

Windows Phone 8 End to End

Twitter: @arielbh

Ariel Ben Horesh

www.arielbh.com

Co-founder - CodeValue

26/2/2013



Steps for an App

1. Idea – Zillion of them, most of them stupid. Can't help you there.
2. Design – Panorama/Pivot, Gaming, HTML based.
3. Architecture – MVVM, IoC, Frameworks, Layers, PCL, WAMS
4. Development – XAML, async-await, Background Tasks, Phone Integration, Store integration, WP Toolkit.
5. Testing – Unitests! (improving)
6. Store – It builds? Ship it.

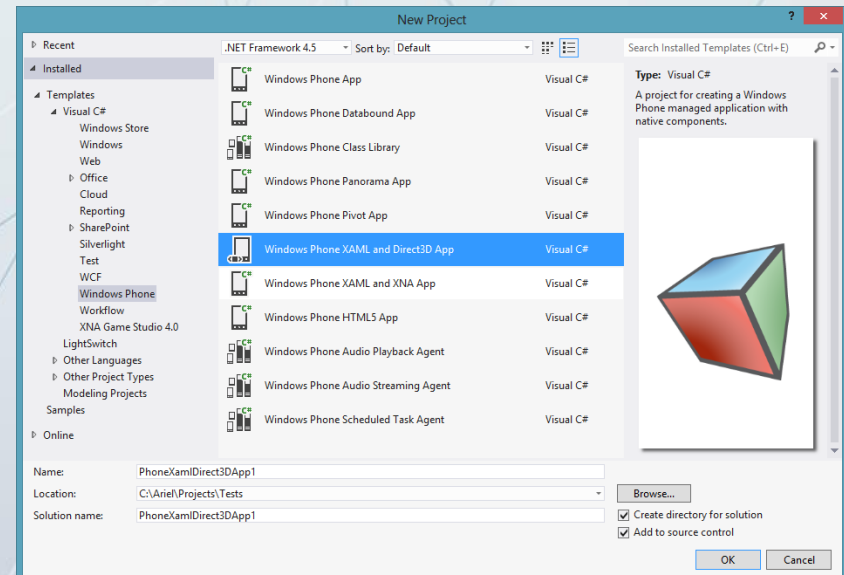
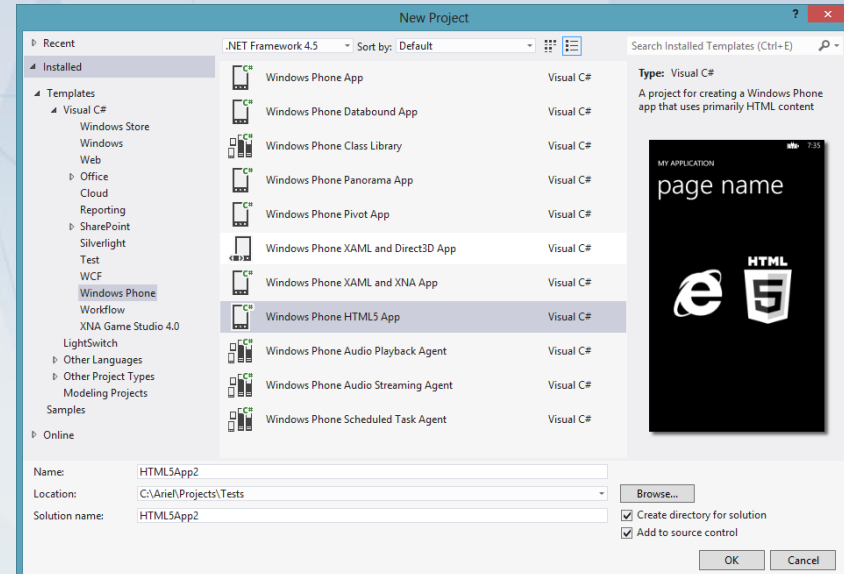
Idea

- My Stupid idea for today:
 - My Friend just got Married yesterday (that's why I'm hangovered here).
 - A Wedding ... That's useful.



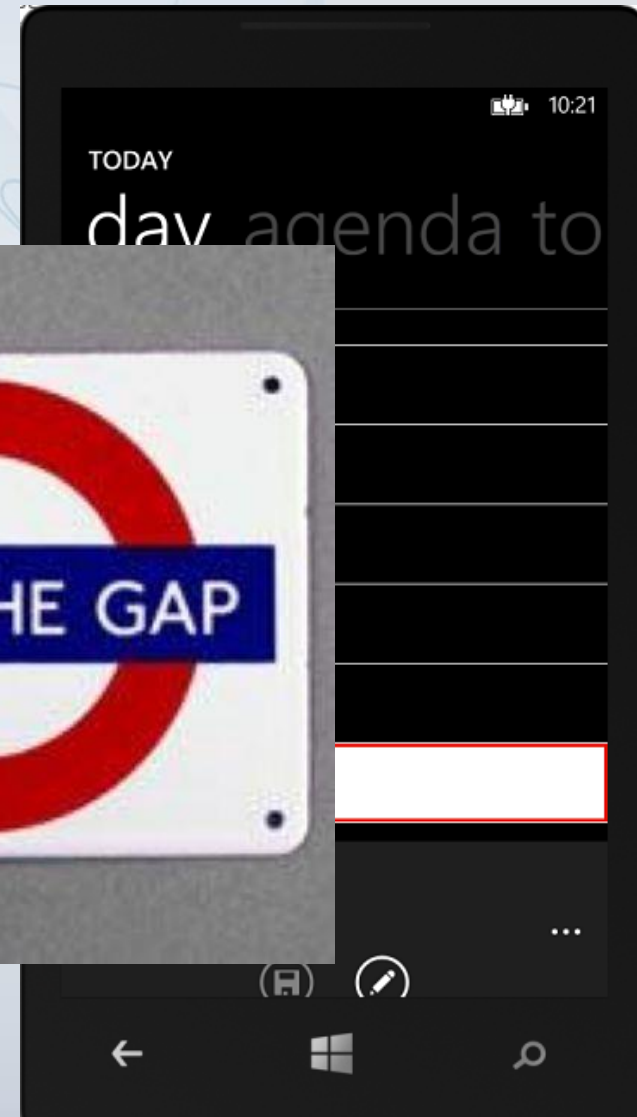
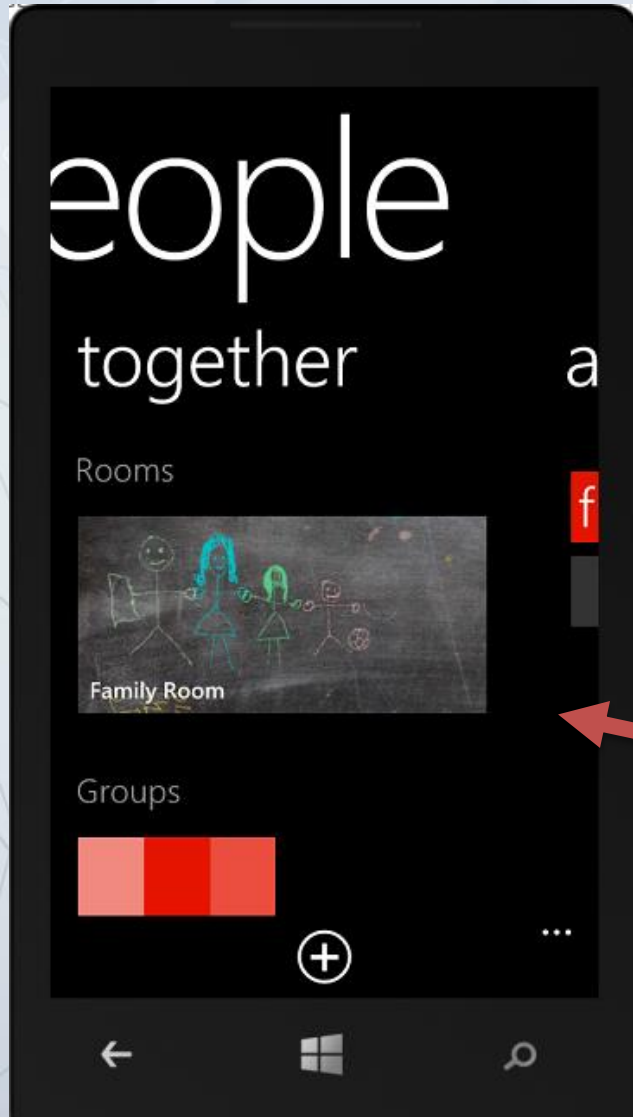
Design (1)

- Clear the table
 - HTML
 - By using WebBrowser Control (IE 10).
 - Mostly on your own.
 - Phone specific is still C# (no WinJS, yet).
 - XAML + XNA/ XAML + Direct3D
 - Games obviously.
 - XNA has no future— better off with Direct3D or different approach such as Unity3d.



Design (2)

- Pivot VS Panorama.



Design (3)

- Use the AppBar (“Application Bar”) for “Context” operations.
- Think about Navigation, Think about CustomMessageBox. Those enable flow in your application of the user.
- Remember various view-states: Landscape and Portrait.
- Follow the Windows Phone Guidelines.

[http://msdn.microsoft.com/en-us/library/hh202915\(v=VS.92\).aspx](http://msdn.microsoft.com/en-us/library/hh202915(v=VS.92).aspx)

Architecture (1)

- It Depend – But This is a common pattern.
- If you want to have live tiles (and you want), you will gonna need agents.

WP8

Agents

Data Access + Model
(PCL?)

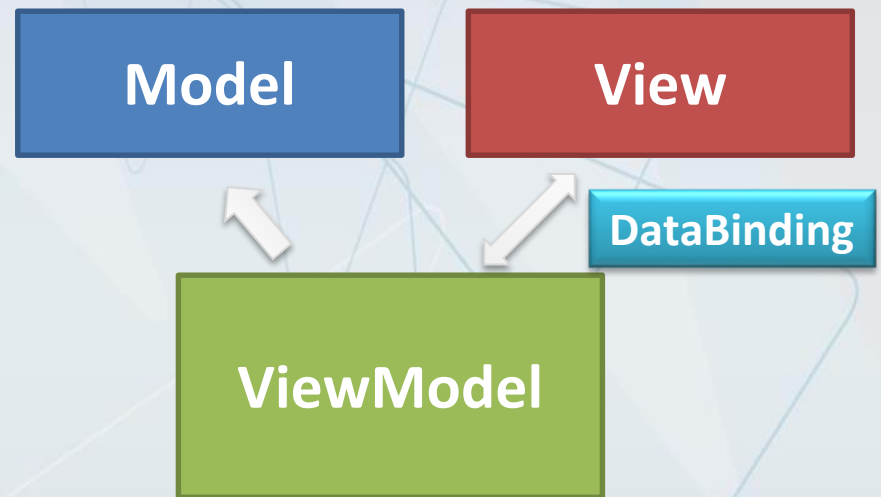
Server Side

Windows
8



Architecture (2)

- MVVM - Use a framework, any framework. **And use MVVM.**
 - **Model:** Responsible for managing and delivering data.
 - **View:** Responsible for formatting and displaying data.
 - **ViewModel:** Responsible for shaping, sorting and filtering data for a view(s).



Architecture (3)

- By default Navigation is View-First.

```
App.RootFrame.Navigate(new Uri("/Views/MyView.xaml?id=" +  
    HttpUtility.UrlEncode(SelectedObject.ID),  
    UriKind.Relative));
```

- You can find frameworks that helps you handle navigation in the ViewModel is preferred.
- Sometimes you will have to use the Code-Behind. But you can use Behaviors to go around it.

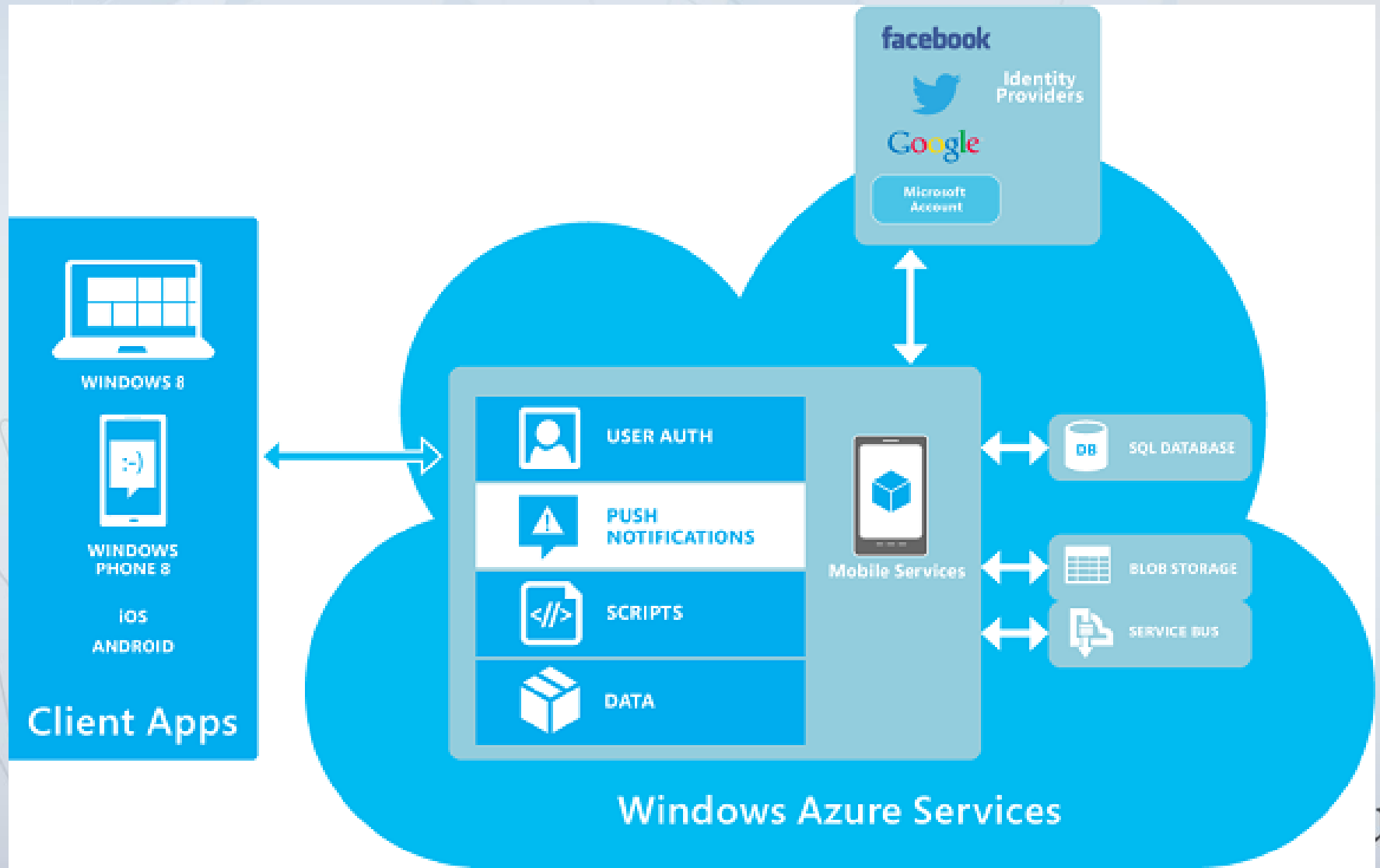
Architecture (4)

- Inversion of Control
 - Again. It depend on how complicated is your app.
 - Many options: Autofac, Tinyloc, SimpleIoC, Funq. (no Unity, yet)
 - Alternative: static members in App. Use Interfaces.



Architecture (5)

- Use Windows Azure Mobile Services if relevant.



Development – Motherf*cker



Our Values

They Claim To Value	They Really Value	We Fucking Do
Individuals and interactions	<i>Tons of billable hours</i>	Programming, Motherfucker
Working software	<i>Tons of pointless tests</i>	Programming, Motherfucker
Customer collaboration	<i>Bleeding clients dry</i>	Programming, Motherfucker
Responding to change	<i>Instability and plausible deniability</i>	Programming, Motherfucker

We think the shit on the left, is really just the con in the middle, and that we really need to just do the thing on the right...Programming, Motherfucker.

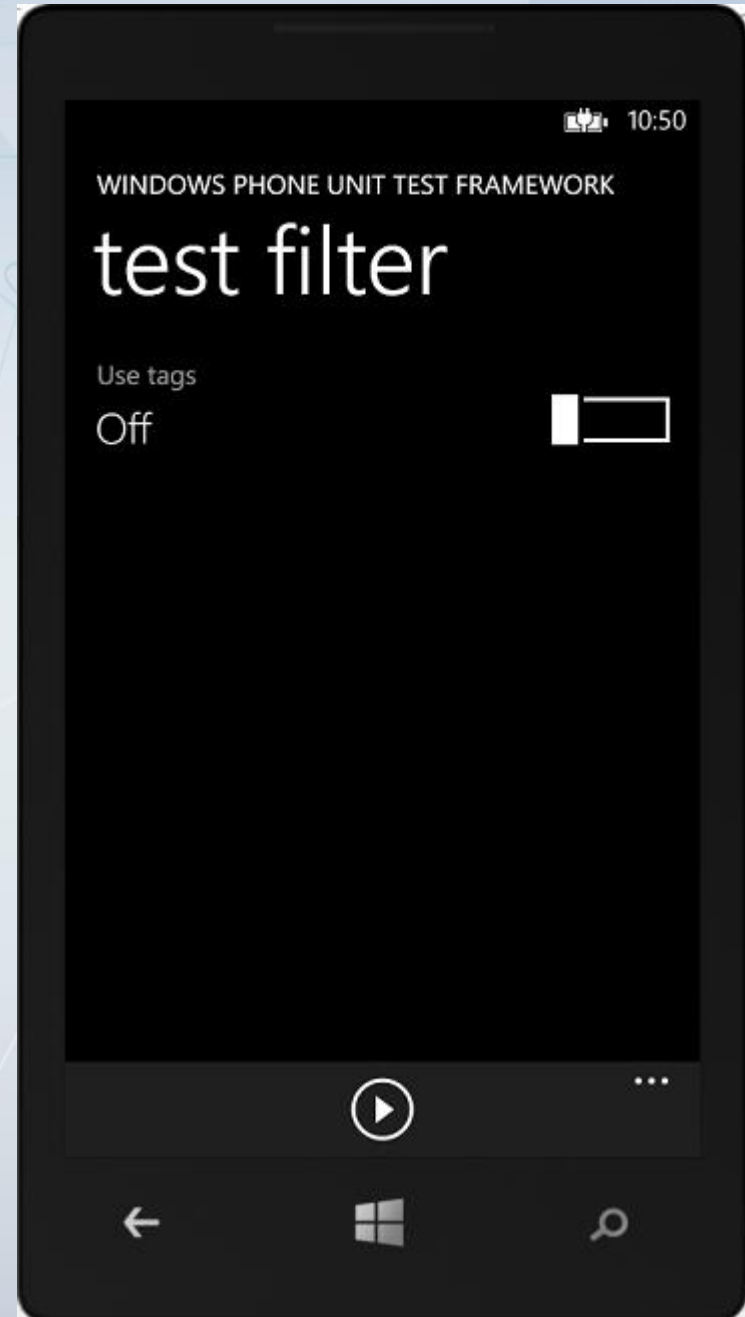
Signed,
Zed A. Shaw
And The Programming Motherfuckers

This bad ass website and T-shirt designed by



Testing

- Test like no tomorrow.
- Use Windows Phone 8 Unitests project.
- Should become better with VS2012 update.



Store

- Deploy quick. Increment as you go.
- Use API for InApp and purchases. (Also Store Mock)
- Get Bug reports and exceptions from the store.
- Allow users to interact with your (better then bad reviews).
- Use Store Test Kit Before publishing.



Summary

- Windows Phone 8 is a great opportunity.
- Reuse your .NET skills effectively.
- Great Phones. OS is getting better.
- Market is increasing.
- Israel is very Microsoft oriented.

Q & A



Thank You

arielbh@CodeValue.net

054-6160650