



Nic Wise

[Fastchicken](#)

nicw@fastchicken.co.nz

@fastchicken

Fast UI Creation with MonoTouch Dialog

Xamarin
EVOLVE 2013

Nic Wise



e. nicw@fastchicken.co.nz
t/a. @fastchicken
b. <http://www.fastchicken.co.nz>

Full source and slides will be on GitHub.

Overview

- Overview - what are we trying to solve here?
- The iOS way - UITableView(Controller)
- The easy way - MonoTouch.Dialog
- Elements
- Styling

Lists





List

Date

Reminders



Buy milk

↗ Leaving: Home



KICKSTARTER

Hi YOLKR Backers :) we hope you are having a great week! It is sure going fast for us!! We wanted to share with you a sneak peek of some packaging concepts and branding that we are playing with, not final but will give you a little insight. We think they are pretty cool, the first image is more real



SCIENCE: Ruining Everything Since...

1,708% Funded

Update #39 · 14 days ago

I call that a win. My sincere thanks to everyone. Every time we put out a new big thing, I think "this is the time nobody wants it!" Thank you for amply assuaging that concern. It means a lot to me, and it means this is gonna be a really freakin' thick book. I really am doing my best here every day

11:13

Cancel Income Save

Description Photo printing

Client Stuart Lodge 

Date 12 Apr 2013

Amount 45.60

Method Card 

Invoice payment 

Reference INV667

Date 12 Apr 2013

Receipt image 
Image has been selected

Notes 

11:24

Trip

Start 12 Apr 2013

End 19 Apr 2013

Duration 8 days

Budget 1000.00

Home Currency GBP 

USD 1.54 USD = 1 GBP 

Add new currency 

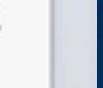
Download latest rates

11:18

Cancel Edit Done

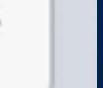
Austin things to do

Location

Starts Sat, 13 Apr 2013 

Ends Sat, 13 Apr 2013

Repeat Never 

Invitees None 

Alert 10 minutes before (23:50) 

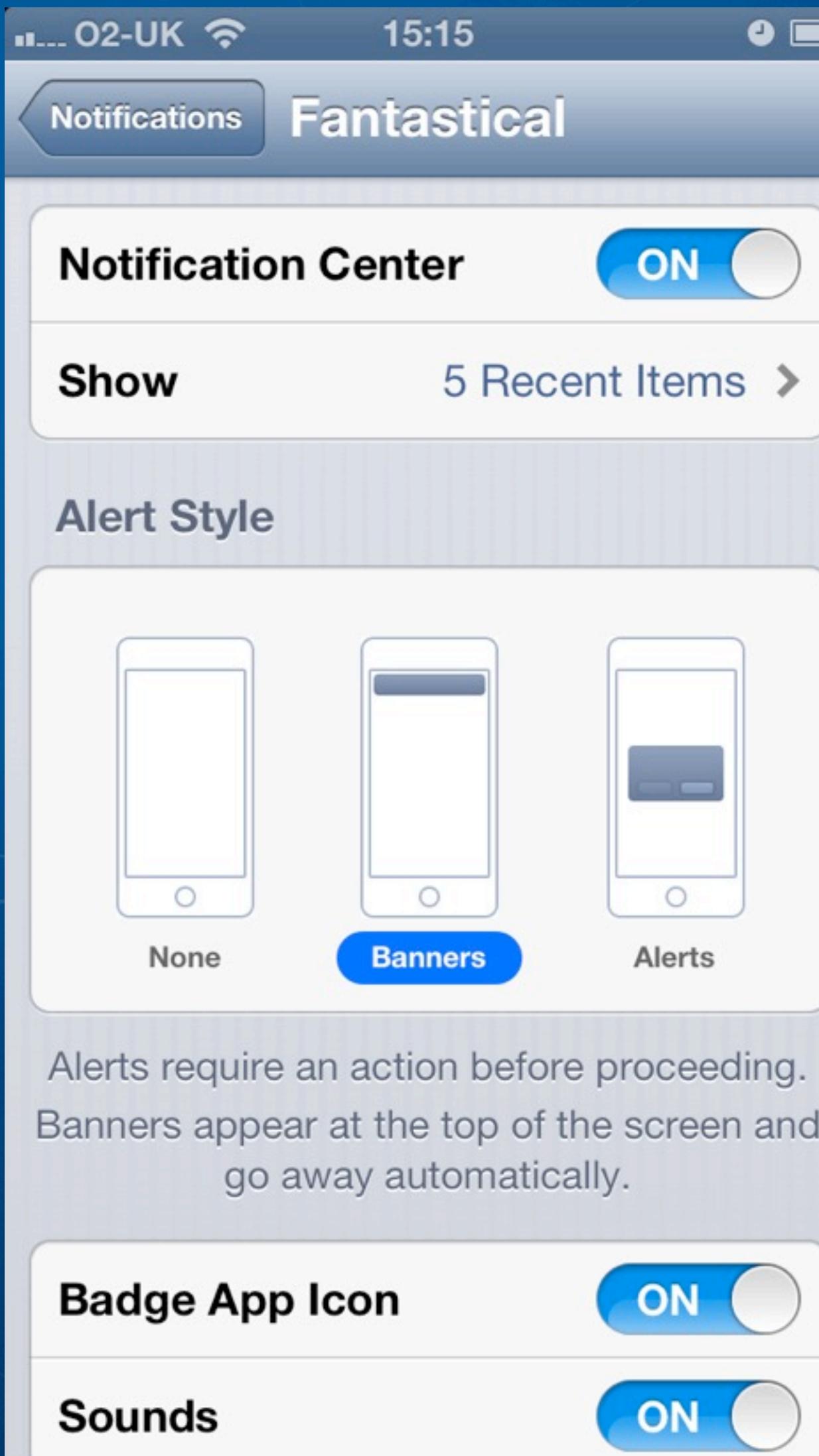
Calendar Nic Wise 

Availability Free 

ToyJoy
Bat bridge - congress Bridge

Lists are the core mobile interaction

UITableView(Controller)



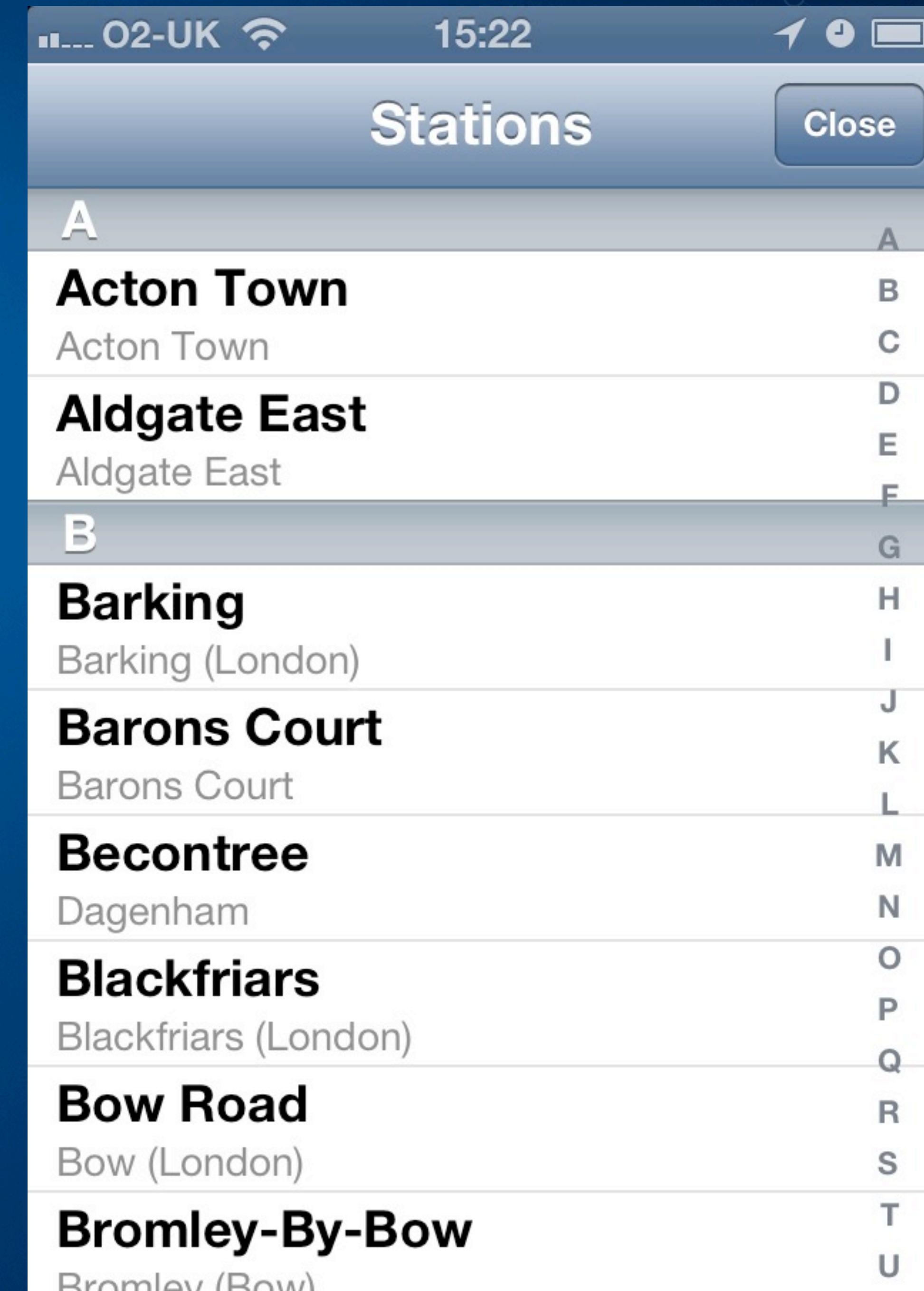
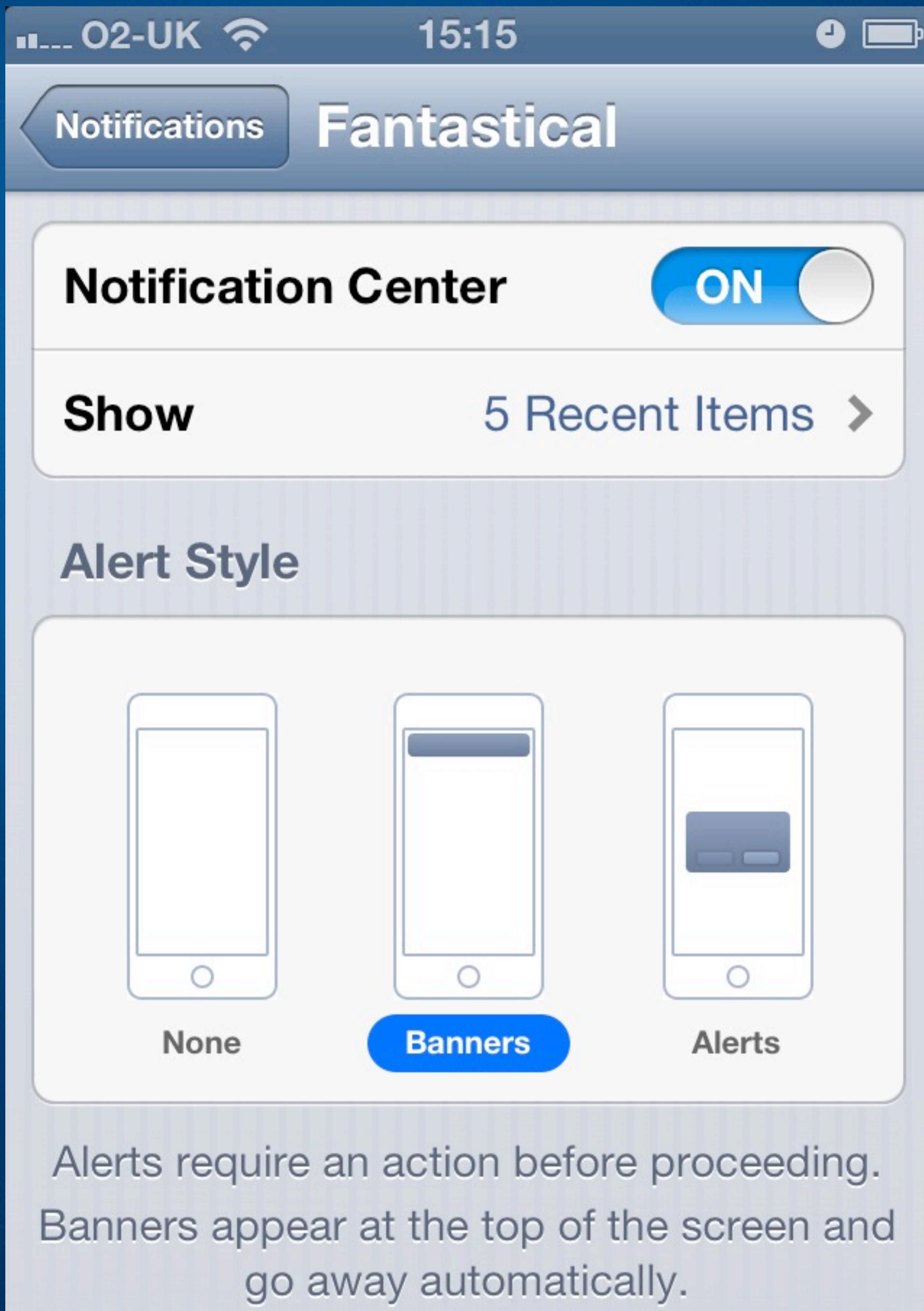
Navigation Bar

Section header

Cell

Section footer

Section





UITableView

- Based around a callback / delegate model
- “How many sections do you have”
- “Give me cell 4 for section 2”
- A cell is just a view container
- Cells are recycled
- You have to maintain your own model

**Seriously powerful.
A little tedious to work with.**

That's all a bit repetitive...

“Although the widget is pretty powerful, creating UIs with it is a chore and a pain to maintain.

... my fingers developed calluses, and at night I kept thinking that there should be a better way.”

Miguel de Icaza

<http://tirania.org/blog/archive/2010/Feb-23.html>

MonoTouch.Dialog

- Make it easy to create forms and list-based views
- Make it flexible enough to do anything that UITableView can do
- Flip the API model from callback to model-driven

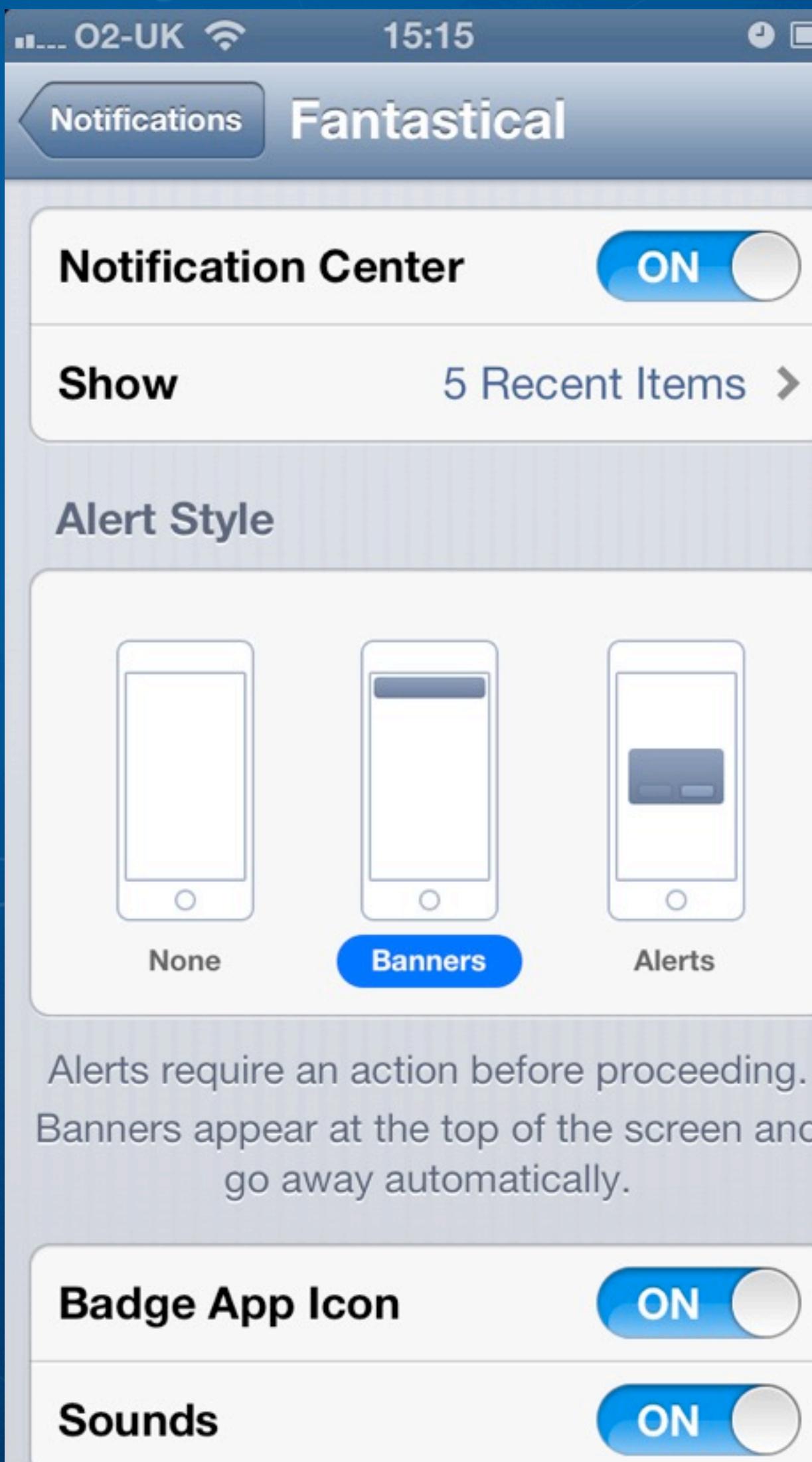
Concepts

- `DialogViewController`
- `Element`
- `RootElement`
- `Section`

Building with Elements

- TheRootElement is at the top of the tree, it contains...
 - ... Section(s), which contain ...
 - ... Elements (which wrap cells)

MonoTouch.Dialog



RootElement

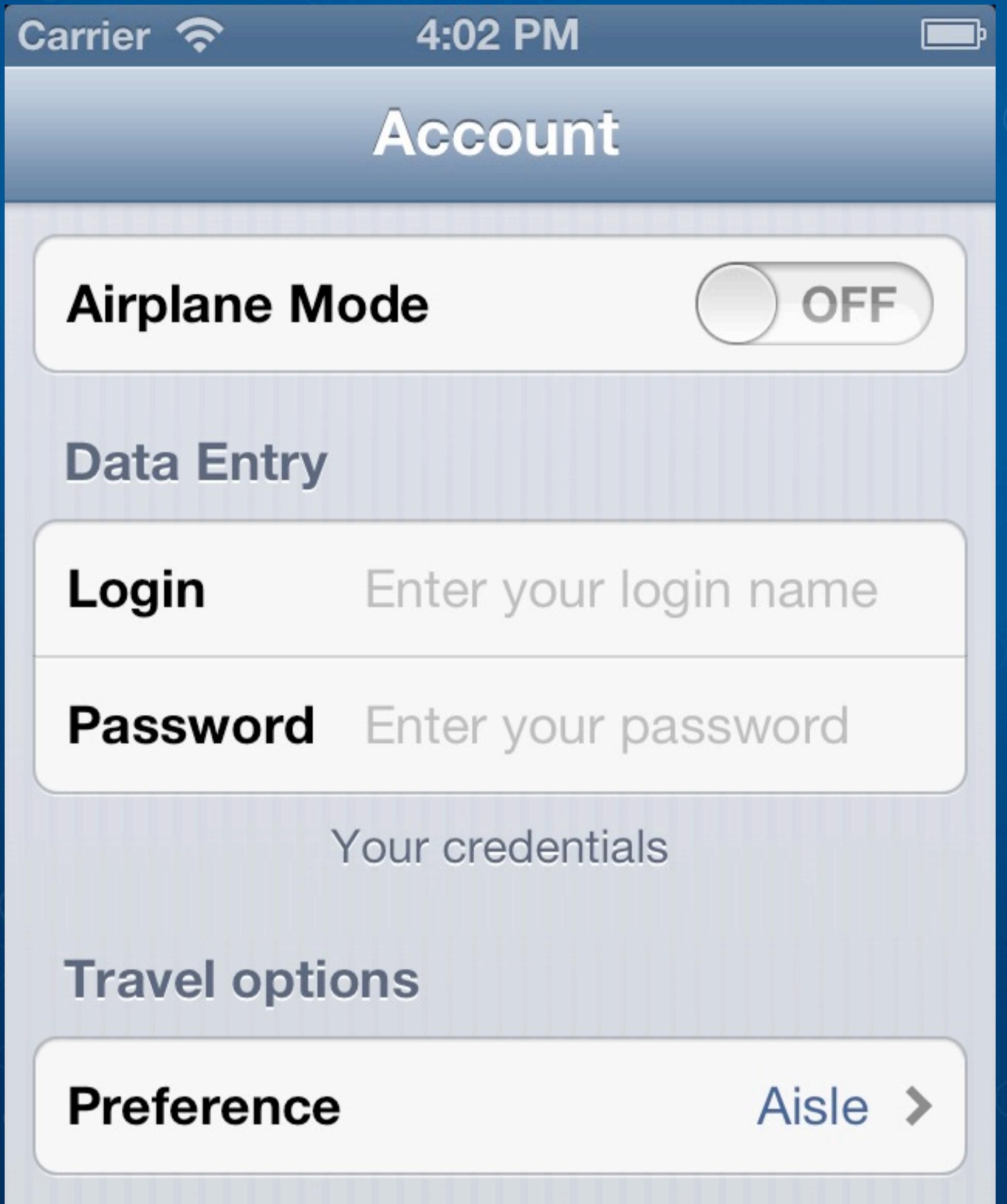
Section Element
Element

Section Element

Section Element
Element

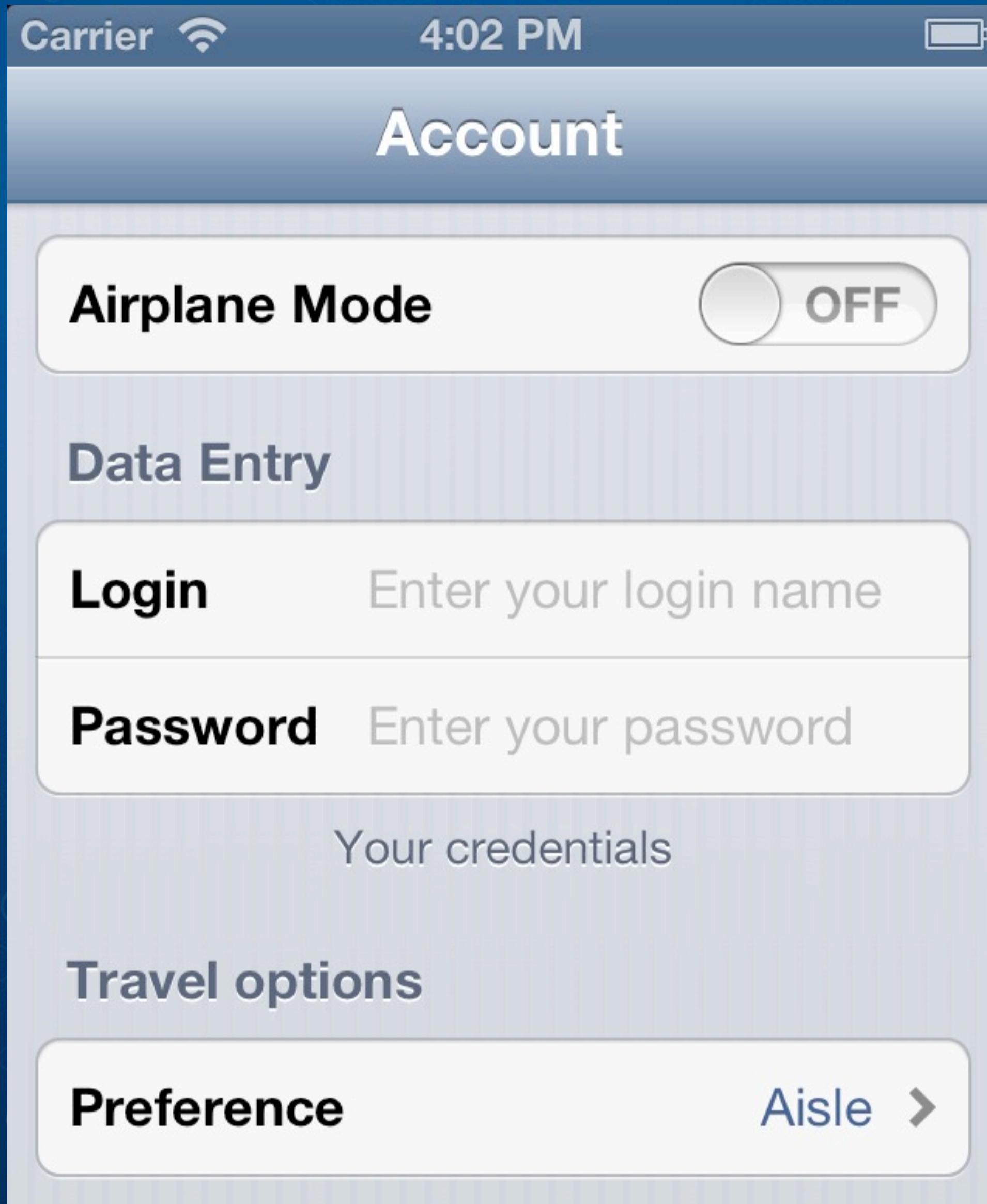
**So, how do we put this all
together?**

Manually building a form



```
public override void LoadView ()  
{  
    base.LoadView ();  
  
    RadioGroup preferenceGroup = new RadioGroup (0);  
  
    Root = new RootElement ("Account")  
    {  
        new Section  
        {  
            new BooleanElement("Airplane Mode", false)  
        }, new Section("Data Entry", "Your credentials")  
        {  
            new EntryElement("Login", "Enter your login name", ""),  
            new EntryElement("Password", "Enter your password", "", true)  
        }, new Section()  
        {  
            new RootElement("Preference", preferenceGroup)  
            {  
                new Section()  
                {  
                    new RadioElement("Aisle"),  
                    new RadioElement("Window"),  
                    new RadioElement("Centre")  
                }  
            }  
        };  
    };  
}
```

Reflection API



```
public class AccountInfoModel
{
    [Section]
    public bool AirplaneMode;

    [Section("Data Entry", "Your Credentials")]
    [Entry("Enter your email")]
    public string Login;

    [Caption("Password")]
    [Password("Enter your password")]
    public string Password;

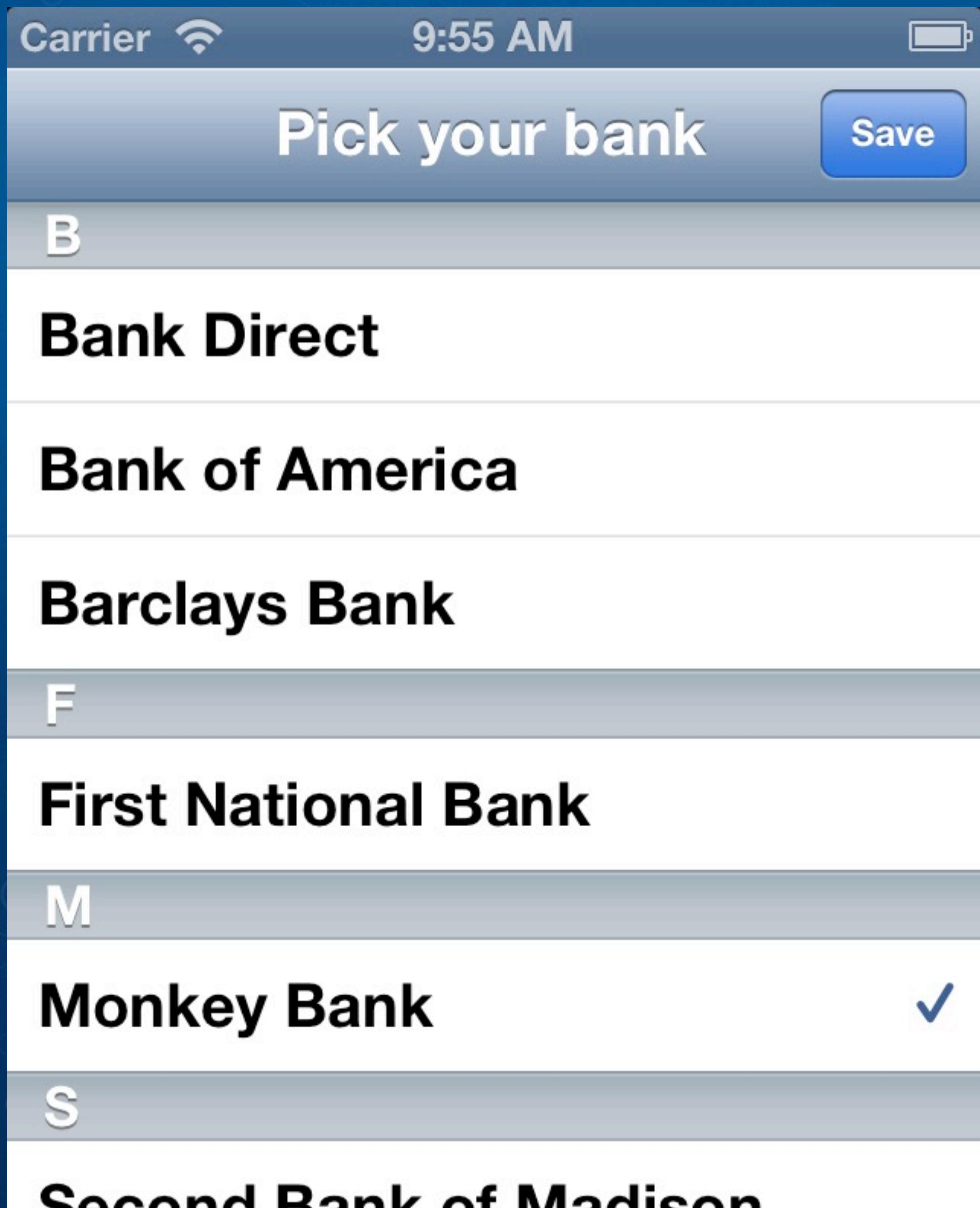
    [Section("Travel Options")]
    public SeatPreference Preference;

    [Section]
    [OnTap("Tapped")]
    public string PressMe;

    public void Tapped()
    {
        var alert = new UIAlertView ("Tapped", "Yes, you tapped it",
                                   null, "Ok");
        alert.Show ();
    }

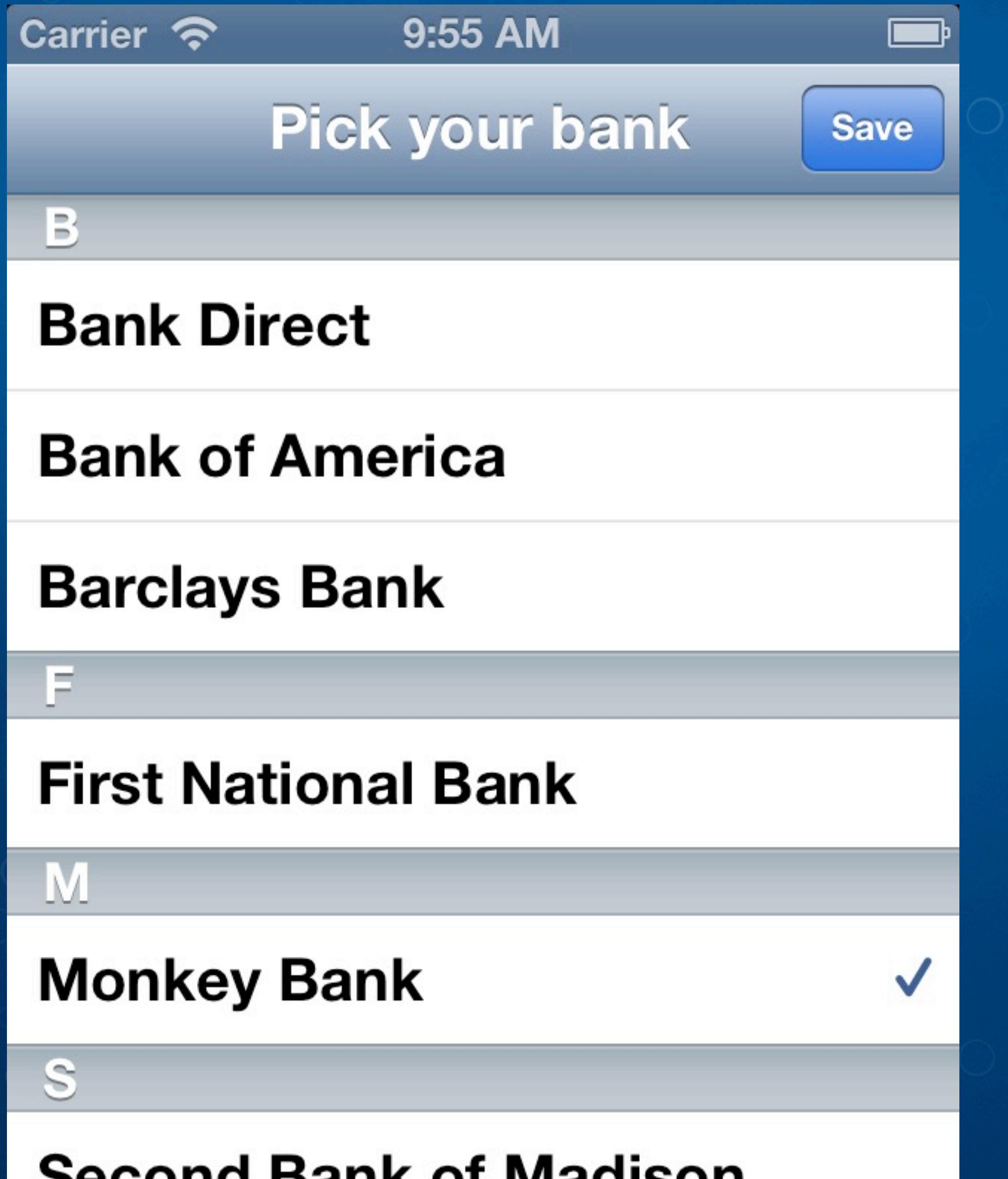
    public enum SeatPreference
    {
        Aisle,
        Window,
        Centre
    }
}
```

Lists



```
RadioGroup bankSelection;  
  
public override void LoadView ()  
{  
    base.LoadView ();  
  
    bankSelection = new RadioGroup (0);  
  
    Section section = null;  
    string lastLetter = "";  
    Root = new RootElement("Account", bankSelection);  
  
    foreach(var name in data)  
    {  
        string currentFirstLetter = name.Substring(0,1);  
        if (currentFirstLetter != lastLetter)  
        {  
            if (section != null && section.Count > 0) Root.Add (section);  
  
            lastLetter = currentFirstLetter;  
            section = new Section(currentFirstLetter);  
        }  
  
        section.Add (new RadioElement(name));  
    }  
  
    if (section != null && section.Count > 0) Root.Add (section);  
}
```

LINQ API



```
RadioGroup bankSelection;
List<string> dummyData;

public override void LoadView ()
{
    base.LoadView ();

    Root = new RootElement("Account", bankSelection)
    {
        from name in dummyData
        group name by name.Substring(0,1) into grp
        orderby grp.Key
        select new Section(grp.Key)
        {
            from elementValue in grp
            select new RadioElement(elementValue)
        }
    };
}
```

Mix and Match
This is an && decision, not an ||



TheRootElement tree is Mutable

Elements

StringElement Value

StyledStringElement VALUE

StyledStringElement VALUE

 **ImageStringElement**

HtmlElement >

StyledMultilineElement
Foo
Bar
Baz
Buzz

EntryElement hello
Hello x

LoadMoreElement

 Loading!



RootElement Item 2 >

Nic Wise 4:29 PM
evolve@xamarin.com
All the good things about Evolve, in 5 >
one email

CheckboxElement ✓

RadioElement ✓

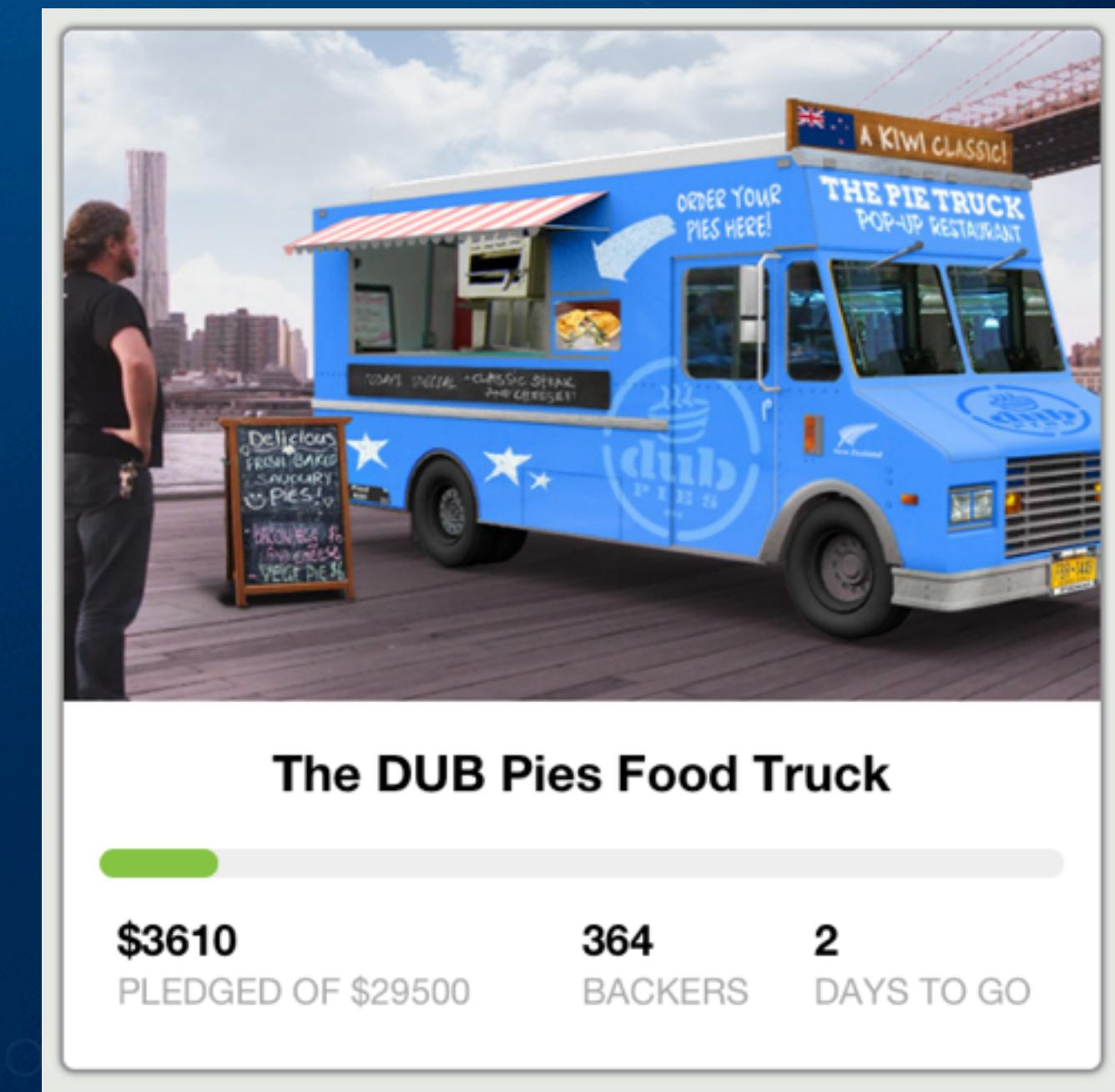
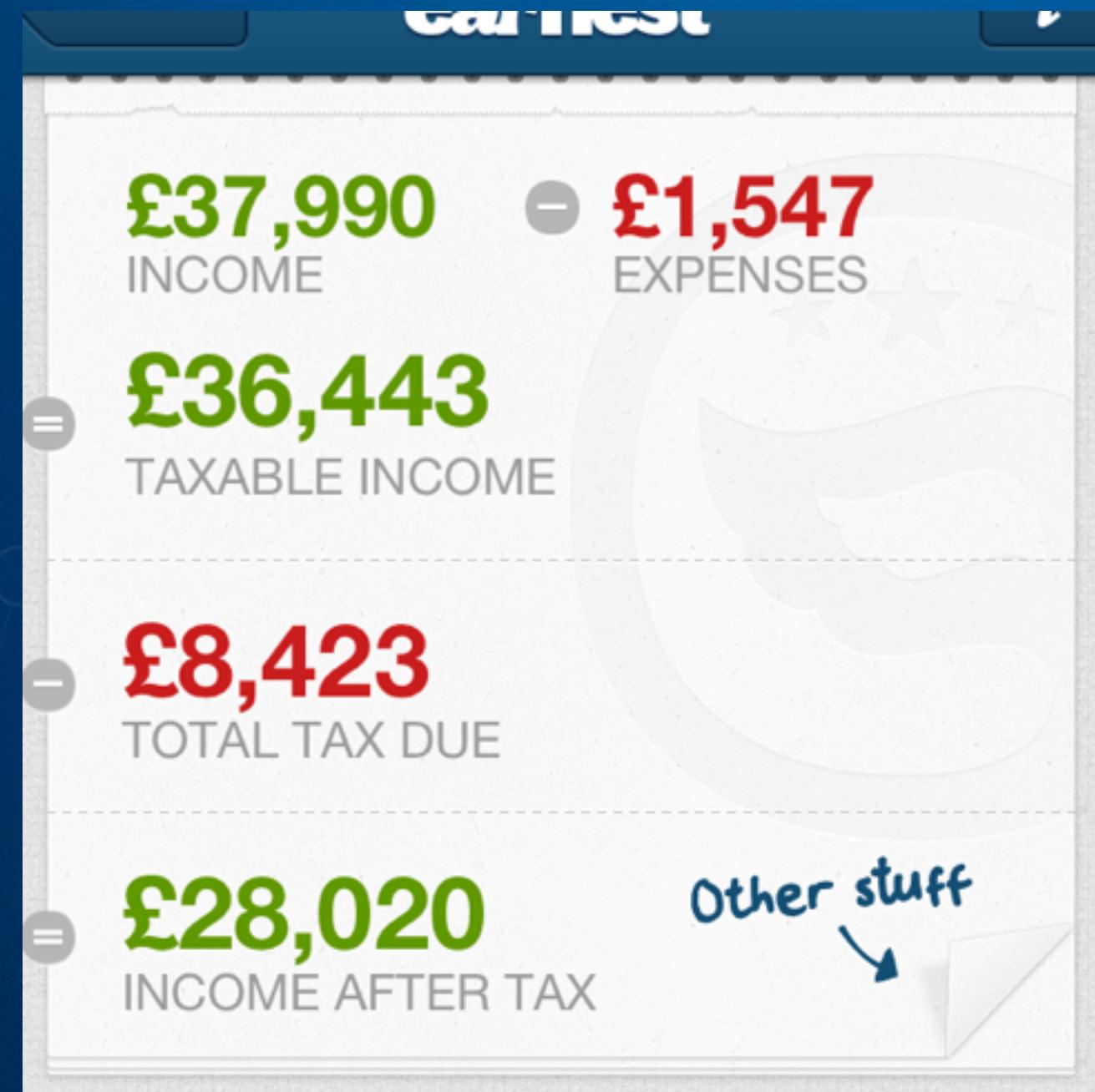
BooleanElement ON

TimeElement 4:10 PM >

DateTimeElement 4/6/13 3:41... >

DateElement Apr 6, 2013 >

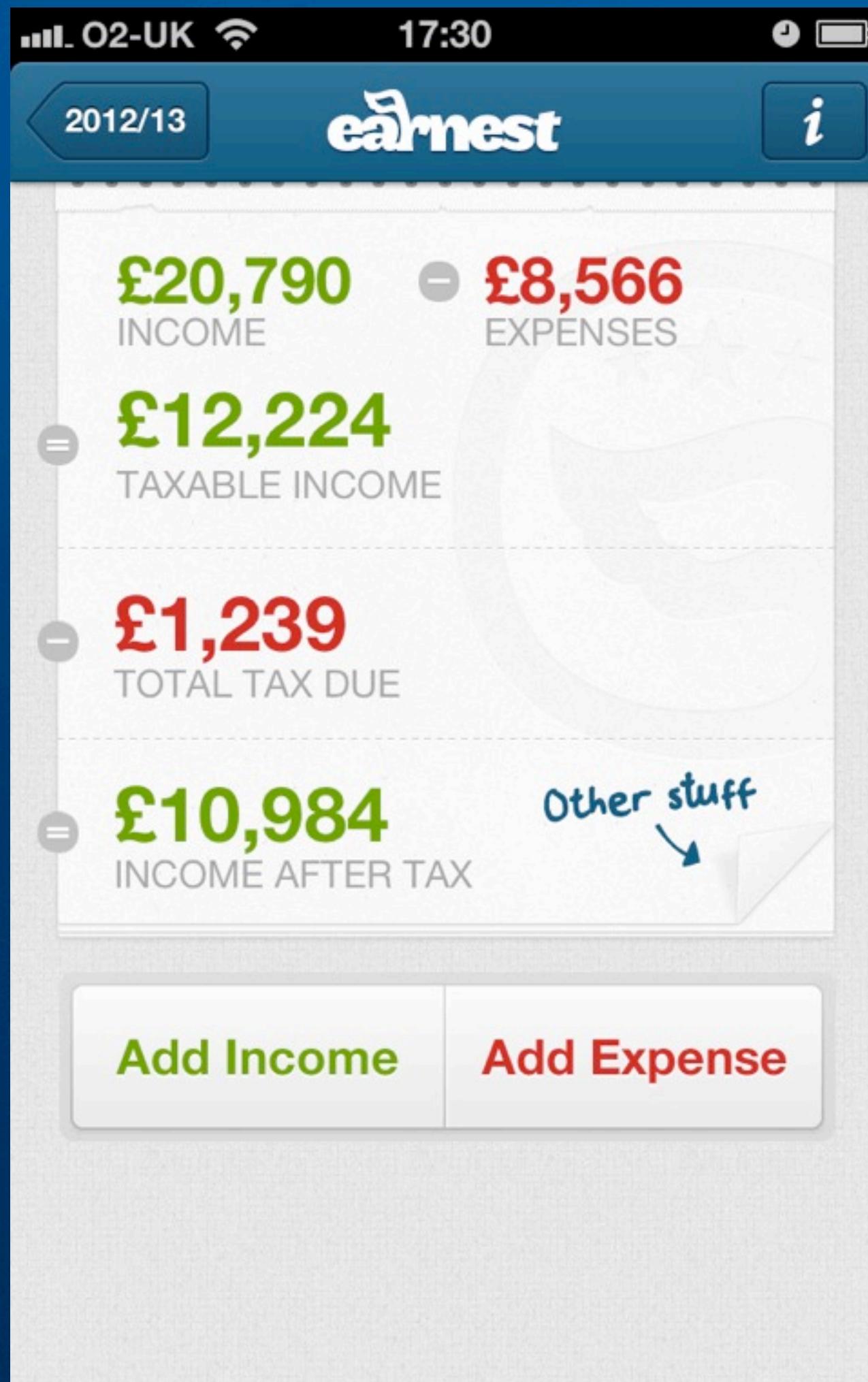
Building custom elements is easy



Building Custom Elements

- Customize an existing Element
- Use UIViewElement
 - Easy! Not as resource friendly
- Use a UITableViewCell descendant
 - Slightly harder. Total control.

Custom Elements

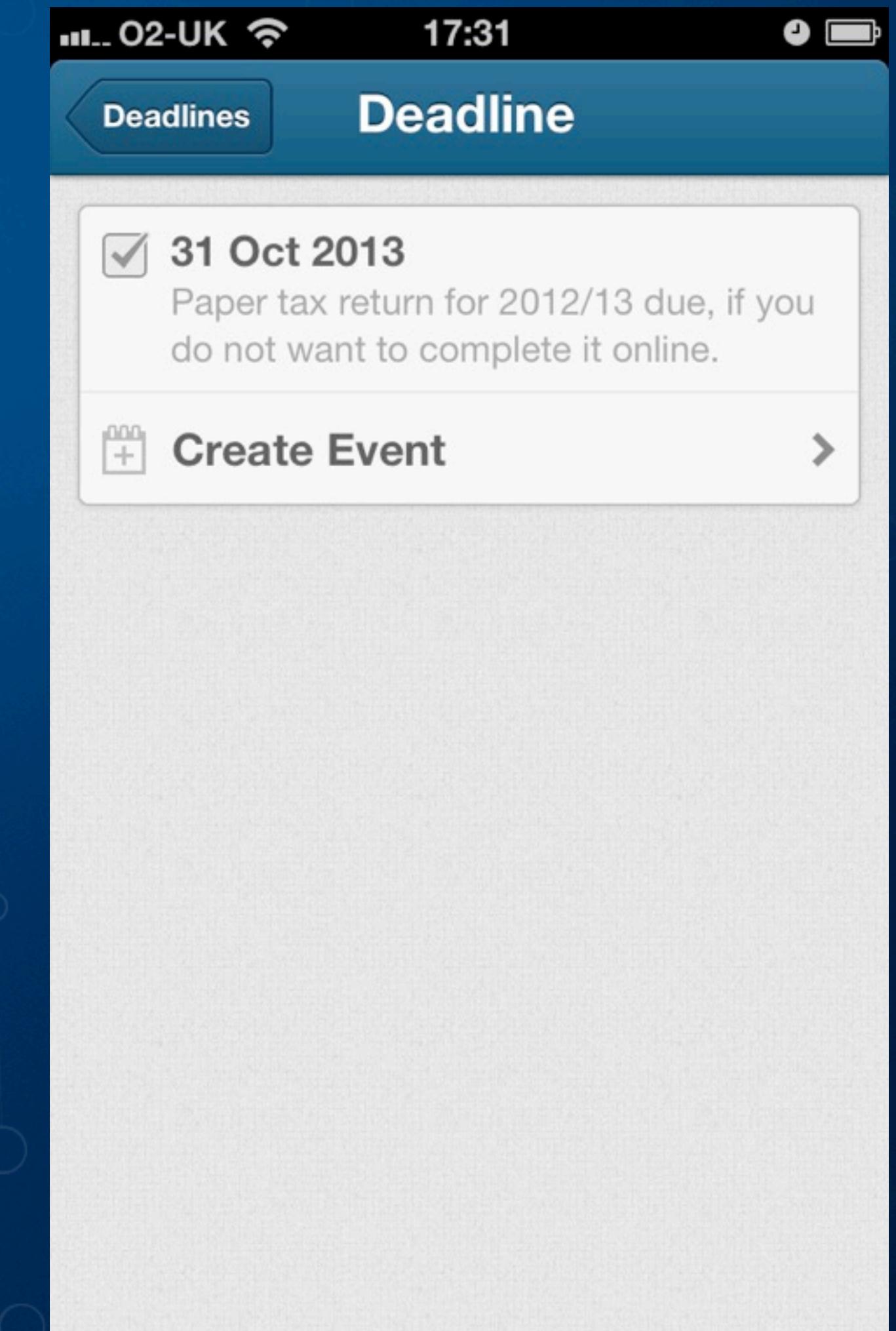


The screenshot shows the '2012/13 Records' screen. It includes a search bar and a date filter set to '18 Nov 2012'. Below this, a list of expenses is shown:

Printing costs	£100.27
Sold 3 photos	£21.18
Hokkaido print	£22.83
Poster print	£38.90
Postcard	£2.34

Below the expense list, another date filter is set to '17 Nov 2012', and the list continues:

Camera batteries	£62.13
Poster	£35.21
Postcards x2	£10.10

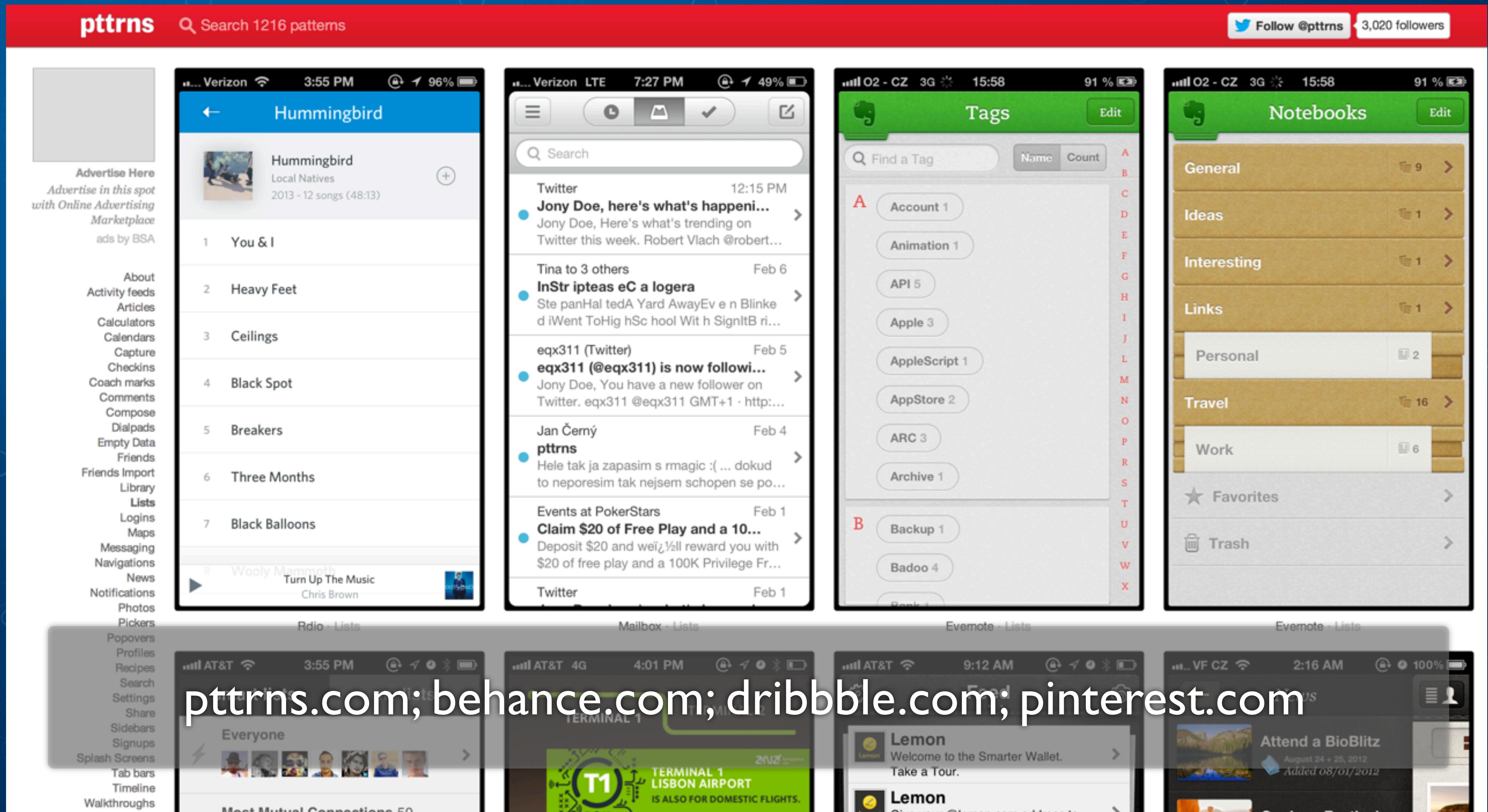




Deep Customization

- Changing the background image for all Grouped cells
- May require a change to the base MonoTouch.Dialog source
- How / where

Inspiration



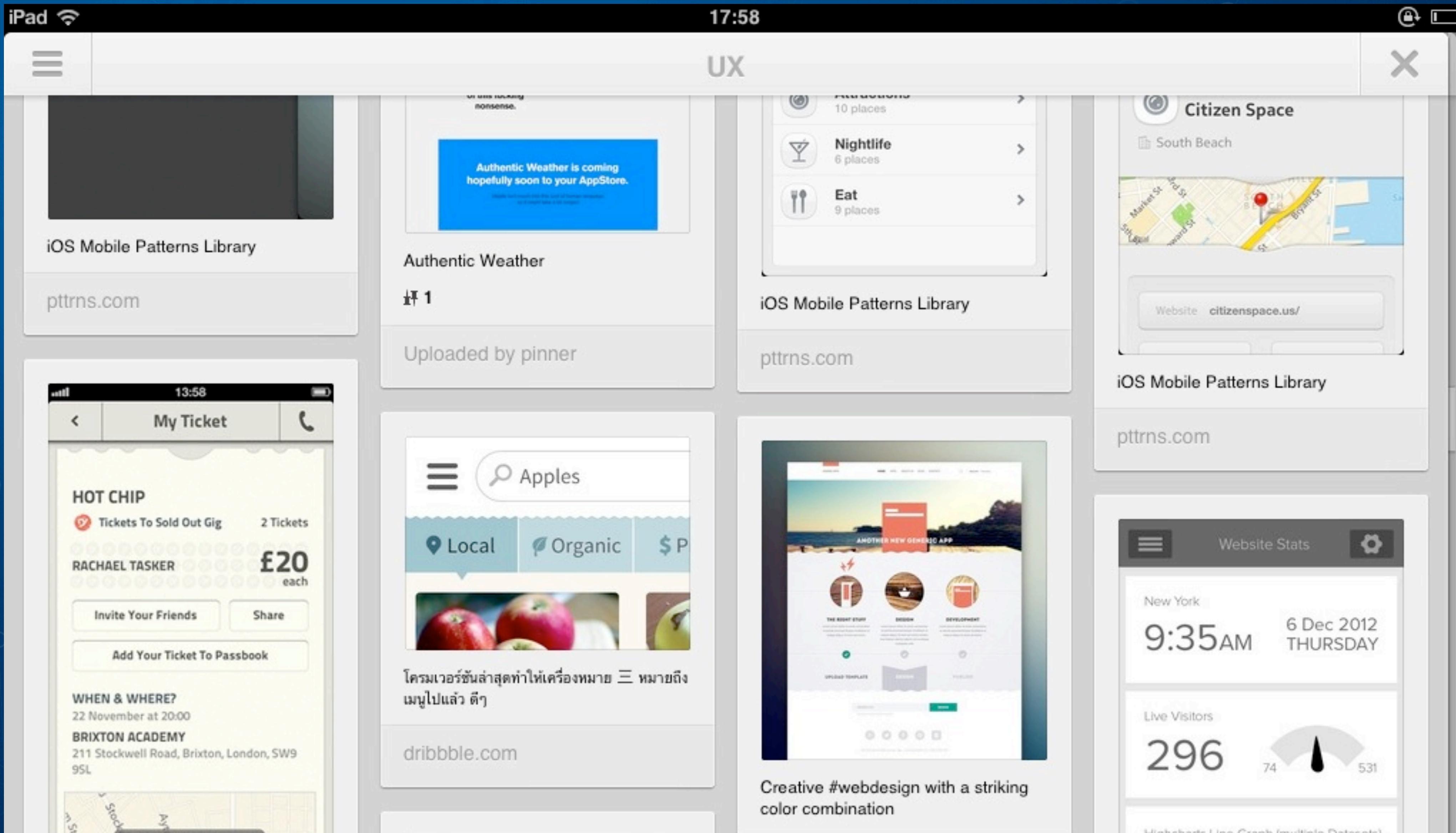
Maintaining your own version

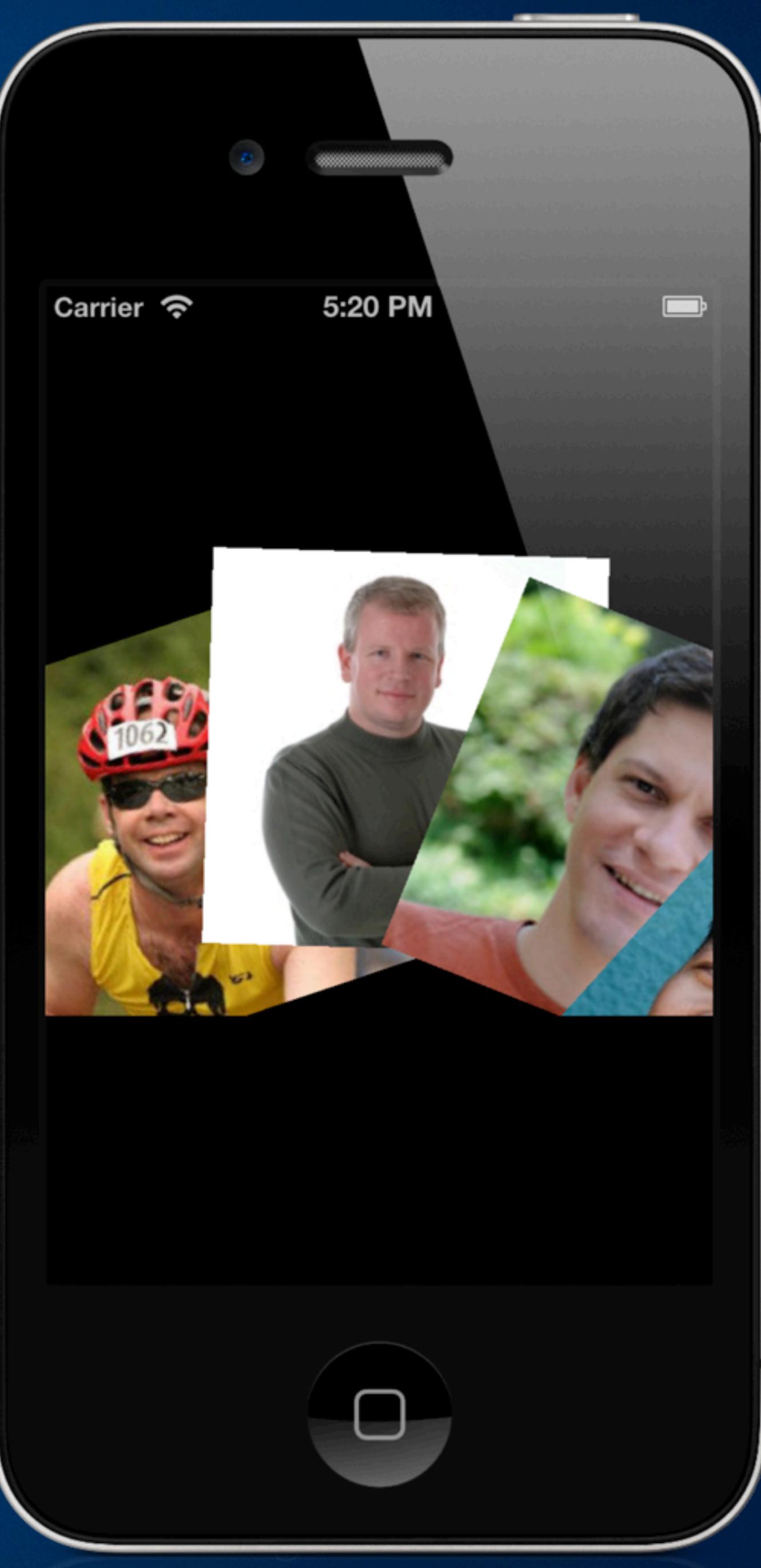
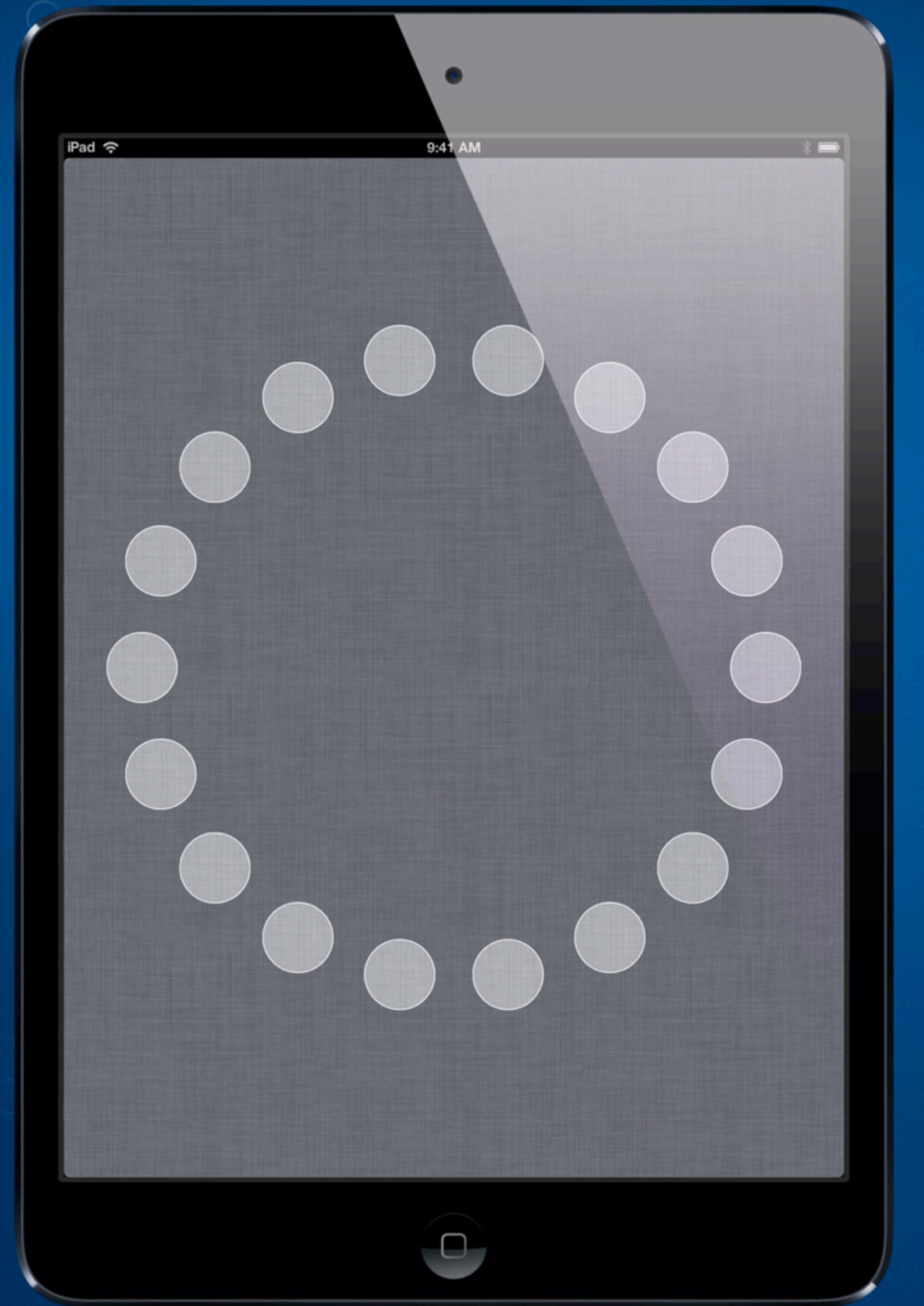
- Avoid it if you can (makes life a lot easier)
- Try to pull + merge as frequently as possible
- Absolutely get the code down and look at it - understand how the framework works

MonoTouch.Dialog

- You can build anything with it you can do with a UITableView
- Very quick to understand and use
- Very mature, still maintained, ships with Xamarin.iOS
- Full customization of any part of the table if you are prepared to mess with the source a little

UICollectionView





Q&A

THANK YOU



Resources

- Code and sides:
 - <https://github.com/nicwise/EvolveMonoTouchDialog>
- MonoTouch.Dialog:
 - <https://github.com/migueldeicaza/MonoTouch.Dialog>

UICollectionView Resources

- http://docs.xamarin.com/guides/ios/user_interface/introduction_to_collection_views
- <https://github.com/mpospese/CircleLayout>
- <https://github.com/slodge/RotatingCollectionView>
- <https://github.com/chiahsien/UICollectionViewWaterfallLayout>
- <https://github.com/schwa/Coverflow>

Design Inspiration

- <http://www.mobiledesignpatterngallery.com>
- <http://pttrns.com>
- <http://dribbble.com>
- <http://behance.com>

