async void Button1_Click(object sender, RoutedEventArgs e) {
    MessageDialog md = new MessageDialog("This is a MessageDialog", "Title");
    bool? result = null;
    md.Commands.Add(new UICommand("OK", new UICommandInvokedHandler((cmd) => result = true)));
    md.Commands.Add(new UICommand("Cancel", new UICommandInvokedHandler((cmd) => result = false)));

    await md.ShowAsync();
    if (result == true) {
        // do something
    }
    Button1.Content = result.ToString();
}
```

Microsoft

# My MessageBox “wrapper” class

```
class MessageBox {

    public static async Task<MessageBoxResult> ShowAsync(string messageBoxText, string caption, MessageBoxButton button) {

        MessageDialog md = new MessageDialog(messageBoxText, caption);
        MessageBoxResult result = MessageBoxResult.None;
        md.Commands.Add(new UICommand("OK", new UICommandInvokedHandler((cmd) => result = MessageBoxResult.OK)));
        if (button == MessageBoxButton.OKCancel) {
            md.Commands.Add(new UICommand("Cancel", new UICommandInvokedHandler((cmd) => result = MessageBoxResult.Cancel)));
        }
        var op = await md.ShowAsync();
        return result;
    }

    public static async Task<MessageBoxResult> ShowAsync(string messageBoxText) {
        return await MessageBox.ShowAsync(messageBoxText, null, MessageBoxButton.OK);
    }
}

public enum MessageBoxButton {
    OK = 0, OKCancel = 1,
}

public enum MessageBoxResult {
    None = 0, OK = 1, Cancel = 2,
}
```

Microsoft

# MessageBox vs MessageDialog

- WP7 MessageBox.Show()

```
private void Button1_Click(object sender, RoutedEventArgs e) {
    var result = MessageBox.Show("This is MessageBox", "Title", MessageBoxButton.OKCancel);
    if (result == MessageBoxResult.OK) {
        // do something
    }
    Button1.Content = result.ToString();
}
```

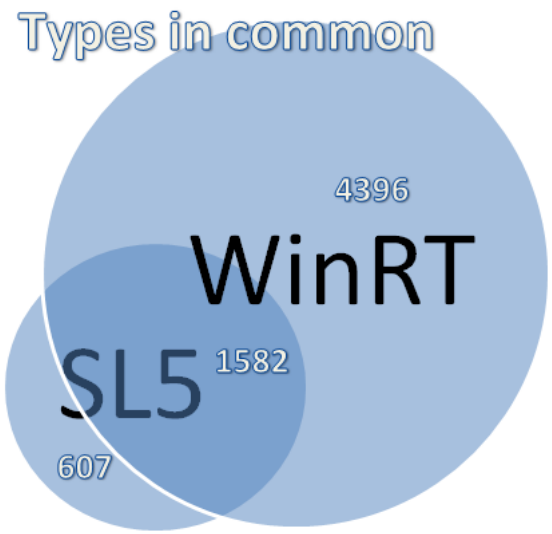
- Windows 8 MessageBox.ShowAsync()

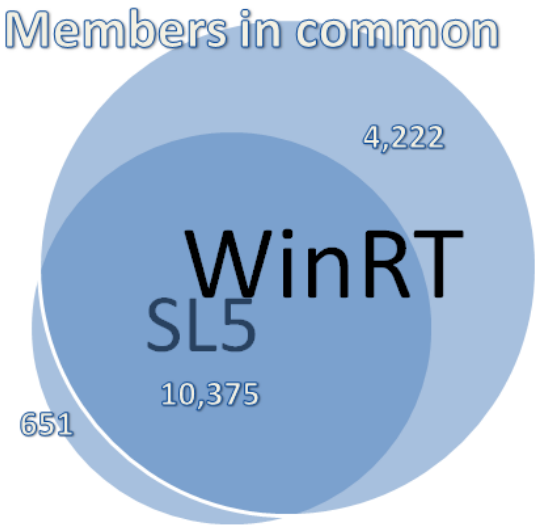
```
private async void Button1_Click(object sender, RoutedEventArgs e) {
    var result = await MessageBox.ShowAsync("This is MessageBox", "Title", MessageBoxButton.OKCancel);
    if (result == MessageBoxResult.OK) {
        // do something
    }
    Button1.Content = result.ToString();
}
```

Microsoft

# More code changes

- Project Genome
  - http://programmerpayback.com/2011/11/17/the-winrt-genome-project/




developers
build what's next now

Microsoft

# Tombstoning

- Windows Phone
  - IsolatedStorageSettings.ApplicationSettings
- Windows 8
  - SupensionManager class from Sample apps
  - Subscribe on Suspending, Resuming and Exiting events of the App object

# Windows Phone – App.xaml.cs

```
private void Application_Launching(object sender, LaunchingEventArgs e) { RestoreSettings(); }

private void Application_Activated(object sender, ActivatedEventArgs e) {
    if (!e.IsApplicationInstancePreserved) RestoreSettings();
}

private void Application_Deactivated(object sender, DeactivatedEventArgs e) { SaveSettings(); }

private void Application_Closing(object sender, ClosingEventArgs e) { SaveSettings(); }

private void RestoreSettings() {
    object value;
    if (IsolatedStorageSettings.ApplicationSettings.TryGetValue("Question", out value)) {
        this.ViewModel.Question = value as Models.Question;
    }
}

private void SaveSettings() {
    IsolatedStorageSettings.ApplicationSettings["Question"] = this.ViewModel.Question;
}

private MainViewModel ViewModel {
    get { return this.Resources["MainViewModelDataSource"] as MainViewModel; }
}
```

Microsoft

# Windows 8 – App.xaml.cs

```
protected async override void OnLaunched(LaunchActivatedEventArgs args) {
    App.Current.Resources["MainViewModelDataSource"] = new MainViewModel();

    if (args.PreviousExecutionState == ApplicationExecutionState.Terminated) {
        await this.RestoreSettings();
    }

    this.Suspending += App_Suspending;
    this.Resuming += App_Resuming;
    this.Exiting += App_Exiting;

    Window.Current.Content = new Views.MainPage();
    Window.Current.Activate();
}

private async void App_Resuming(object sender, object e) {
    await RestoreSettings();
}

private async Task RestoreSettings() {
    await SuspensionManager.RestoreAsync();
    object value;
    if (SuspensionManager.SessionState.TryGetValue("Question", out value)) {
        this.ViewModel.Question = value as Models.Question;
    }
}
```

Microsoft

# Windows 8 – App.xaml.cs

```
private async void App_Exiting(object sender, object e) {
    await SaveSettings();
}

private async void App_Suspending(object sender, Windows.ApplicationModel.SuspendingEventArgs e) {
    await SaveSettings();
    e.SuspendingOperation.GetDeferral().Complete();
}

private async Task SaveSettings() {
    SuspensionManager.SessionState["Question"] = this.ViewModel.Question;
    await SuspensionManager.SaveAsync();
}


private MainViewModel ViewModel {
    get { return this.Resources["MainViewModelDataSource"] as MainViewModel; }
}
```

Microsoft

# Dependency & Attached Properties

- Dependency & Attached Properties are not typed (Bug?)
  - Workaround: always use Object
- Register() parameter changes

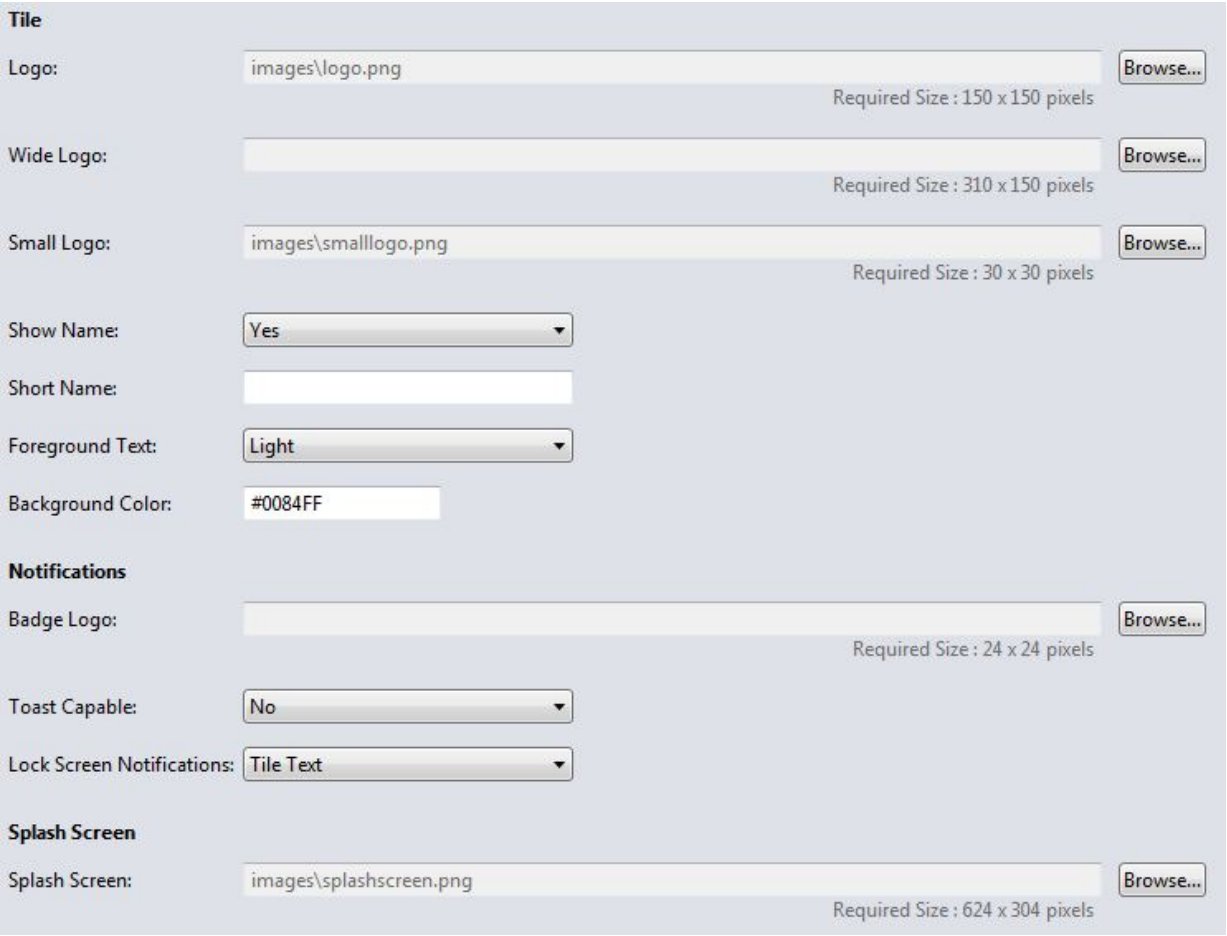
```
// Windows 8
public static readonly DependencyProperty TextProperty =
    DependencyProperty.Register("Text",
                                "Object",                          // Always use "Object" (String)
                                typeof(MyUserControl).FullName, // String not Type
                                new PropertyMetadata(string.Empty, OnTextPropertyChanged));
```

- Download propdw8 and propaw8
  - http://www.reflectionit.nl/downloads/Windows8Snippets.zip

# Artwork

- Formats
  - PNG
  - JPG
- StoreLogo
  - 56x56
- Expresion Design
  - Slices

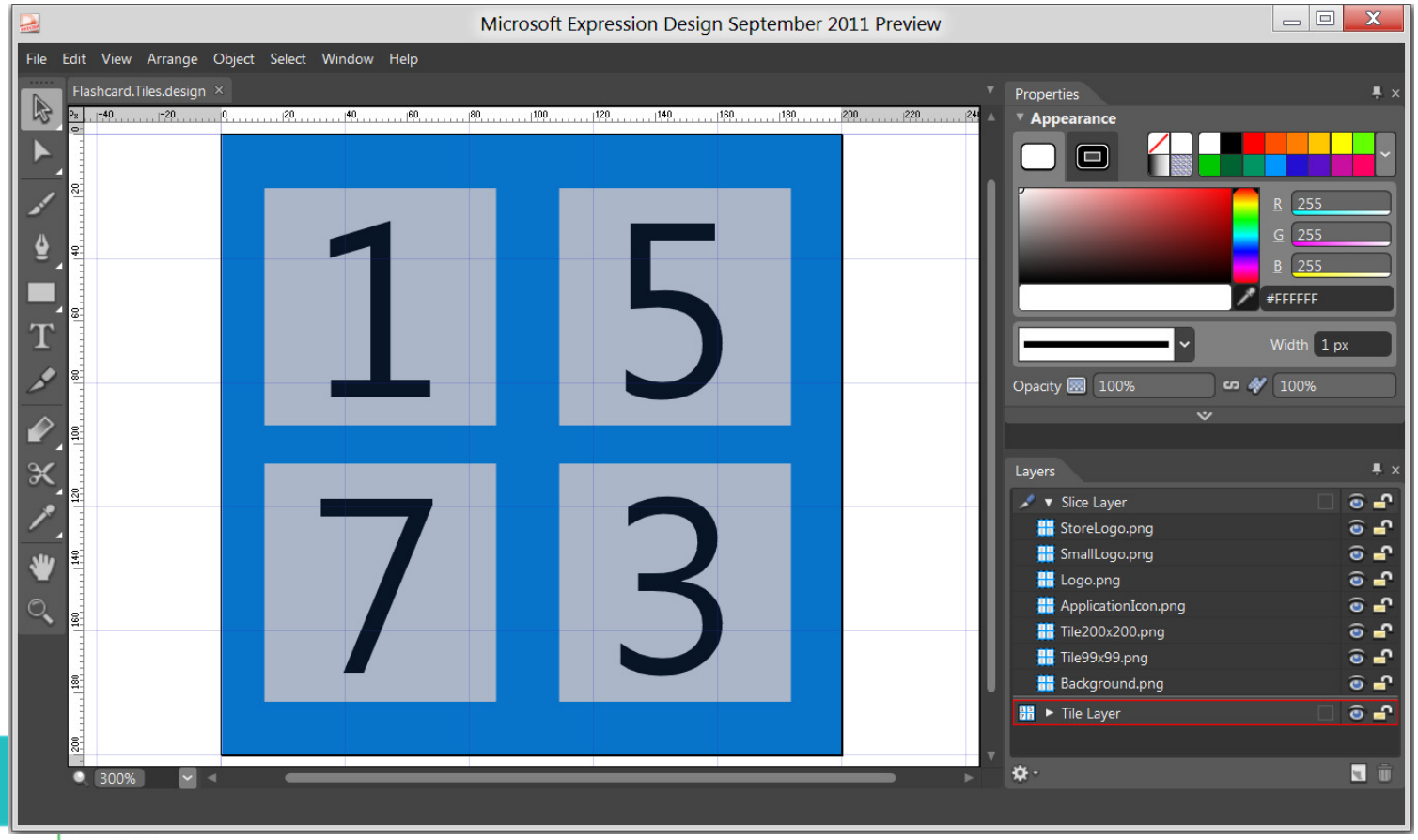
package.appxmanifest




developers
build what's next now

Microsoft

# Artwork

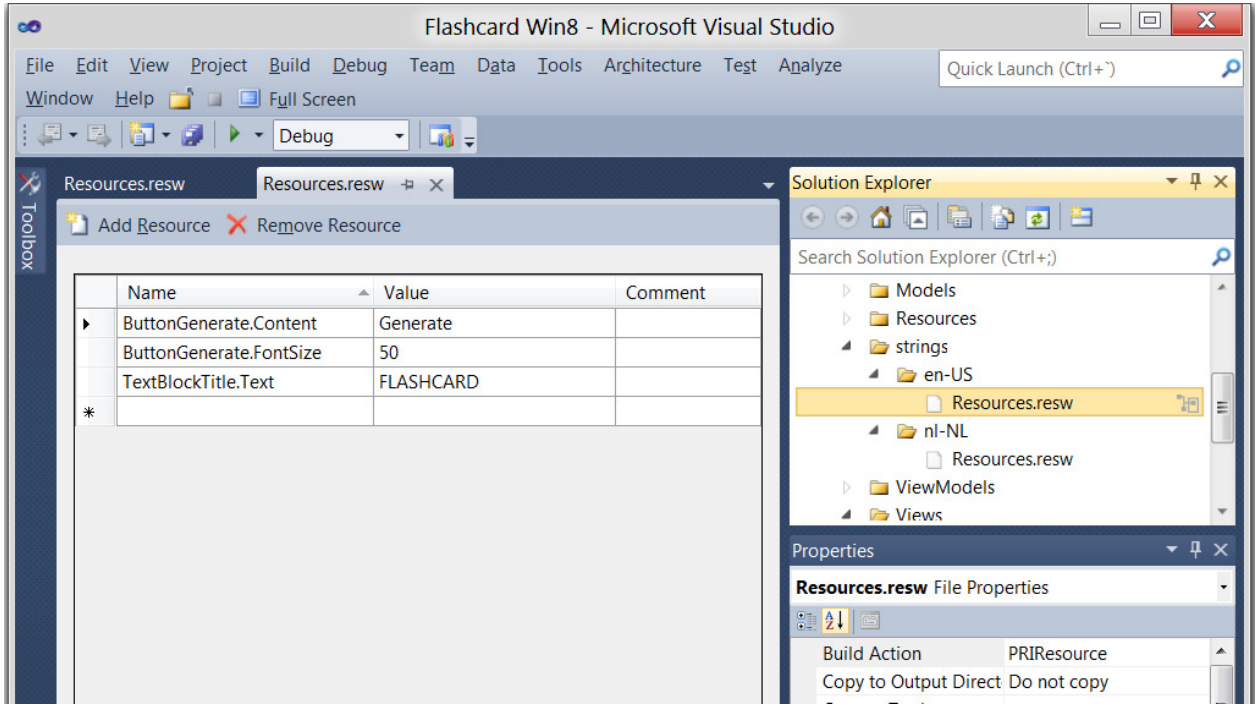

Microsoft Expression Design September 2011 Preview
File Edit View Arrange Object Select Window Help
Flashcard.Tiles.design
1 5
7 3
300%
Properties
Appearance
R 255
G 255
B 255
#FFFFFF
Width 1 px
Opacity 100%
100%
Layers
Slice Layer
StoreLogo.png
SmallLogo.png
Logo.png
ApplicationIcon.png
Tile200x200.png
Tile99x99.png
Background.png
Tile Layer


Start
amsterdam
1 5
7 3
4 7
3 5
1
BLOO
Desktop
//build/
windows

# Localization

- x:UID

```
<TextBlock x:Name="ApplicationTitle"
           Style="{StaticResource PhoneTextNormalStyle}"
           x:Uid="TextBlockTitle"
           Text="FLASHCARD" />
```




developers
build what's next now

Microsoft

# Localization

Flashcard Win8 - Resources.resw

Resources.resw

Add Resource  Remove Resource

| Name | Value |
|---|---|
| ButtonGenerate.Content | Genereren |
| TextBlockTitle.Text | CIJFERSPEL |

Language

Control Panel ▸ Clock, Language, and Region ▸ Language

Search Control Panel

Control Panel Home

Advanced settings

Change your language preferences

Add languages you want to use to this list. The language at the top of your list is your primary language (the one you want to see and use most often).

Add a language  Remove  Move up  Move down

| | | |
|---|---|---|
| Nederlands (Nederland) | Keyboard layout: United States-International | Options... |
| English (United States) | Windows Display Language: On<br>Keyboard layout: US | Options... |

See also

Fonts

Formats

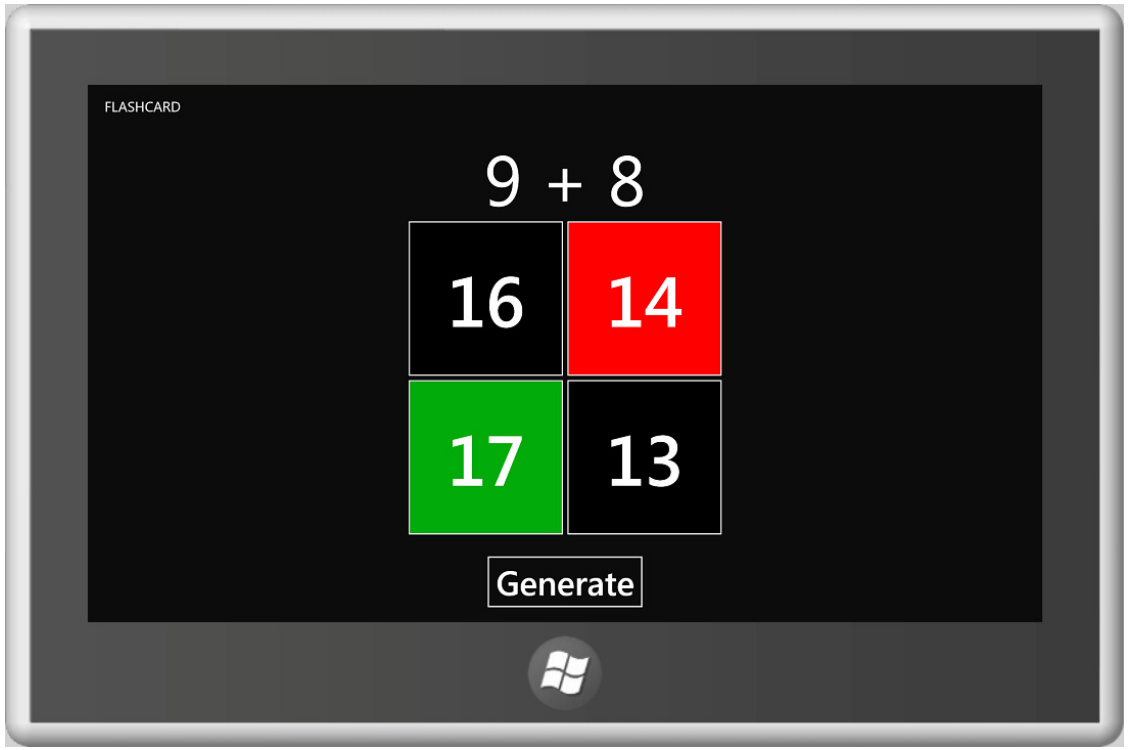
Location

Administrative

developers
build what's next now

# Screen Layout

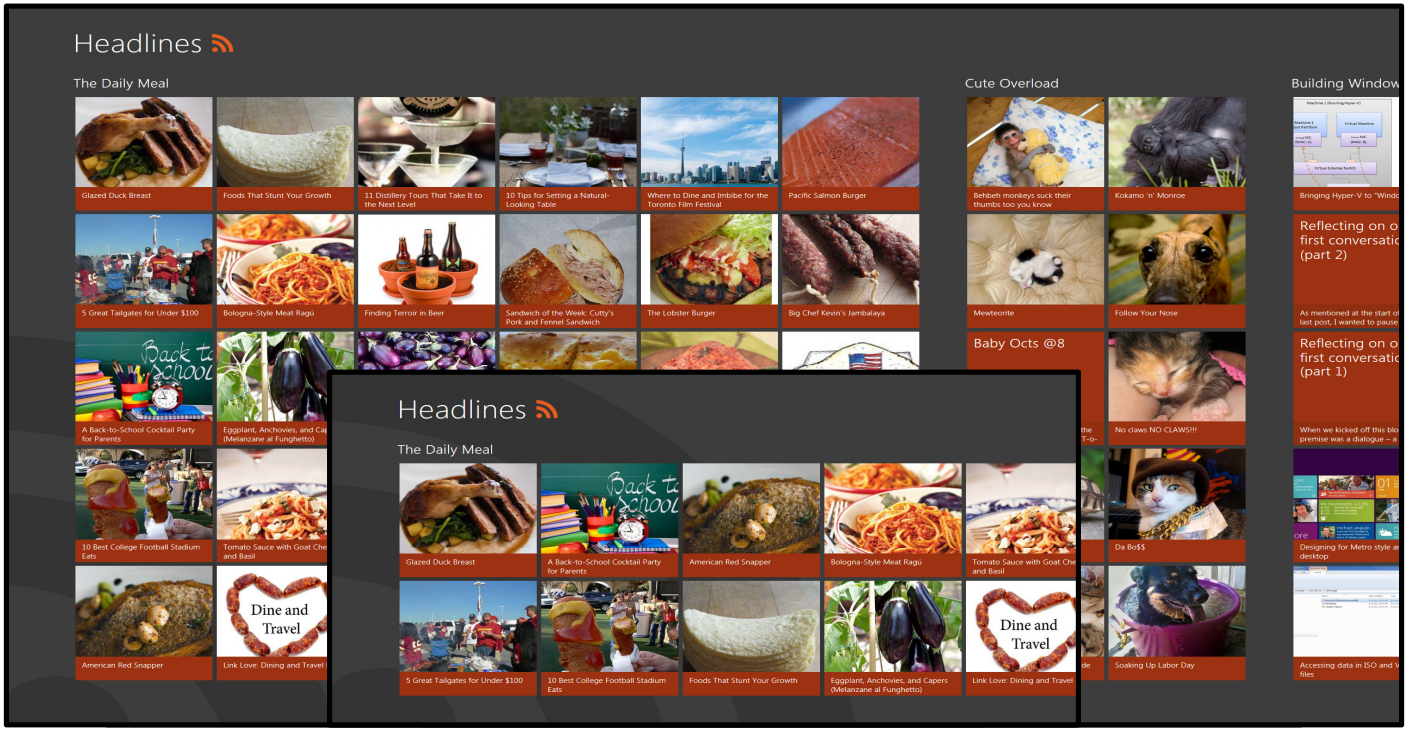
- Sizes
- Orientation
- Layout




developers
build what's next now

Microsoft

# Screen sizes

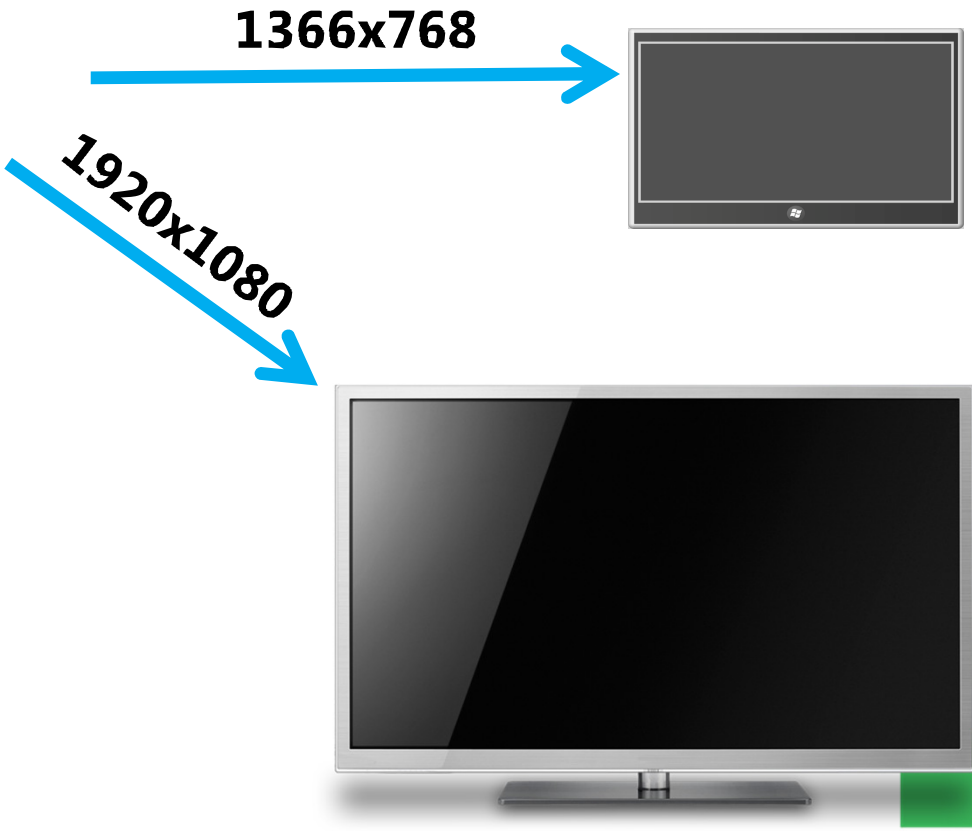



developers
build what's next now

Microsoft

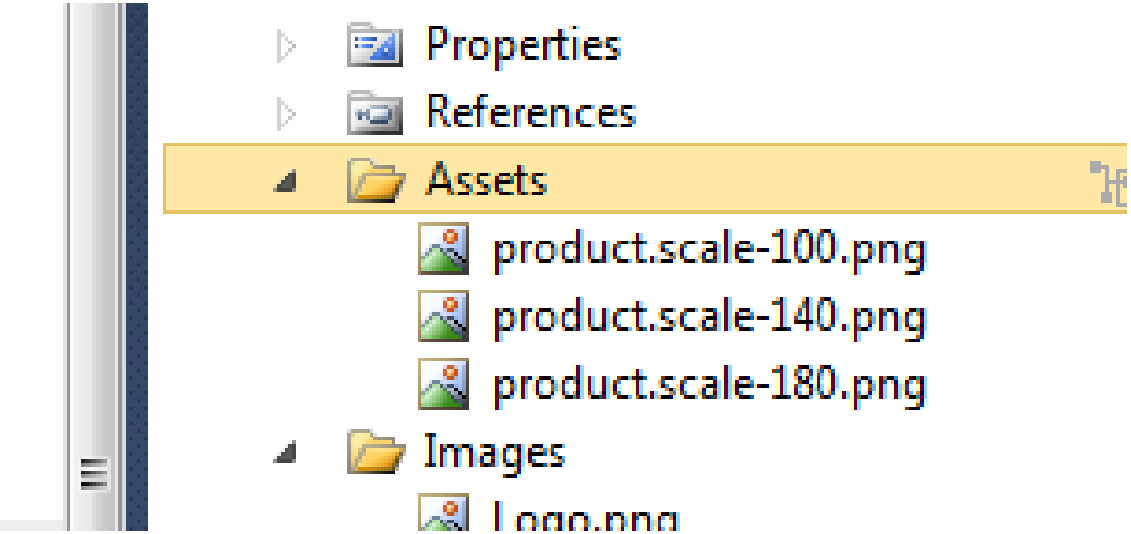
# Detecting Screen Resolution

- Current Resolution:
  - Window.Current.Bounds
- Event:
  - Window.Current.SizeChanged



# Image Source Example

```
<Grid x:Name="LayoutRoot" Background="#FF0C0C0C">
    <Image Source="Assets/product.png" />
</Grid>
```



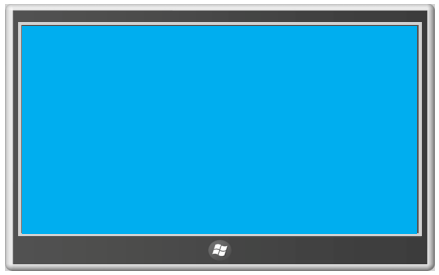

developers
build what's next now

Microsoft

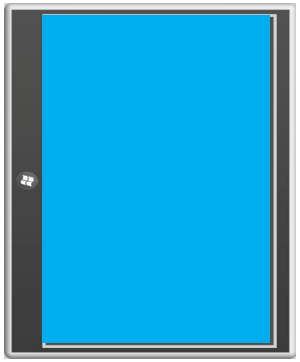
# Detecting Orientation

- Current (Namespace: Windows.Graphics.Display)
  - DisplayProperties.CurrentOrientation
- Event:
  - DisplayProperties.OrientationChanged

Landscape



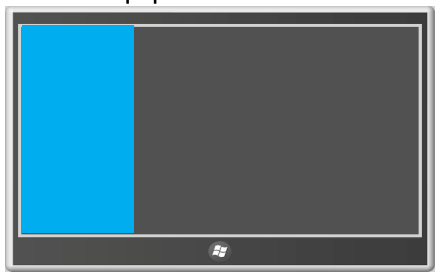
Portrait



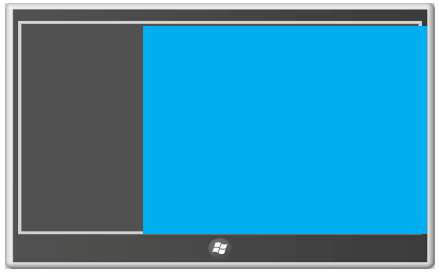
# Detecting Layout

- Current (Namespace: Windows.UI.ViewManagement)
  - ApplicationLayout.Value
- Event:
  - ApplicationLayout.GetForCurrentView().LayoutChanged

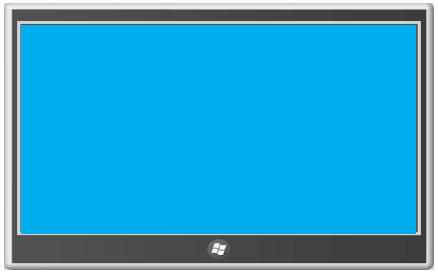
Snapped (~25%)



Filled (~75%)



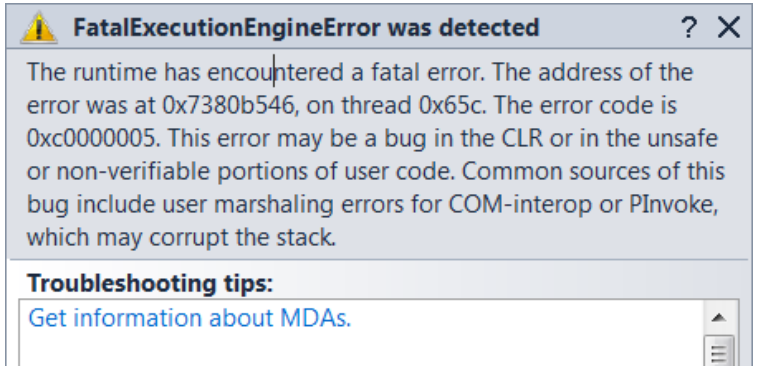
Full Screen



# Screen Layout - VisualStateManager

```
private void Page_LayoutChanged(object sender, ApplicationLayoutChangedEventArgs e) {
    SetCurrentViewState(this);
}

private void Page_OrientationChanged(object sender) {
    SetCurrentViewState(this);
}

private void SetCurrentViewState(Control viewStateAwareControl) {
    VisualStateManager.GoToState(viewStateAwareControl, this.GetViewState(), false);
}

private String GetViewState() {
    var orientation = DisplayProperties.CurrentOrientation;
    if (orientation == DisplayOrientations.Portrait ||
        orientation == DisplayOrientations.PortraitFlipped) return "Portrait";
    var layout = ApplicationLayout.Value;
    if (layout == ApplicationLayoutState.Filled) return "Fill";
    if (layout == ApplicationLayoutState.Snapped) return "Snapped";
    return "Full";
}
```

# Bugs




- TextBox Font rendering
- ElementName Binding
- Events in user controls
- FatalExecutionEngineError in Animations
  - http://social.msdn.microsoft.com/Forums/en-US/winappswithcsharp/thread/8539a645-e84f-4669-87a7-20e3731322a7
- ?

developers
build what's next now

Microsoft

# Simple, clean, & impactful text here.

developers
build what's next now

Microsoft


Questions

# Contact


fons.sonnemans@reflectionit.nl

http://www.twitter.com/fonssonnemans

http://www.linkedin.com/in/fonssonnemans


developers
build what's next now

Microsoft

Microsoft®