# CROSS PLATFORM DEVELOPMENT

Glenn Stephens

# ABOUT ME

- Glenn Stephens, BCompSci (UNE), MBA (USQ), GradCertArts (UNE)
- Began writing software in the 80s
- Worked as Software Developer, Team Leader, Architect, CEO
- Worked in Finance, Education, Security, ehealth
- Currently
  - Senior Content Developer at Microsoft
  - Director, Orchard ebusiness Pty Ltd

https://www.linkedin.com/in/glennthomasstephens/

# STRUCTURE

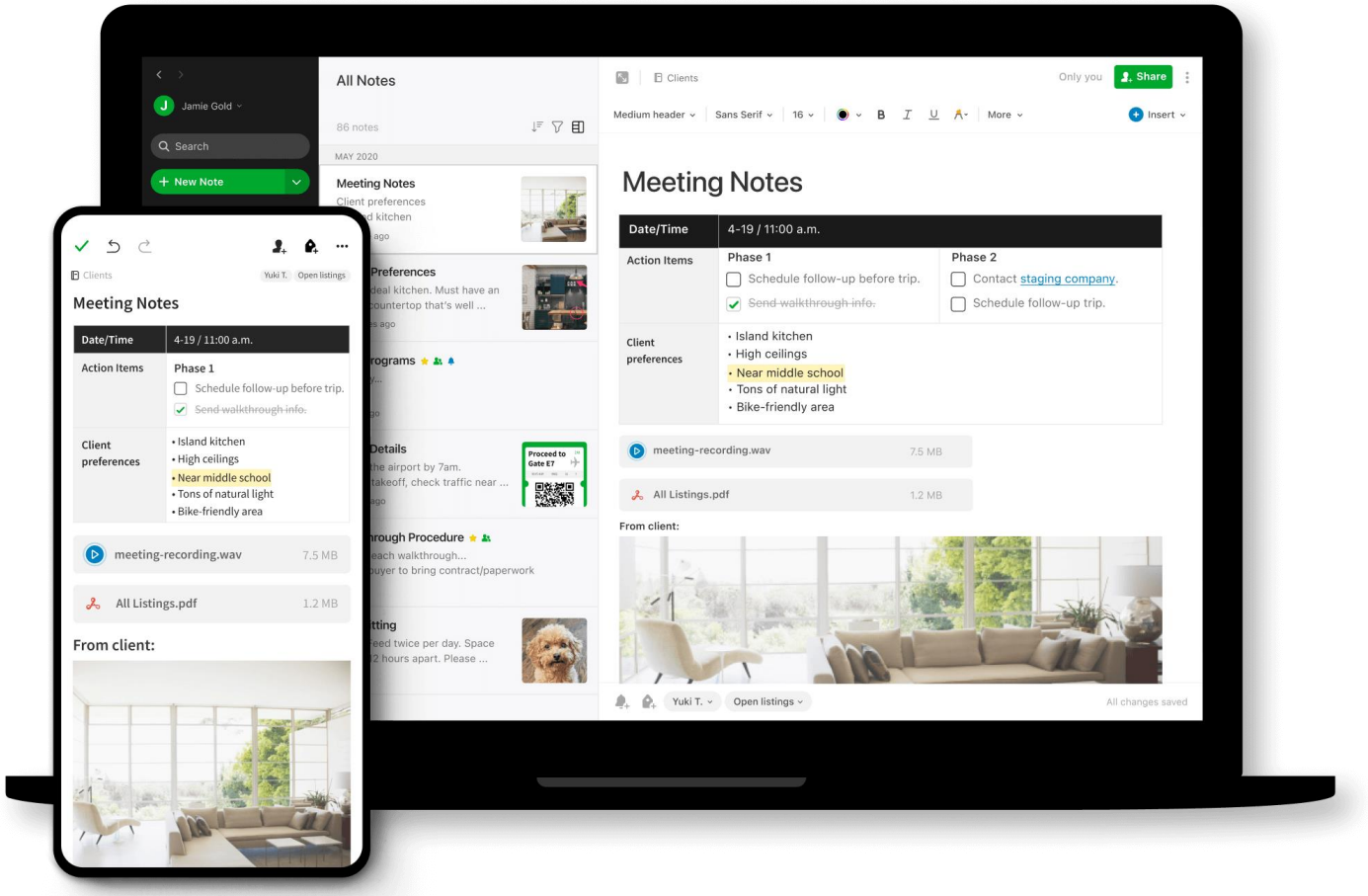
- What is Cross Platform Development
- Cross Platform Benefits
- The Modern Ways to do it
- Q&A

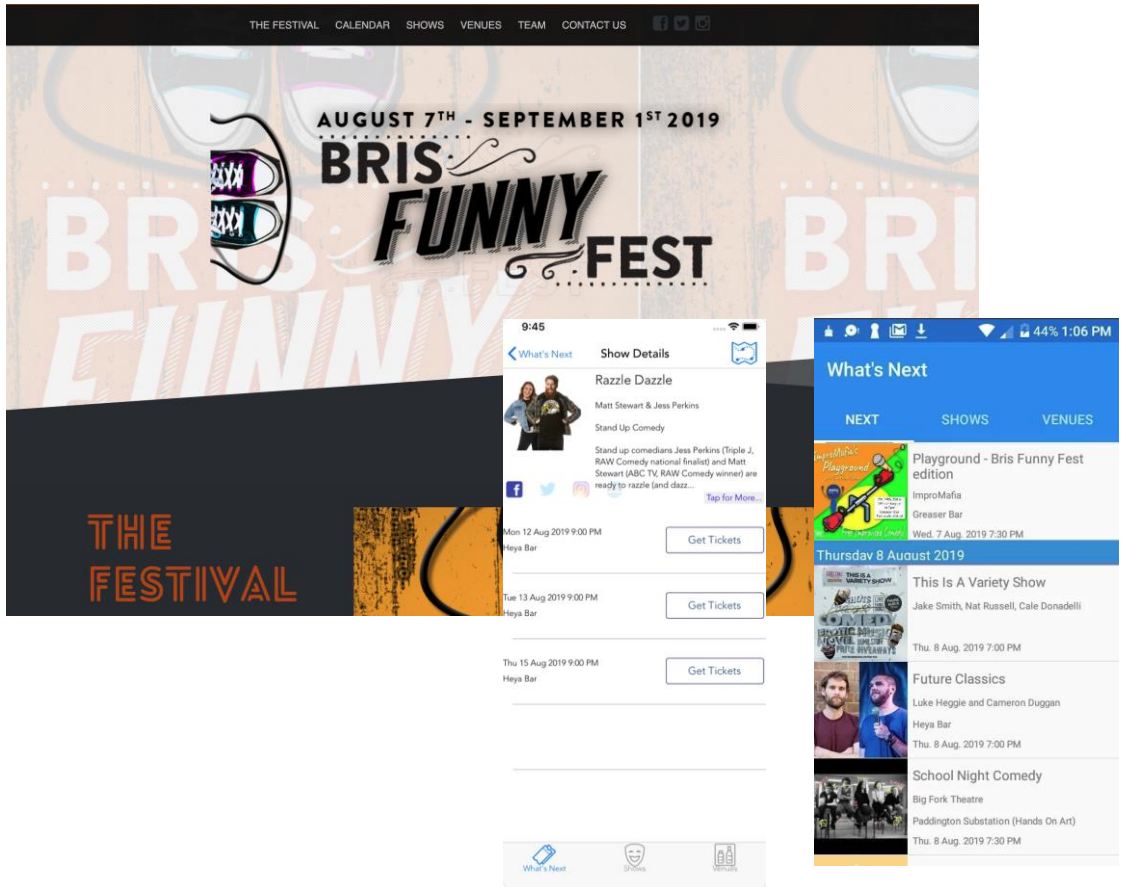
# WHAT IS CROSS PLATFORM DEVELOPMENT?

# CROSS PLATFORM DEVELOPMENT

- Reduces development time
- Reduces testing time
- Often requires code to be more maintainable and well structured

# EXAMPLES


All Notes
Jamie Gold
Search
New Note
86 notes
MAY 2020
Meeting Notes
Client preferences
Clients
Only you
Share
Medium header
Sans Serif
16
More
Insert
Meeting Notes
Date/Time
4-19 / 11:00 a.m.
Action Items
Phase 1
Schedule follow-up before trip.
Send walkthrough info.
Phase 2
Contact staging company.
Schedule follow-up trip.
Client preferences
• Island kitchen
• High ceilings
• Near middle school
• Tons of natural light
• Bike-friendly area
meeting-recording.wav
7.5 MB
All Listings.pdf
1.2 MB
From client:
Yuki T.
Open listings
All changes saved


THE FESTIVAL
CALENDAR
SHOWS
VENUES
TEAM
CONTACT US
AUGUST 7TH - SEPTEMBER 1ST 2019
BRIS FUNNY FEST
THE FESTIVAL
9:45
What's Next
Show Details
Razzle Dazzle
Matt Stewart & Jess Perkins
Stand Up Comedy
Get Tickets
What's Next
NEXT
SHOWS
VENUES
Playground - Bris Funny Fest edition
ImproMafia
Greaser Bar
Wed. 7 Aug. 2019 7:30 PM
Thursday 8 August 2019
This Is A Variety Show
Jake Smith, Nat Russell, Cale Donadelli
Thu. 8 Aug. 2019 7:00 PM
Future Classics
Luke Heggie and Cameron Duggan
Heya Bar
Thu. 8 Aug. 2019 7:00 PM
School Night Comedy
Big Fork Theatre
Paddington Substation (Hands On Art)
Thu. 8 Aug. 2019 7:30 PM

# THE BENEFITS TO CROSS PLATFORM

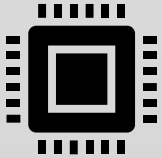
- Efficiency
  - Write for more platforms at once
  - Reach more customers with a smaller total effort
- Cost
  - Cost is reduced
  - Time to market can be quicker

# WHAT IS A PLATFORM?

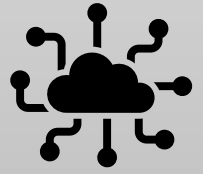
Desktop

Web

Tablet



IoT

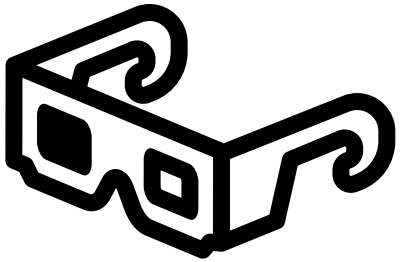


Cloud

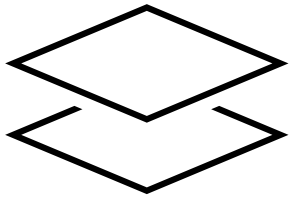
Phone

TV

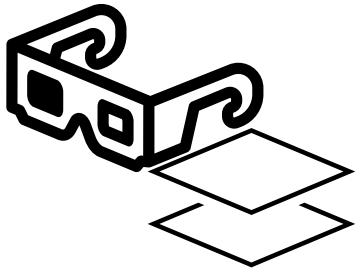
# THE PATHS…



A tool that provides the capabilities



One that allows for Native integrations



One that allows for both

# MODERN TOOLING

- Xamarin
  - https://visualstudio.microsoft.com/xamarin/
- React Native
  - https://reactnative.dev/
- Flutter
  - https://flutter.dev/
- Swift
  - https://developer.apple.com/swift/
- Ionic
  - https://ionicframework.com/
- Unity3D
  - https://unity.com/

# THINKING CROSS PLATFORM

Strategies, Patterns and Tooling

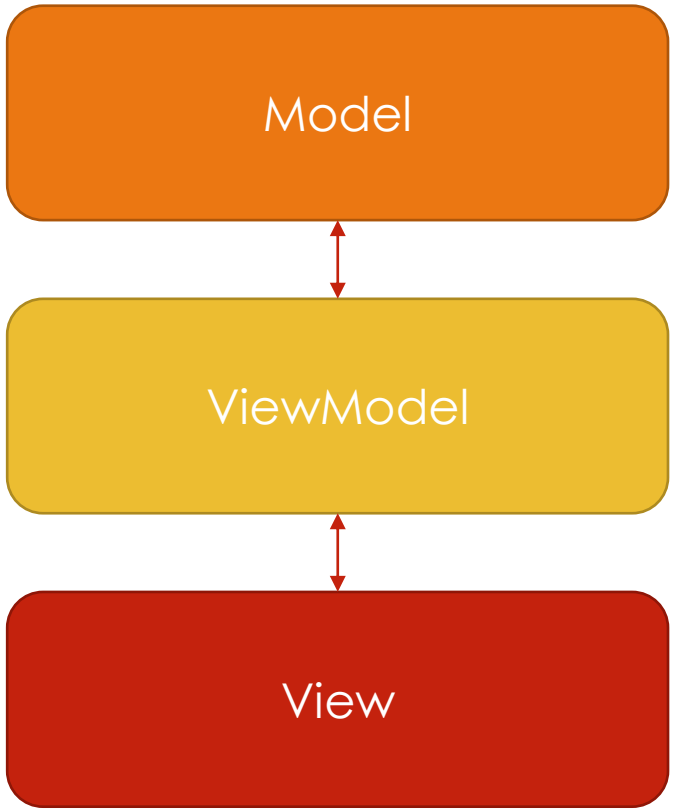
# ARCHITECTURAL PATTERNS

- MVVM
  - Model-View-ViewModel
- MVU
  - Model-View-Update

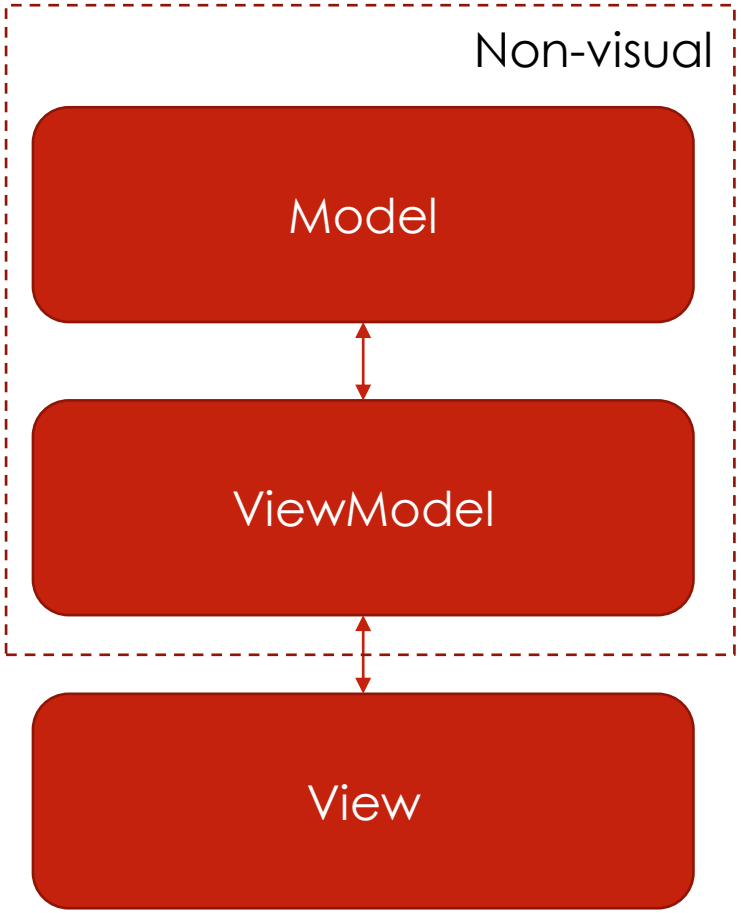
# NATIVE VS SHARABLE CODE

- There will always be percentage of code that is native to your app
- There will always be a percentage of code that is shared between platforms

# MODEL-VIEW-VIEWMODEL



# MODEL-VIEW-VIEWMODEL




- Data and Services used in the app
- REST Clients
- Data Layers
- abstractions

- Contains the data and behavior that represents an app/screen
- Work against abstractions or services in shared code

- Views can be written for the targeted platform or idiom
- This is ideal for the consumer based applications

# MVVM EXAMPLE

```
public class CounterViewModel : ViewModel
{
    int counter;

    public int Counter {
      get => counter;
      set {
        SetPropertyValue(ref counter, value);
      }
    }

    public Command IncrementCounter { get; private set; }

    ...
}
```

C#

# WORKING WITH ABSTRACTIONS

```
public interface IMessageDialog {
    void ShowMessage(string title, string message, string buttonText);
}
```

# WORKING WITH ABSTRACTIONS

```
public class HandleOperationsViewModel : ViewModel
{
    IMessageDialog _msgDialog;

    public HandleOperationsViewModel(IMessageDialog msgDialog) {
      this._msgDialog = msgDialog;
    }

    void DoSomething()
    {
      msgDialog.ShowMessage(”Welcome”, “Thanks for running the app”, “Close”);
    }
}
```

C#

# WORKING WITH ABSTRACTIONS

```
class MessageDialog_iOS : IMessageDialog {
    ...
}

class MessageDialog_Android : IMessageDialog {
    ...
}

class MessageDialog_Blazor : IMessageDialog {
 ...
}
```

C#

# DEMO

A simple MVVM example running on iOS and Android

# MODEL-VIEW-UPDATE

```
type Model = { Count: int }
type Msg = | Increment of int | IncrementRandom | CmdIncrementRandom
let init() = { Count = 0 }, Cmd.none
let private random = Random()
let private incrementRandom () = (Increment(random.Next(0, 100)))
let update msg model =
  match msg with
    | Increment value -> { model with Count = model.Count + value }, Cmd.none
    | IncrementRandom -> model, Cmd.ofMsg CmdIncrementRandom
    | CmdIncrementRandom -> model, Cmd.ofMsg (incrementRandom())
```

F#

```
let view (model: Model) dispatch =
  View.ContentPage
    (content =
      View.StackLayout
        (children =
          [ View.Label(text = sprintf "Current Count: %d" model.Count)
            View.Button(
              text = "Increment",
              command = (fun () -> dispatch (Increment 1)))
            View.Button(
              text = "Increment Random",
              command = (fun () -> dispatch IncrementRandom))
          ]))

let program = Program.mkProgram init update view
```

F#

# DEMO

A simple MVU app running on iOS and Android

# Q&A