

ShazamKit

How to use ShazamKit on iOS and visionOS

Roland Schmitz, 29.02.2024, CocoaHeads Aachen

iOS, visionOS, ShazamKit, iPhone, iPad, AirPods, App Store, Shazam, and Apple Vision Pro are trademarks of Apple Inc.

Agenda

- Introduction
- ShazamKit
- visionOS adaption

Introduction

Motivation

- Played **Hitster** with my family on Christmas
 - <https://hitstergame.com/de-de/>
- Watched a **WWDC23 video** about ShazamKit after Christmas
 - <https://developer.apple.com/wwdc23/10051>
- Connected both and started working on "Music Uncovered"
 - Nonstop Shazam
 - Show title, artist and release year
 - Always-on song display

Introduction

"Music Uncovered" History

- 2023/12/27: Watched a **WWDC23 Video** about ShazamKit
- Started working on "Music Uncovered"
- 2024/01/08: **TestFlight** approval
- Created a **web site** with Publish, Swift, Markdown and GitHub Pages
- 2024/01/24: **App Store** approval for **iOS** and published it 🦊
- Created native version for visionOS
- 2024/01/28: **App Store** approval for **visionOS** and published it 🧐
- 2024/02/02: "**Thank You**" email from Apples Susan Prescott 😊

ShazamKit

What is it?

"ShazamKit lets you enrich your app experience by letting users find out a song's name, who sang it, the genre, and more."

<https://developer.apple.com/shazamkit/>

ShazamKit

What is it?

- Listens to your Music
- Compares with a catalog
- Identifies a matching song from the catalog
- Manage the users Shazam library

ShazamKit

What is it?

- Listens to your Music
 - using the microphone of the users device
 - using the audio stream to the users headphones or AirPods
- Compares with a catalog
- Identifies a matching song from the catalog
- Manage the users Shazam library

ShazamKit

What is it?

- Listens to your Music
- Compares with a catalog
 - Shazams catalog
 - Your own catalog
- Identifies a matching song from the catalog
- Manage the users Shazam library

ShazamKit

What is it?

- Listens to your Music
- Compares with a catalog
- Identifies a matching song from the catalog
 - Match once
 - Match continuously
- Manage the users Shazam library

ShazamKit

What is it?

- Listens to your Music
- Compares with a catalog
- Identifies a matching song from the catalog
- Manage the users Shazam library
 - Per app list of songs which can be used for favorites
 - Automatically synced to the user's devices
 - Add, remove and retrieve songs from the library

ShazamKit

Quiz

- When did the first Shazam service start?
- How much did Apple pay for Shazam?

ShazamKit

Evolution

- 1999 Shazam Entertainment Limited was founded
- 2002 Shazam starts with a phone service via # 2580 in the UK
- 2008 First iOS version of Shazam in the App Store
- 2017/2018 Apple buys Shazam for 400 million US\$
- 2021 First ShazamKit Release
- 2022 ShazamKit CLI to create custom catalogs at scale
- 2023 SHManagedSession and support music streamed to AirPods

ShazamKit

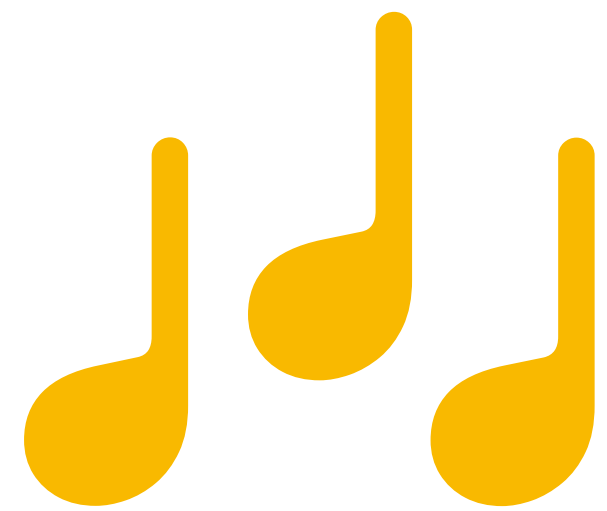
Other Sound Processing Frameworks

- Framework "Sound Analysis"
 - Classify various sounds by analyzing audio files or streams.
- Framework "Speech"
 - Perform speech recognition on live or prerecorded audio, and receive transcriptions, alternative interpretations, and confidence levels of the results.

ShazamKit

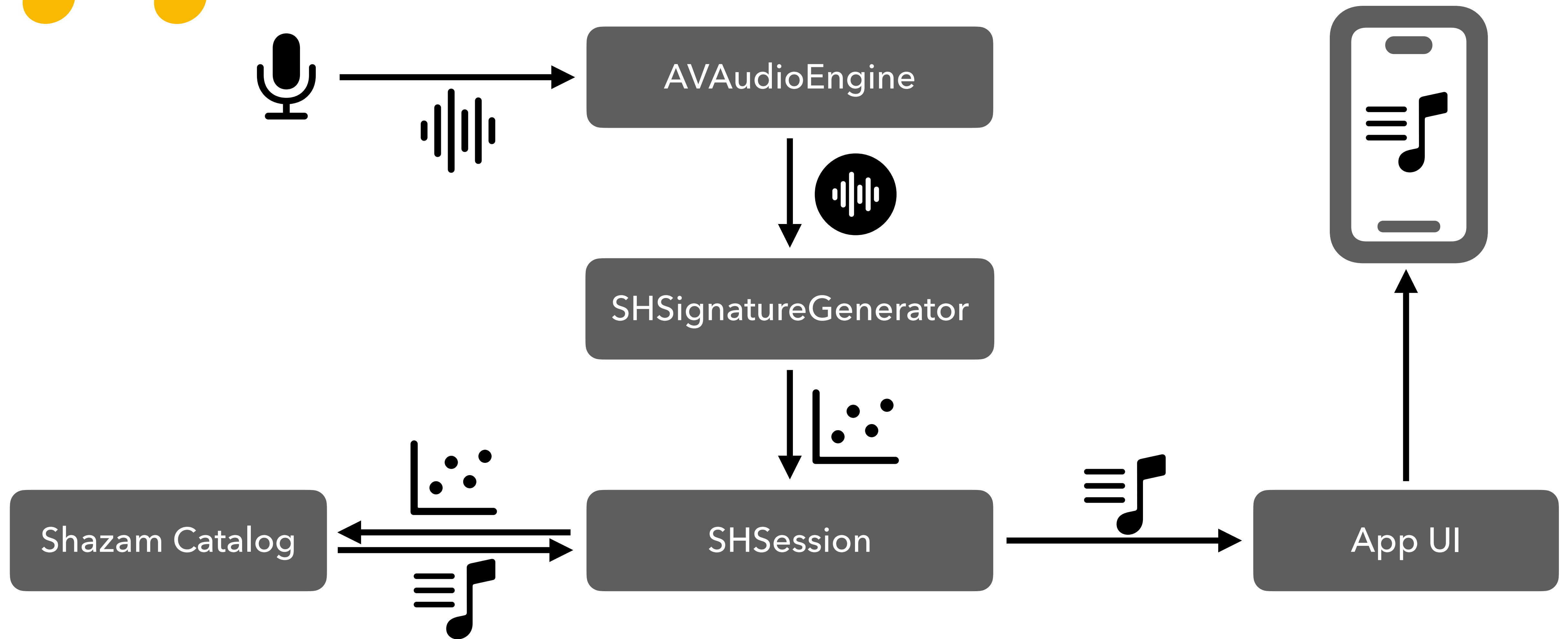
Which platforms are covered?

- iOS 15.0+, iOS 17.0+ for the newest API
- macOS 12.0+
- Mac Catalyst 15.0+
- tvOS 15.0+
- watchOS 8.0+
- visionOS 1.0+
- iPadOS 15.0+ , iOS 17.0+ for the newest API
- Android minSdkVersion 21+



ShazamKit

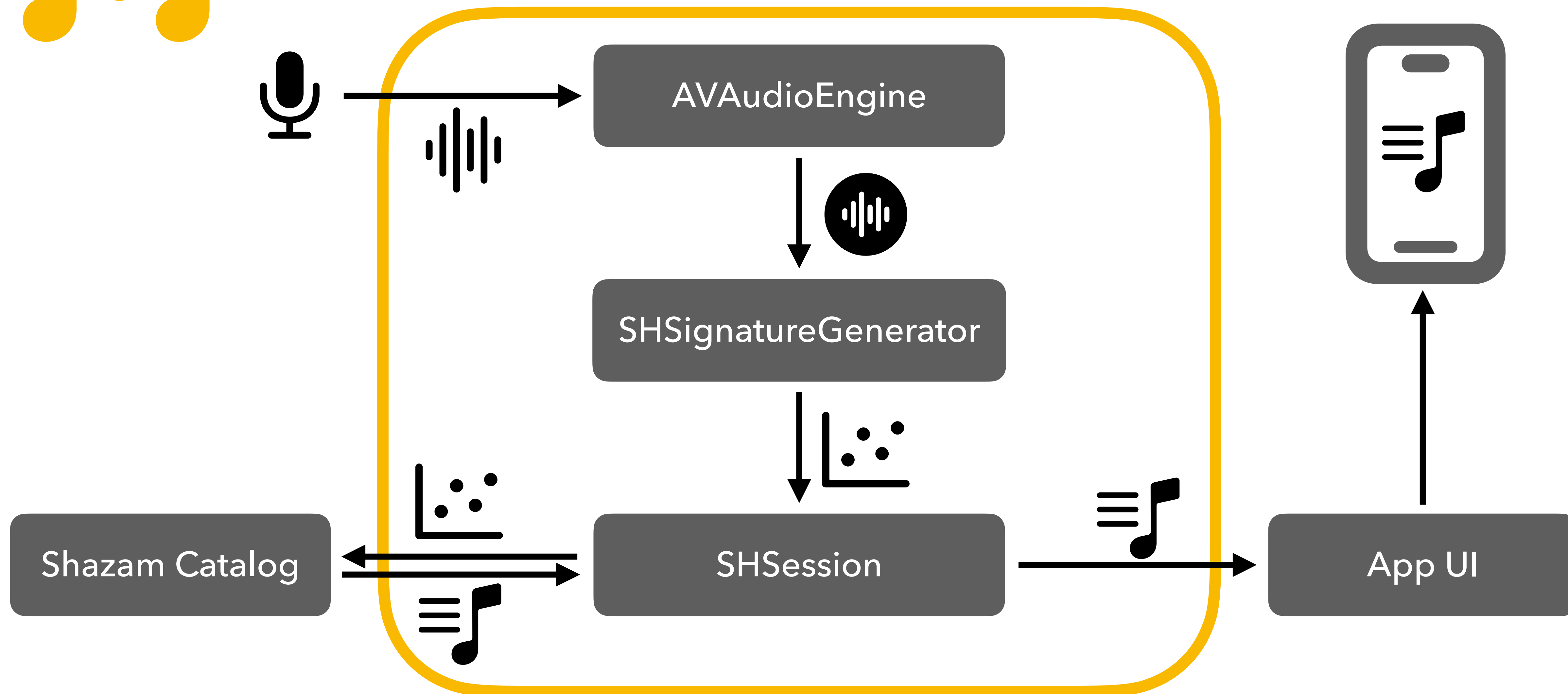
How does it work?





ShazamKit

SHManagedSession (2023)



ShazamKit

How do we use it?

- Create a managed session
- Get state
- Start prerecording (optional)
- Start matching once
- Start matching nonstop
- Stop

ShazamKit

How do we use it?

- Create a managed session
 - `@State private var shazamSession = SHManagedSession()`
- Get state
- Start prerecording (optional)
- Start matching once
- Start matching nonstop
- Stop

ShazamKit

How do we use it?

- Create a managed session
- Get state
 - `let state: SHManagedSession.State = shazamSession.state`
- Start prerecording (optional)
- Start matching once
- Start matching nonstop
- Stop

ShazamKit

How do we use it?

- Create a managed session
- Get state
- Start prerecording (optional)
 - `await shazamSession.prepare()`
- Start matching once
- Start matching nonstop
- Stop

ShazamKit

How do we use it?

- Create a managed session
- Get state
- Start prerecording (optional)
- Start matching once
 - `let result: SHSession.Result = await shazamSession.result()`
- Start matching nonstop
- Stop

ShazamKit

How do we use it?

- Create a managed session
- Get state
- Start prerecording (optional)
- Start matching once
- Start matching nonstop
 - **`for await result in shazamSession.results { ... }`**
- Stop

ShazamKit

How do we use it?

- Create a managed session
- Get state
- Start prerecording (optional)
- Start matching once
- Start matching nonstop
- Stop
 - **`shazamSession.cancel()`**

ShazamKit

Usage in a SwiftUI View

```
struct ContentView: View {  
    @State private var shazamSession = SHManagedSession()  
    var body: some View {  
        Text("**State:** \ (String(describing: shazamSession.state)) ")  
        Button("Prepare") { ... }  
        Button("Shazam Once") { ... }  
        Button("Shazam Nonstop") { ... }  
        Button("Stop") { ... }  
    }  
}
```


ShazamKit

Shazam Once

```
Button("Shazam Once") {  
    Task {  
        switch await shazamSession.result() {  
        case .match(let match):  
            print("- Title: \(match.mediaItems.first?.title ?? "-")")  
            print("- Artist: \(match.mediaItems.first?.artist ?? "-")")  
        case .noMatch(_): ...  
        case .error(let error, _): ...  
        }  
    }  
}
```

ShazamKit

Shazam Nonstop

```
Button("Shazam Nonstop") {
    Task {
        for await result in shazamSession.results {
            switch result {
            case .match(let match):
                print("- Title: \(match.mediaItems.first?.title ?? "-")")
                print("- Artist: \(match.mediaItems.first?.artist ?? "-")")
            case .noMatch(_): ...
            case .error(let error, _): ...
            }
        }
    }
}
```

ShazamKit

How to access properties of a matched media item?

```
if let mediaItem: SHMatchedMediaItem = match.mediaItems.first {  
    let title = mediaItem.title  
    let album = mediaItem.songs.first?.albumTitle  
    let artwork = mediaItem.songs.first?.artwork  
    let currentOffset = mediaItem.predictedCurrentMatchOffset  
}
```

ShazamKit

Properties of media items

- Properties contained in class SHMediaItem
 - shazamID: String?
 - title: String?
 - subtitle: String?
 - artist: String?
 - genres: [String]
 - appleMusicID: String?
 - appleMusicURL: URL?
 - webURL: URL?
 - artworkURL: URL?
 - videoURL: URL?
 - explicitContent: Bool
 - isrc: String?

ShazamKit

Properties of media items

- Music library properties contained in media item songs array:
 - artwork: Artwork?
 - albumTitle: String?
 - artistName: String
 - artistURL: URL?
 - attribution: String?
 - composerName: String?
 - duration: TimeInterval?
 - genreNames: [String]
 - hasLyrics: Bool
 - isrc: String?
 - releaseDate: Date?
 - title: String
 - composers: MusicItemCollection<Artist>?
 - musicVideos: MusicItemCollection<MusicVideo>?
 - etc.

ShazamKit

Properties of media items

- Properties contained in media item song artwork
 - url(width: Int, height: Int) -> URL?
 - backgroundColor: CGColor?
 - primaryTextColor: CGColor?
 - secondaryTextColor: CGColor?
 - tertiaryTextColor: CGColor?
 - quaternaryTextColor: CGColor?

ShazamKit

Properties of media items

- Dynamic properties contained in class SHMatchedMediaItem
 - frequencySkew: Float
 - matchOffset: TimeInterval
 - predictedCurrentMatchOffset: TimeInterval

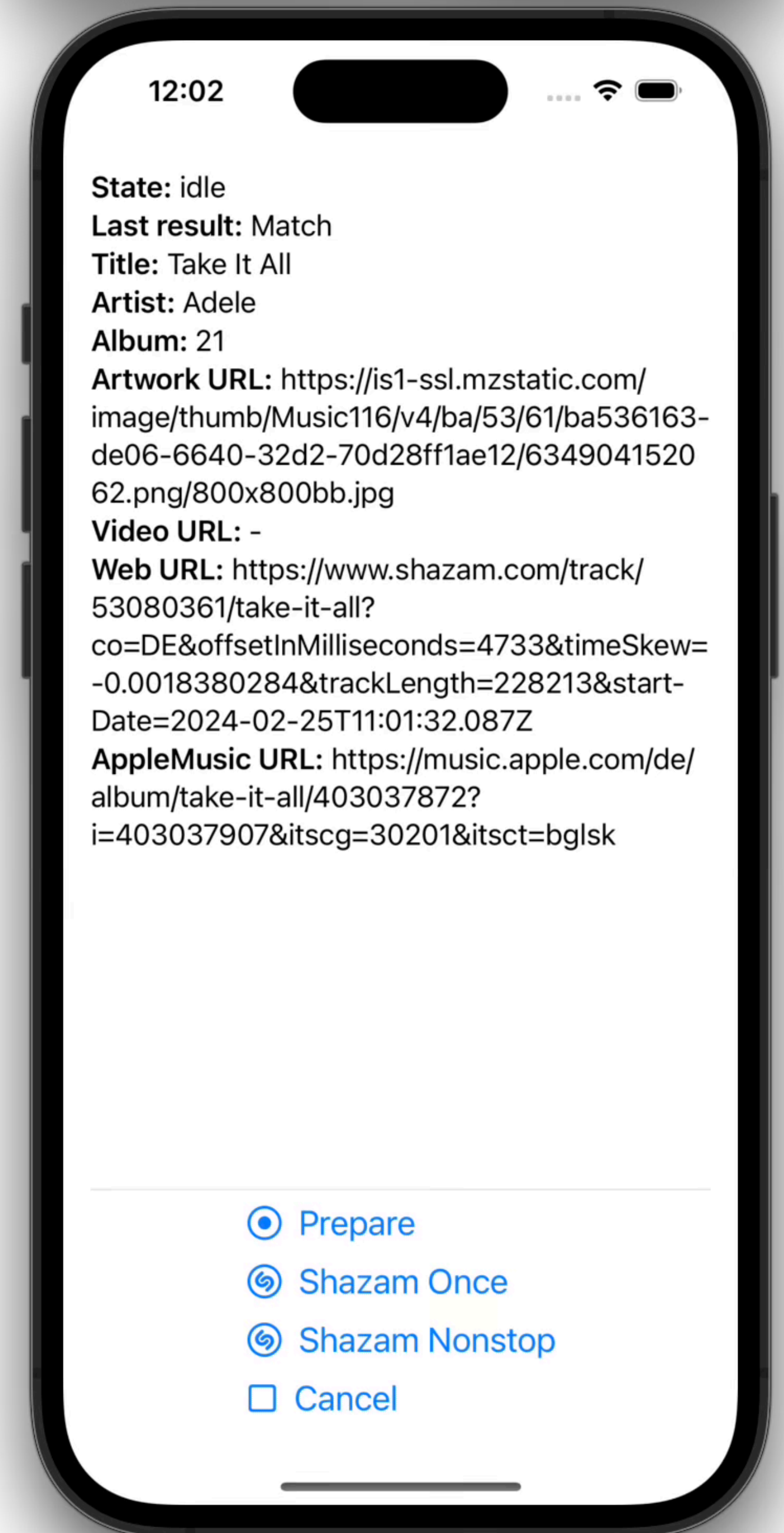
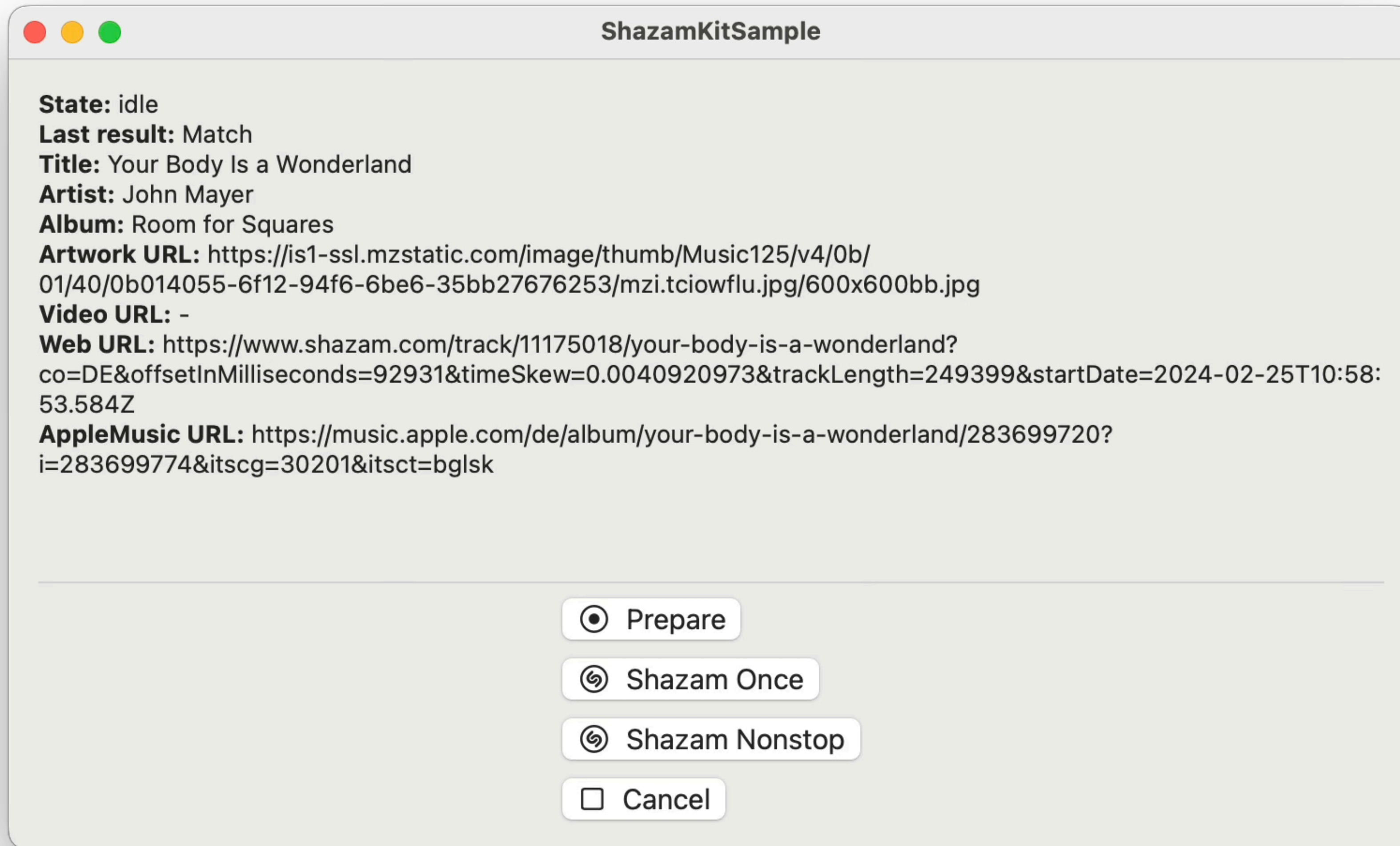
ShazamKit

Demo

- Sample App on GitHub
 - <https://github.com/roland-schmitz/ShazamKitSample>

ShazamKit

Demo



ShazamKit

Why does it not work? Preconditions.

- Check for errors. ShazamKit tells you why it doesn't work.
 - `if case .error(let error) = result { print("error: \(error)") }`
- "Shazam Nonstop" with `shazamSession.results` is not working on Intel Macs
 - Use "Shazam Once" with `shazamSession.result()` or use an Apple Silicon Mac
- Add microphone usage description to info.plist
 - Privacy - Microphone Usage Description
- Add media library usage description to info.plist
 - Privacy - Media Library Usage Description
- Ask for microphone permissions when using SHSession
 - Happens automatically, when using SHManagedSession
- Enable ShazamKit App Service for your App with your developer account
 - <https://developer.apple.com/account/resources/identifiers/bundleId/>

ShazamKit

Certificates, Identifiers & Profiles

[← All Identifiers](#)

Edit your App ID Configuration

Platform

iOS, iPadOS, macOS, tvOS, watchOS, visionOS

App ID Prefix

Description

-

Bundle ID

You cannot use special characters such as @, &, *, "

Capabilities

App Services

ENABLE

NAME

DESCRIPTION



MusicKit ⓘ

Access the Apple Music catalog and make personalized requests for authorized users.



ShazamKit ⓘ

Match audio signatures against Shazam's music catalog.



WeatherKit ⓘ

Access the Apple Weather service.

ShazamKit

Other issues

- CarPlay
 - Audio routing to car speakers stops while using ShazamKit
 - Will create a Feedback

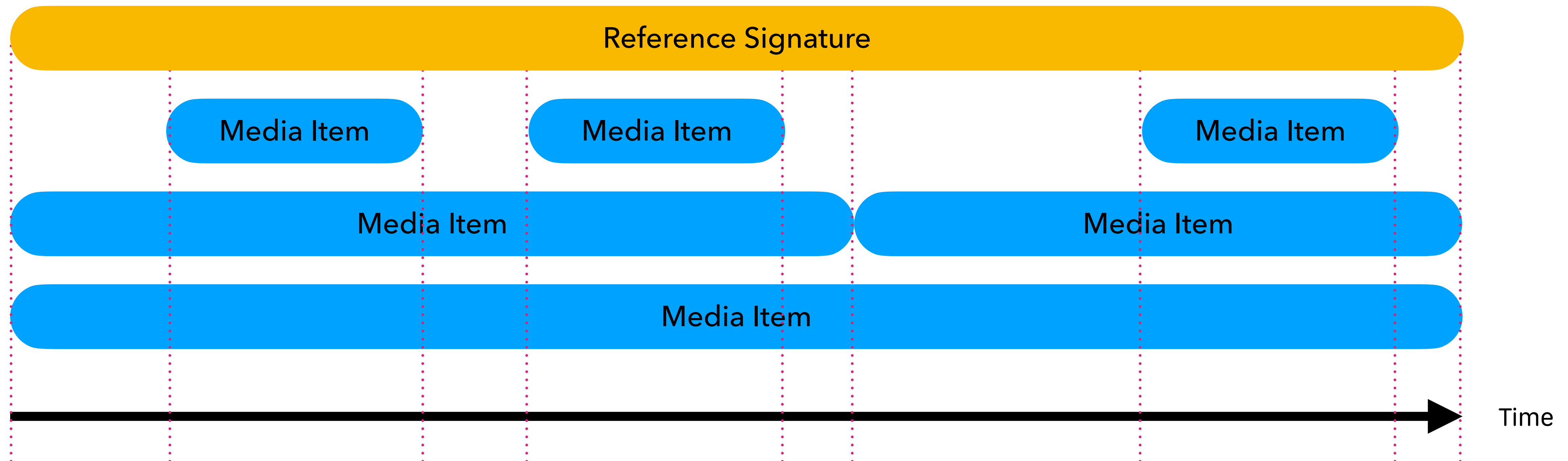
ShazamKit

Creating custom catalogs

- ShazamKit CLI is preinstalled on macOS
 - create signatures
 - create catalogs by combining signatures with media items
 - media items can be defined in a CSV file
 - columns are properties
 - rows are media items with specific time ranges

ShazamKit

Creating custom catalogs



ShazamKit

Creating custom catalogs

- ShazamKit CLI is preinstalled on macOS
 - `shazam signature -i mymedia.mp4 -o mymedia.shazamsignature`
 - `shazam custom-catalog create --help`
 - `shazam custom-catalog create --signature-asset mymedia.shazamsignature --media-items mymedia.csv --output mymedia.shazamcatalog`
 - `shazam custom-catalog update --signature-asset mymedia2.mp4 --media-items mymedia2.csv --input mymedia.shazamcatalog`

ShazamKit

Creating custom catalogs

- Or use Swift APIs:
 - SHSignatureGenerator
 - `.signature(from: AVURLAsset (url: url))`
 - SHMediaItem
 - `.init(properties: [.title: "Title", .timeRanges: [0.0.. 5.0])`
 - SHCustomCatalog
 - `.addReferenceSignature(signature, representing: mediaItems)`
 - `.write(to: URL)`
- More infos in the WWDC22 video:
 - <https://developer.apple.com/wwdc22/10028>

ShazamKit

How to match with a custom catalog

- `let catalog = SHCustomCatalog()`
 - `catalog.add(from customCatalogURL: URL)`
- `let shazamSession = SHManagedSession(catalog: catalog)`



ShazamKit

visionOS



Daydreamer's Son

Mark Gillespie

Album

Barefoot and Naked

Released in

2006



Nonstop



Once

ShazamKit

visionOS

- Add an "Apple Vision (visionOS)" destination to your target
- Conditional compilation to enable device specific features
 - only use `UIImpactFeedbackGenerator` on iOS
 - Only use depth on visionOS
- Create a new app icon with 3 layers
- Provide a separate archive for the App Store

ShazamKit

visionOS

- Use `UIImpactFeedbackGenerator` only on iOS

```
#if os(iOS)
    UIImpactFeedbackGenerator(style: .medium).impactOccurred()
#endif
```

ShazamKit

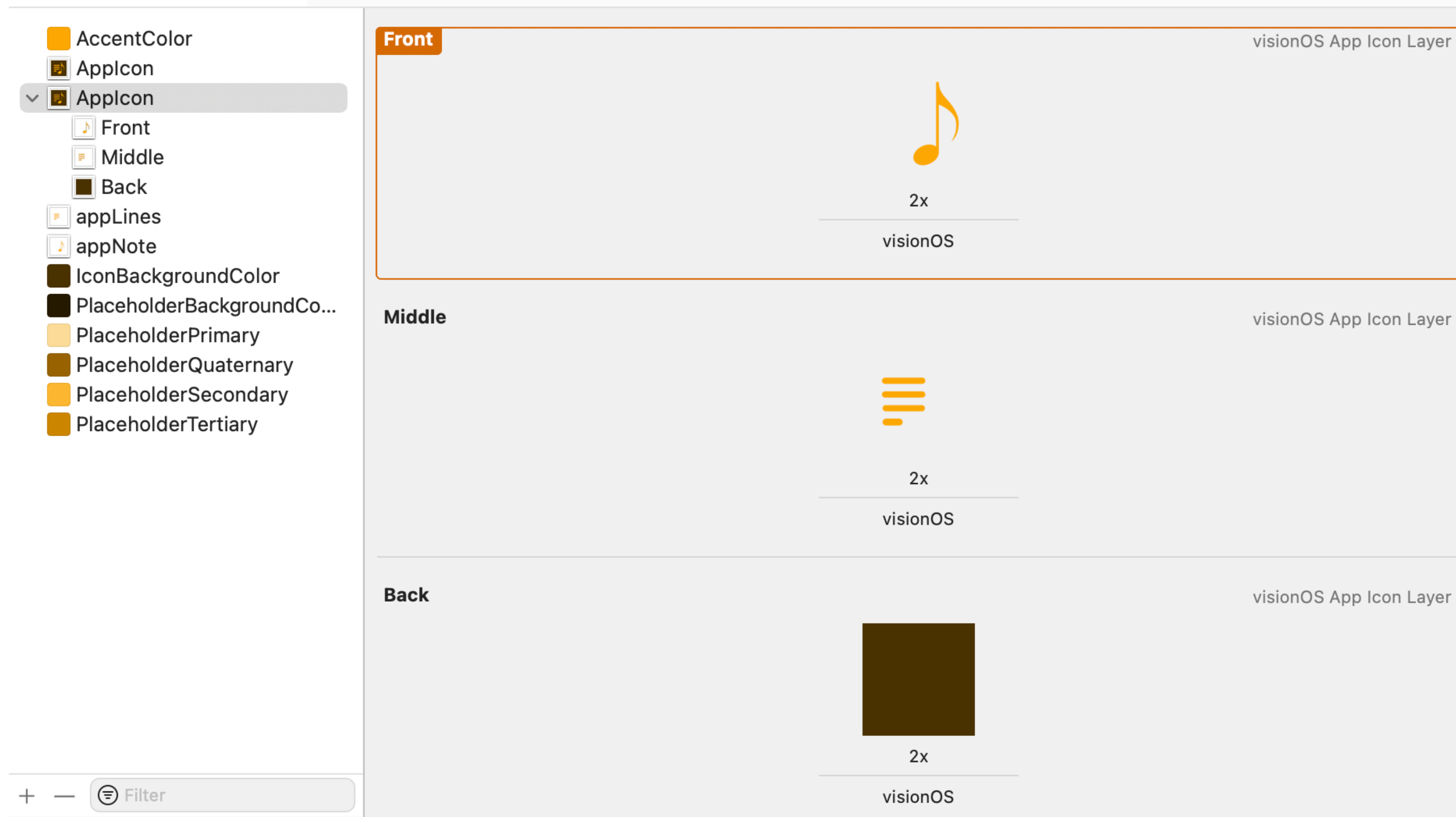
visionOS

- Use depth only on visionOS

```
extension View {  
    func visionOsOffset(z: Double) -> some View {  
        #if os(visionOS)  
            return offset(z: z)  
        #else  
            return self  
        #endif  
    }  
}
```

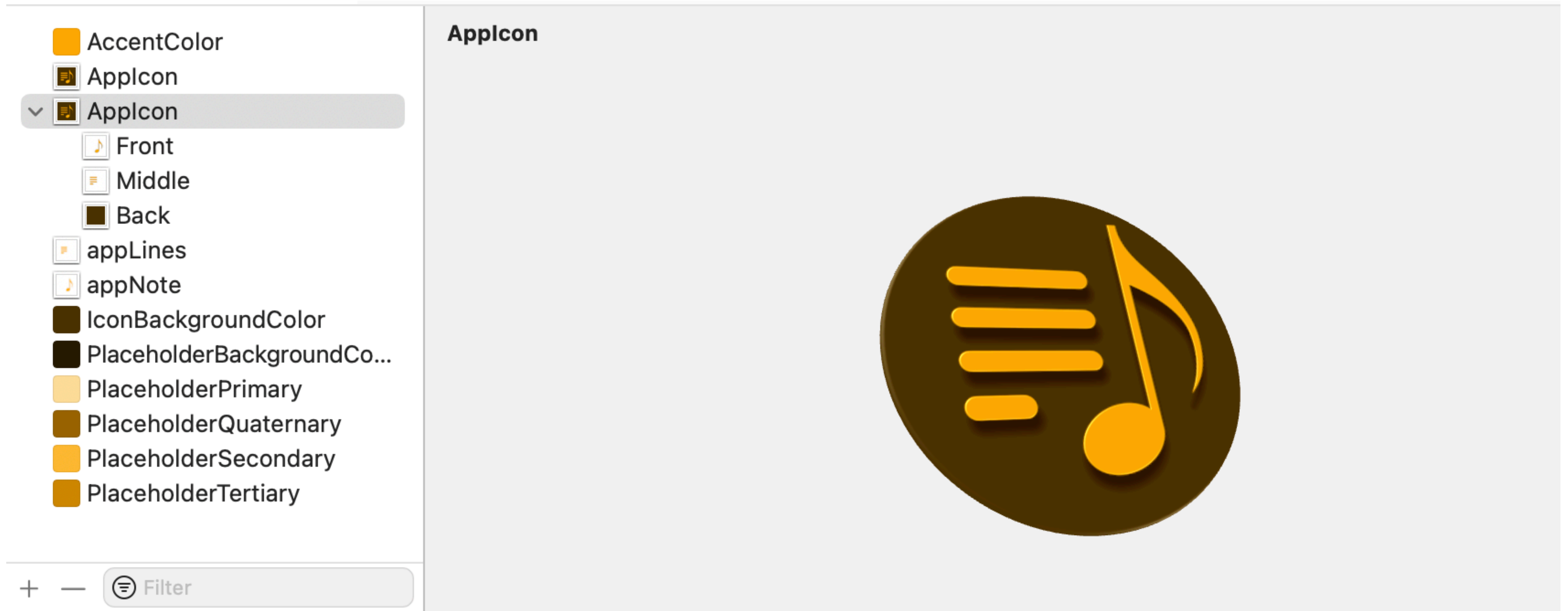
ShazamKit

visionOS



ShazamKit

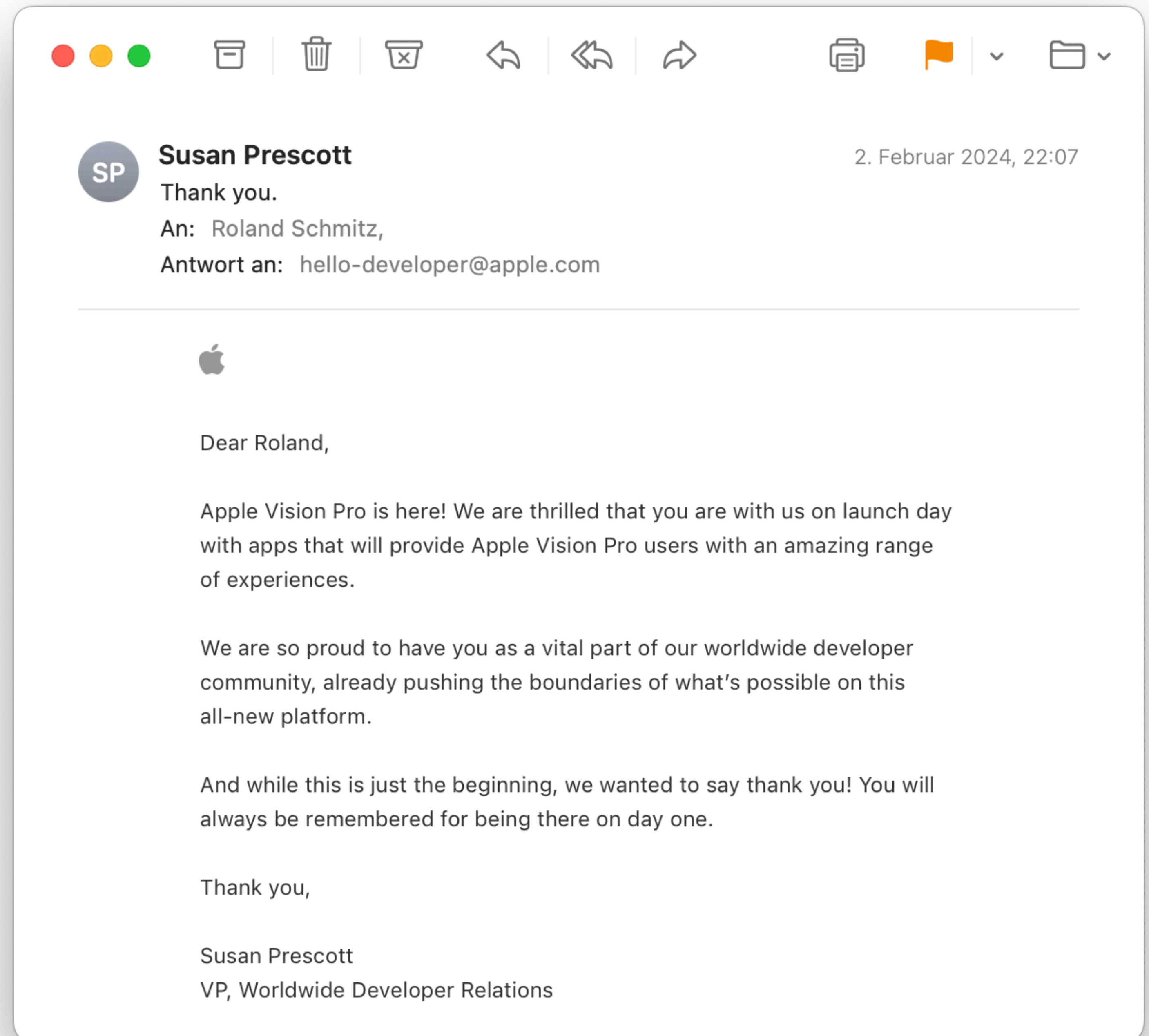
visionOS



ShazamKit

visionOS

- **Music Uncovered** was in the visionOS App Store on day one.



ShazamKit

visionOS

- Next steps
 - Use visionOS specific UI components like Ornaments
 - Create visionOS specific view layout and avoid too many GeometryReaders
 - Try it on a real device! Soon, as we have one in the office.

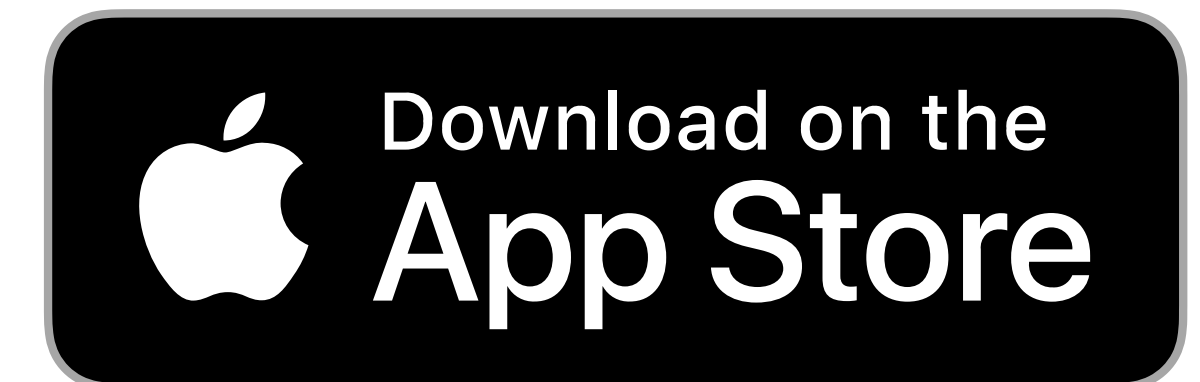
ShazamKit

Thank You

- Roland Schmitz
- Telegram: @Roland_Schmitz
- www.linkedin.com/in/roland-schmitz-8683766



Music Uncovered



<https://apple.co/49dIY9E>



ShazamKit

Interesting Links

- <https://www.appgefahren.de/shazam-jetzt-auch-songs-beim-hoeren-mit-kopfhoerern-identifizieren-351674.html>
- <https://www.heise.de/news/Shazam-in-AirPods-und-Apples-Headset-Bald-vielleicht-keine-Zukunftsmusik-mehr-7270622.html>
- <https://www.patentlyapple.com/2022/09/apple-files-a-patent-for-a-possible-new-dimension-to-shazam-for-a-wide-array-of-devices-including-an-mr-headset.html>