Android, iOS and Hybrid Applications

Mobile-Development

Introduction

- Fabrizio Niedda
- fabrizio@encodo.ch
- Bachelor
- Encodo Systems AG, Winterthur
- Mobile Development in Banking, Pharma, User Assistance etc.

Introduction

- Name?
- Where do you work?
- What technologies are you experienced in?
- Any experience in Mobile Development?
- What do you want to learn?

General

- From 21:00 open session - questions, problems etc.
- One written exam at the end of the semester
- App that you create during this course will be rated

GENERAL

- Documentation can be found on

https://github.com/FabrizioNiedda/hfu2020docs

- Example code can be found on

https://github.com/FabrizioNiedda/hfutodoapp

- MS documentation root

https://docs.microsoft.com/en-us/xamarin/xamarin-forms/

DAY 1

- Overview Mobile Development
- Xamarin Forms basics
- Setup Xamarin Sample-App
- Cross Platform Basics
- Navigation Basics
- Document your app/idea and get it approved

Day 2

- MVVM
- XAML for Forms
- Controls and differences to WPF
- Bindings
- Commands

Day 3

- Dialogs
- Styling
- Inversion of Control (IOC)
- Testing

Day 4

- Notifications
  - Local
  - PUSH/Remote

Day 5

- Hybrid Applications
- Interoperability with the native part
  - Design a possible interface
  - Present your approach
- Create a small working sample

Day 6

- Basics (Block Cyphers, PK-Infrastructure)
- Mobile Security
- Biometrics

Day 7

- Local Databases (SQLite)
- Logging & Crashes

- Continue working on your app

Day 8

- Written exam
- Continue working on your app
  - Short review with each one

Day 9

- Present your apps from the project week

Overview Mobile Development

Mobile Development				
	IDE	Languages	Frameworks	Build/Deployment
Android	Android Studio Eclipse NetBeans	Java Kotlin C++	Dagger data-bind Crashlytics Google Play Service Support Library gson jdeferred	PlayStore Gradle Maven
iOS	xCode AppCode	Swift Objective-C	Alamofire CryptoSwift SwiftyJSON SwiftyRSA PromiseKit	Testflight AppStore CocoaPods
Cross	Visual Studio JB Rider WebStorm Visual Studio Code	C#/F# TypeScript/JS Dart etc. JavaScript HTML/CSS Less/Sass React/Redux Angular	React Native Ionic PhoneGap (Cordova) NativeScript	fastlane HockeyApp Artifactory Jenkins Teamcity

Setup

- Install Visual Studio Xamarin Tools
- One of the following:
  - Setup an emulator (Android) (simulator (iOS))
  - Setup your device to allow debugging
- (Install SmartGit (https://www.syntevo.com/smartgit/))
- Provide me with your github username

Questions?

- Short break
- Everyone send me his username

Git

- Pull vs Fetch
- Push vs Commit
- Local vs Remote
- Merge vs Rebase

- Demo
- Clone the “hfu2020docs” & “hfutodoapp” repository
- FabrizioNiedda is my account

https://www.syntevo.com/smartgit/

Overview

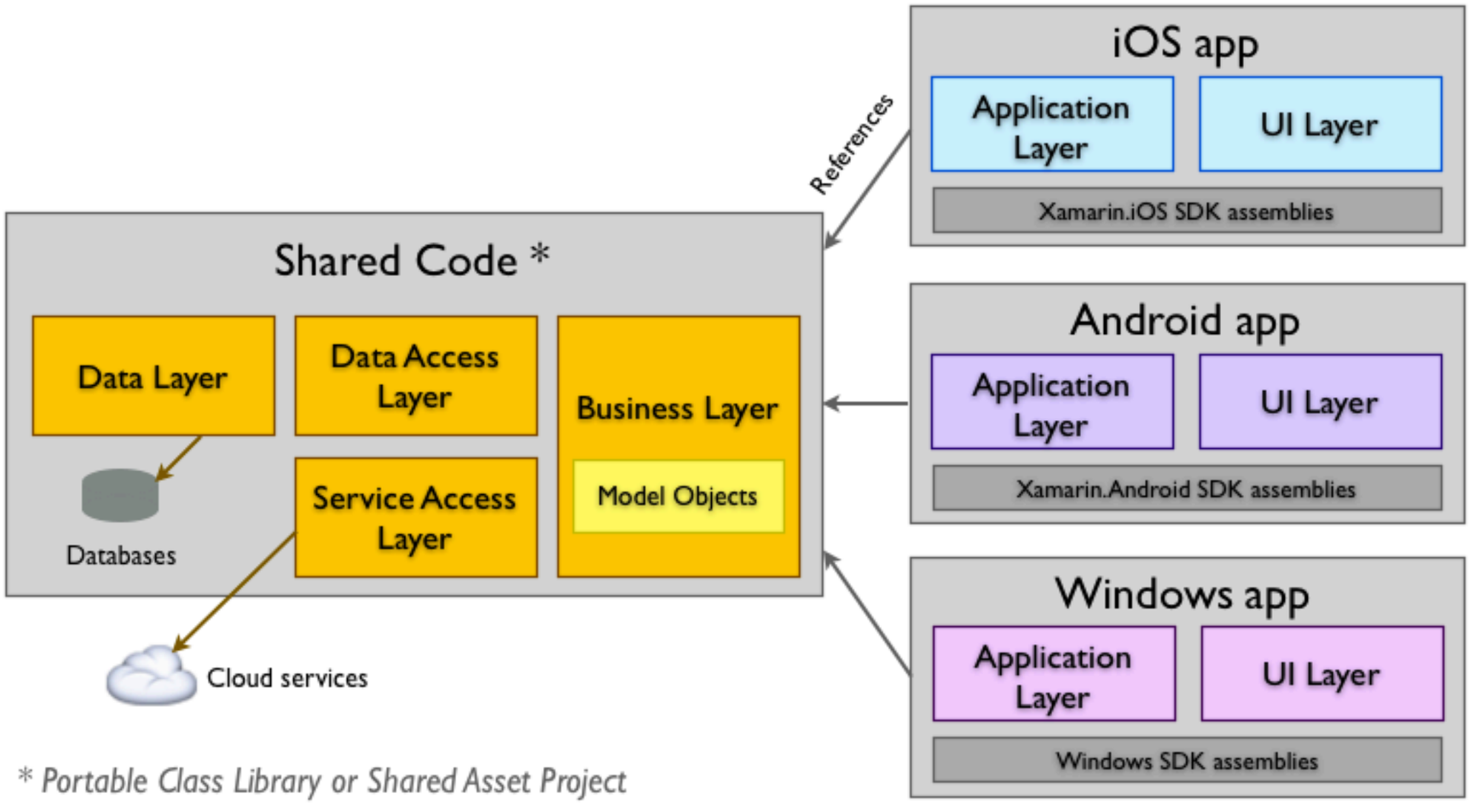
- Xamarin was created by Mono
- Use C# to develop on Android and iOS native
- Acquired and integrated by MS

Overview – Xamarin "Classic"

- Xamarin.iOS & Xamarin.Android (Classic)
  - They basically map the platform 1:1
  - Support new APIs in 24 hours
  - Share the "Core" libraries
  - No shared UI - unless you use a Hybrid approach

https://docs.microsoft.com/en-us/xamarin/cross-platform/app-fundamentals/building-cross-platform-applications/understanding-the-xamarin-mobile-platform

OVERVIEW – XAMARIN CLASSIC



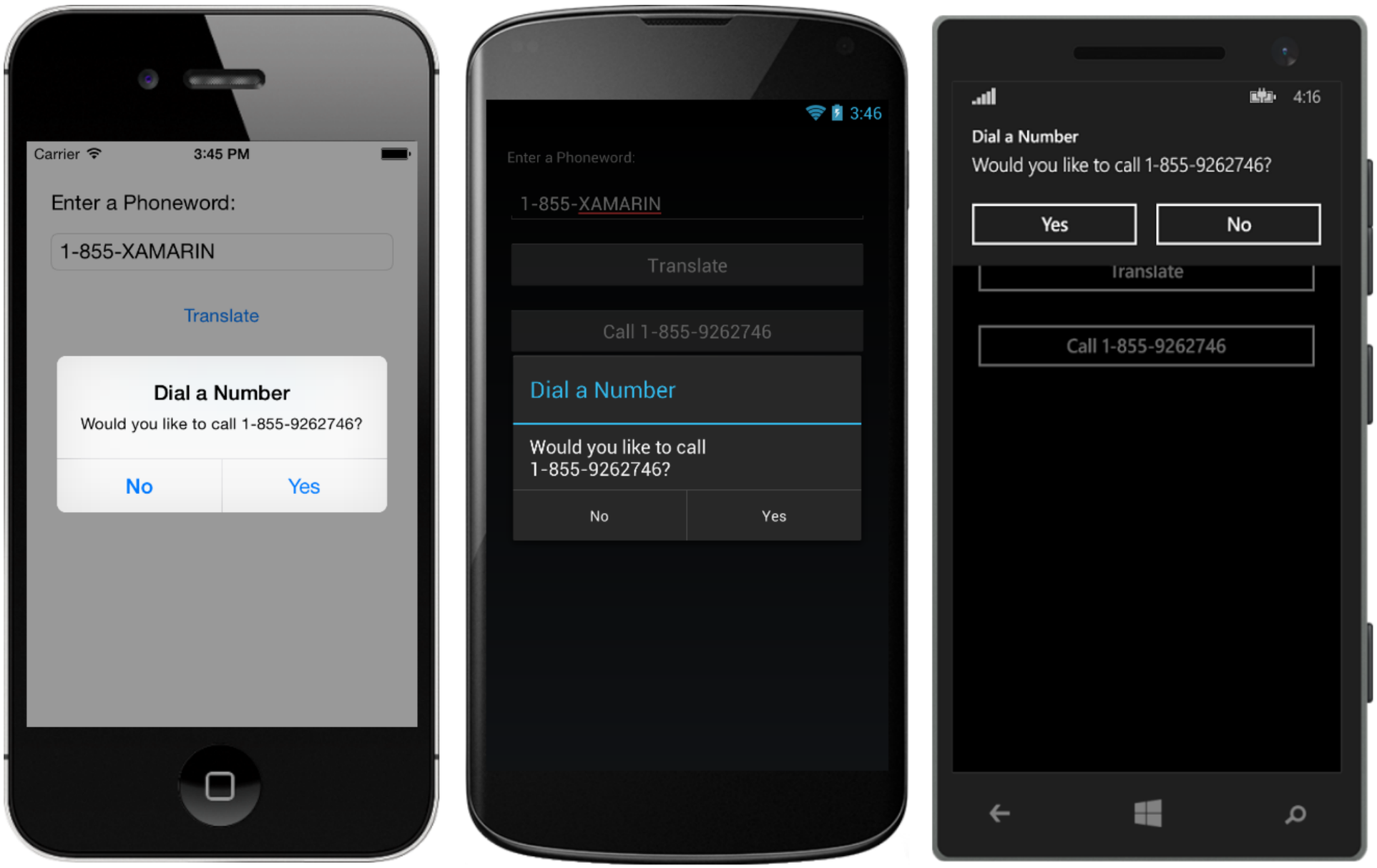
* *Portable Class Library or Shared Asset Project*

Overview – Xamarin Forms

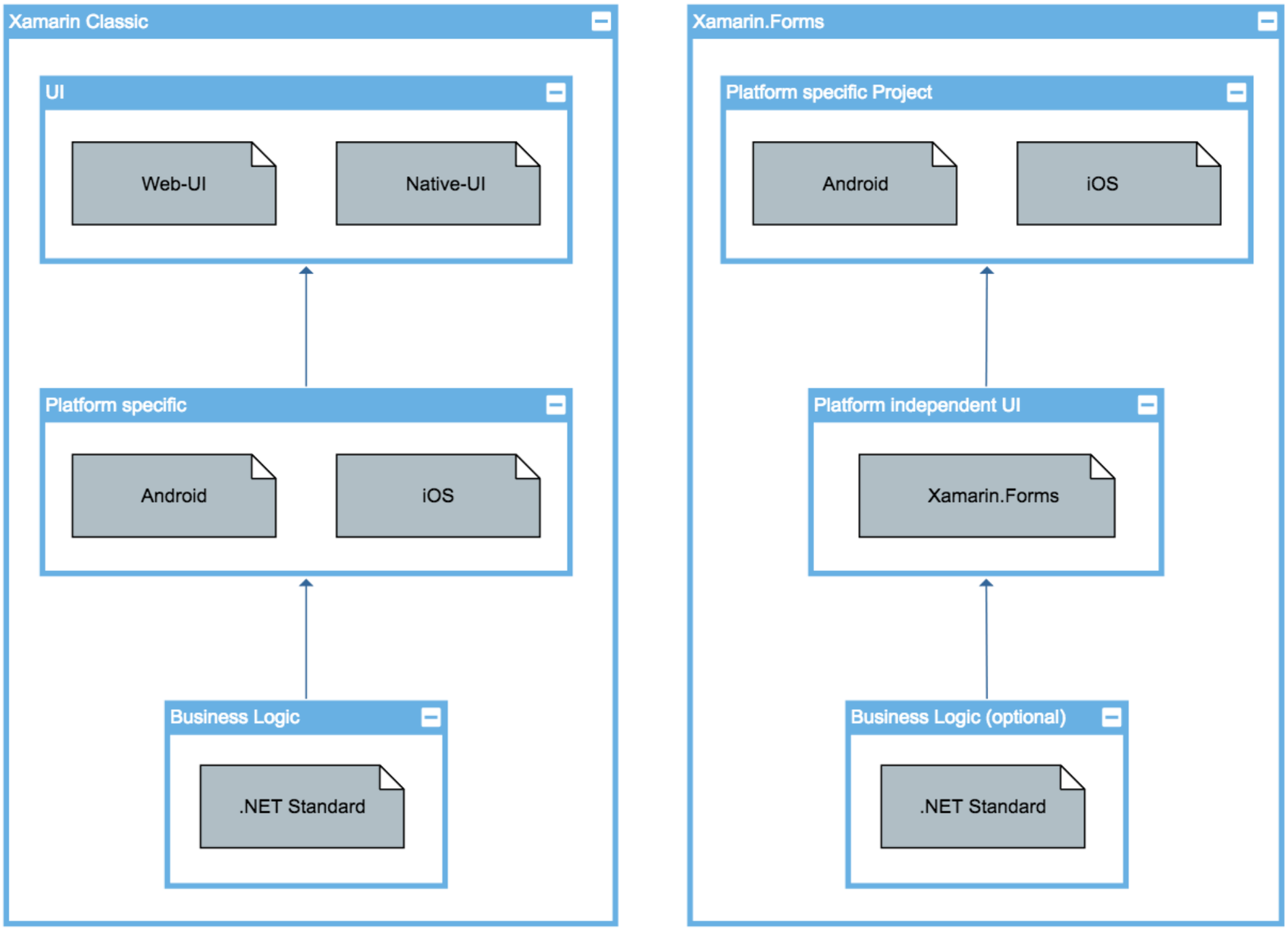
- Xamarin.Forms
  - Share UI code and write it in XAML
  - Use DependencyService (IoC) to access platform features
  - Abstraction of features (Dialogs, Notifications, etc.)
  - Built in support for navigation

https://docs.microsoft.com/en-us/xamarin/xamarin-forms/

Overview – Xamarin Forms



Overview – Architecture



.NET Standard

- Version 2.1 is the latest
  - The higher the version the more APIs are available
- Think of it as an “interface” it does not contain actual code
  - Mono implements .NET Standard for iOS/Android
  - .NET Framework implements it for Windows
- You can find the definitions on github

https://github.com/dotnet/standard/blob/master/docs/versions/netstandard2.1.md

.NET Standard

- Some popular APIs:
  - File (System.IO)
  - Collections & LINQ
  - Task & async await
  - Http (Client) (System.NET)

Questions?

Create a Project

- Pick Xamarin.Forms (with .NET Standard)
- You'll see three projects (Shared, iOS, Android)
- Shared one is a .NET Standard project

- Walkthrough
- Everyone get it running!

Xamarin Forms: Navigation

- Wrap your start page in a “NavigationPage”
- Push other pages on top of the Stack

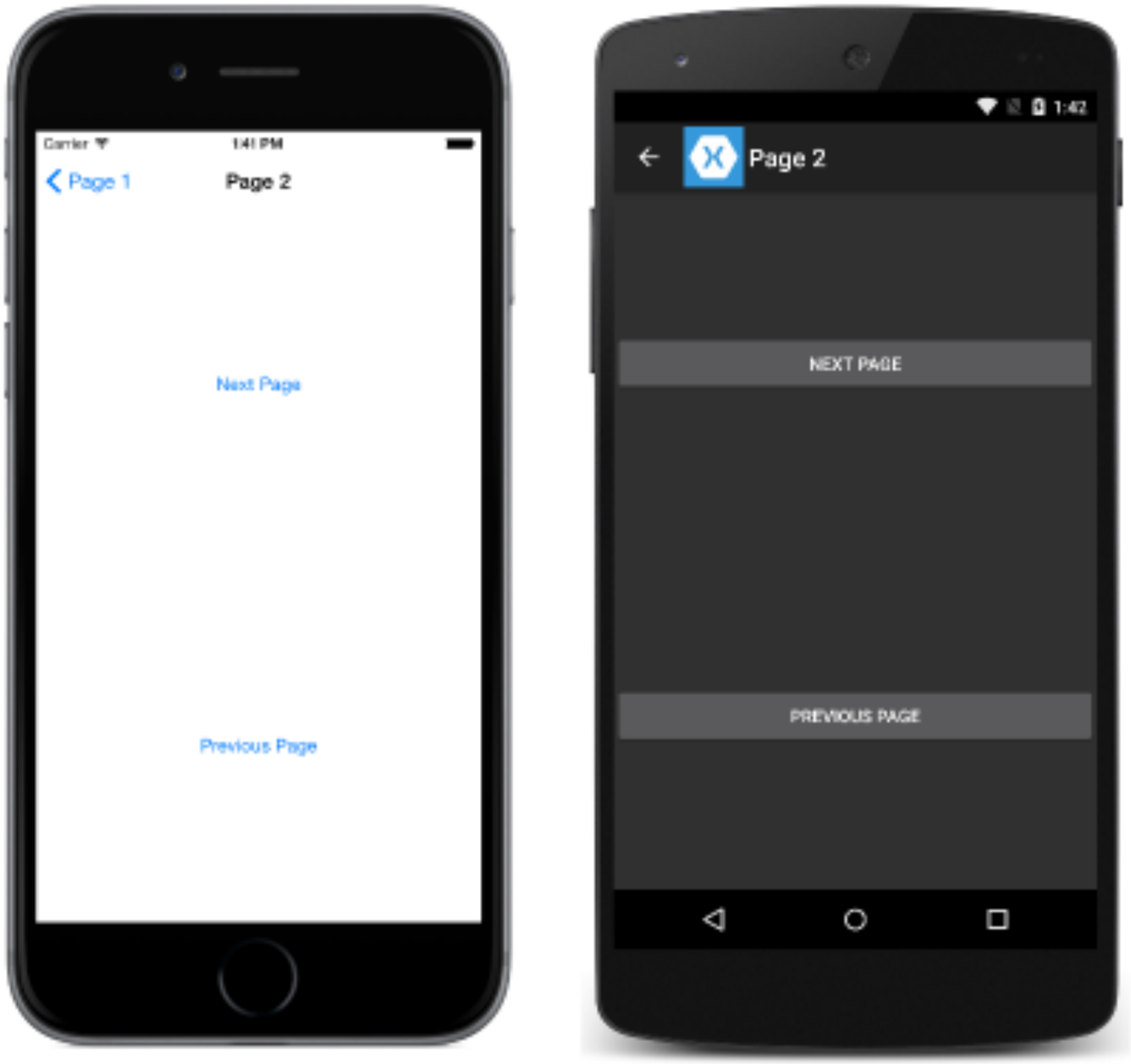
```
// App.xaml.cs
MainPage = new NavigationPage(new MainPage());

// Navigate to the "ListPage"
this.Navigation.PushAsync(new ListPage());

// Pop the top view and return to the previous one.
this.Navigation.PopAsync();
```

https://docs.microsoft.com/en-us/xamarin/xamarin-forms/app-fundamentals/navigation/hierarchical

Navigation


Carrier
1:41 PM
Page 1
Page 2
Next Page
Previous Page
1:42
Page 2
NEXT PAGE
PREVIOUS PAGE

Navigation

- Example
- Checkout branch “day1/navigation”

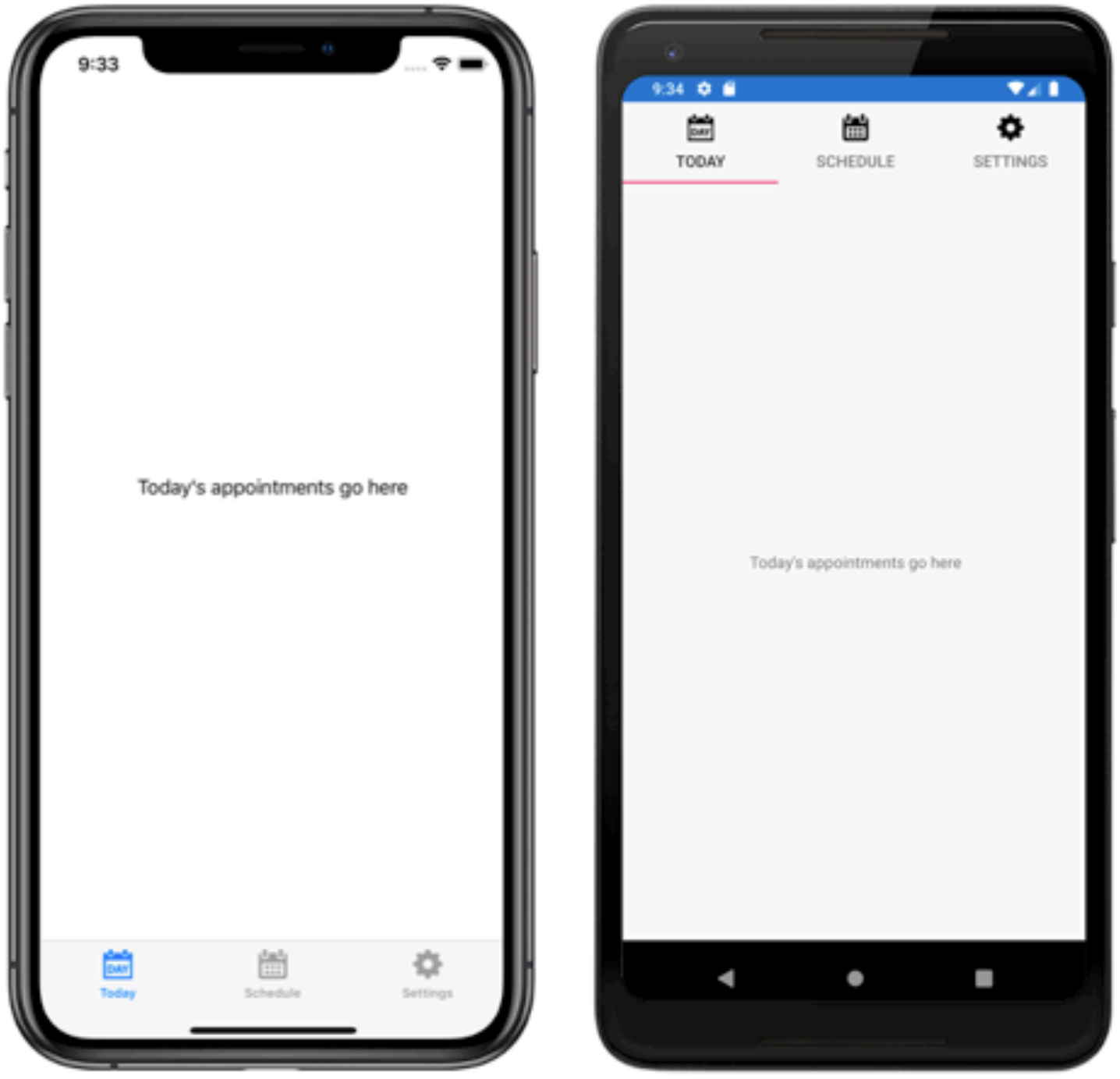
Xamarin Forms Tabs

- Replace the Main Page with a “TabbedPage”
- Populate pages as tabs in the “TabbedPage.Children”

```
<TabbedPage
    xmlns="http://xamarin.com/schemas/2014/forms"
    xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
    xmlns:local="clr-namespace:todo"
    x:Class="todo.MainPage"
    Title="Main Page"
    SelectedTabColor="Firebrick">
    <TabbedPage.Children>
        <ContentPage Title="Tab 1" IconImageSource="schedule.png" BackgroundColor="White">
            <Button Text="Some Button" />
        </ContentPage>
        <ContentPage Title="Tab 2">
            <Button Text="Some other Button" />
        </ContentPage>
        <local:TodoListPage Title="Tab 2" />
    </TabbedPage.Children>
</TabbedPage>
```

https://docs.microsoft.com/en-us/xamarin/xamarin-forms/app-fundamentals/navigation/tabbed-page

Tabs


9:33
Today's appointments go here
Today
Schedule
Settings
9:34
TODAY
SCHEDULE
SETTINGS
Today's appointments go here

Tabs

- Example
- Checkout branch “day1/tabNavigation“

Think about a Project

- It should include
  - Notifications
  - Some login scenario
    - Alerts (Yes/No Dialog)
    - Some input fields
    - A list or something similar
- TODO-App for example

Setup the Application Layers

- Setup the necessary navigation pages
- Push to github
- Invite me to your project
- Test the navigation

Additional Tasks

- Check out the Master-Detail pattern

https://docs.microsoft.com/en-us/xamarin/xamarin-forms/app-fundamentals/navigation/master-detail-page

- Check out the Carousel

https://docs.microsoft.com/en-us/xamarin/xamarin-forms/app-fundamentals/navigation/carousel-page