

# Avalonia UI

A SHORT INTRODUCTION



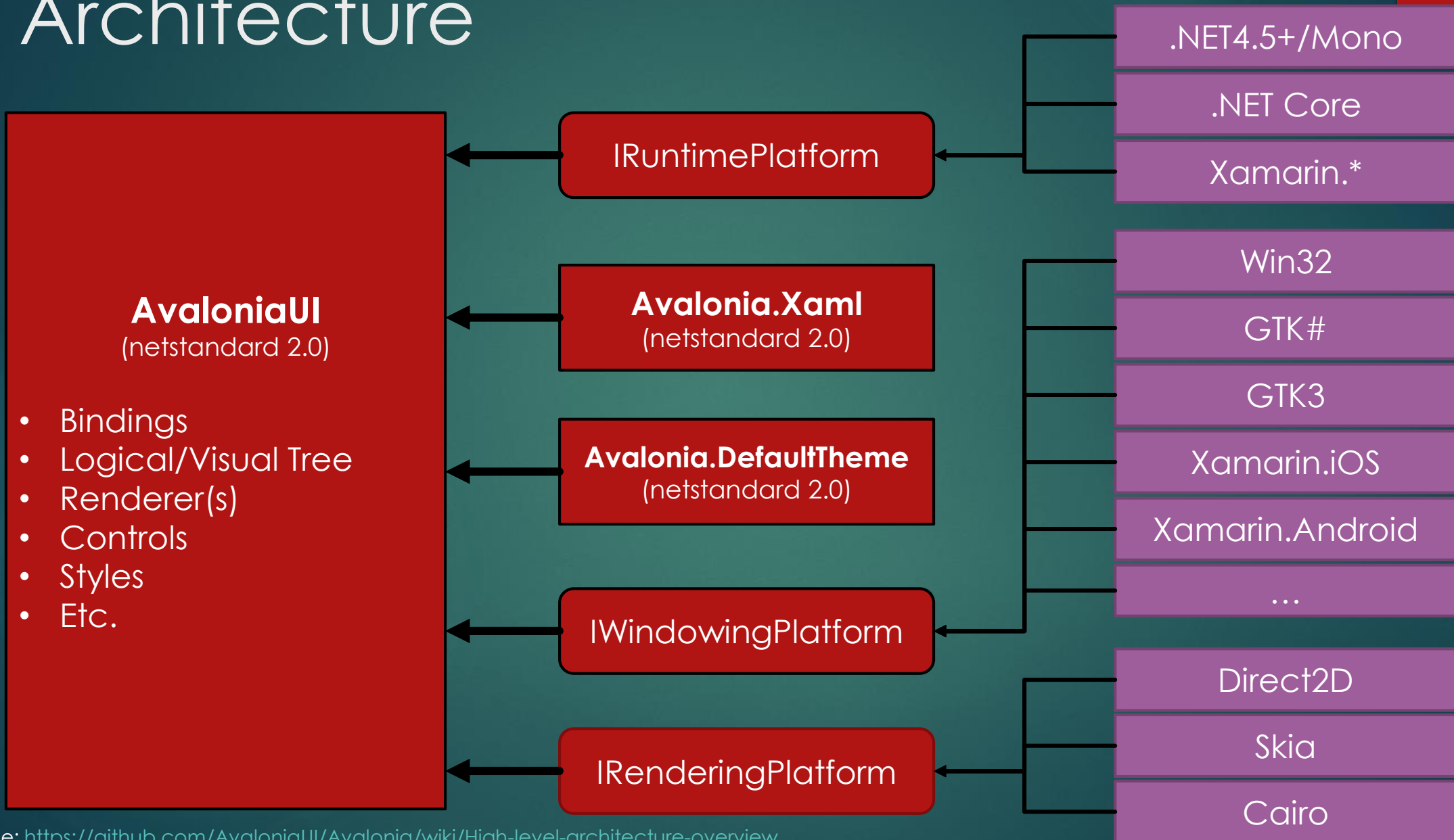
Source: <http://avaloniaui.net>

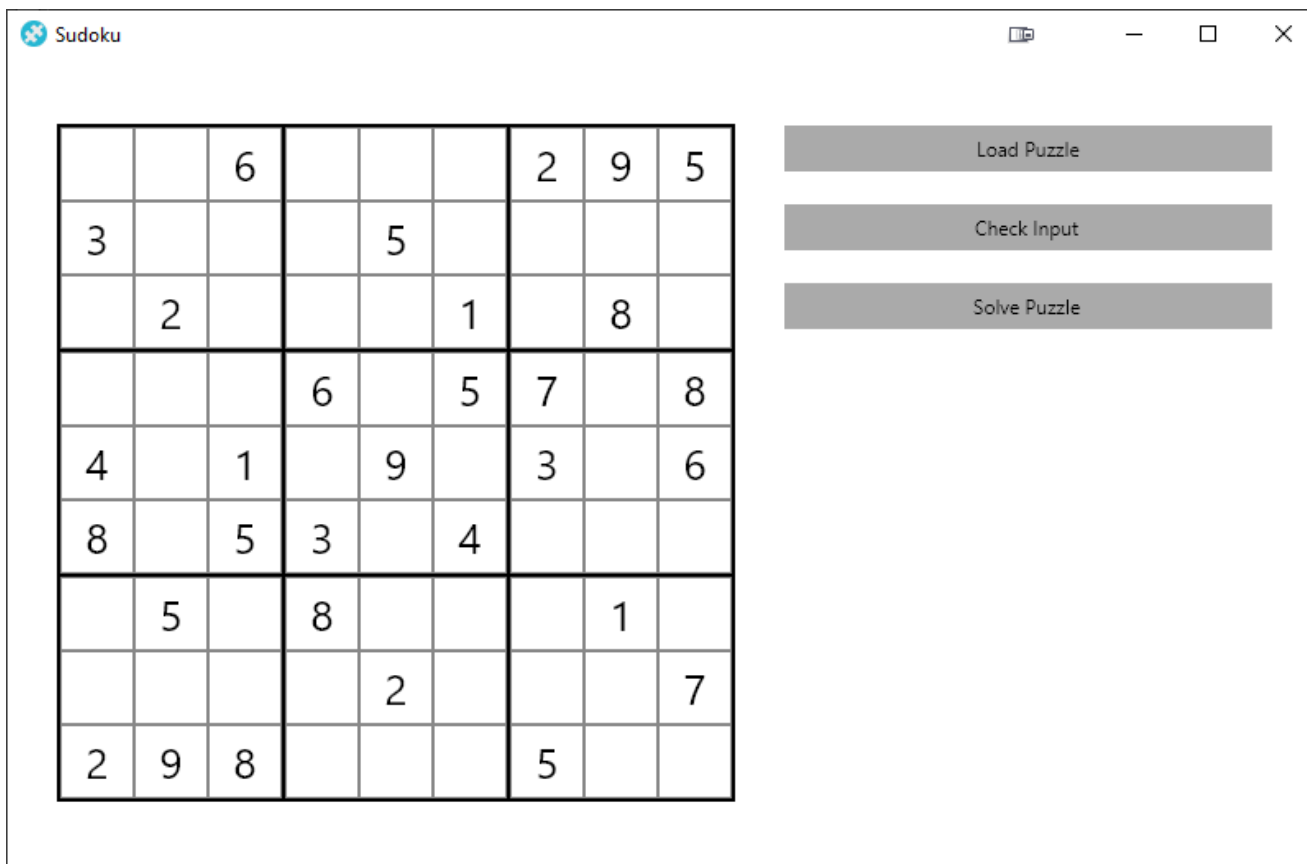
# What is Avalonia UI



- ▶ Cross Platform UI Framework for .NET Framework, .NET Core and Mono (Windows, Linux, OSX, Android and iOS)
- ▶ XAML
- ▶ MVVM Binding through Reactive UI
- ▶ Open Source (MIT) ☺

# Architecture

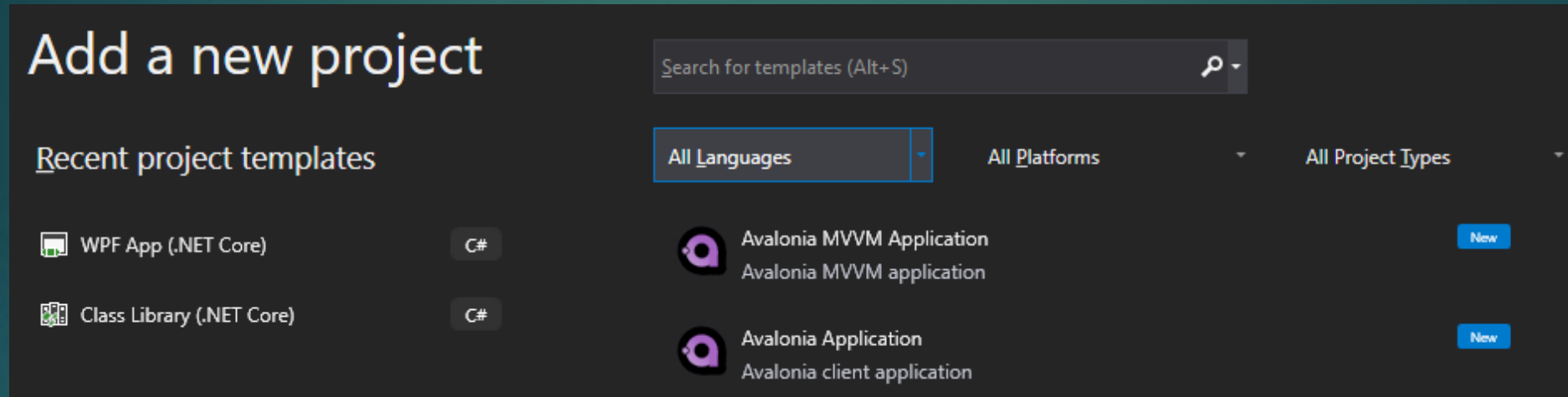




# Implementing a Sudoku App

# Creating the project

## ► Option 1: Visual Studio Extensions

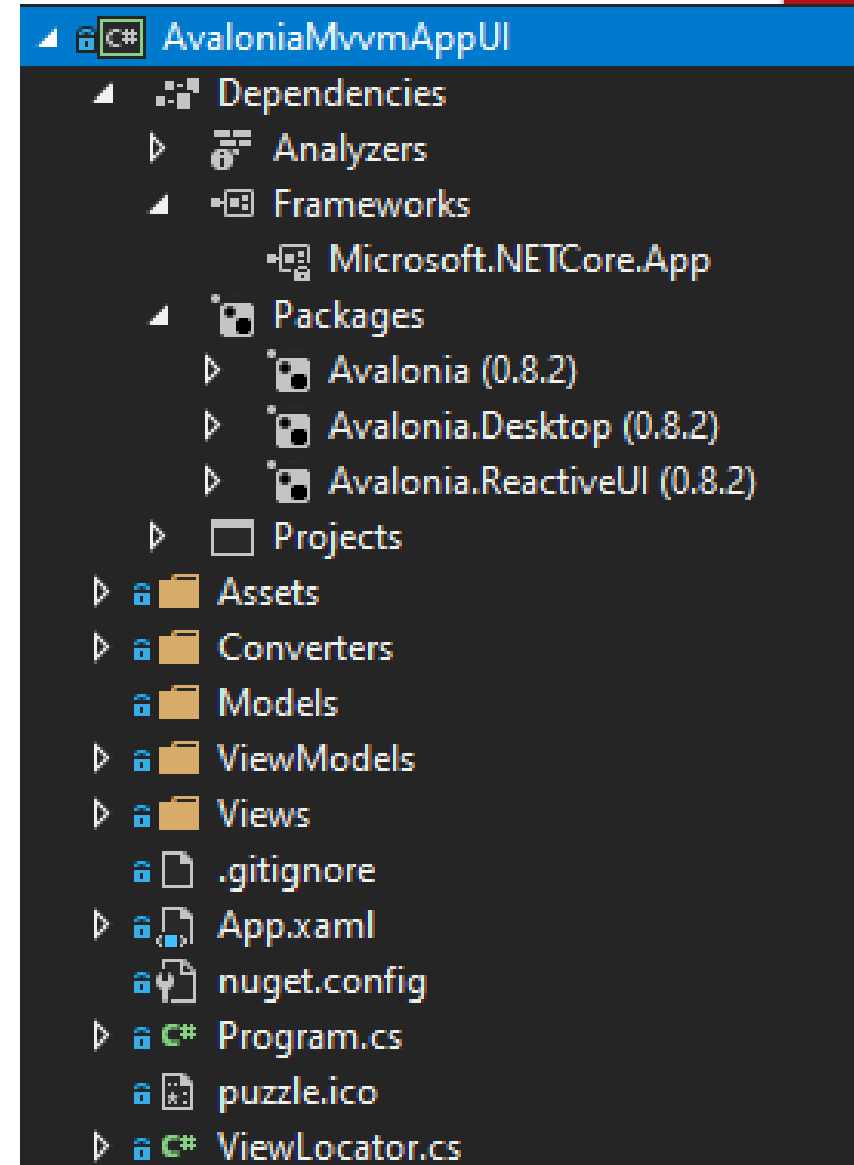


## ► Option 2: .NET Core Template (available on Github)

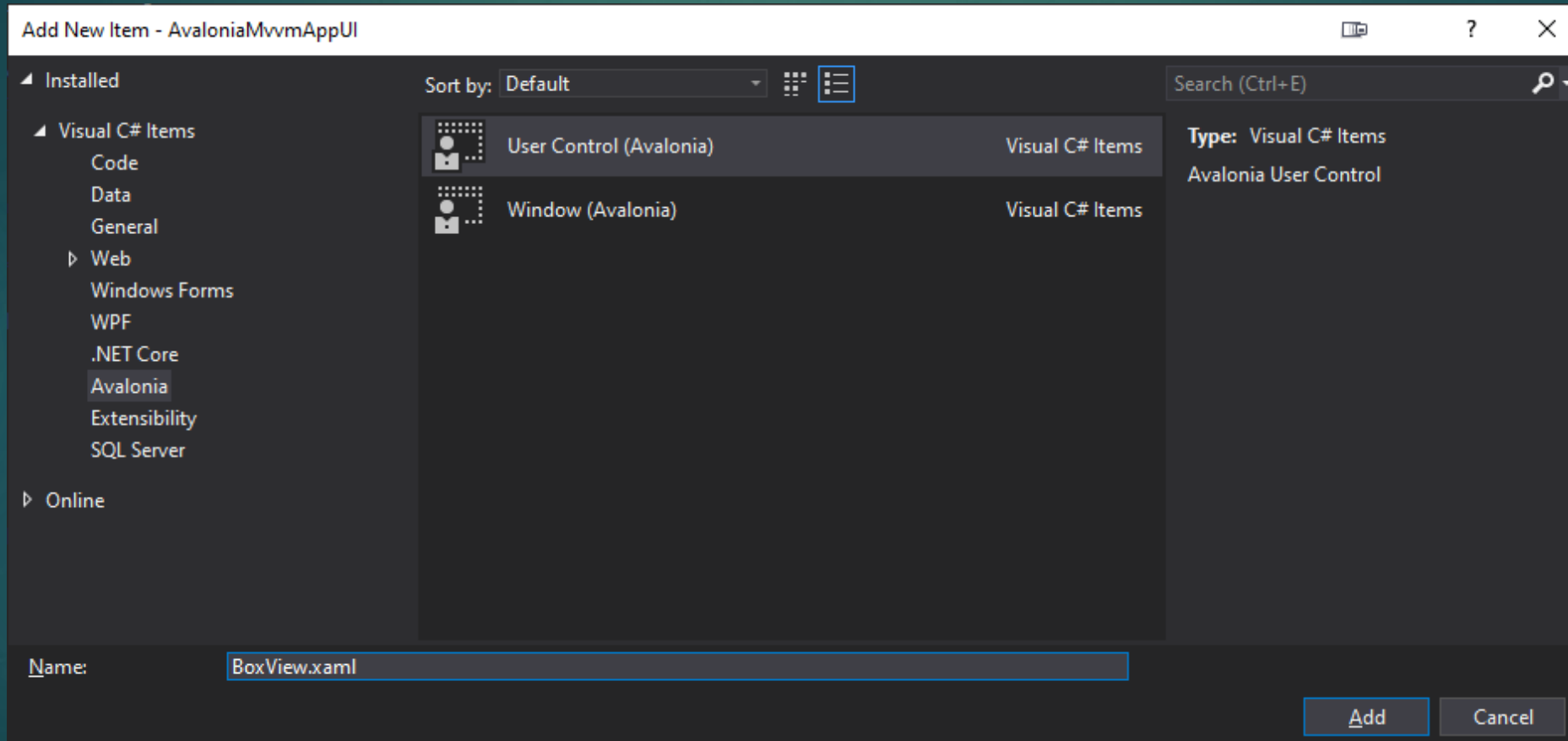
```
> dotnet new avalonia.mvvm -o AvaloniaMvvmAppUI
```

# Project Structure

- ▶ References to
  - ▶ Avalonia
  - ▶ Avalonia.Desktop
  - ▶ Avalonia.ReactiveUI
- ▶ Folders for
  - ▶ Models
  - ▶ ViewModels
  - ▶ Views
- ▶ ViewLocator

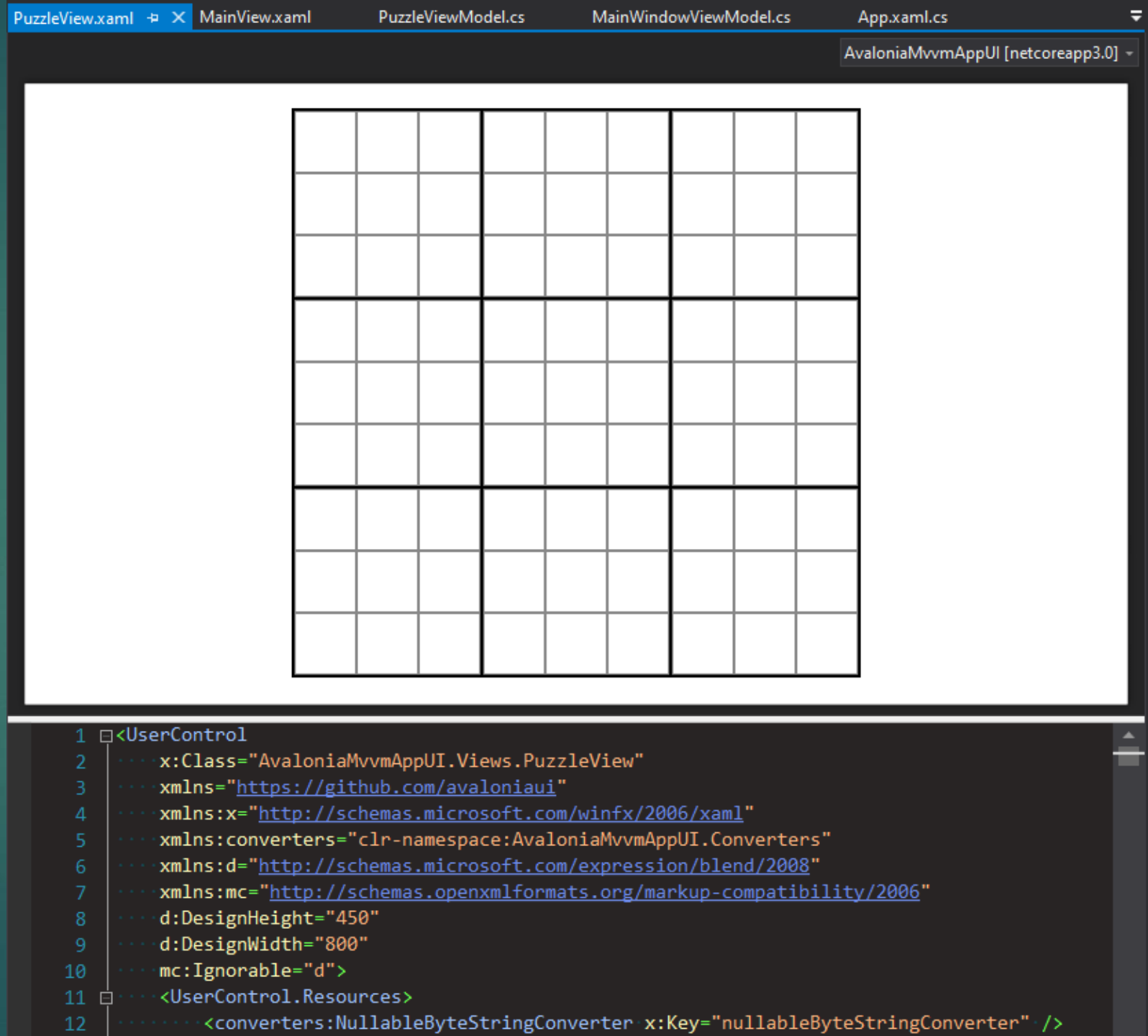


# Creating the Control (Visual Studio)



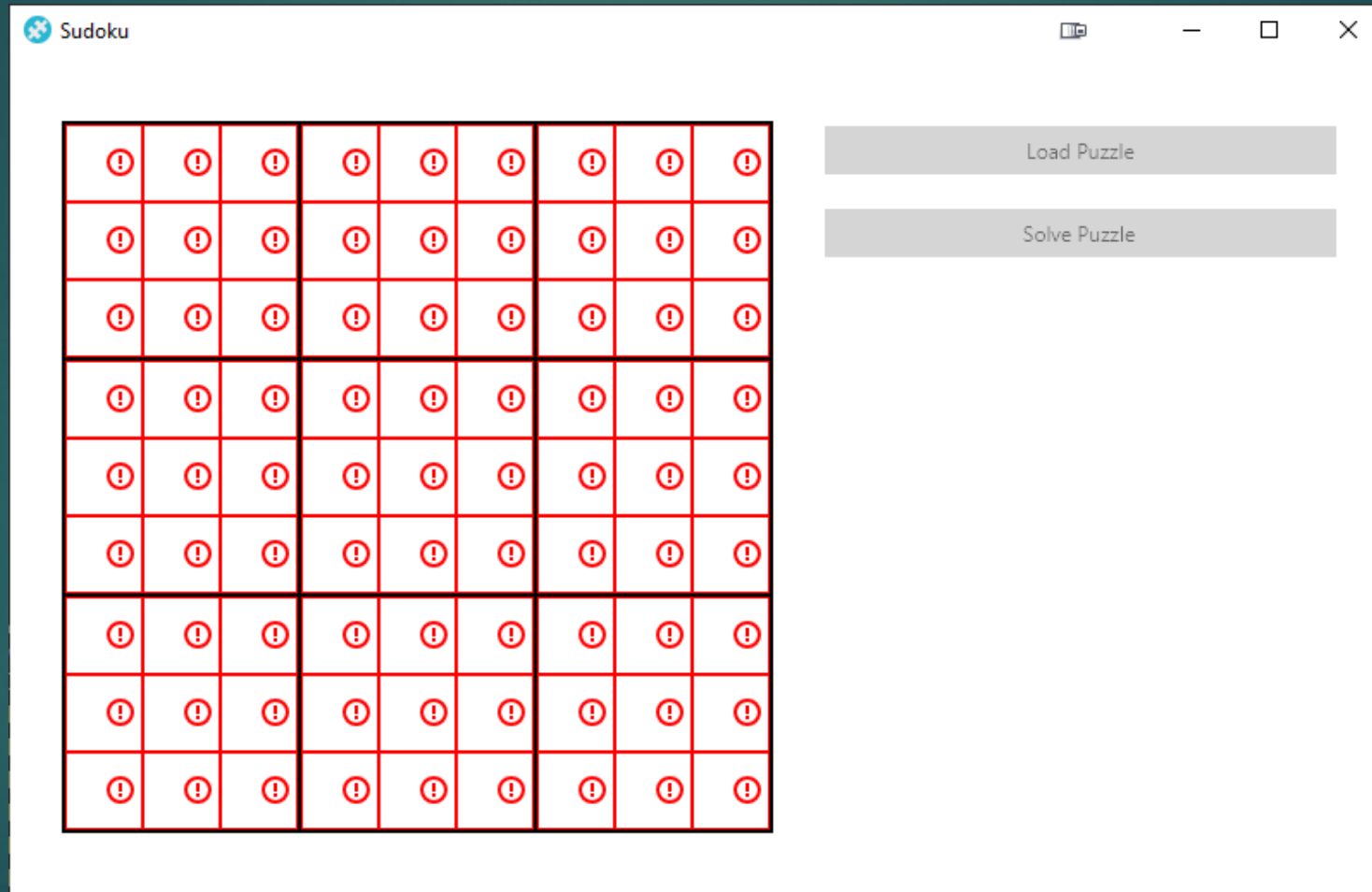
# Creating the Control (XAML)

- Styles
- Converter
- DataBinding





# Data Binding

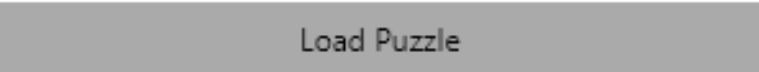


# Command

- ▶ No RelayCommand necessary

```
...  
<Button Command="{Binding LoadPuzzle}" Content="Load Puzzle" ... />  
...
```

```
public class MainWindowViewModel : ViewModelBase  
{  
    public void LoadPuzzle()  
    {  
        this.PuzzleViewModel.Load(PuzzleProvider.SamplePuzzle1);  
    }  
    ...  
}
```

A rectangular button with a light gray background and a thin white border. The text "Load Puzzle" is centered on the button in a dark gray font.

# Supported Controls

- ▶ Border
- ▶ Button
- ▶ Calendar
- ▶ Canvas
- ▶ Carousel
- ▶ CheckBox
- ▶ ContentControl
- ▶ ContextMenu
- ▶ Decorator
- ▶ DockPanel
- ▶ Expander
- ▶ Grid
- ▶ GridSplitter
- ▶ Image
- ▶ ItemsControl
- ▶ LayoutTransformControl
- ▶ ListBox
- ▶ Menu
- ▶ Panel
- ▶ ProgressBar
- ▶ RadioButton
- ▶ RepeatButton
- ▶ ScrollViewer
- ▶ Separator
- ▶ Slider
- ▶ StackPanel
- ▶ TabControl
- ▶ TextBlock
- ▶ Textbox
- ▶ ToolTip
- ▶ TreeView
- ▶ UserControl
- ▶ Window
- ▶ WrapPanel

# Roadmap



## Stuff we might want but not need to release 1.0 (Read: up for grabs)

- Ribbon
- DataGrid (see <https://github.com/AvaloniaUI/DataGrid/>)

## Stuff we aren't sure we need

- Universal like support (one app runs on all) IBD
- QML-style markup

Source:  
<https://github.com/AvaloniaUI/Avalonia/wiki/The-Plan>

# Conclusion

## Pro

- ▶ Tutorial
- ▶ Very WPF like
- ▶ Documentation
- ▶ Visual Studio Extensions
- ▶ MVVM support
- ▶ Debugging (code only)
- ▶ Binding to public Method (no Command required)
- ▶ Cross Platform

## Con

- ▶ Not yet ready for comercial usage
- ▶ Limited auto completion
- ▶ No implicit type conversion
- ▶ No Triggers
- ▶ No DataGrid and other controls
- ▶ No Helper Tools like Snoop
- ▶ Controls are limited
- ▶ No out of the box Internationalization support

# Try it out! 😊

- ▶ <https://avaloniaui.net>
- ▶ <https://github.com/mmarkovic/SudokuPlayground>

