



# TECH DAYS 2015

BREAKING NEW GROUND

# Building adaptive Windows 10 XAML apps

from Raspberry Pi to Surface Hub

Andy Wigley

Microsoft Developer Evangelist

t: andy\_wigley    b: <http://andywigley.com>



# Agenda

The Microsoft design language

How Windows makes design easier

What am I designing?

Techniques to adapt

Adaptive Tooling

- Visual States Triggers
- Relative Panel

Phone



Phablet



Small Tablet



Large Tablet



2-in-1s  
(Tablet or Laptop)



Classic  
Laptop



Desktops  
& All-in-Ones

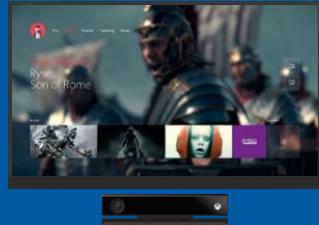


Windows 10

Surface Hub



Xbox



Holographic



IoT



# The Microsoft Design Language





# Microsoft design language principles

## Keep it Simple

Isn't it nice when **things just work**, when the next step is intuitive, and people are inspired?

## Think Universal

**Better technology** for anyone is better technology for everyone. It's an attitude.

## Design as One

Work together and do amazing things.  
Many teams, **one ecosystem**, happy customers.

## Make it Personal

We **design for real people**, not requiring people can fit into our design. Personal means human.

## Create Delight

Attention of detail equals **moments of delight**. Sometimes delight is so perfect it is invisible.



# Typeography

Display 3

Display 2

Display 1

Title 1

Body 1

Caption 1

Typeface      Weight      Size      Line spacing      Word spacing      Tracking

Segoe UI      Light      62 pt      80 pt      100%      -10

Segoe UI      Light      46 pt      56 pt      100%      -10

Segoe UI      Light      34 pt      40 pt      100%      0

Segoe UI      Light      24 pt      28 pt      100%      0

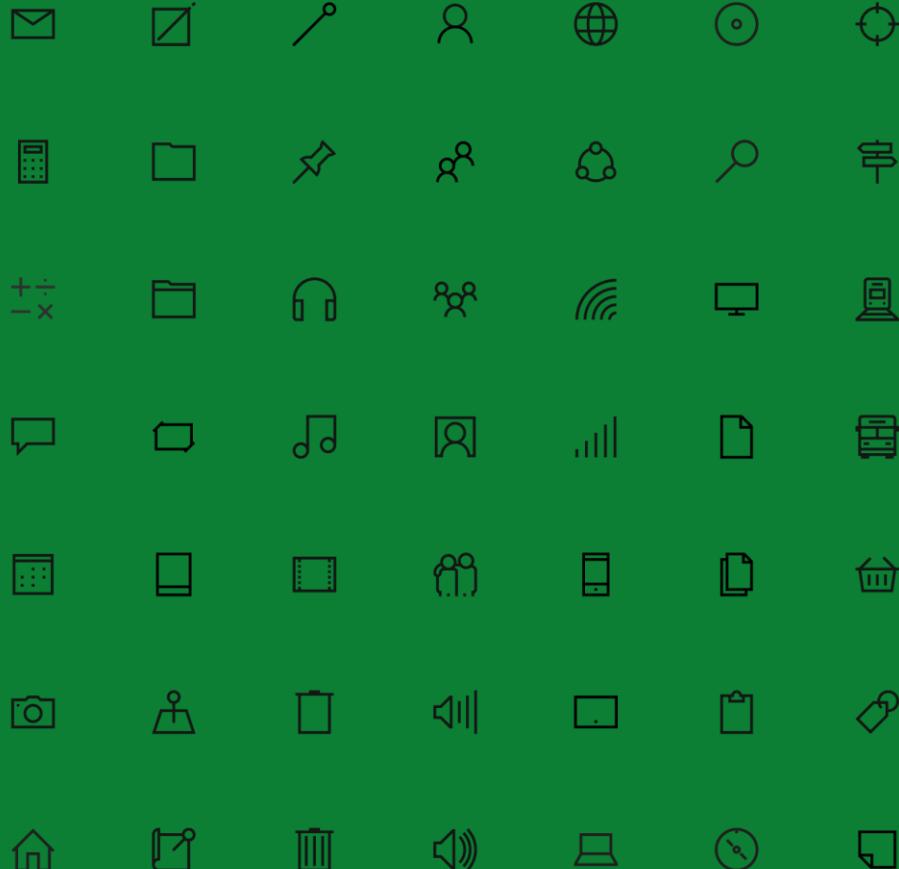
Segoe UI      Regular      14.6 pt      20 pt      100%      0

Segoe UI      Regular      12 pt      16 pt      100%      0



# Iconography

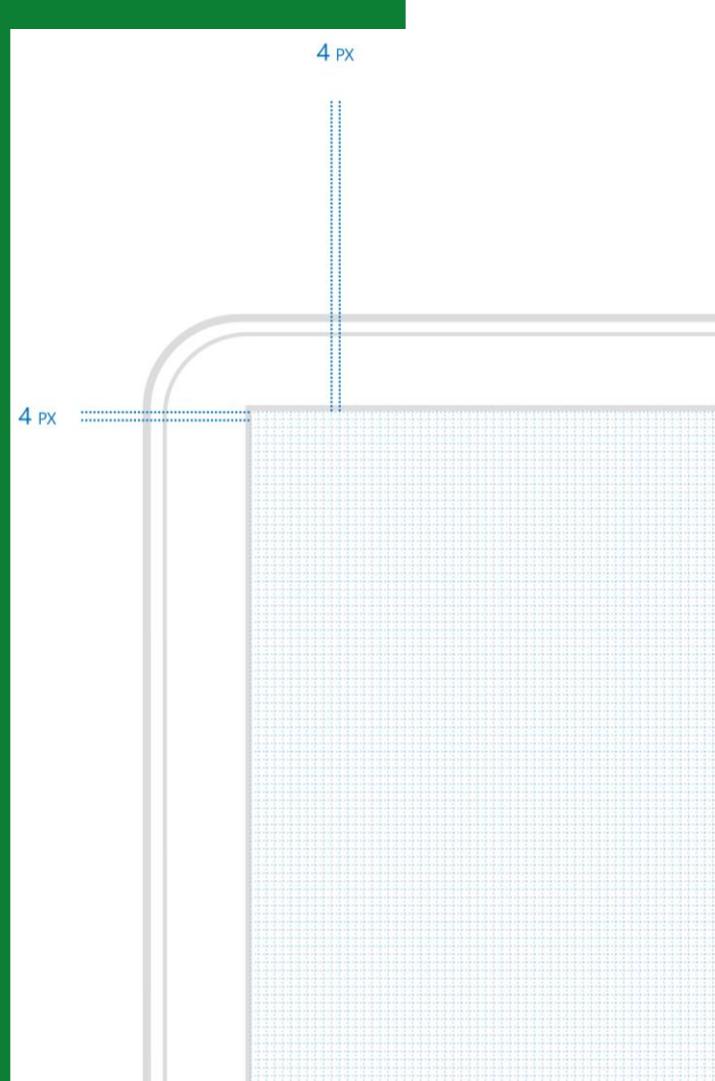
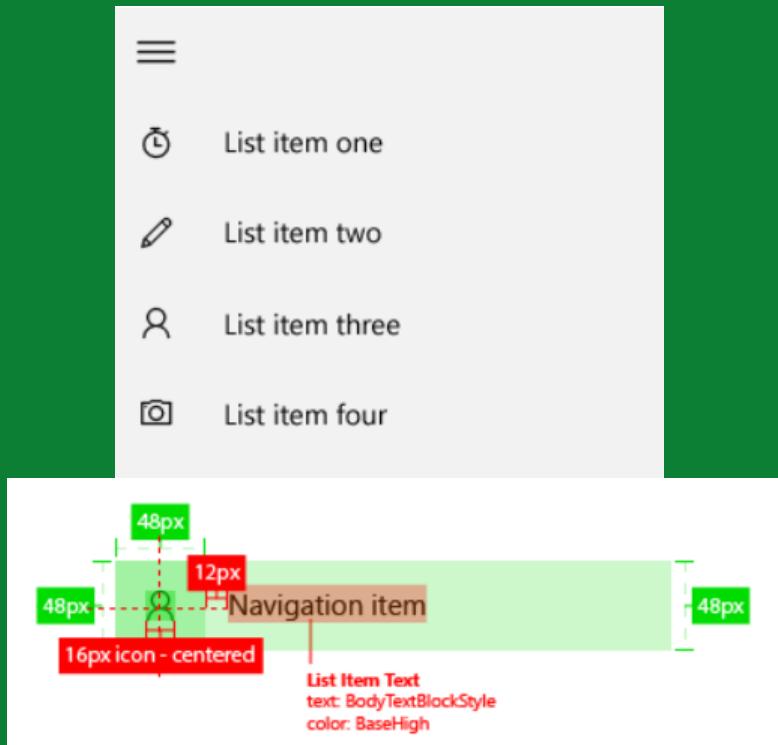
2px



For most text, use  
15 epx Segoe UI Regular



# Four is the magic number



4 PX

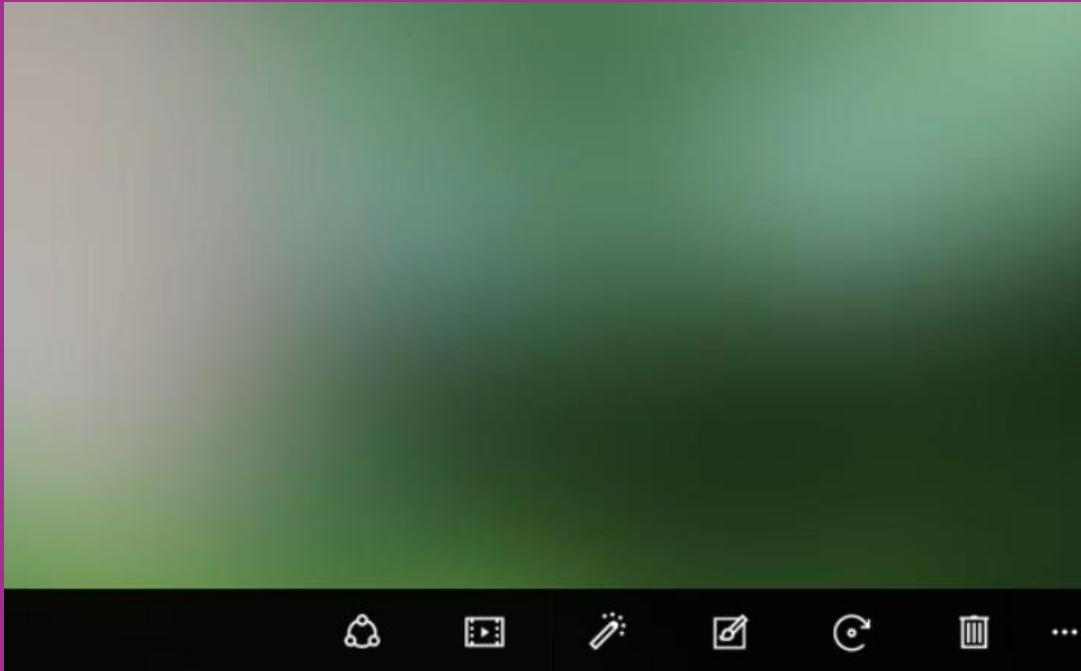
4 PX

A photograph of a young man and woman sitting on a set of concrete steps outside a modern building with large glass windows. They are both smiling and looking at a black tablet device held by the woman. She is wearing a teal and white striped sweater and blue jeans. He is wearing a colorful, horizontally striped long-sleeved shirt and grey sweatpants. A red backpack is slung over his shoulder. The background shows the building's exterior and some greenery.

# How Windows makes design easier

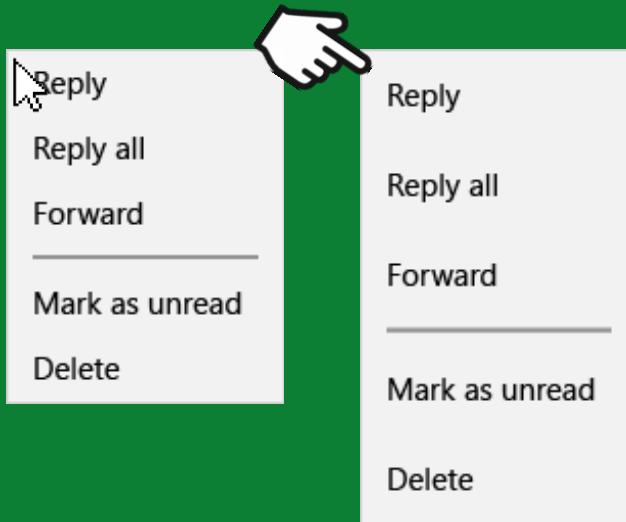


# Adaptive controls





# Input intelligence

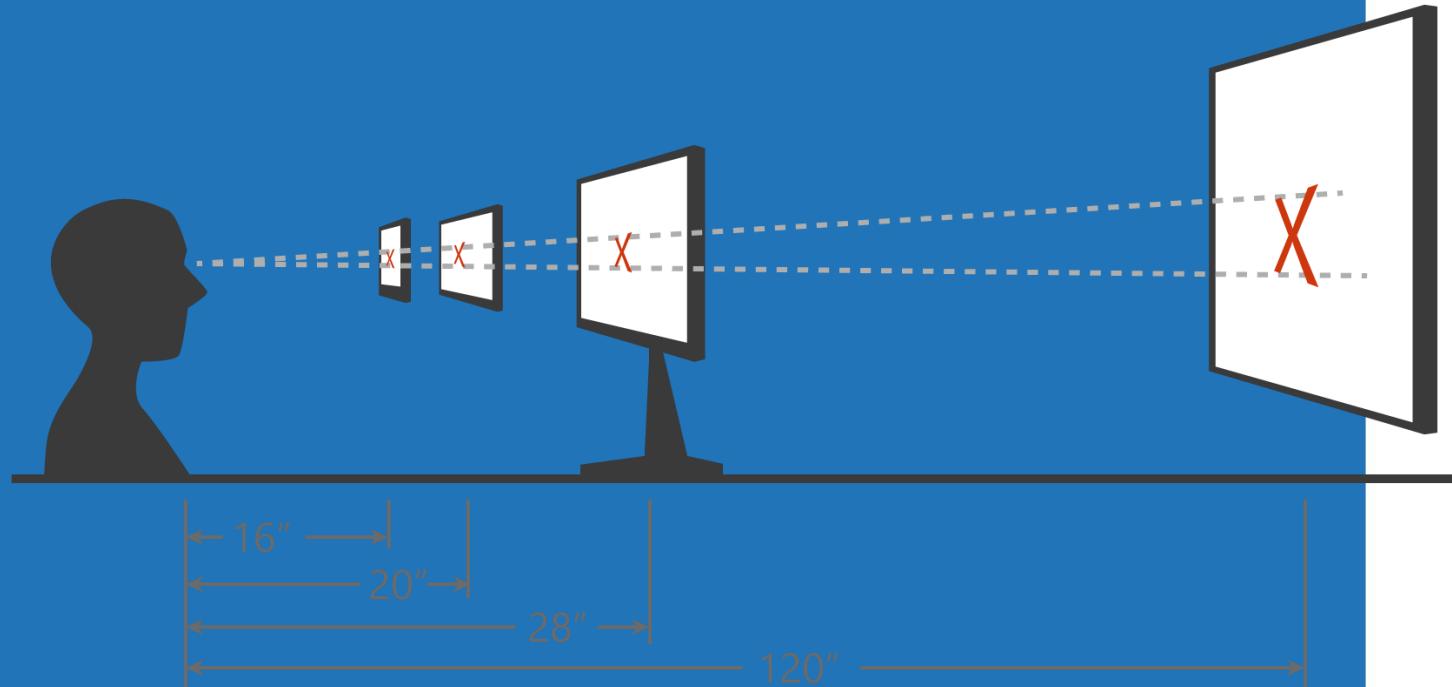


Month	Day	Year
September	20	2010
October	21	2011
November	22	2012
December	23	2013
January	24	2014
February	25	2015
March	26	2016
April	27	2017
Mav	28	2018

October	20	2010
November	21	2011
December	22	2012
January	23	2013
February	24	2014
March	25	2015
April	26	2016
May	27	2017
June	28	2018



# Scaling algorithm



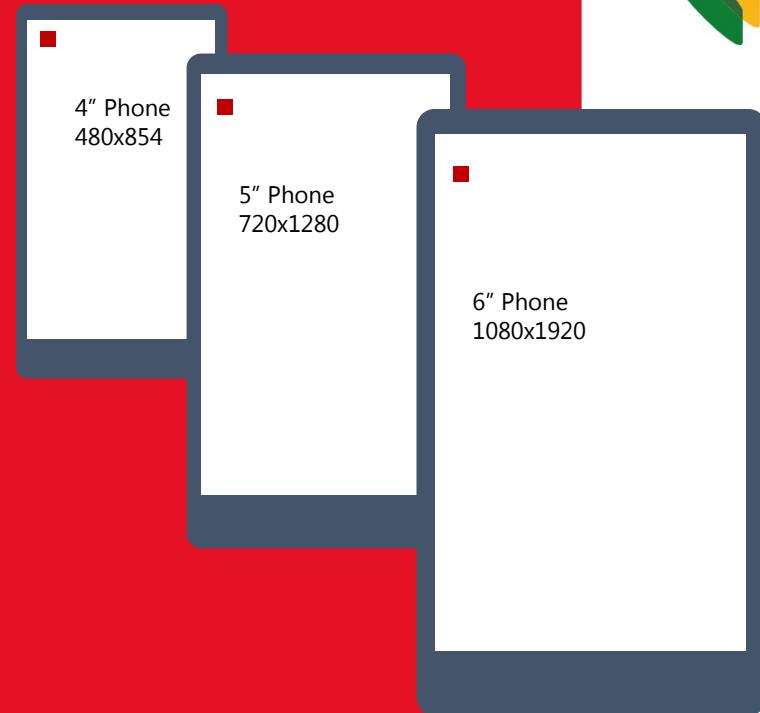
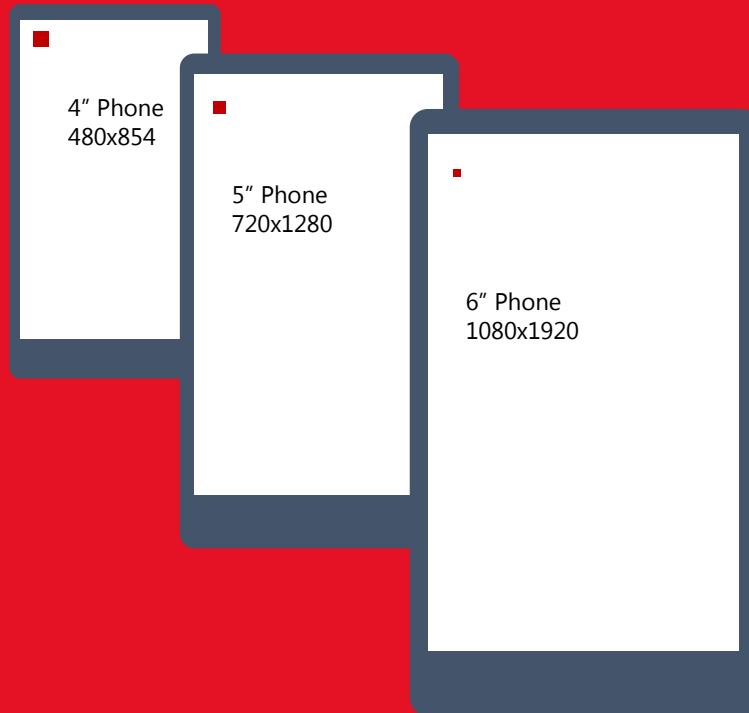
# Universal Windows Platform apps

for Universal Windows Platform Apps



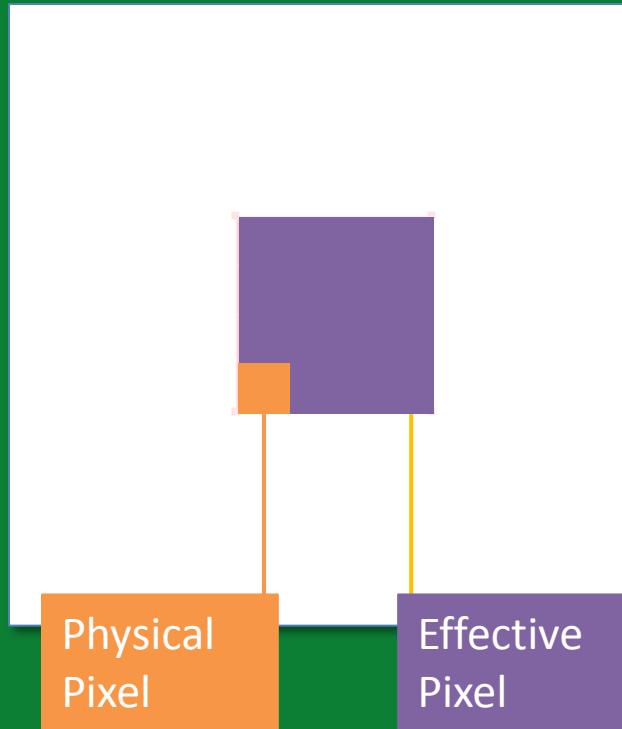
Windows 10

# Physical versus effective pixel





# Effective pixel



Ignore scale, resolution, & dpi.  
Design in Effective Pixels

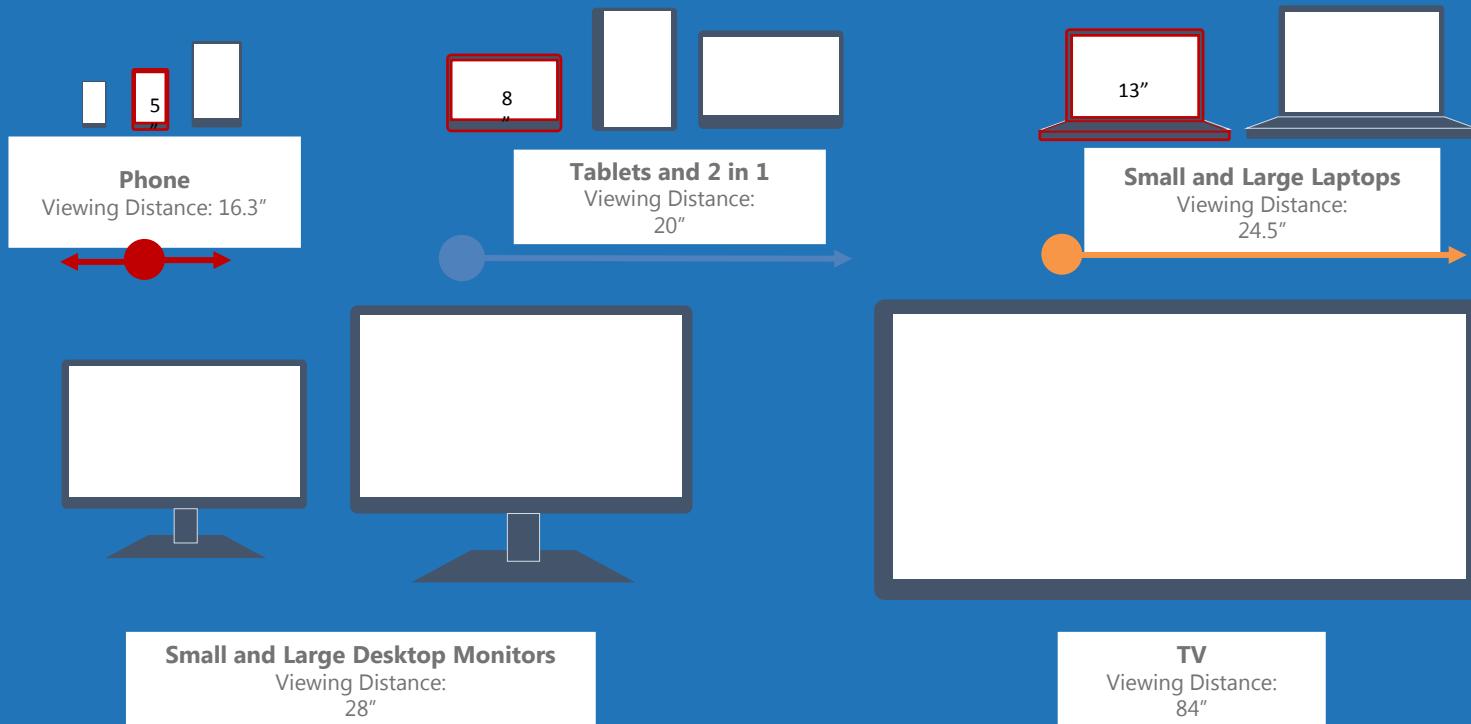


Windows



# What am I designing?

# Planning your design



# Snap points



epx

320

548

720

1024

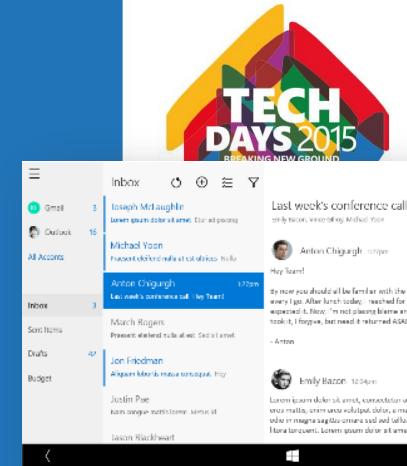
phone

- limited landscape support
- one frame at a time
- sys tray on the left
- steering wheel on the right
- 4+ actions on the bottom
- tabs are centered

- limited landscape support
- one frame at a time
- sys tray on the left
- steering wheel on the right
- 4+ actions on the bottom
- single column stops scaling
- tabs are centered

phablet  
& tablet

- full landscape support
- two frames
- actions at the top
- one “...” visible - TBD
- tabs are left aligned
- Show search field if search was represented as an icon on smaller devices



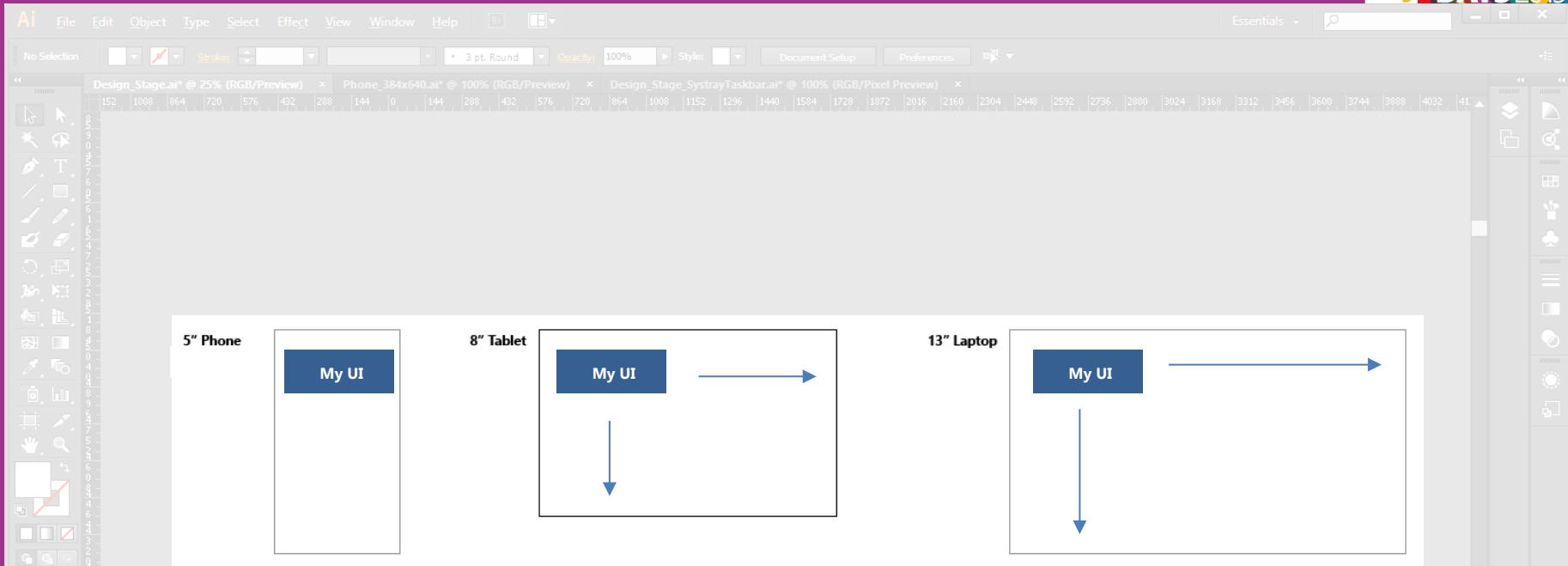
desktop

- full landscape support
- three frames
- compact nav pane
- actions at the top
- one “...” visible
- tabs left aligned

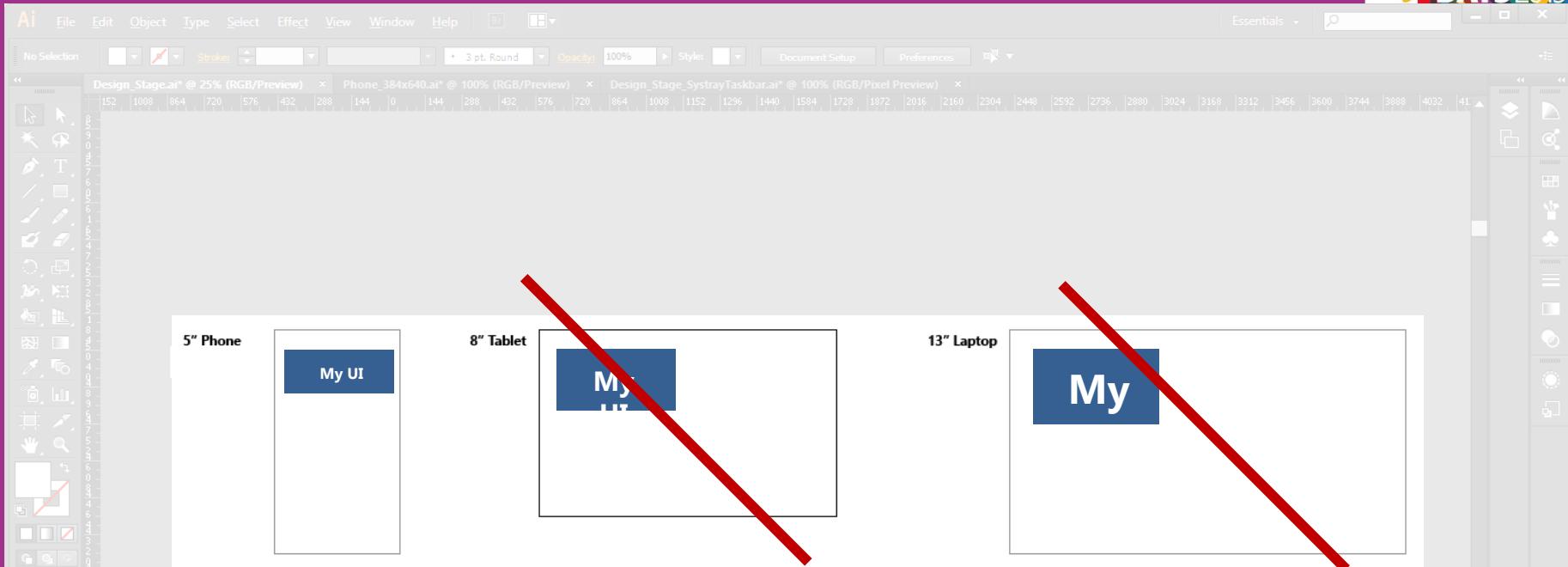
# Demo

## The Weather App

# Flow, not size



# Flow, not size



# Check variants



Ai File Edit Object Type Select Effect View Window Help Document Setup Preferences Essentials

No Selection

Design.Stage.ai\* @ 25% (RGB/Preview) × Phone\_384x640.ai\* @ 100% (RGB/Preview) × Design.Stage.SystrayTaskbar.ai\* @ 100% (RGB/Pixel Preview) ×

152 1008 864 720 576 432 288 144 0 144 288 432 576 720 864 1008 1152 1296 1440 1584 1728 1872 2016 2160 2304 2448 2592 2736 2880 3024 3168 3312 3456 3600 3744 3888 4032 411

4" Phone 360x640

PROFILE WHAT'S NEW CONNECT HISTORY 9  
Alexandra Lane  
Microsoft, Google, Facebook, Skype  
1 new photo  
Alexandra was tagged in Aaron Collier's photo  
Facebook, 1 hour ago  
Message (02) 456 7890  
Call mobile (098) 989 092  
Call work (099) 012 3456  
Call with app viber  
Video call

5" Phone

PROFILE WHAT'S NEW CONNECT HISTORY 9  
Alexandra Lane  
Microsoft, Google, Facebook, Skype  
1 new photo  
Alexandra was tagged in Aaron Collier's photo  
Facebook, 1 hour ago  
Message (02) 456 7890  
Call mobile (098) 989 092  
Call work (099) 012 3456  
Call with app viber  
Video call

8" Tablet

13" Laptop

6" Phone 480x854

PROFILE WHAT'S NEW CONNECT HISTORY 9  
Alexandra Lane  
Microsoft, Google, Facebook, Skype  
1 new photo  
Alexandra was tagged in Aaron Collier's photo  
Facebook, 1 hour ago  
Message (02) 456 7890  
Call mobile (098) 989 092  
Call work (099) 012 3456  
Call with app viber

8" Tablet Portrait 640x1024

The image shows a Adobe Illustrator interface with multiple document windows open. The top window displays a 4-inch phone screen with a social media feed for 'Alexandra Lane'. A red arrow points from the word 'Lane' in the feed to the profile picture. Below it are documents for a 5" Phone, 8" Tablet, 13" Laptop, and 6" Phone, all showing blank white spaces. The bottom window shows an 8" Tablet in portrait mode with a 640x1024 resolution.

# As you design

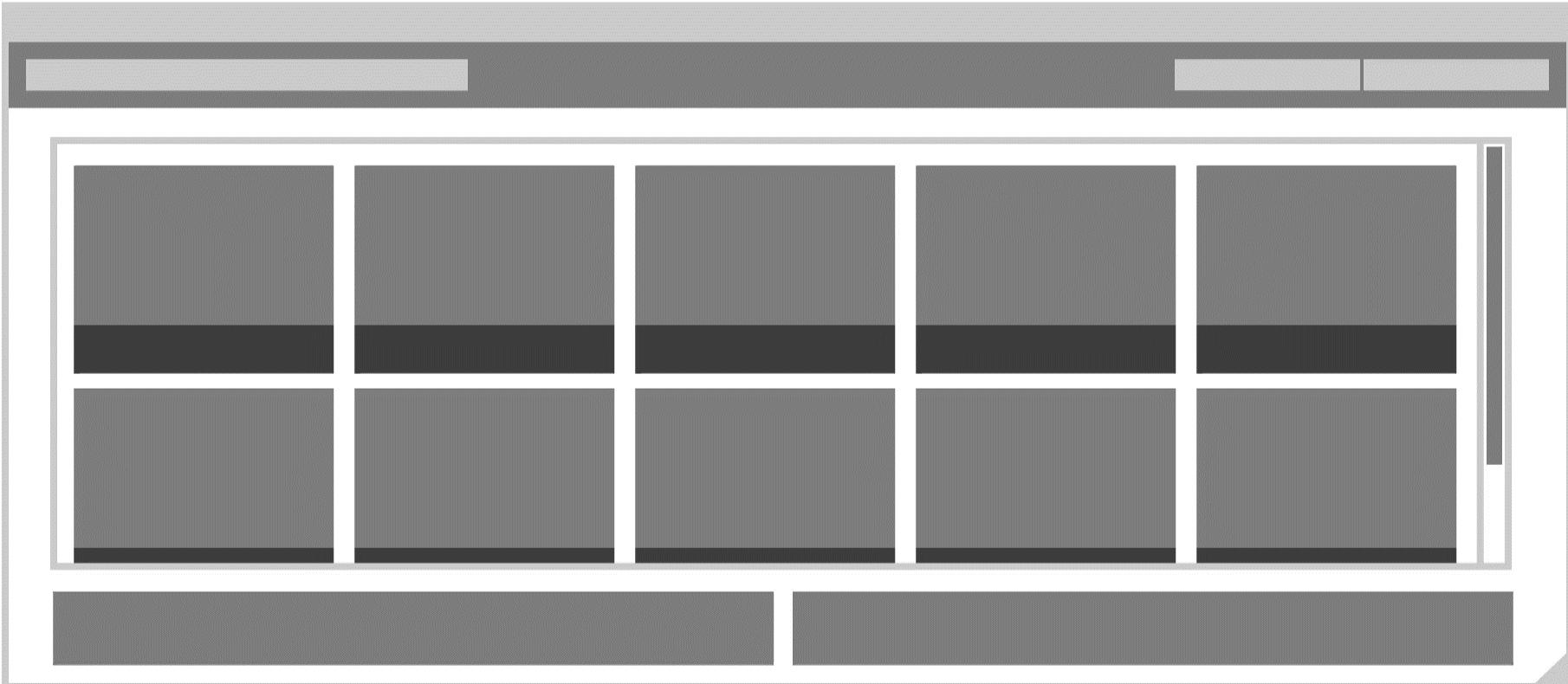
1. Adapt to size change
2. Adapt to input change
3. Build with effective pixels
4. Count on the scaling algorithm



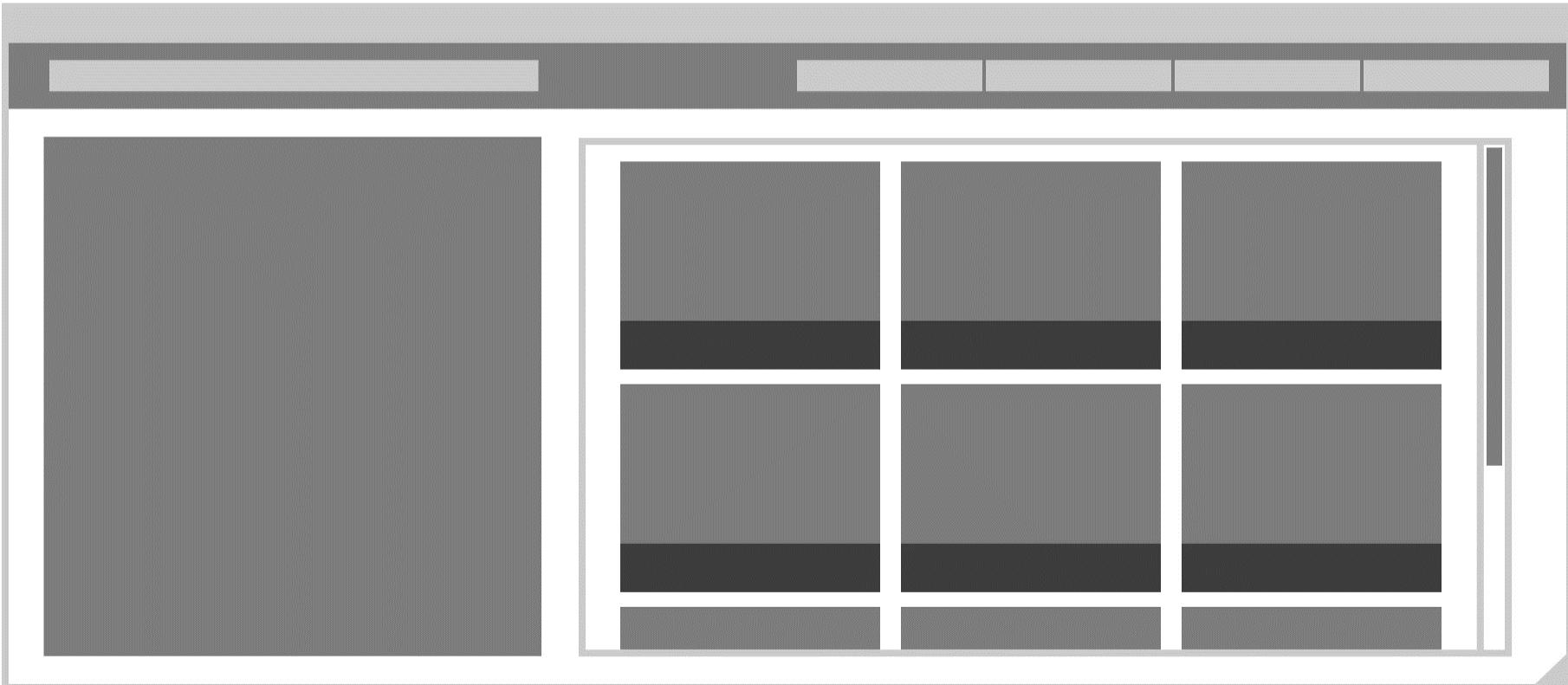
# Adaptive UI design techniques



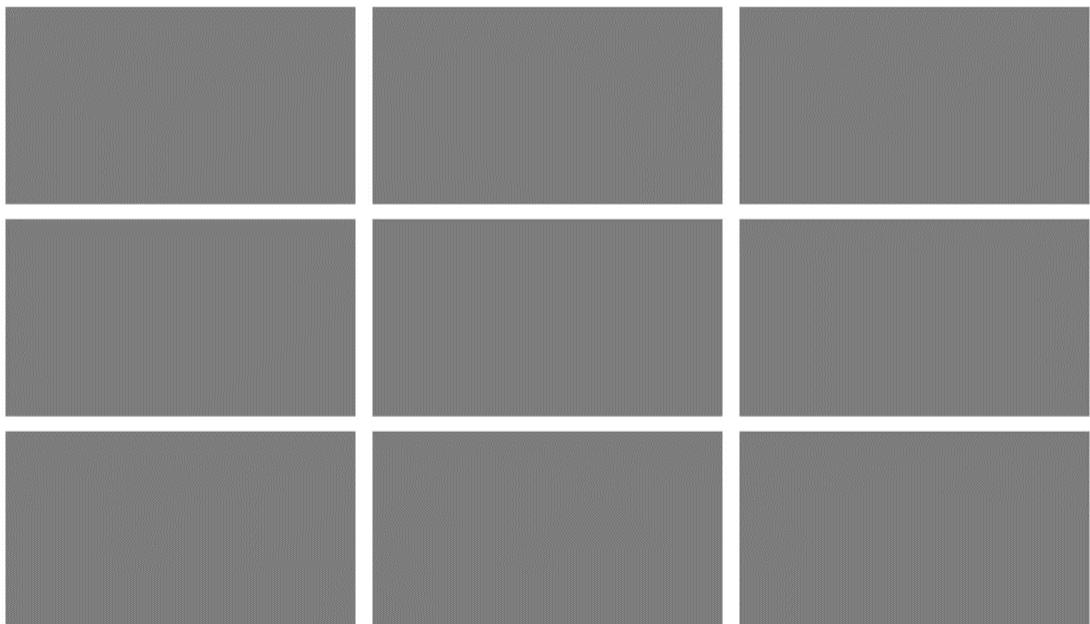
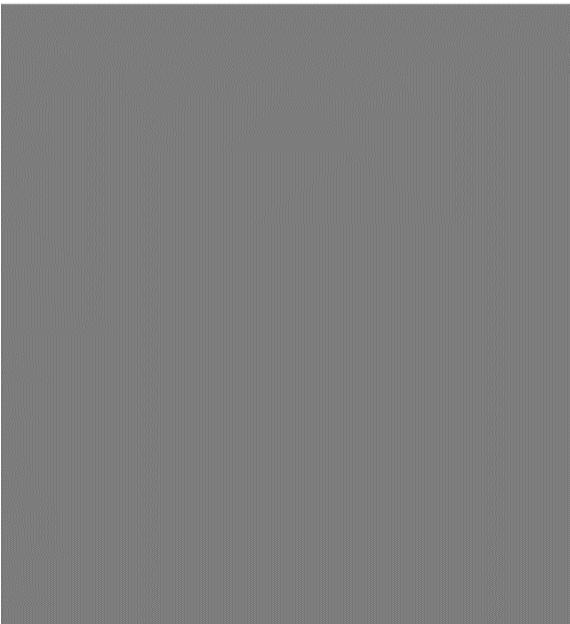
# Responsive



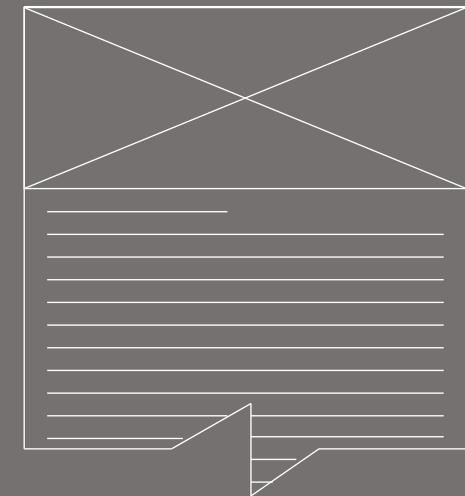
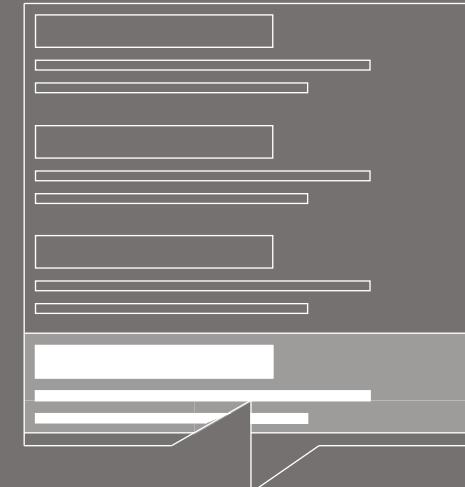
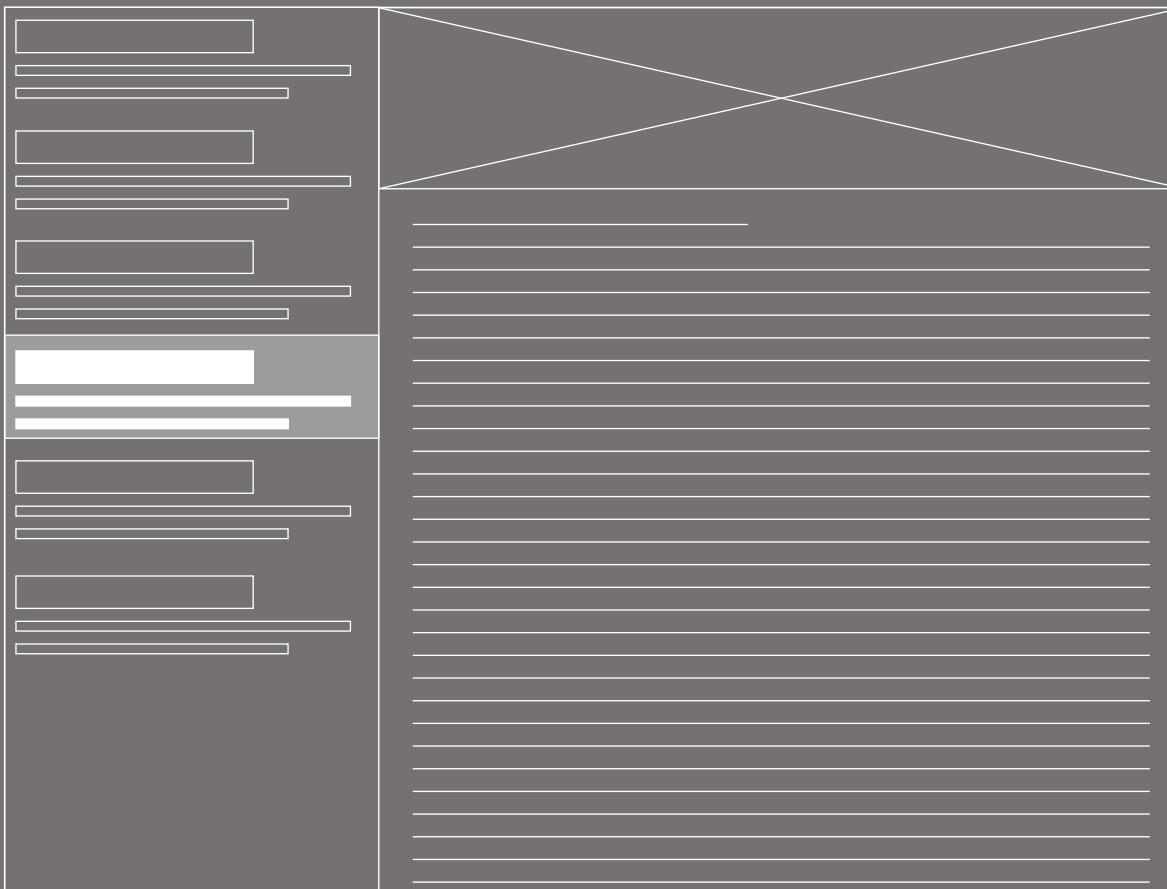
# Adaptive



# Scaling



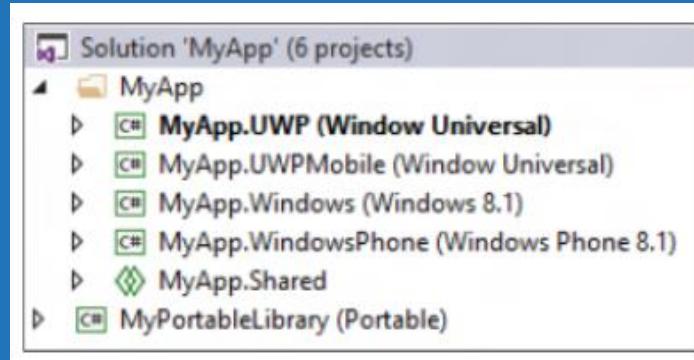
# ReArchitect/tailored design

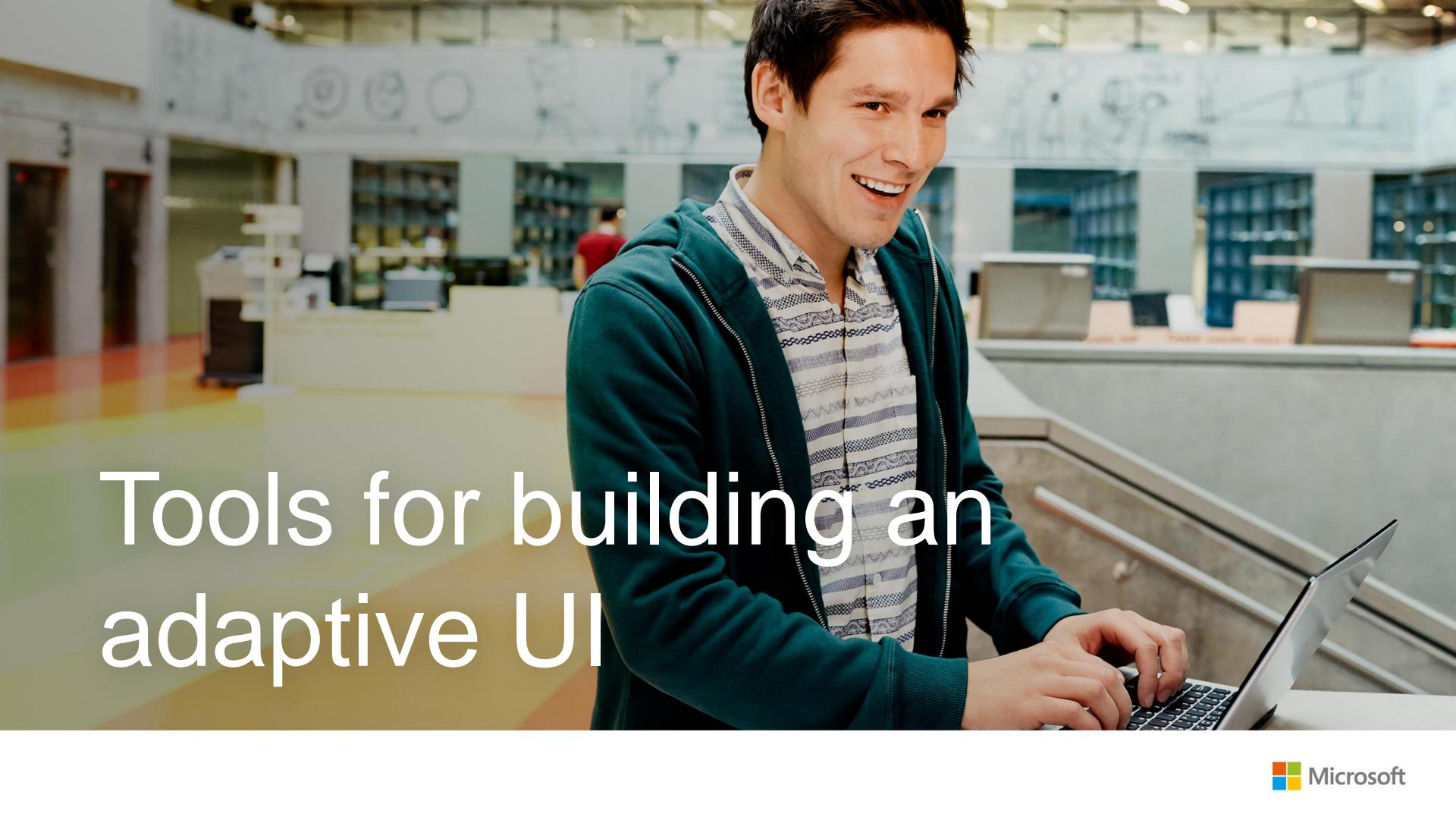




# Dedicated, targeted apps

Nothing is stopping you from creating a multi-headed solution



A photograph of a young man with dark hair, smiling broadly. He is wearing a teal zip-up hoodie over a white and blue striped shirt. He is seated at a desk, looking down at a silver laptop. The background shows a modern office environment with large windows, bookshelves, and other people in the distance.

# Tools for building an adaptive UI



# Visual States

Define XAML views

Unique layout for distinct states

Simplify animation

Automatically implement state transitions

Build in Blend

Design and preview states and transitions

# Demo

## Visual States

Visual states let designers  
define many different layouts of a  
view





# Select a visual state in code

VisualStateManager.Goto(element, state, transition)

```
public MainPage()
{
    this.InitializeComponent();
    this.SizeChanged += (s, e) =>
    {
        var state = "VisualState000min";
        if (e.NewSize.Width > 500)
            state = "VisualState500min";
        else if (e.NewSize.Width > 800)
            state = "VisualState800min";
        else if (e.NewSize.Width > 1000)
            state = "VisualState1000min",
        VisualStateManager.GoToState(this, state, true);
    };
}
```

# Demo

## Adaptive triggers



# Adaptive triggers

Code-free state transition

Two built-in triggers:

MinWindowHeight (Taller than this)

MinWindowWidth (Wider than this)

```
<VisualState x:Name="VisualState500min">
    <VisualState.StateTriggers>
        <AdaptiveTrigger MinWindowWidth="501" />
    </VisualState.StateTriggers>
</VisualState>
```

Adaptive triggers allow selection of visual states from XAML



Windows

# Custom triggers support your special scenarios



Windows



# Custom adaptive triggers

## Build to handle special cases

```
public class DeviceFamilyTrigger : StateTriggerBase
{
    private string _deviceFamily;
    public string DeviceFamily
    {
        get { return _deviceFamily; }
        set
        {
            var qualifiers = Windows.ApplicationModel.Resources.Core
                .ResourceContext.GetForCurrentView().QualifierValues;
            if (qualifiers.ContainsKey("DeviceFamily"))
                SetActive(qualifiers["DeviceFamily"] == (_deviceFamily = value));
            else
                SetActive(false);
        }
    }
}
```

# Windows Insiders FTW!



dotMorten / WindowsStateTriggers

Watch 4   Star 26   Fork 8

A collection of custom visual state triggers

32 commits   1 branch   0 releases   3 contributors   Code

branch: master + [WindowsStateTriggers](#) / +

Updated code summary

dotMorten authored 27 days ago

src   Updated code summary  
 .gitignore   Initial commit  
 LICENSE   Updated to MIT license  
 README.md   Update README.md

README.md

<http://github.com/dotMorten/WindowsStateTriggers>

Morten Nielsen  
Windows State Triggers  
Windows Developer MVP (@dotMorten)

A collection of custom visual state triggers

A screenshot of a GitHub repository page for "WindowsStateTriggers". The repository has 32 commits and 1 branch. It shows a list of files including src, .gitignore, LICENSE, and README.md. A commit from dotMorten is shown as the latest, made 27 days ago. Below the repository stats, there's a section for README.md. The main content area displays a grid of six cards, each representing a different type of custom visual state trigger. The cards are: DeviceFamilyAdaptiveTrigger (Hello Windows Desktop!), NetworkConnectionStateTrigger (Internet is available), OrientationStateTrigger (Portrait mode), IsInWindowStateTrigger / IsFalseStateTrigger (Checkbox is checked, Checkbox is unchecked, Check me checkbox), IsTypePresentStateTrigger (Hardware Backbutton not available), and EqualsStateTrigger (Checkbox value is not indeterminate, Check me checkbox, Slider value is 0 slider). A circular profile picture of Morten Nielsen is in the bottom right corner.

# Demo

## Custom adaptive triggers



# Visual state setters

Setting a simple, scalar value

Great when you think of ENUM values like Visibility, Stretch

Does not invoke a storyboard

Does not require ObjectAnimationUsingKeyFrames

```
<VisualState.Setters>
    <Setter Target="MyText01.FontSize" Value="24" />
    <Setter Target="MyImage.Stretch" Value="UniformToFill" />
    <Setter Target="MyImage.Height" Value="150" />
</VisualState.Setters>
```



# Introducing the Relative Panel

## Windows XAML layout controls

Grid

Stack  
Panel

Canvas

Scroll  
Viewer

Border

View Box

Wrap  
Grid

Relative  
Panel

Relative Panel is a XAML layout control. It arranges children by declaring relationships between them.



# XAML's RelativePanel control

A child or two act as anchor elements

They are relative to the panel

Other children are relative to the anchors

`RelativePanel.Above = "ElementName"`

`RelativePanel.RightOf = "ElementName"`

`RelativePanel.Below = "ElementName"`

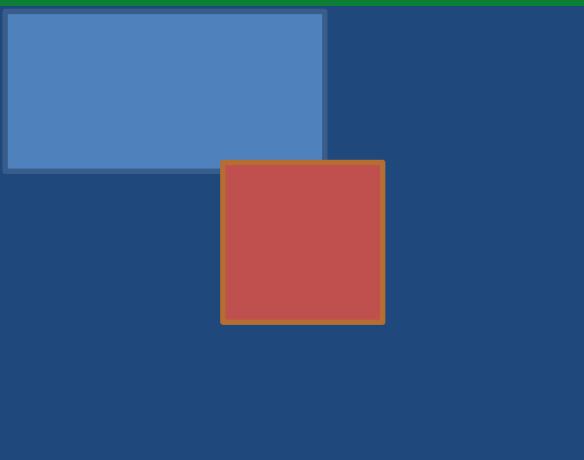
`RelativePanel.LeftOf = "ElementName"`

RelativePanel simplifies adaptive UI

A simple Visual State setter can rearrange the UI

One element can move a family of related element

# Align with panel



```
<RelativePanel>

    <Rectangle x:Name="RedRect"
        Height="100" Width="100" Fill="Red"

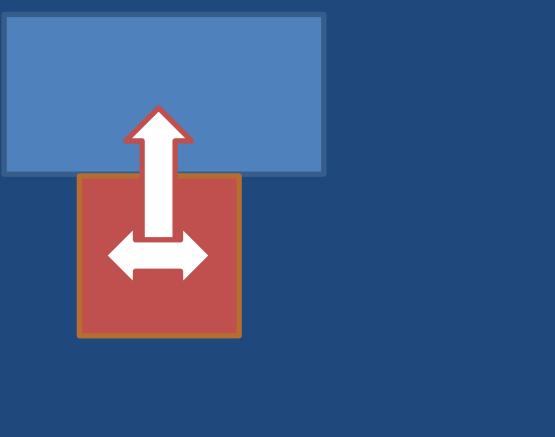
            RelativePanel.AlignHorizontalCenterWithPanel="True"
            RelativePanel.AlignVerticalCenterWithPanel="True" />

    <Rectangle x:Name="BlueRect"
        Height="100" Width="200" Fill="Blue" />

</RelativePanel>
```



# Align with sibling (below, center)



```
<RelativePanel>

    <Rectangle x:Name="BlueRect"
        Height="100" Width="100" Fill="Blue" />

    <Rectangle x:Name="RedRect"
        Height="100" Width="100" Fill="Red"
        RelativePanel.Below="BlueRect"
        RelativePanel.AlignHorizontalCenterWith="BlueRect" />

</RelativePanel>
```

Developers have many tools  
to build an adaptive UI



Windows

# Review

The Microsoft design language

How Windows makes design easier

What am I designing?

Techniques to adapt

Adaptive Tooling

Visual States Triggers

Relative Panel





# Related sessions (28-5)

Timeslot	Session	Speaker
12:15- 13:15	The Windows 10 App platform: an introduction to the Universal Windows Platform	Andy Wigley
12:45- 13:45	App lifecycle in Windows 10	Dave Smits
13:00- 14:00	Design for Universal Windows Apps	Martin Tirion
14:15- 15:15	Building adaptive Windows 10 XAML apps: from Raspberry Pi to Surface Hub	Andy Wigley
15:30- 16:30	Universal Windows Platform app to app communication	Martin Tirion
15:30- 16:30	Universal Windows Apps: 21st Century WIMP – Adding Ink, Speech and More	Mike Taulty
17:30- 18:30	Windows 10 Universal Windows Apps for Enterprise	Matthijs Hoekstra



# Related sessions (29-5)

Timeslot	Session	Speaker
08:15-09:15	Universal Windows Apps: Background >= Foreground	Mike Taulty
12:15-13:15	The Windows 10 App platform: an introduction to the Universal Windows Platform	Andy Wigley
13:45-14:45	What's new in XAML for Universal Windows Apps	Martin Tirion
15:00-16:00	Windows 10 for Makers: Raspberry Pi, Arduino, AllJoyn & Microsoft Band	Rajen Kishna
16:15-17:15	Maps and Location in Windows 10	Joost van Schaik
16:15-17:15	Integrate with Windows 10	Dave Smits

# Your feedback is important!

Scan the QR Code and let us know via the TechDays App.



Laat ons weten wat u van de sessie vindt via de TechDays App!  
Scan de QR Code.

MVA

Microsoft  
Virtual  
Academy

Bent u al lid van de Microsoft Virtual Academy?! Op MVA kunt u altijd iets nieuws leren over de laatste technologie van Microsoft. Meld u vandaag aan op de MVA Stand. MVA biedt 7/24 gratis online training on-demand voor IT-Professionals en Ontwikkelaars.





# TECH DAYS 2015

BREAKING NEW GROUND