

Framework

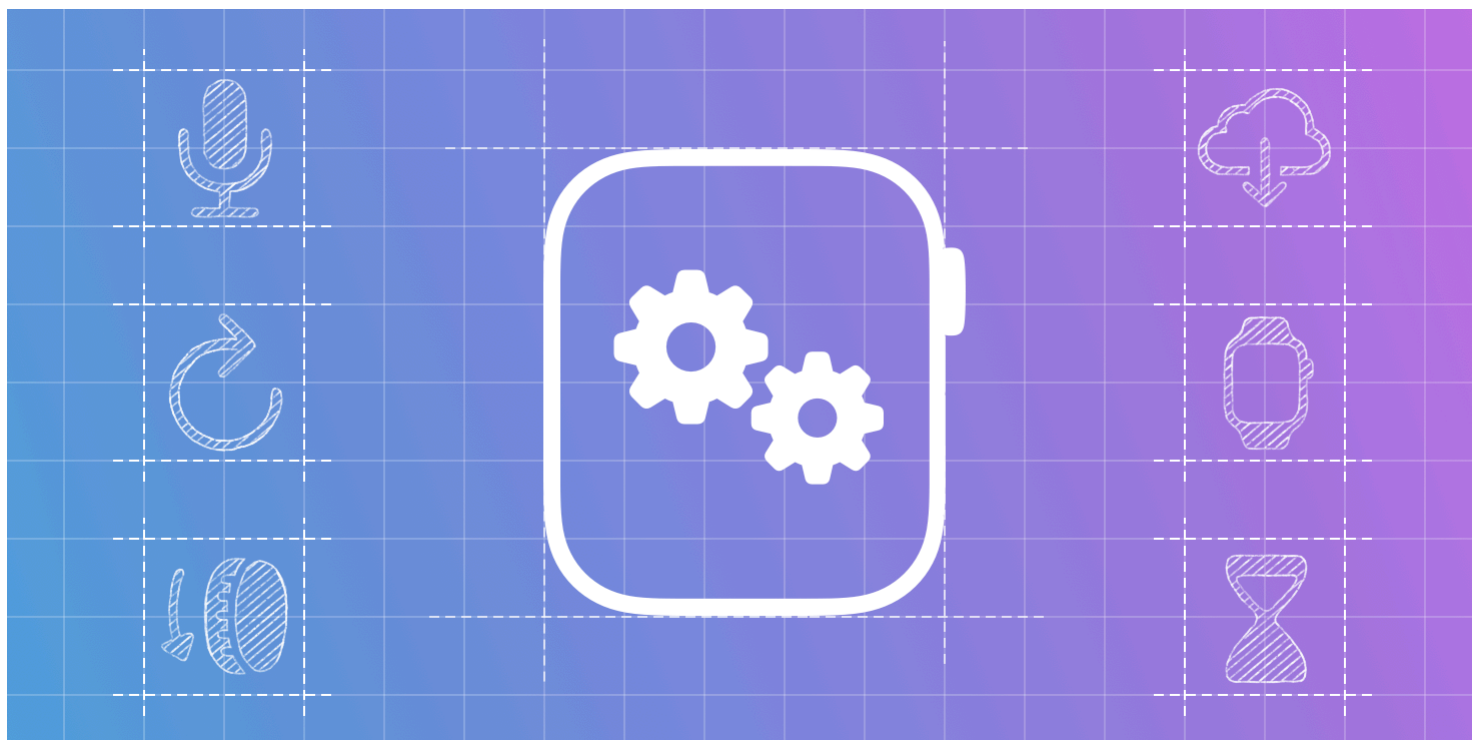
WatchKit

Build watchOS apps that use features the app delegate monitors or controls, such as background tasks and extended runtime sessions.

watchOS 2.0+

Overview

The WatchKit framework provides infrastructure for creating watchOS apps, including an extension delegate that manages background tasks, extended runtime sessions, and Siri intents. The framework also performs other support tasks, such as accessing information about the user's Apple Watch.



You can also use WatchKit to design your app's user interface in a storyboard, connecting UI elements to an interface controller.

Note

Building your app with SwiftUI gives you more control over the user interface than designing it in a storyboard. When creating a new watchOS app, strongly consider using [SwiftUI](#). For more information, see [Building a watchOS app](#).

For more information on building watchOS apps, see [watchOS apps](#).

Topics

App structure



Setting up a watchOS project

Create a new watchOS project or add a watch target to an existing iOS project.

`class WKApplication`

The centralized point of control and coordination for apps with a single watchOS app target.

`protocol WKApplicationDelegate`

A collection of methods that manages the app-level behavior for a single-target watchOS app.

`class WKExtension`

The centralized point of control and coordination for extension-based apps running in watchOS.

`protocol WKExtensionDelegate`

A collection of methods that manages the app-level behavior of a WatchKit extension.

`func WKApplicationMain(Int32, UnsafeMutablePointer<UnsafeMutablePointer<CChar>?>, String?) -> Int32`

Creates the application object and the application delegate, and sets up the app's event cycle.

`class WKInterfaceDevice`

An object that provides information about the user's Apple Watch.

`WKPrefersNetworkUponForeground`

A Boolean value that indicates whether an app requires network access on launch.

Runtime management

- ☰ Background execution
Manage background sessions and tasks.
- ☰ Life cycles
Receive and respond to life-cycle notifications.
- 📄 Using extended runtime sessions
Create an extended runtime session that continues running your app after the user stops interacting with it.

`class WKExtendedRuntimeSession`

A session that continues to run your app after the user has stopped interacting.

- { } Interacting with Bluetooth peripherals during background app refresh
Keep your complications up-to-date by reading values from a Bluetooth peripheral while your app is running in the background.

User interface

- ☰ Storyboard support
Connect your code to storyboard elements using interface controllers, interface objects, and event handlers.

`struct NowPlayingView`

A view that displays the system's Now Playing interface so that the user can control audio.

Errors

`struct WatchKitError`

An error reported by WatchKit.