

Lessons learned from migrating to .NET MAUI

Damian Antonowicz
damianantonowicz.pl

Demant

damianantonowicz.pl



@D_Antonowicz



Why migrate to .NET MAUI?

Why migrate to .NET MAUI?

- Xamarin support will end on 1 May 2024
 - no more fixes, updates or technical assistance
- Xamarin will support only (from 11 October 2023):
 - Android API 34 (was API 33)
 - Xcode 15 SDKs (was 14 SDKs)



*prediction

Sources:

<https://dotnet.microsoft.com/en-us/platform/support/policy/xamarin>
<https://developer.apple.com/ios/submit/>
https://en.wikipedia.org/wiki/Android_version_history

Building Xamarin

Azure DevOps Agent	Xamarin SDKs	Xcode	Android API	Can build Xamarin using Xcode 15?	Can build Xamarin using Android API 34?
macOS 12					
macOS 13					

Sources:

<https://github.com/actions/runner-images/blob/main/images/macos/macOS-13-Readme.md>

<https://github.com/actions/runner-images/blob/main/images/macos/macOS-12-Readme.md>

<https://github.com/actions/runner-images/issues/8045>

<https://github.com/actions/runner-images/issues/8647>

Building Xamarin

Azure DevOps Agent	Xamarin SDKs	Xcode	Android API	Can build Xamarin using Xcode 15?	Can build Xamarin using Android API 34?
macOS 12					
macOS 13					

Sources:

<https://github.com/actions/runner-images/blob/main/images/macos/macOS-13-Readme.md>

<https://github.com/actions/runner-images/blob/main/images/macos/macOS-12-Readme.md>

<https://github.com/actions/runner-images/issues/8045>

<https://github.com/actions/runner-images/issues/8647>

Building Xamarin

Azure DevOps Agent	Xamarin SDKs	Xcode	Android API	Can build Xamarin using Xcode 15?	Can build Xamarin using Android API 34?
macOS 12		13.1 – 14.2			
macOS 13		14.1 – 15.0.1			

Sources:

<https://github.com/actions/runner-images/blob/main/images/macos/macOS-13-Readme.md>

<https://github.com/actions/runner-images/blob/main/images/macos/macOS-12-Readme.md>

<https://github.com/actions/runner-images/issues/8045>

<https://github.com/actions/runner-images/issues/8647>

Building Xamarin

Azure DevOps Agent	Xamarin SDKs	Xcode	Android API	Can build Xamarin using Xcode 15?	Can build Xamarin using Android API 34?
macOS 12		13.1 – 14.2	27 - 34		
macOS 13		14.1 – 15.0.1	27 - 34		

Sources:

<https://github.com/actions/runner-images/blob/main/images/macos/macOS-13-Readme.md>

<https://github.com/actions/runner-images/blob/main/images/macos/macOS-12-Readme.md>

<https://github.com/actions/runner-images/issues/8045>

<https://github.com/actions/runner-images/issues/8647>

Building Xamarin

Azure DevOps Agent	Xamarin SDKs	Xcode	Android API	Can build Xamarin using Xcode 15?	Can build Xamarin using Android API 34?
macOS 12	✓	13.1 – 14.2	27 - 34	✗	
macOS 13	✗	14.1 – 15.0.1	27 - 34	✗	

Sources:

<https://github.com/actions/runner-images/blob/main/images/macos/macOS-13-Readme.md>

<https://github.com/actions/runner-images/blob/main/images/macos/macOS-12-Readme.md>

<https://github.com/actions/runner-images/issues/8045>

<https://github.com/actions/runner-images/issues/8647>

Building Xamarin

Azure DevOps Agent	Xamarin SDKs	Xcode	Android API	Can build Xamarin using Xcode 15?	Can build Xamarin using Android API 34?
macOS 12	✓	13.1 – 14.2	27 - 34	✗	✓
macOS 13	✗	14.1 – 15.0.1	27 - 34	✗	✗

Sources:

<https://github.com/actions/runner-images/blob/main/images/macos/macOS-13-Readme.md>

<https://github.com/actions/runner-images/blob/main/images/macos/macOS-12-Readme.md>

<https://github.com/actions/runner-images/issues/8045>

<https://github.com/actions/runner-images/issues/8647>

Which .NET MAUI version to use?

Which .NET MAUI version to use?

- .NET MAUI is part of .NET from version 6
- .NET support policy

Version	Release type	End of support
.NET 6	Long Term Support (LTS)	November 2024
.NET 7	Standard Term Support (STS)	May 2024
.NET 8	Long Term Support (LTS)	November 2027

- .NET MAUI support policy

Version	End of support
.NET MAUI 6	May 2023
.NET MAUI 7	May 2024
.NET MAUI 8	May 2025

- “A major version of .NET MAUI receives support for a minimum of **6 months** after a successor (the next major release) ships.”
- “To receive technical support from Microsoft, you need to be on the latest servicing level (update) for a .NET MAUI release.”
- “Servicing updates are made available for supported versions of .NET MAUI.”

Sources:

<https://dotnet.microsoft.com/en-us/platform/support/policy/maui>

<https://dotnet.microsoft.com/en-us/platform/support/policy>

<https://dotnet.microsoft.com/en-us/platform/support/policy/dotnet-core>

Which .NET MAUI version to use?

- .NET MAUI releases in 2023

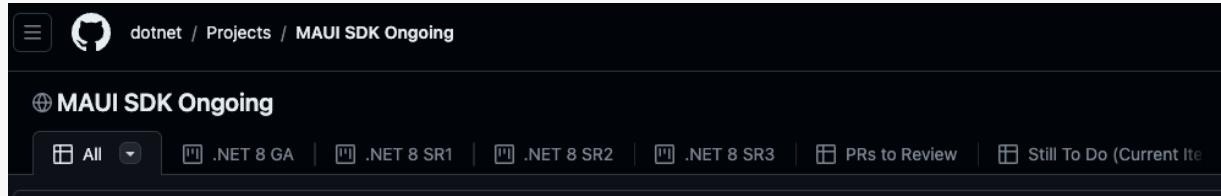
Version	Number of releases
.NET MAUI 6	3
.NET MAUI 7	8
.NET MAUI 8	11

- Issues that won't be backported to .NET MAUI 7

- <https://github.com/dotnet/maui/issues/11251> - „text alignment is not respected”
- <https://github.com/dotnet/maui/issues/10977> - „CascadeInputTransparent="False" is not working on ContentView”

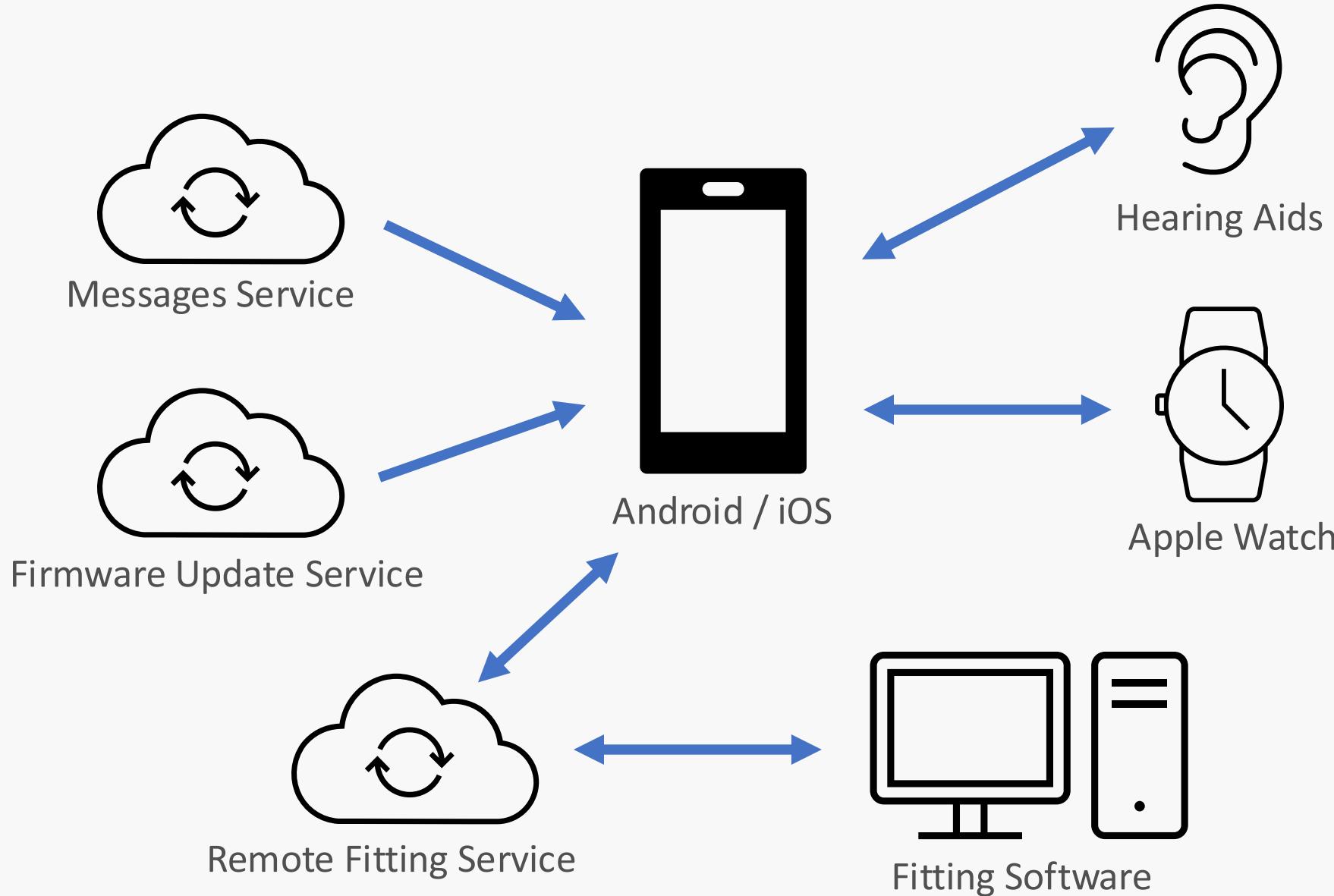
Which .NET MAUI version to use?

Blogpost	Date
Announcing .NET MAUI in .NET 8	November 2023
Improvements & Changes in Android resource generation in .NET 8	November 2023
.NET 8 Performance Improvements in .NET MAUI	October 2023
Announcing .NET MAUI in .NET 8 Release Candidate 2: More Quality	October 2023
Announcing .NET MAUI in .NET 8 Release Candidate 1: Quality	September 2023
Announcing .NET MAUI in .NET 8 Preview 7: Keyboard Accelerators	August 2023
Announcing .NET MAUI in .NET 8 Preview 6: Hello VS Code & VS for Mac	July 2023
Announcing .NET MAUI in .NET 8 Preview 5	June 2023
Announcing .NET MAUI in .NET 8 Preview 3	April 2023

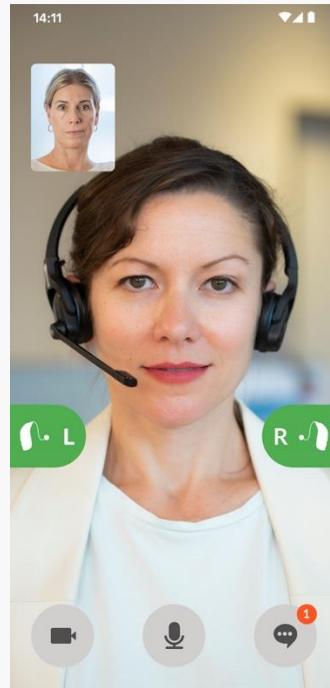
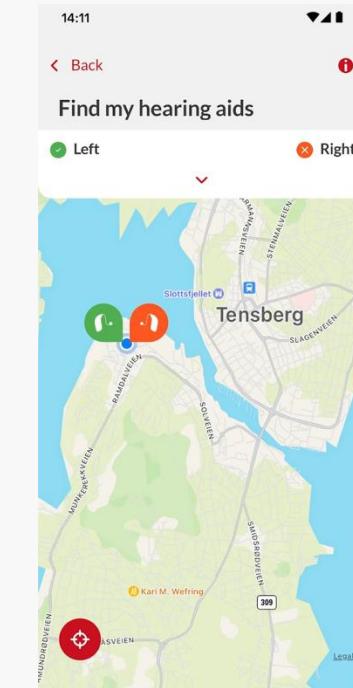
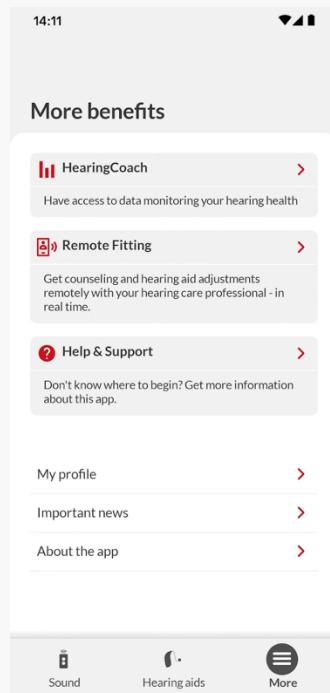
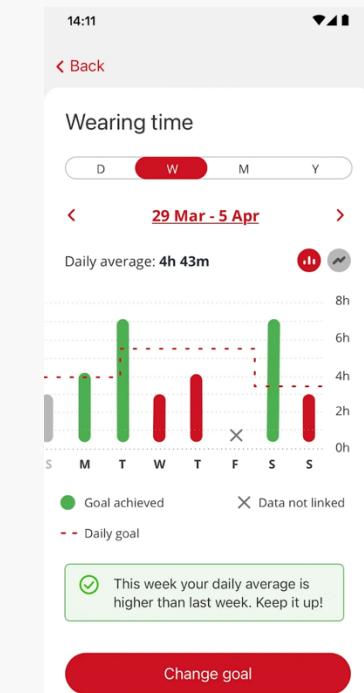
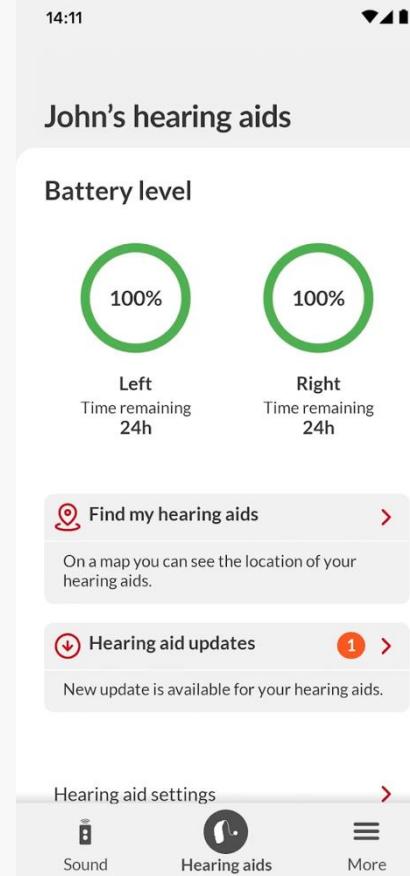
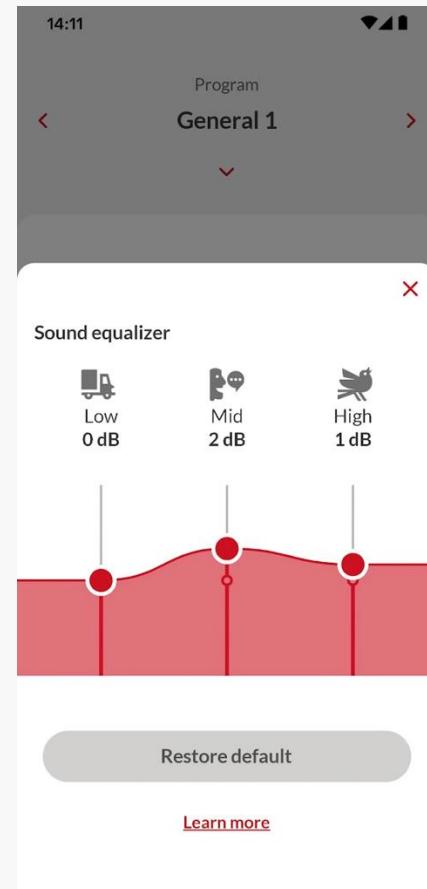
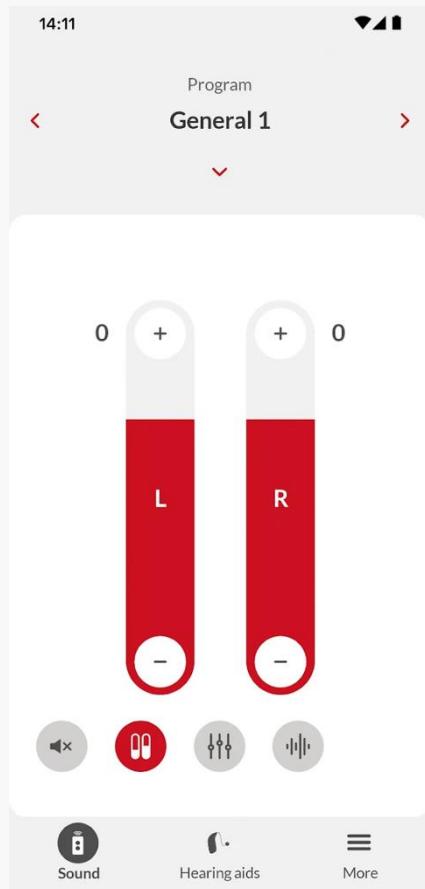


Demant context

Demant context



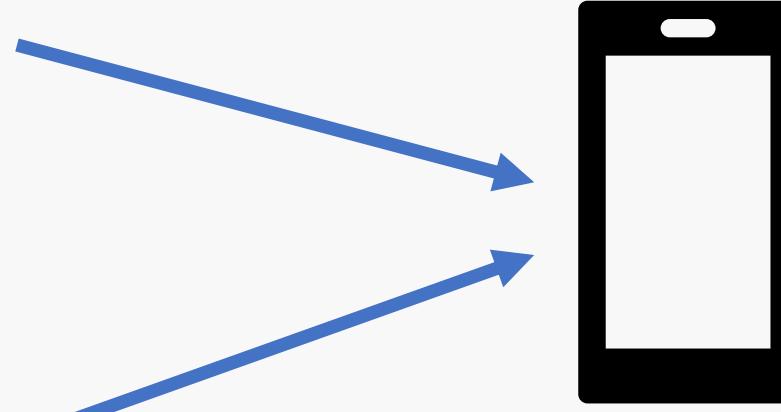
Demant context



Demant context



1 team



6 teams



mobile app for Android & iOS

How to plan migration?

How to plan migration?

- Sync meetings
- Technical debt
 - what will happen if not paid?
 - impacted areas / features
 - what needs to be done
- .NET Upgrade Assistant
- <https://learn.microsoft.com/en-us/dotnet/maui/migration/>
- Architecture Decision Log (ADL)
 - owner
 - workgroup
 - stakeholders
 - solution proposals with pros & cons
 - outcome

How to plan migration?

- **ADLs:**

- How to migrate Rg.Plugins.Popup?
- How to migrate Map control?
- How to migrate splashscreens?
- How to migrate app icon?
- How to migrate Xamarin.Essentials?
- How to migrate Xamarin.Community.Toolkit?
- How to make Lottie animations work with .NET MAUI?
- How to use built-in .NET MAUI IoC container with Dryloc?
- How to migrate image assets?
- How to migrate custom renderers?
- How to migrate effects?
- Should Border be used in favor of Frame?
- How to re-design reloading dynamic font sizes?
- How to migrate Secure Storage?
- How to migrate preferences?

How to plan migration?

• How to migrate Rg.Plugins.Popup?

- **Problem:** Rg.Plugins.Popup does not provide packages for .NET MAUI
- **Outcome:** use Mopups
 - „*Mopups is a replacement for the "Rg.Plugins.Popups" plugin for Xamarin. Mopups intends to provide a similar experience to this plugin, however also clean up the code base and provide forward looking enhancements.*”

• How to migrate Map control?

- **Problem:** API & namespaces have changed in .NET MAUI
- **Outcome:** update source code & use .NET MAUI 7

Xamarin.Forms API	.NET MAUI API	Comment
Xamarin.Forms.Maps.Map.HasScrollEnabled	Microsoft.Maui.Controls.Maps.Map.IsEnabled	
Xamarin.Forms.Maps.Map.HasZoomEnabled	Microsoft.Maui.Controls.Maps.Map.IsZoomEnabled	
Xamarin.Forms.Maps.Map.TrafficEnabled	Microsoft.Maui.Controls.Maps.Map.IsTrafficEnabled	
Xamarin.Forms.Maps.Map.MoveToLastRegionOnLayoutChange		No .NET MAUI equivalent.
Xamarin.Forms.Maps.Pin.Id	Microsoft.Maui.Controls.Maps.Pin.MarkerId	
Xamarin.Forms.Maps.Pin.Position	Microsoft.Maui.Controls.Maps.Pin.Location	
Xamarin.Forms.Maps.MapClickedEventArgs.Position	Microsoft.Maui.Controls.Maps.MapClickedEventArgs.Location	
Xamarin.Forms.Maps.Position	Microsoft.Maui.Devices.Sensors.Location	Members of type Xamarin.Forms.Maps.Position have changed to the Microsoft.Maui.Devices.Sensors.Location type.
Xamarin.Forms.Maps.Geocoder	Microsoft.Maui.Devices.Sensors.Geocoding	Members of type Xamarin.Forms.Maps.Geocoder have changed to the Microsoft.Maui.Devices.Sensors.Geocoding type.

Sources:

<https://www.nuget.org/packages/Rg.Plugins.Popup/>

<https://github.com/LuckyDucko/Mopups>

<https://learn.microsoft.com/en-gb/dotnet/maui/migration/multi-project-to-multi-project#map-changes>

How to plan migration?

- How to migrate splashscreens?

- **Solution proposals:**
 - #1 - use .NET MAUI approach
 - #2 - use already implemented platform approach in Xamarin.Forms
 - **Outcome:** #2

- How to migrate app icon?

- **Solution proposals:**
 - #1 - use .NET MAUI approach
 - #2 - use already implemented platform approach in Xamarin.Forms
 - **Outcome:** #2

Sources:

<https://learn.microsoft.com/en-us/dotnet/maui/user-interface/images/splashscreen?tabs=android>

<https://learn.microsoft.com/en-us/dotnet/maui/user-interface/images/app-icons?tabs=android>

How to plan migration?

- How to migrate Xamarin.Essentials?
 - **Outcome:** remove Xamarin.Essentials NuGet & update csproj to have `<UseMauiEssentials>true</ UseMauiEssentials>`
- How to migrate Xamarin.Community.Toolkit?
 - **Outcome:** use CommunityToolkit.Maui
- How to make Lottie animations work with .NET MAUI?
 - **Problem:** Com.Airbnb.Xamarin.Forms.Lottie does not provide packages for .NET MAUI
 - **Outcome:** use SkiaSharp.Extended.UI.Maui
- How to use built-in .NET MAUI IoC container with DryIoc?
 - **Solution proposals:**
 - #1 – migrate all references to .NET MAUI container
 - #2 – integrate .NET MAUI container with DryIoc
 - **Outcome: #2**
 - **Code:**
 - ```
var builder = MauiApp.CreateBuilder();
builder.ConfigureContainer(ContainerConfiguration.Setup());
```

Sources:

<https://www.nuget.org/packages/SkiaSharp.Extended.UI.Maui/>

<https://github.com/Baseflow/LottieXamarin>

# How to plan migration?

- How to migrate image assets?

- **Solution proposals:**
    - #1 – use current custom implementation generating PNGs from SVGs
    - #2 – use .NET MAUI approach
  - **Outcome:** #2

- How to migrate custom renderers?

- **Solution proposals:**
    - #1 – reuse custom renderers in .NET MAUI
    - #2 – migrate custom renderers to .NET MAUI handlers
    - #3 – decision to reuse or migrate should be taken in specific use case
  - **Outcome:** #3

Sources:

<https://learn.microsoft.com/en-gb/dotnet/maui/migration/custom-renderers>

<https://learn.microsoft.com/en-us/dotnet/maui/user-interface/system-theme-changes>

<https://learn.microsoft.com/en-us/dotnet/maui/user-interface/images/images>

# How to plan migration?

- How to migrate effects?
  - **Outcome:** reuse effects
- Should Border be used in favor of Frame control?
  - **Problem:** "The Frame class existed in Xamarin.Forms and is present in .NET MAUI for users who are *migrating* their apps from Xamarin.Forms to .NET MAUI. If you're *building a new* .NET MAUI app it's *recommended to use Border instead*, and to set shadows using the Shadow bindable property on VisualElement. For more information, see *Border and Shadow*."
  - **Solution proposals:**
    - #1 – use Border everywhere
    - #2 – use Frame everywhere
    - #3 – use Border where Frame will be buggy
  - **Outcome:** #3

Sources:

<https://learn.microsoft.com/en-gb/dotnet/maui/migration/effects>

# How to plan migration?

- How to re-design reloading dynamic font sizes?

- **Solution proposals:**
    - #1 – use .NET MAUI approach
    - #2 – use current custom implementation
  - **Outcome:** #1

- How to migrate Secure Storage?

- **Solution proposals:**
    - #1 – keep using Xamarin.SecureStorage in app based on .NET MAUI
    - #2 – migrate data from Xamarin.SecureStorage to Maui.SecureStorage
    - #3 – migrate data from Xamarin.SecureStorage to SQLite database
  - **Outcome:** #3

Sources:

<https://learn.microsoft.com/en-us/dotnet/maui/user-interface/fonts>

<https://learn.microsoft.com/en-us/dotnet/maui/migration/secure-storage?view=net-maui-8.0>

# How to plan migration?

- How to migrate Preferences?

- **Solution proposals:**
  - #1 – keep using Xamarin.Preferences in app based on .NET MAUI
  - #2 – migrate data from Xamarin.Preferences to Maui.Preferences
  - #3 – migrate data from Preferences to SQLite database
- **Outcome:** #3

Sources:

<https://learn.microsoft.com/en-us/dotnet/maui/migration/app-properties?view=net-maui-8.0>

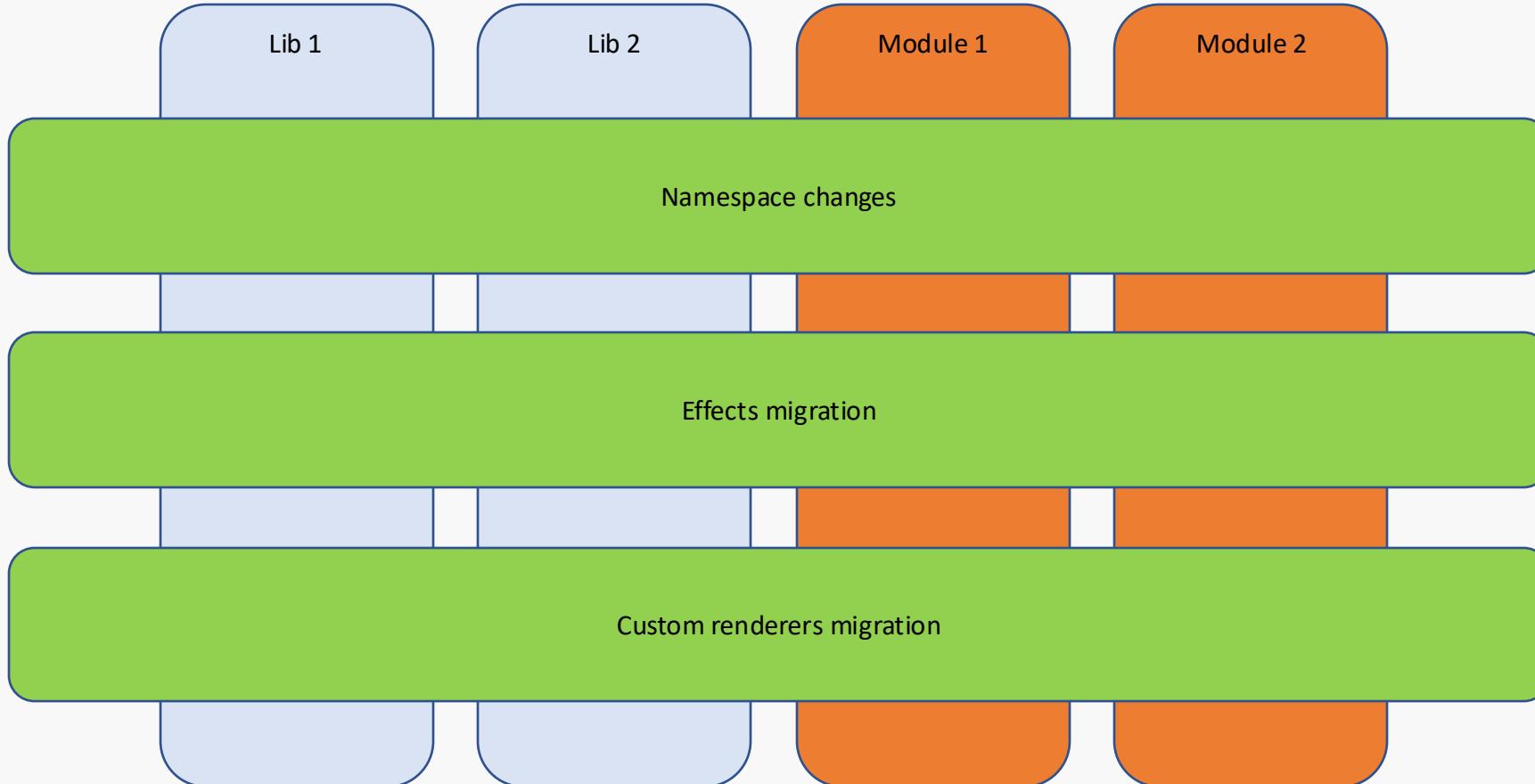
# How to plan migration?

- iOS privacy manifest

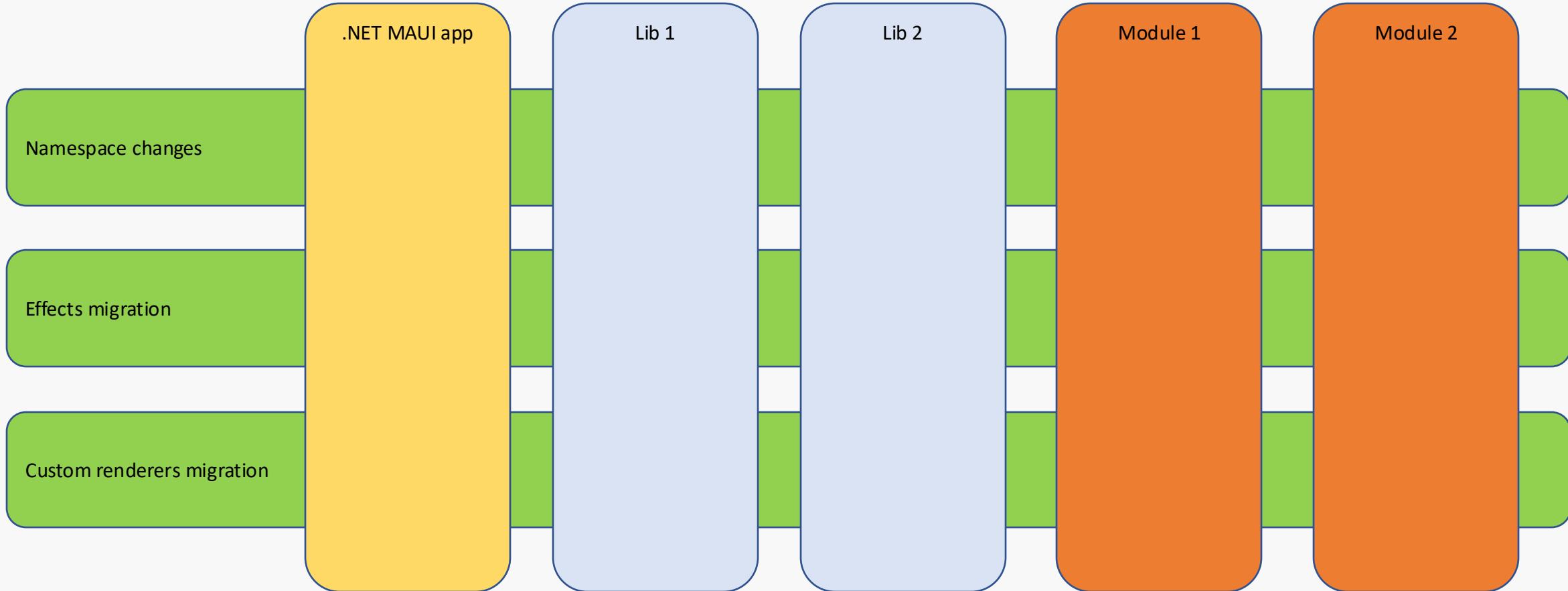
Sources:

<https://learn.microsoft.com/en-gb/dotnet/maui/ios/privacy-manifest?view=net-maui-8.0#required-reasons-api-use-in-net-maui>

# How to plan migration?



# How to plan migration?



# Testing

- Test Plan for regression
- Test Cases ready to be automated

# Performance

# Performance

- .NET MAUI is on par with Xamarin.Forms performance or slightly better
- our mobile app based on .NET MAUI is mostly on par with version based on Xamarin.Forms
- Recommended csproj configuration:
  - <UseInterpreter>False</UseInterpreter>
  - <AndroidLinkMode>SdkOnly</AndroidLinkMode>
  - <RunAOTCompilation>True</RunAOTCompilation>
  - <AndroidEnableProfiledAot>True</AndroidEnableProfiledAot>
  - <AndroidStripILAfterAOT>True</AndroidStripILAfterAOT>
  - <AndroidUseDefaultAotProfile>True</AndroidUseDefaultAotProfile>

Migration is finished!

Questions?

# Thanks!

Damian Antonowicz  
[damianantonowicz.pl](http://damianantonowicz.pl)