Build Adaptive UIs with Xamarin.Forms

Greg Lutz

Xuni Product Manager, GrapeCity

Overview

With Xamarin.Form’s natural ability to target devices of all sizes, building adaptive UIs is more important than ever.

Learn how to build adaptive UIs that scale across iPhones, Android tablets and Windows PCs while reusing as much code as possible.

Takeaways include adaptive best practices and tips for XAML & C# development.

Agenda

Why adaptive UI

Tips and techniques for building adaptive Xamarin.Forms apps

XAML best practices

Why Adaptive UI

The goal of adaptive UI is to adapt its layout to the needs of the user.

In our case Adaptive UI will mean adaption to different sized devices.

Xamarin.Forms Adaptive UI

With Xamarin.Forms we can develop once for all devices using a single C# codebase.

- Across different platforms (iOS, Android, Windows)
- Across different devices (phones, tablets, desktop)

*This means our UI needs to be adaptive.

- Maximize sharing
- Reduce code
- Deliver better UX

The Goal

Our goal is to build as close to a single UI as possible that is adaptive for all devices.

The straight-forward alternative:

```
if(phone) → launch phone version of app
if(tablet) → launch tablet version of app
if(desktop) → launch desktop (UWP) version of app
```

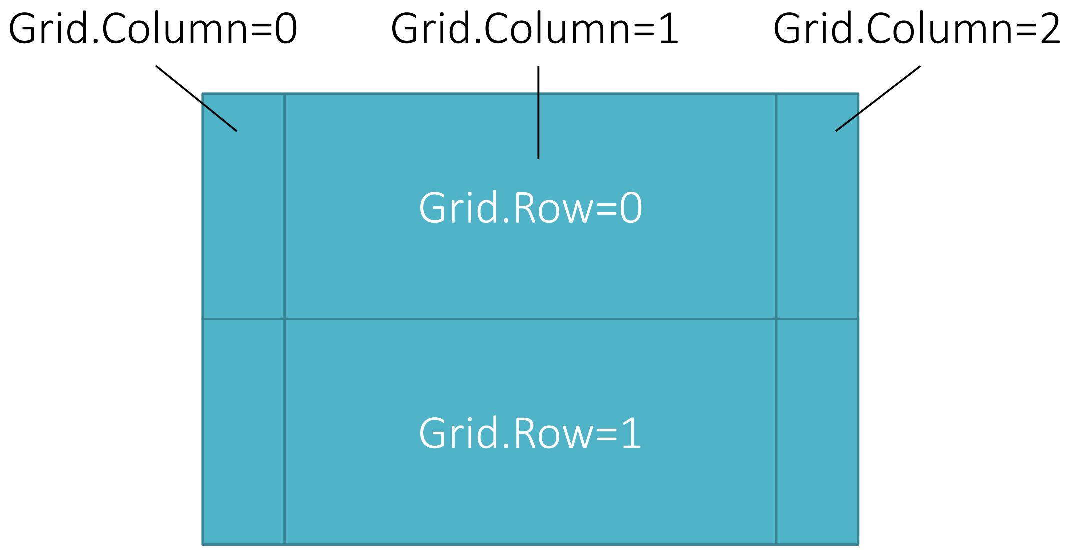
Adaptive Basics

ChartBuilder sample

Use Basic Layout Controls

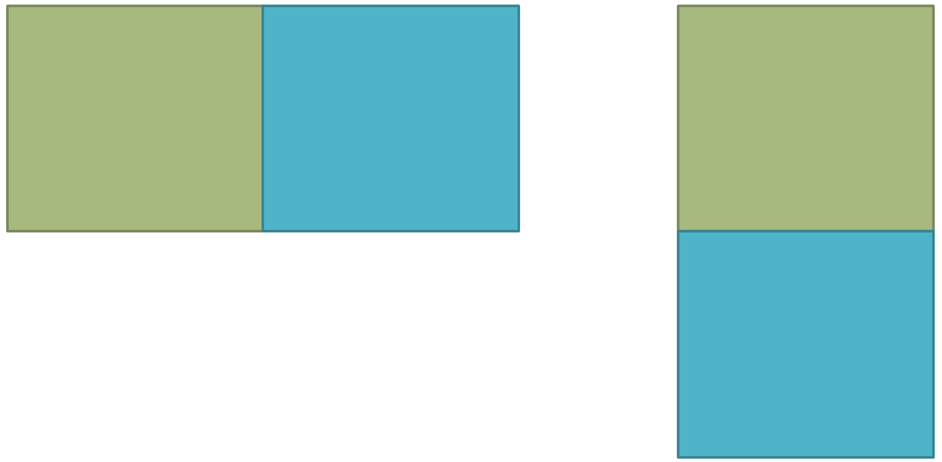
GRID

Column and Row placement



STACKPANEL

Horizontal or Vertical stacking



Let children stretch and fill the space

Use ScrollView for Overflow

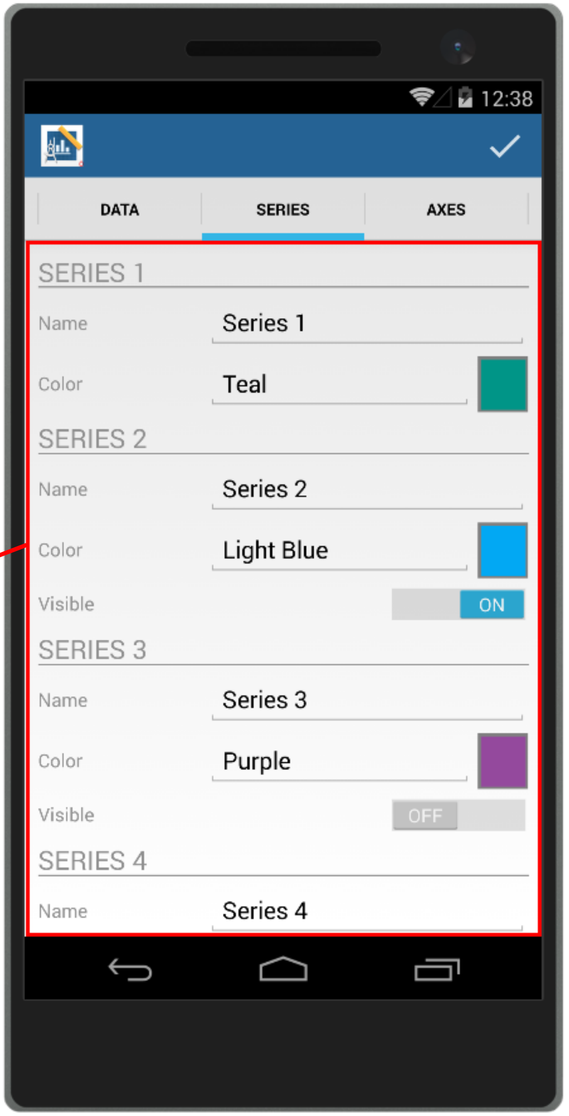
Enables scrolling for view content that overflows available screen real estate

Vertical and horizontal scrolling

Makes a page instantly usable on all devices

Good safety-net for those tiny devices you couldn't predict users may have

```
<ScrollView>
    <Grid />
</ScrollView>
```



SERIES 1

Name	Series 1	
Color	Teal	🟩

SERIES 2

Name	Series 2	
Color	Light Blue	🟦
Visible		ON

SERIES 3

Name	Series 3	
Color	Purple	🟪
Visible		OFF

SERIES 4

Name	Series 4

Use Device.Idiom

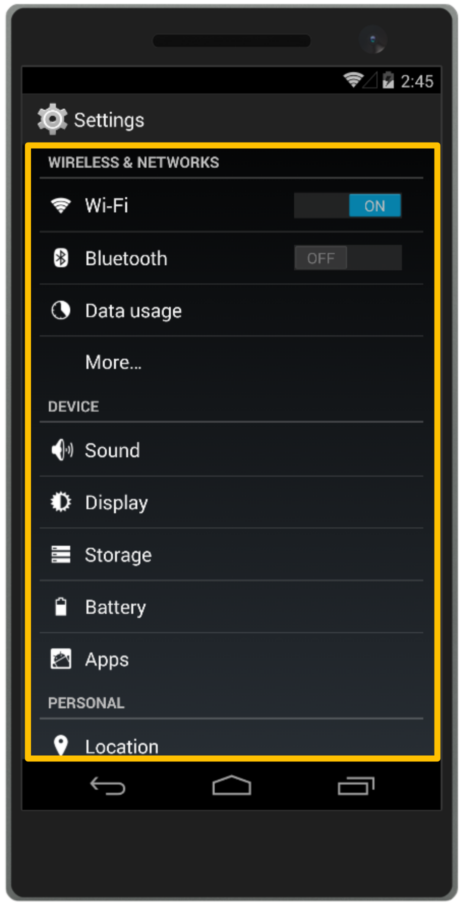
Xamarin.Forms.Device.Idiom enum contains information about what type of device the app is being run on

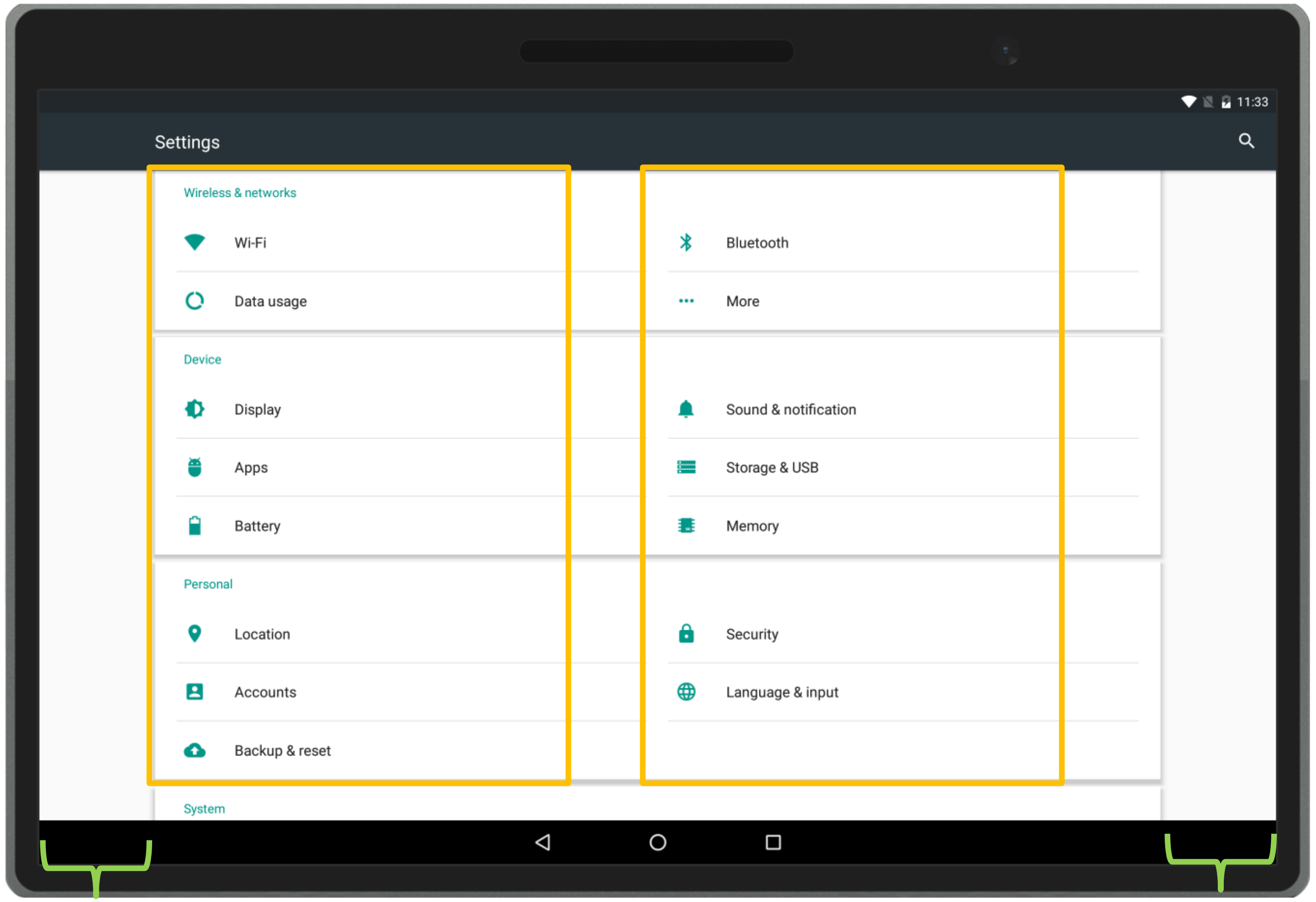
- Phone (any OS)
- Tablet (any OS)
- Desktop (UWP on Windows 10)

```
if (Xamarin.Forms.Device.Idiom == TargetIdiom.Phone)
{
    // apply phone only code
}
else if (Xamarin.Forms.Device.Idiom == TargetIdiom.Tablet)
{
    // apply tablet only code
}
```

One Column

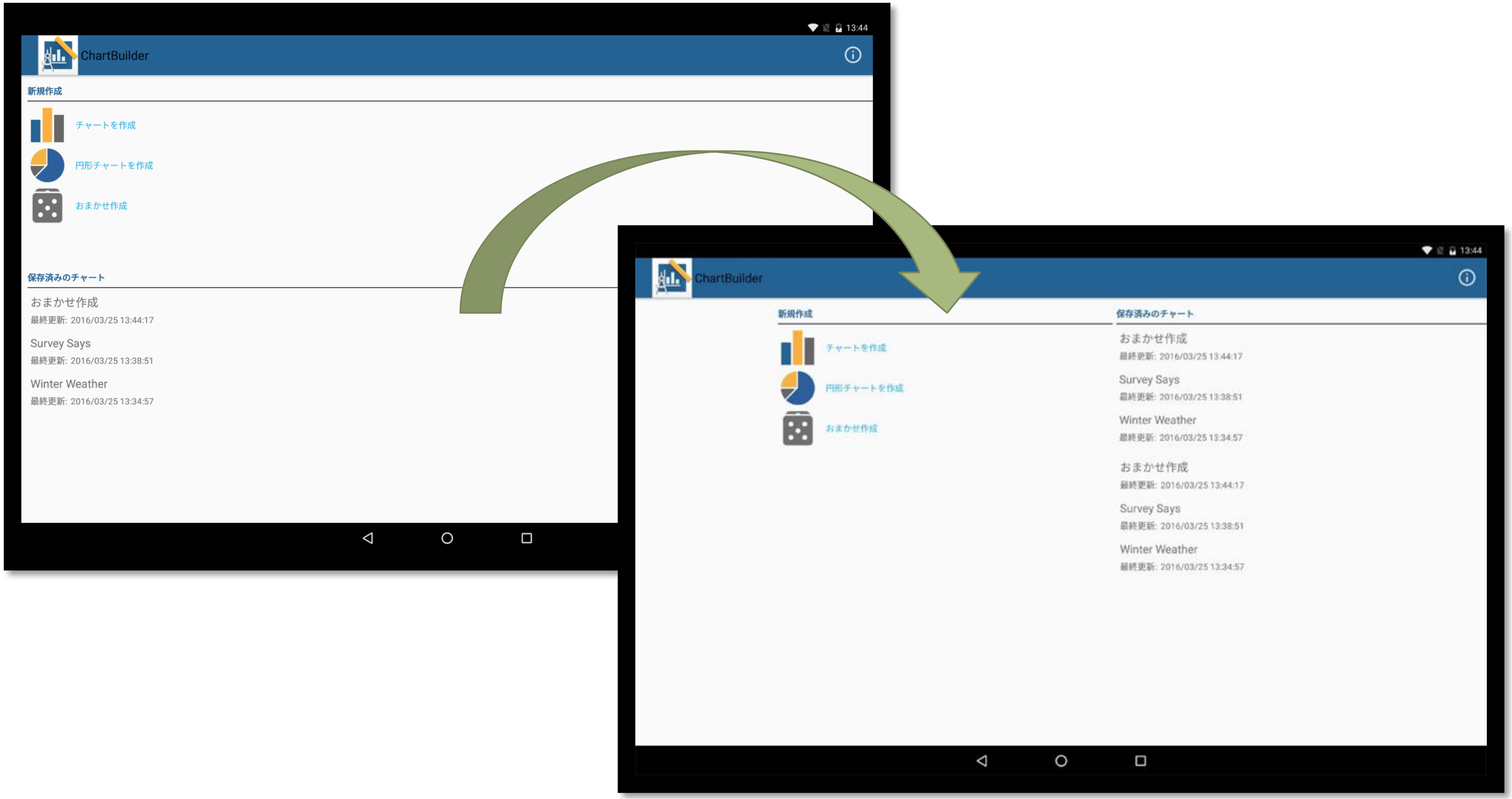
Two Columns





Minimal Left & Right Padding

More Left & Right Padding


ChartBuilder

新規作成

チャートを作成

円形チャートを作成

おまかせ作成

保存済みのチャート

おまかせ作成
最終更新: 2016/03/25 13:44:17

Survey Says
最終更新: 2016/03/25 13:38:51

Winter Weather
最終更新: 2016/03/25 13:34:57

ChartBuilder

新規作成

チャートを作成

円形チャートを作成

おまかせ作成

保存済みのチャート

おまかせ作成
最終更新: 2016/03/25 13:44:17

Survey Says
最終更新: 2016/03/25 13:38:51

Winter Weather
最終更新: 2016/03/25 13:34:57

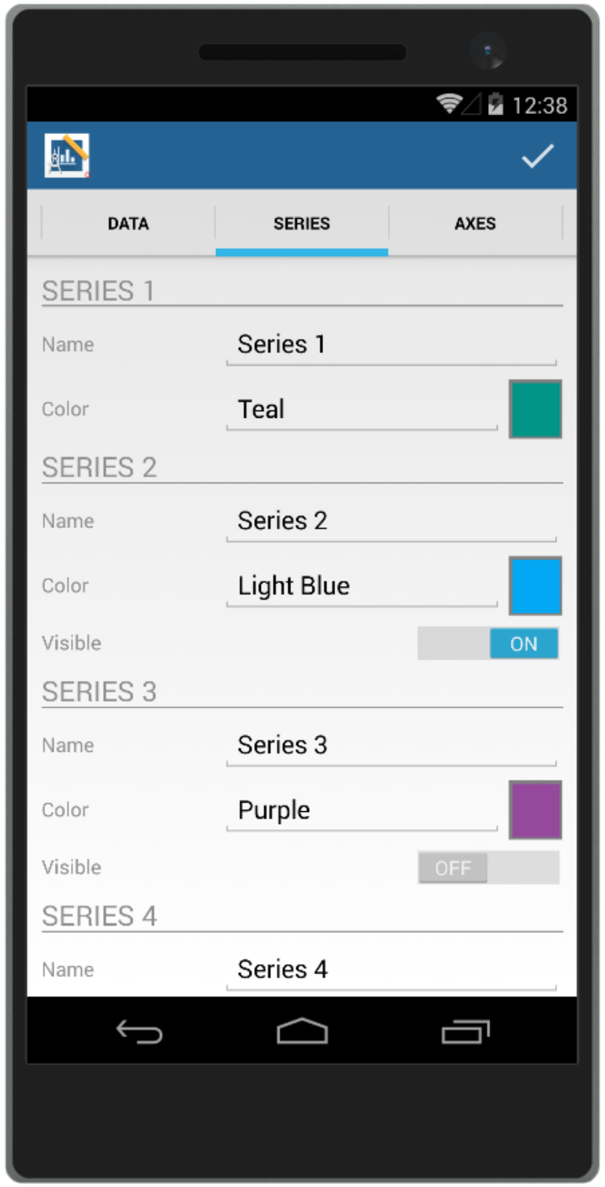
おまかせ作成
最終更新: 2016/03/25 13:44:17

Survey Says
最終更新: 2016/03/25 13:38:51

Winter Weather
最終更新: 2016/03/25 13:34:57

How it’s done

```
if (Xamarin.Forms.Device.Idiom == TargetIdiom.Phone)
{
    // apply phone only code
    mainLayout.Orientation = StackOrientation.Vertical;
}
else if (Xamarin.Forms.Device.Idiom == TargetIdiom.Tablet)
{
    // apply tablet only code
    mainLayout.Orientation = StackOrientation.Horizontal;
}
```


12:38

DATA | SERIES | AXES

SERIES 1

Name: Series 1

Color: Teal

SERIES 2

Name: Series 2

Color: Light Blue

Visible: ON

SERIES 3

Name: Series 3

Color: Purple

Visible: OFF

SERIES 4

Name: Series 4

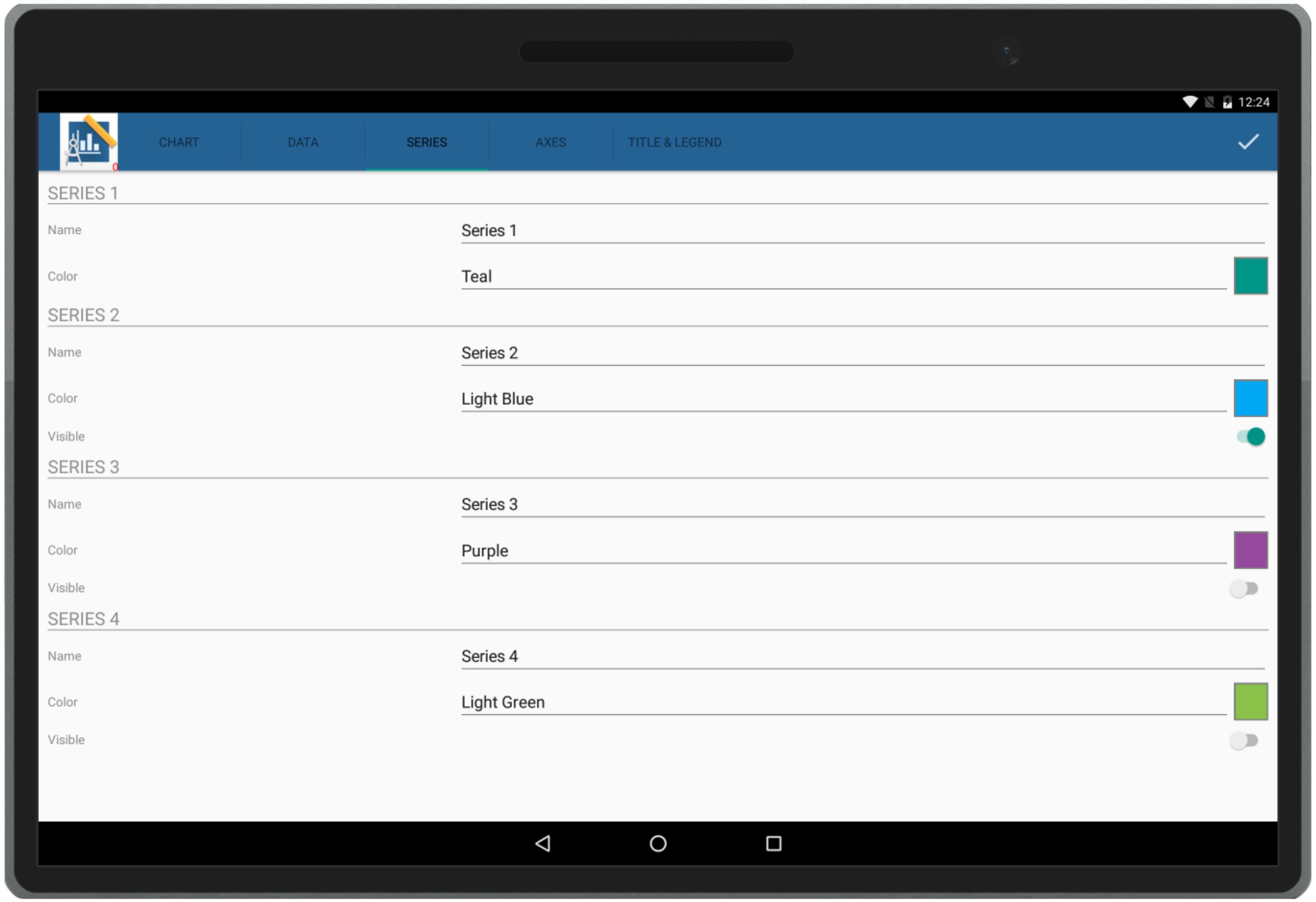

12:24

CHART | DATA | SERIES | AXES | TITLE & LEGEND

SERIES 1

Name: Series 1

Color: Teal

SERIES 2

Name: Series 2

Color: Light Blue

Visible

SERIES 3

Name: Series 3

Color: Purple

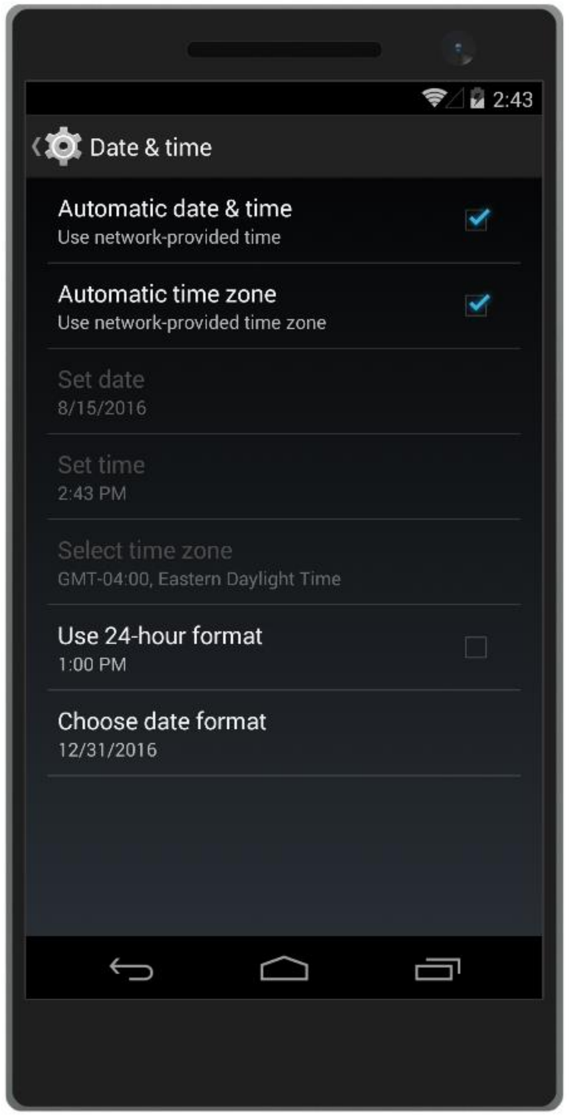
Visible

SERIES 4

Name: Series 4

Color: Light Green

Visible



2:43

Date & time

Automatic date & time
Use network-provided time ✓

Automatic time zone
Use network-provided time zone ✓

Set date
8/15/2016

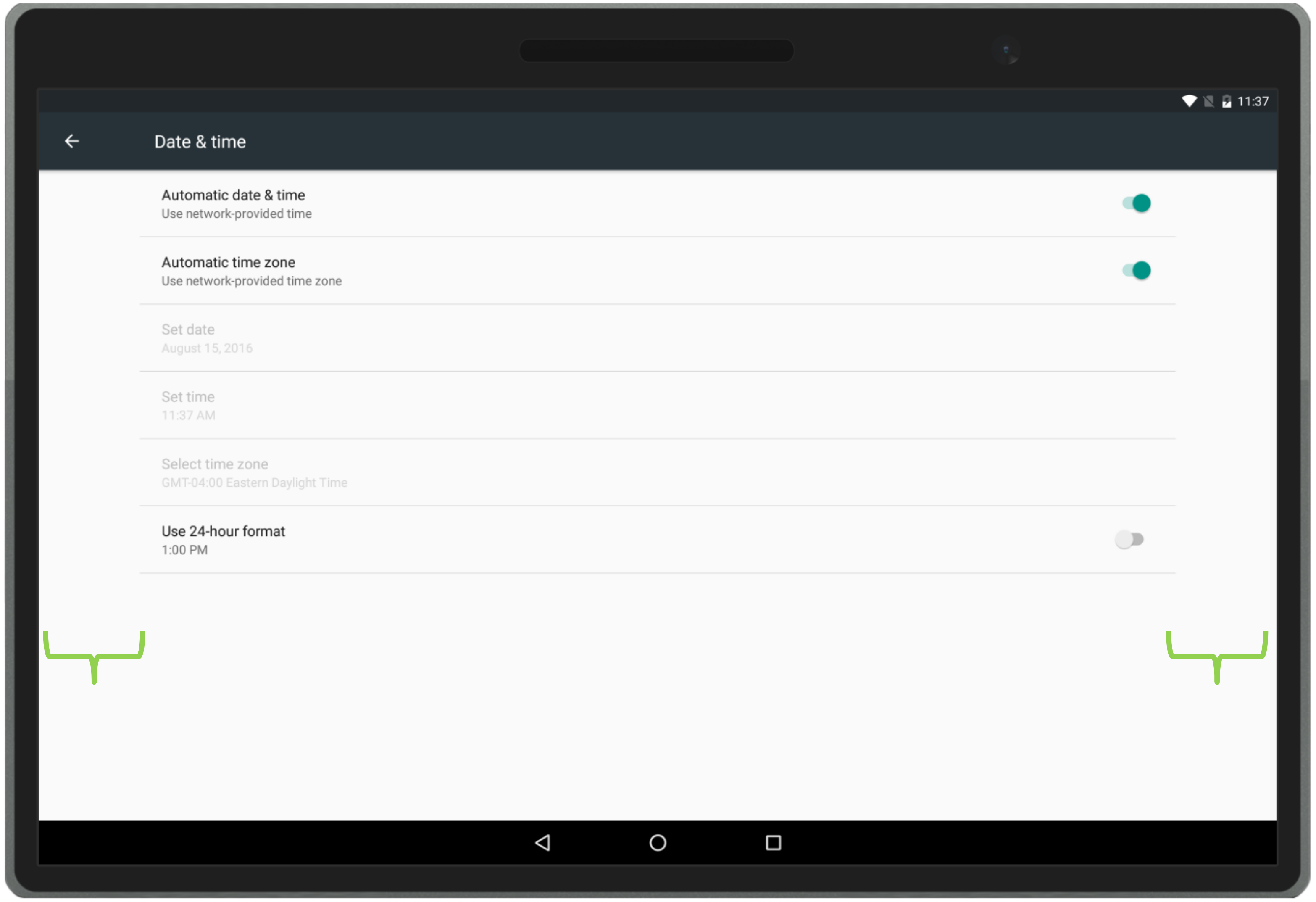
Set time
2:43 PM

Select time zone
GMT-04:00, Eastern Daylight Time

Use 24-hour format
1:00 PM ☐

Choose date format
12/31/2016

Minimal Left & Right Padding



11:37

Date & time

Automatic date & time
Use network-provided time

Automatic time zone
Use network-provided time zone

Set date
August 15, 2016

Set time
11:37 AM

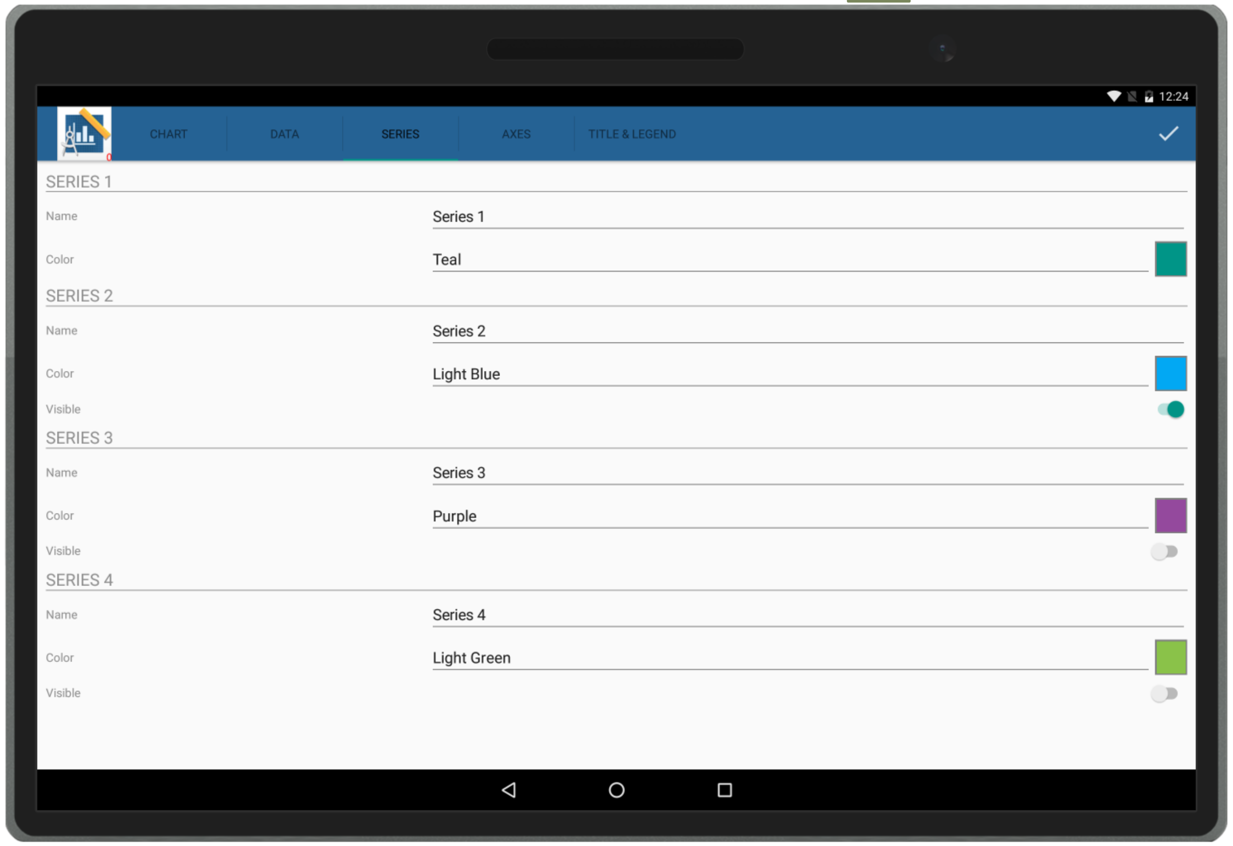
Select time zone
GMT-04:00 Eastern Daylight Time

Use 24-hour format
1:00 PM

More Left & Right Padding

Before

After



SERIES 1

Name	Series 1	
Color	Teal	🟩

SERIES 2

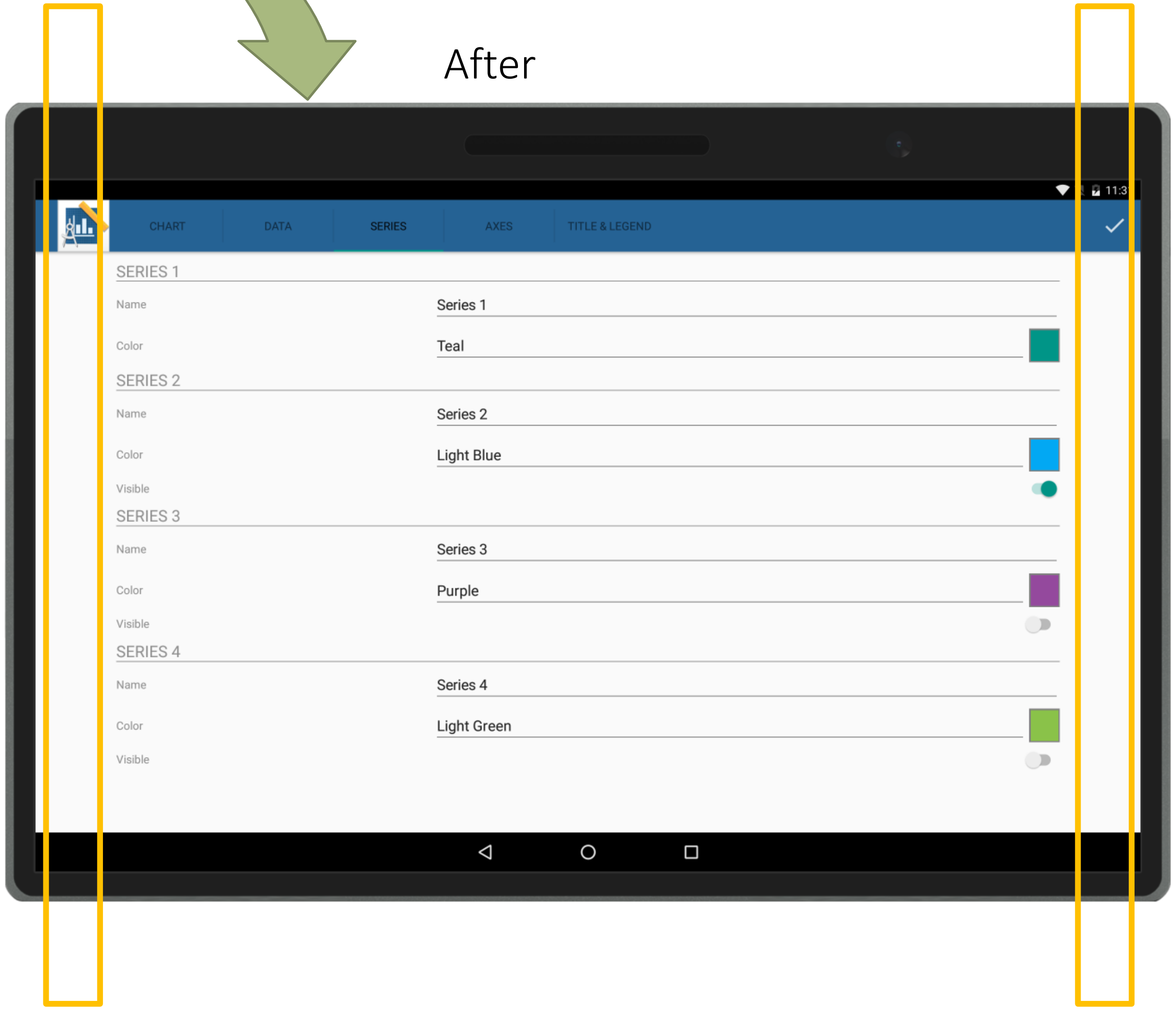
Name	Series 2	
Color	Light Blue	🟦
Visible		●

SERIES 3

Name	Series 3	
Color	Purple	🟪
Visible		○

SERIES 4

Name	Series 4	
Color	Light Green	🟩
Visible		○



SERIES 1

Name	Series 1	
Color	Teal	🟩

SERIES 2

Name	Series 2	
Color	Light Blue	🟦
Visible		●

SERIES 3

Name	Series 3	
Color	Purple	🟪
Visible		○

SERIES 4

Name	Series 4	
Color	Light Green	🟩
Visible		○

Code Demos

Device.Idiom example in code, XAML and global styles.

Styles

XAML feature that allows you to encapsulate a set of values for an element and apply it to all similar elements

Styles avoid repeating XAML and allow reuse of XAML

- **Implicit** styles apply to all elements of the specified type
- **Explicit** styles have a key and only apply to the element instances you specify

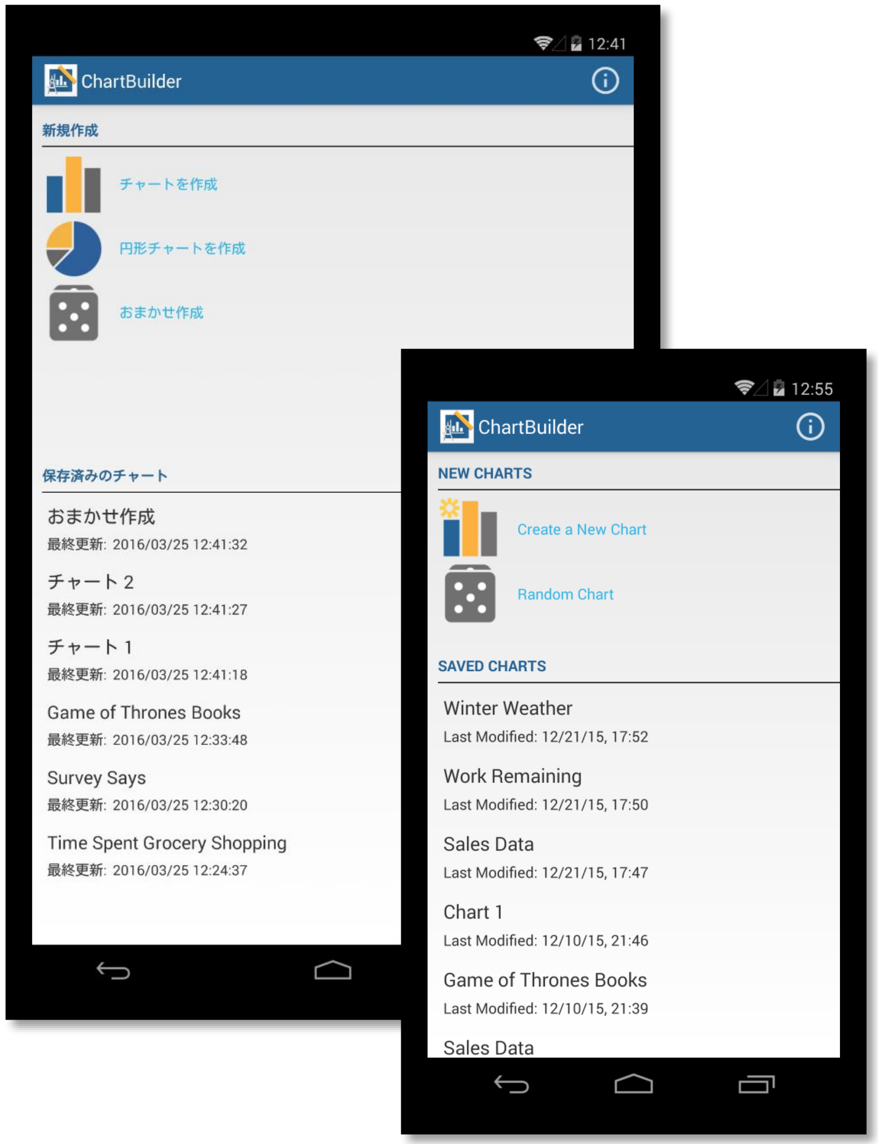
Handling Device Orientation

Listen to your page’s SizeChanged event and compare the page’s width to height.

```
private void MyPage_SizeChanged(object sender, EventArgs e)
{
    if(App.Current.MainPage.Width > App.Current.MainPage.Height)
    {
        // device is landscape
    }
    else
    {
        // device is portrait (or square)
    }
}
```

Portrait (Height > Width)

Landscape (Width > Height)



12:41

ChartBuilder

新規作成

チャートを作成

円形チャートを作成

おまかせ作成

保存済みのチャート

おまかせ作成
最終更新: 2016/03/25 12:41:32

チャート 2
最終更新: 2016/03/25 12:41:27

チャート 1
最終更新: 2016/03/25 12:41:18

Game of Thrones Books
最終更新: 2016/03/25 12:33:48

Survey Says
最終更新: 2016/03/25 12:30:20

Time Spent Grocery Shopping
最終更新: 2016/03/25 12:24:37

12:55

ChartBuilder

NEW CHARTS

Create a New Chart

Random Chart

SAVED CHARTS

Winter Weather
Last Modified: 12/21/15, 17:52

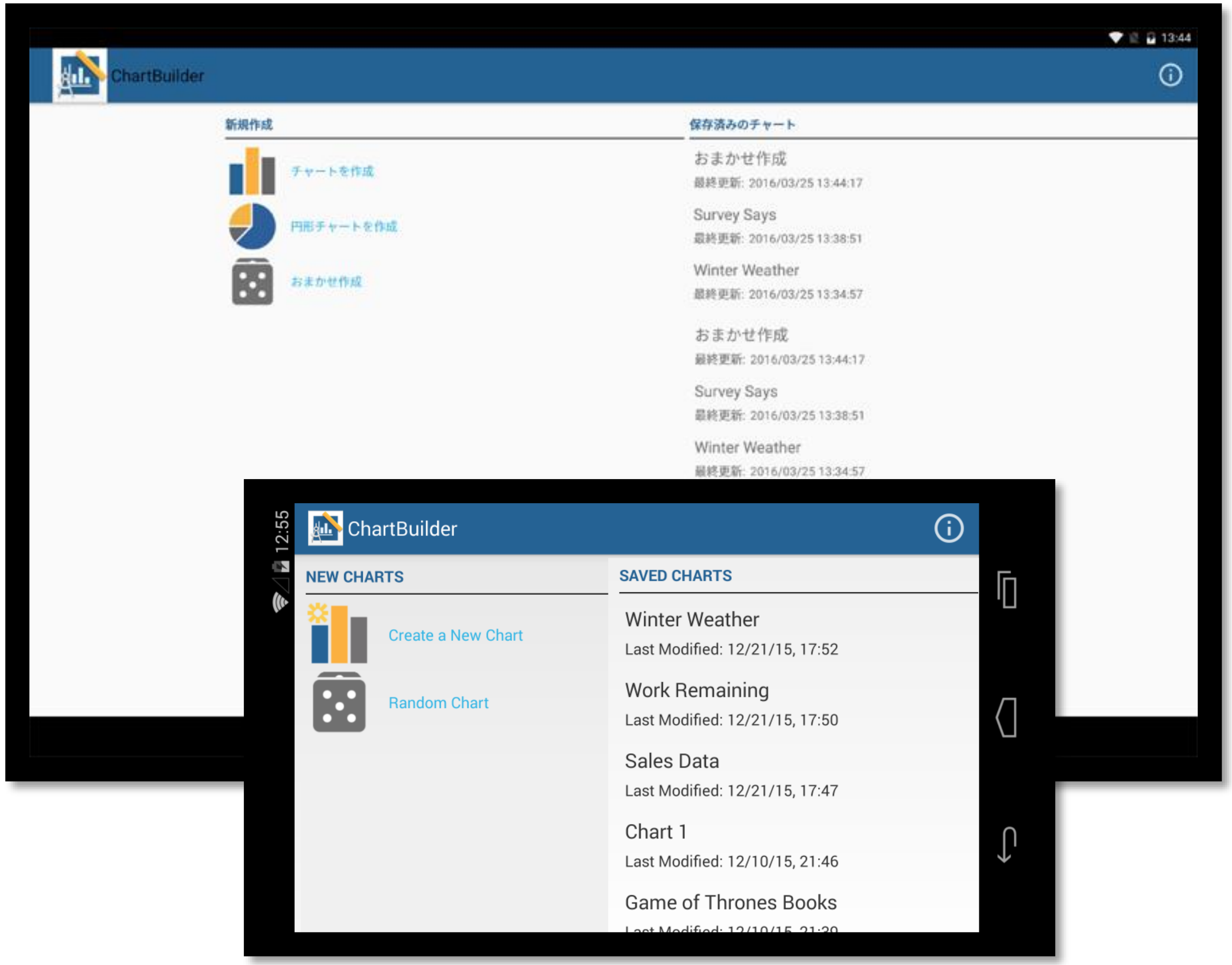
Work Remaining
Last Modified: 12/21/15, 17:50

Sales Data
Last Modified: 12/21/15, 17:47

Chart 1
Last Modified: 12/10/15, 21:46

Game of Thrones Books
Last Modified: 12/10/15, 21:39

Sales Data



13:44

ChartBuilder

新規作成

チャートを作成

円形チャートを作成

おまかせ作成

保存済みのチャート

おまかせ作成
最終更新: 2016/03/25 13:44:17

Survey Says
最終更新: 2016/03/25 13:38:51

Winter Weather
最終更新: 2016/03/25 13:34:57

おまかせ作成
最終更新: 2016/03/25 13:44:17

Survey Says
最終更新: 2016/03/25 13:38:51

Winter Weather
最終更新: 2016/03/25 13:34:57

12:55

ChartBuilder

NEW CHARTS

Create a New Chart

Random Chart

SAVED CHARTS

Winter Weather
Last Modified: 12/21/15, 17:52

Work Remaining
Last Modified: 12/21/15, 17:50

Sales Data
Last Modified: 12/21/15, 17:47

Chart 1
Last Modified: 12/10/15, 21:46

Game of Thrones Books

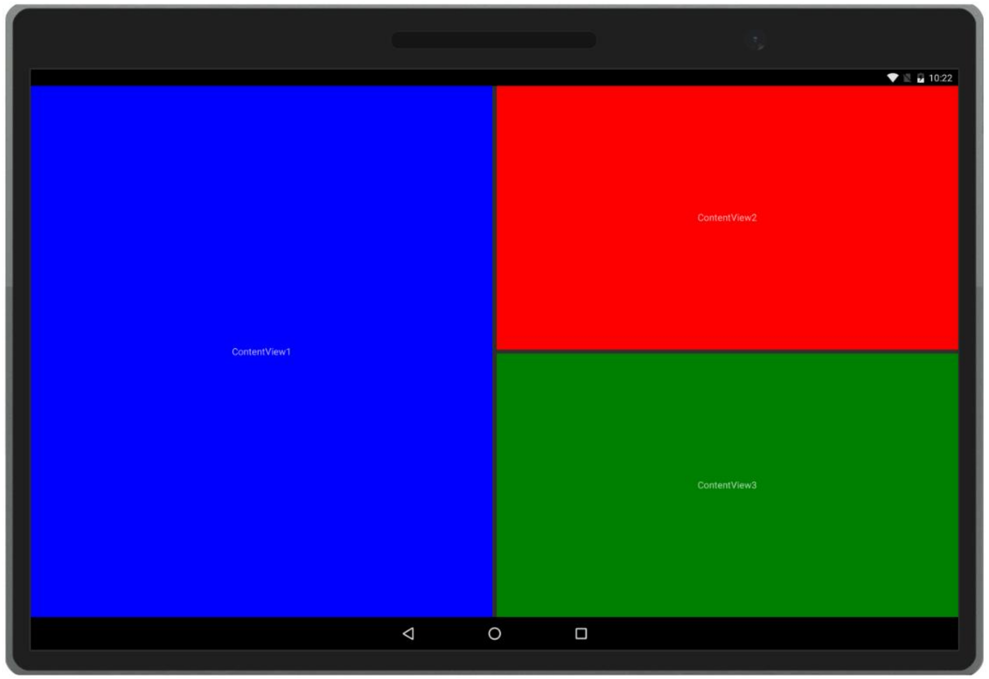
More Adaptive Techniques

ContentViews and MasterDetailPage

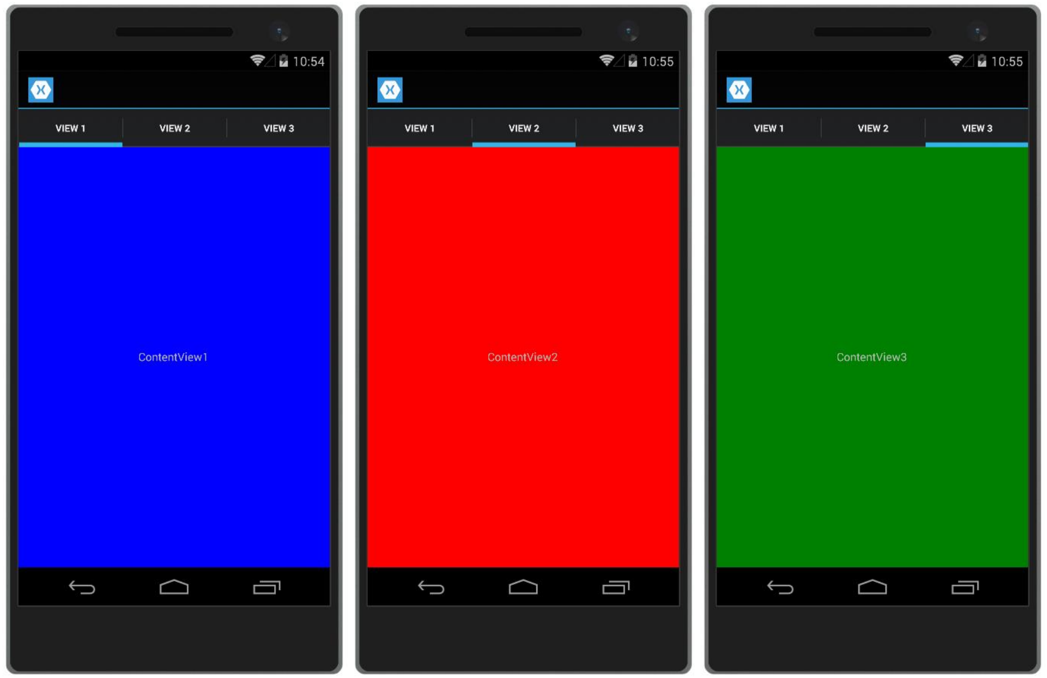
ContentViews

Similar to User Controls, ContentViews allow you to create reusable parts of your Xamarin.Forms UI

One Page



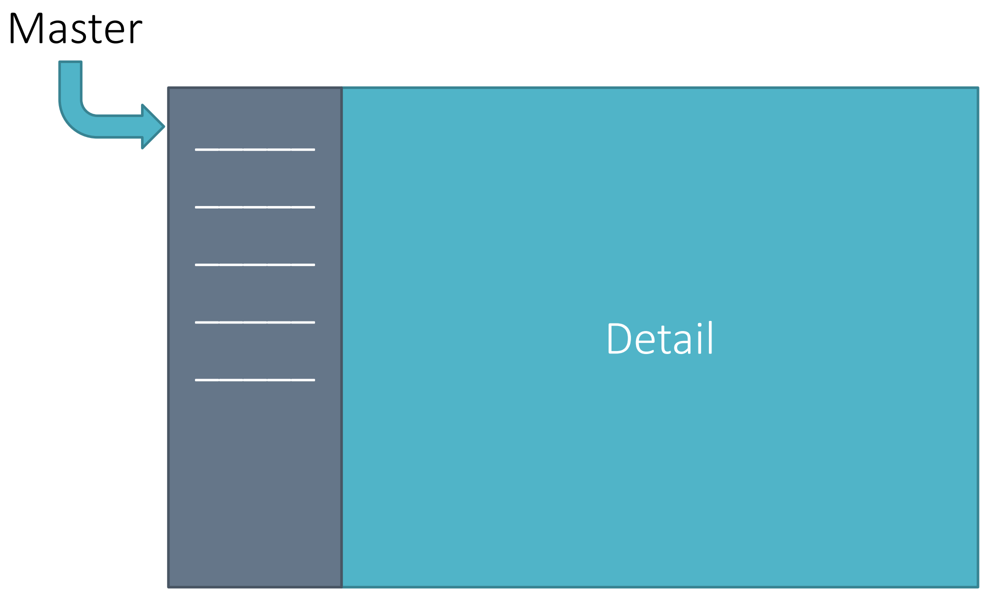
Tabbed Pages

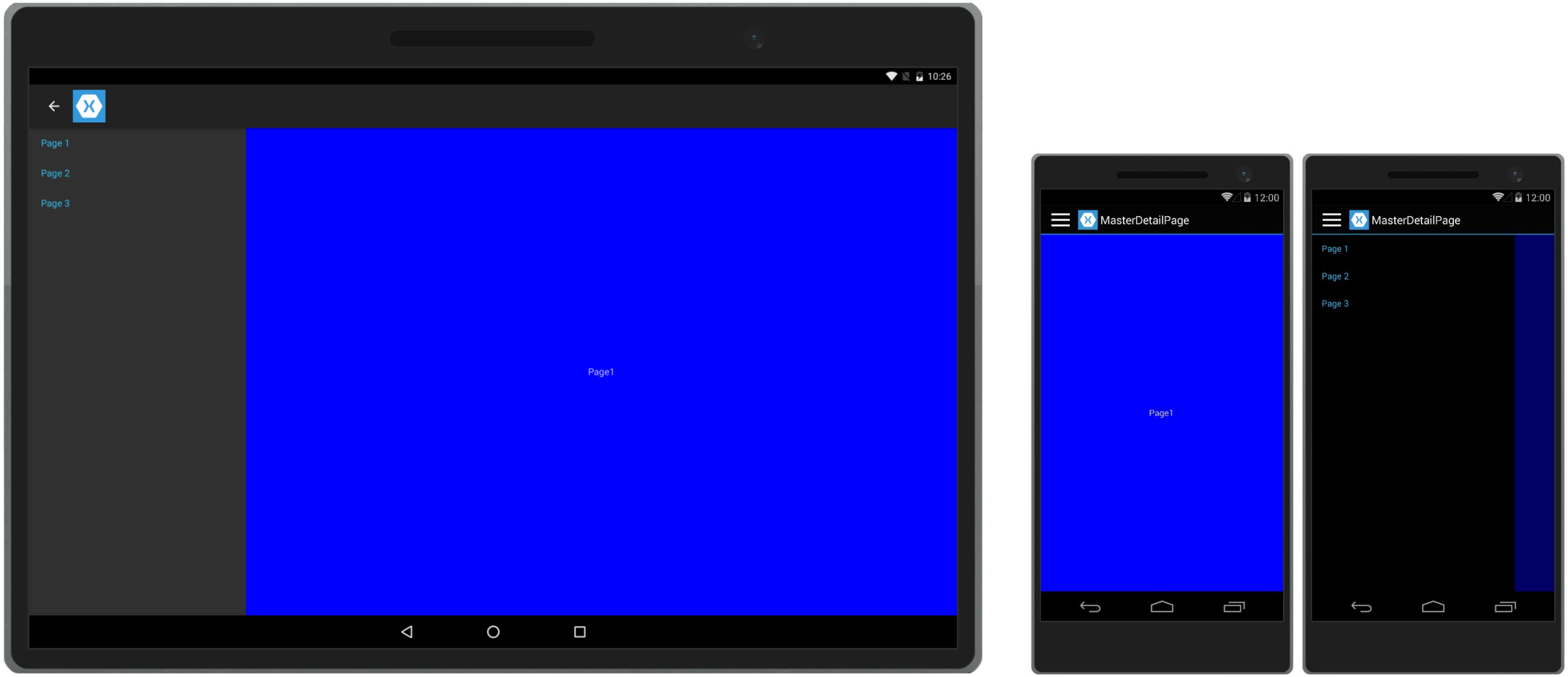


MasterDetailPage

Single page template that manages two pages of content: Master and Detail

Master is often used like a navigation list (hamburger menu)




10:26
Page 1
Page 2
Page 3
Page1
12:00
MasterDetailPage
Page1
12:00
MasterDetailPage
Page 1
Page 2
Page 3

RelativeLayout

Place constraints on children to determine layout

Alternative to Grid/StackLayout – easier to use and read?

```
<RelativeLayout>
  <BoxView Color="Red"
           RelativeLayout.WidthConstraint="{ConstraintExpression Type=RelativeToParent,
           Property=Width, Factor=0.25}"
           RelativeLayout.HeightConstraint="{ConstraintExpression Type=RelativeToParent,
           Property=Height, Factor=0.25}"
           RelativeLayout.XConstraint= "{ConstraintExpression Type=RelativeToParent,
           Property=Width, Factor=0.25}"
           RelativeLayout.YConstraint= "{ConstraintExpression Type=RelativeToParent,
           Property=Height, Factor=0.25}"
           />
</RelativeLayout>
```

Recap

Use basic Grids and StackLayouts

Use ScrollView for overflow

Use Device.Idiom for device-specific changes

You can handle device orientation changes in code comparing Page.Width/Height in the SizeChanged event

Use ContentViews for rearrangeable & reusable parts

MasterDetailPage provides an adaptive navigation layout

Resources

Device.Idiom

https://blog.xamarin.com/bringing-xamarin-forms-apps-to-tablets/

https://developer.xamarin.com/guides/xamarin-forms/platform-features/device/

App.xaml

https://blogs.msdn.microsoft.com/devfish/2016/06/24/global-resources-in-xamarin-forms-no-app-xaml-create-one/

Questions?

greg.lutz@grapecity.com

You can follow my product @GoXuni / www.goxuni.com

Orchestra Tickets Available $220.00

Mezzanine Tickets Available $116.00

Balcony Tickets Available $75.00

Low: 64.22
High: 65.66
Open: 64.32
Close: 65.37
Vol: 34,560

Volume 43,991

First Name	Last Name	Order Total
Oprah	Myers	
Paul	Paulson	
Fred	Orsted	
Andy	Richards	
Karl	Lehman	
Country:	United States	
City:	Los Angeles	
Address:	741 Main ST	
PostalCode:	36079	
Herb	Trask	
Rich	Stevens	
Zeb	Griswold	
Ulrich	Paulson	
Dan	Myers	
Andy	Ambers	

xuni