

Swift Package Manager

PMDM - iOS Programación multimedia y dispositivos móviles

Ariel Hernández, Febrero 2021

Swift Package Manager

- Paquete (o *namespace*): Conjunto de archivos de código que puede ser compilado, distribuido y compartido con otros proyectos.
- *Package Manager*: Monitoriza cambios en los paquetes usados por un proyecto.
- *Cocoapods* y *Carthage* son generalmente los más famosos y usados.
- **Swift Package Manager**: Anunciado en 2018, desarrollado en y para Swift y Xcode

Swift Package Manager

- Paquete (o *namespace*): Conjunto de archivos de código que puede ser compilado, distribuido y compartido con otros proyectos.
- *Package Manager*: Monitoriza cambios en los paquetes usados por un proyecto.
- *Cocoapods* y *Carthage* son generalmente los más famosos y usados.

Swift Package Manager

- **Swift Package Manager:** Desarrollado en y para Swift y Xcode.
- Desarrollado oficialmente por Apple.
- Anunciado en 2018.
- Open Source.
- Built In en Xcode y macOS.

SPM: Crear paquetes

- Incluimos `GridLayout.swift` y `GridView.swift`
- Por convención todo el código se incluye en una carpeta `Source`
- En la raíz del proyecto se incluye un fichero `Package.swift` con la descripción del proyecto y las instrucciones para su distribución.

```
my-example-package
├── Sources
│   └── Grid
│       ├── GridLayout.swift
│       └── GridView.swift
└── Package.swift
```

SPM: Package.swift

- Describe el paquete a distribuir.
- `products` describe todos “binarios” compilados distribuidos por el paquete, pueden ser [`.library`, `.executable`]

Deben incluir que targets incluye.

- `targets` define “bloques compilados” como unidad. Puede tener dependencias de otros target, o de paquetes externos.
- `dependencies` define otros paquetes de los que depende una o varias Target, la versión puede especificarse exactamente, o en rango.

```
import PackageDescription

let package = Package(
    name: "Grid",
    products: [
        .library(
            name: "Grid",
            targets: ["Grid"]),
    ],
    dependencies: [
        .package(
            url: "http://.../Alamofire",
            exact("5.4.1")),
    ],
    targets: [
        .target(
            name: "Grid",
            dependencies: ["Alamofire"]),

        .testTarget(
            name: "GridTests",
            dependencies: ["Grid"]),
    ]
)
```

SPM: Versionado Semántico

- Define un sistema de versiones de 3 niveles:
v 1.2.3 \Rightarrow MAJOR.MINOR.PATCH
 - Major: Versión incompatible con versiones anteriores.
 - Minor: Versión que incluye funcionalidades nuevas, compatible con otras versiones anteriores (mismo Major)
 - Patch: Versión que arregla errores y mantiene compatibilidad.
- Una versión nueva siempre será mayor que la anterior.

SPM: Versionado Semántico

- Una vez la versión ha sido publicada, no puede ser cambiada, cualquier cambio define una nueva versión
- La versión `0.x.y` está reservada para desarrollo inicial y puede contener cambios incompatibles entre cualquiera de sus versiones “menores”.
- La versión `1.0.0` será la primera publicación oficial .
- Una versión pre-release se denotará por una etiqueta al final: `v 1.2.3-alpha`
- Una versión puede incluir información del commit o binario que la produjo, esta se añade al final con un “+” como prefijo: `v 1.2.3-alpha+295fc79`

Demo

- Swift Package Manager
- Alamofire
- SDWebImage
- App icon