



POLITÉCNICA

ETSIT
UPM

dit
UPM

Desarrollo de Apps para iOS Iconos e Imágenes

IWEB 2015-2016
Santiago Pavón

ver: 2015.09.23

Imágenes

- Las aplicaciones iOS usan varios tipos de iconos e imágenes:
 - Iconos para el escritorio,
 - Iconos para los Ajustes
 - Iconos para el Spotlight
 - Imágenes de lanzamiento
 - Imágenes para iTunes (distribución)
 - y todas las imágenes que usa internamente la aplicación.
- Estos iconos e imágenes deben proporcionarse en varios tamaños en función del tipo de dispositivo, tamaño de la pantalla, tipo de pantalla, versión de iOS, ...
 - iPhone sin pantalla retina de 3.5'
 - iPhone con pantalla retina de 3.5" o 4"
 - iPad mini no retina de 7".
 - iPad con y sin pantalla retina de 9.7"
 - iOS6 o anterior, o iOS 7 o posterior

Convenio de Nombres

- Para un mismo icono o imagen se proporcionan varios ficheros.
 - Cada fichero contiene una versión del icono o de la imagen pero con un tamaño distinto.
 - El nombre de todos estos ficheros es igual pero añadiendo un sufijo diferenciador.
- Se usa:
 - **@2x** para indicar versión para pantalla retina
 - **-568h@2x** es un iPhone Retina de 4".
 - **~ipad** para indicar que es para iPad
 - ...

Iconos

- **Escritorio:**

- iPhone 3: **Icon.png** (57x57 px)
- iPhone 4 Retina: **Icon@2x.png** (114x114 px iOS 5 y 6) (120x120 px iOS 7)
- iPad: **Icon-72.png** (72x72 px iOS 5 y 6) (76x76 px iOS 7)
- iPad Retina: **Icon-72@2x.png** (144x144 px iOS 5 y 6) (152x152 px iOS 7)

- **Spotlight:**

- iPhone 3: **Icon-Small.png** (29x29 px)
- iPhone 4 Retina: **Icon-Small@2x.png** (58x58 px iOS 5 y 6) (80x80 px iOS 7)
- iPad: **Icon-Small-50.png** (50x50 px iOS 5 y 6) (40x40 px iOS 7)
- iPad Retina: **Icon-small-50@2x.png** (100x100 px iOS 5 y 6) (80x80 px iOS 7)

- **Ajustes:**

- iPhone 3: **Icon-Small.png** (29x29 px)
- iPhone 4 Retina: **Icon-Small@2x.png** (58x58 px iOS 5, 6 y 7)
- iPad: **Icon-Small.png** (29x29 px iOS 5, 6 y 7)
- iPad Retina: **Icon-Small@2x.png** (58x58 px iOS 5, 6 y 7)

- **iTunes** (png sin la extensión):

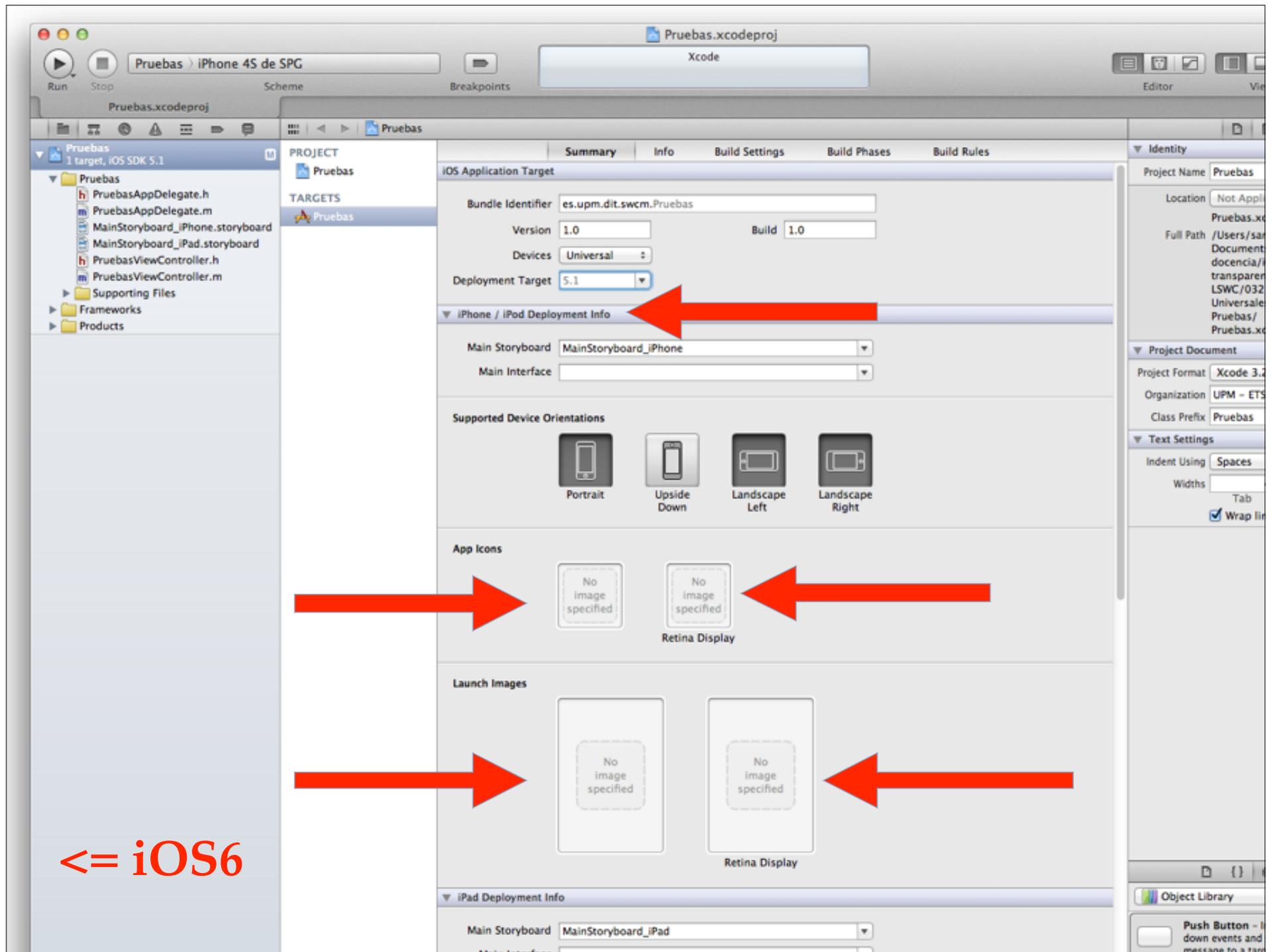
- iTunesArtWork: **512x512 px**
- iTunesArtWork@2x: **1024x1024 px**

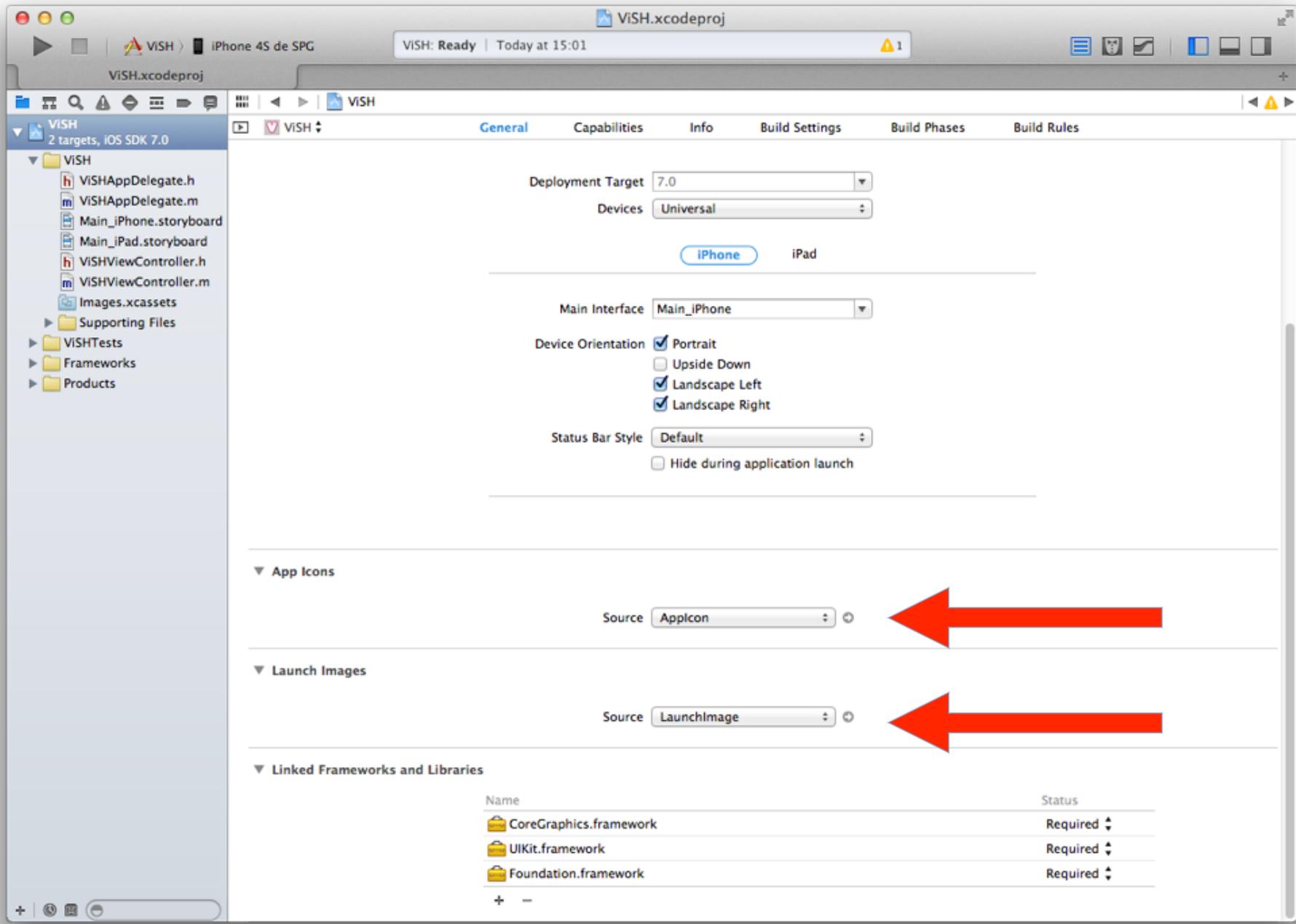
Imágenes de Lanzamiento

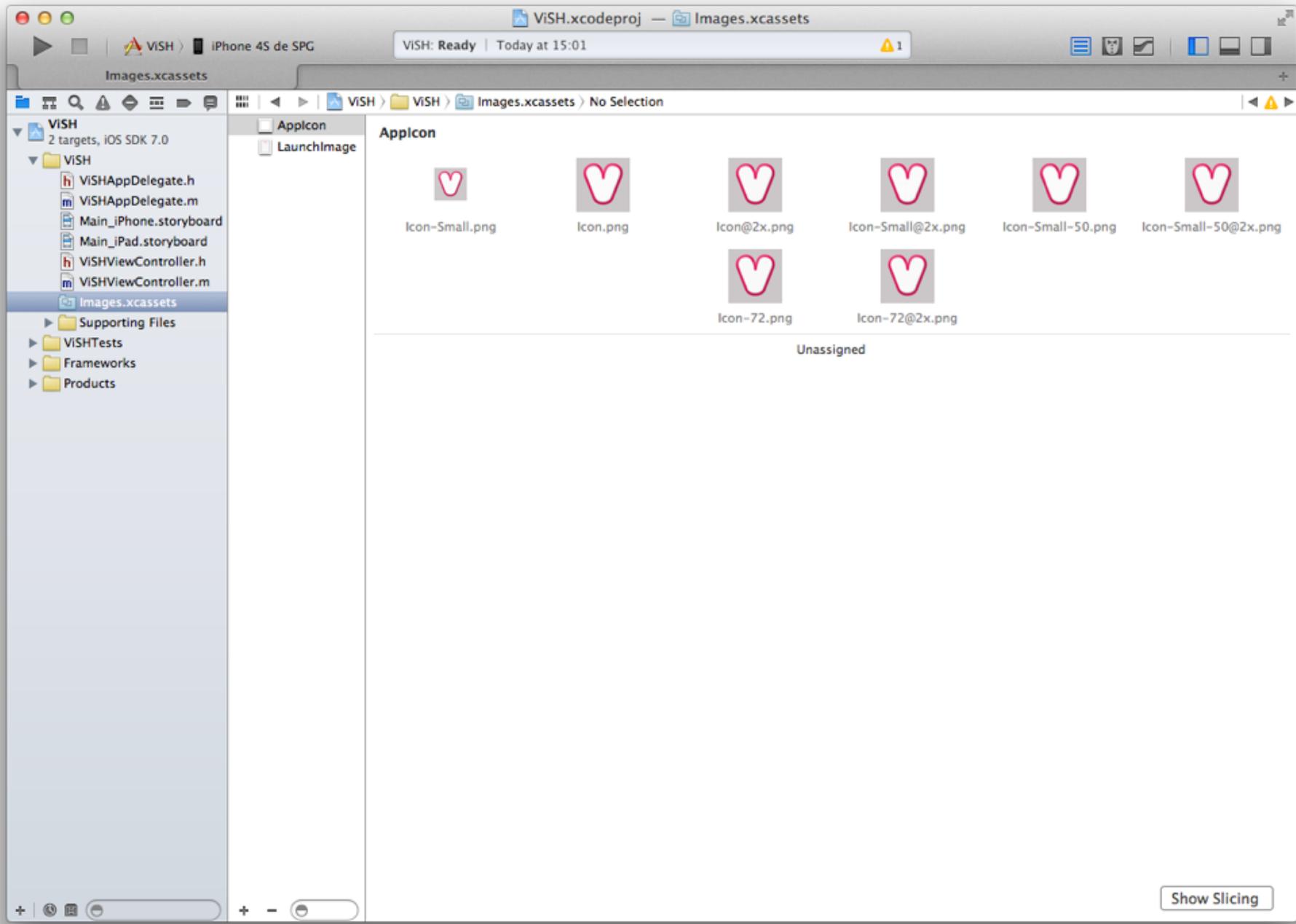
- **Default.png** - iPhone 3
 - (~~320x480 px iOS 5 y 6~~)
- **Default@2x.png** - iPhone Retina 3.5-inch
 - (640x960 px iOS 5, 6 y 7)
- **Default-568h@2x.png** - iPhone Retina 4-inch
 - (640x1136 px iOS 5, 6 y 7)
- **Default-Portrait-ipad.png** - iPad
 - (768x1024 px iOS 5 y 6 sin barra de estado, y iOS 7)
 - (768x1004 px iOS 5 y 6 con barra de estado)
- **Default-Landscape-ipad.png** - iPad
 - (1024x768 px iOS 5 y 6 sin barra de estado, y iOS 7)
 - (1024x748 px iOS 5 y 6 con barra de estado)
- **Default-Portrait@2x-ipad.png** - iPad Retina
 - (1536x2048 px iOS 5 y 6 sin barra de estado, y iOS 7)
 - (1536x2008 px iOS 5 y 6 con barra de estado)
- **Default-Landscape@2x-ipad.png** - iPad Retina
 - (2048x1536 px iOS 5 y 6 sin barra de estado, y iOS 7)
 - (2048x1496 px iOS 5 y 6 con barra de estado)

Iconos e Imágenes de Lanzamiento

- Los iconos e imágenes de lanzamiento a usar se pueden indicar en el fichero Info.plist.
- Pero es más cómodo hacerlo interactivamente editando la configuración del proyecto.
 - Hasta iOS 6 se podían arrastrar las diferentes versiones de los iconos e imágenes hasta su hueco correspondiente en la pantalla de edición del proyecto de Xcode.
 - Con iOS 7 es más fácil usar un catálogo de activos (assets) .







Catálogo de Activos

(Asset Catalog)

¿Para qué sirve?

- Organizar diferentes versiones de una imagen (normal, retina, versión para iPad, versión para iPhone de 3.5 y 4 pulgadas, iconos de escritorio, ajustes, spotlight, iTunes, ...) bajo un mismo nombre.
 - Todos estos ficheros se gestionan más cómodamente si se meten en un catálogo.
 - Tenemos agrupados juntos todos los tamaños de una misma imagen.
 - Los nombres de las imágenes ya no tienen que incluir los sufijos @2x, ~iphone, ~ipad, -568h, ...
- Un catálogo incluye:
 - los iconos e las imágenes iniciales
 - y las imágenes usadas en las pantallas de la aplicación. (*.png)
- Soportados en iOS 6 y posteriores.
- Xcode compila los catálogos generando un binario que disminuye el tiempo de carga en la aplicación.
- Al instalar una app en el terminal solo se descargan las imágenes adecuadas para él.

¿Cómo se usa?

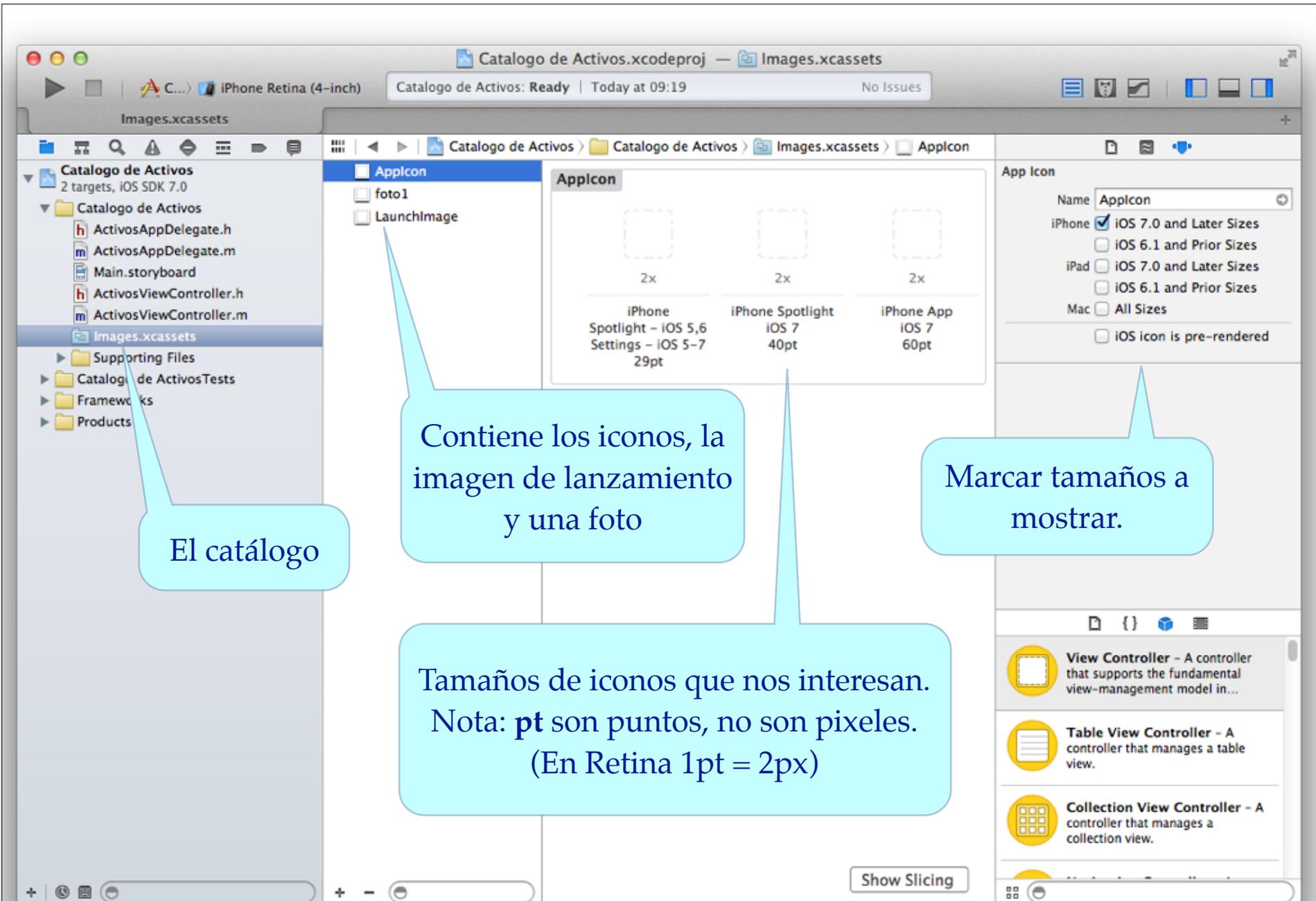
- Las plantillas de Xcode 5 (y posteriores) ya incluyen un catálogo creado.
 - Para crear un catálogo en proyectos antiguos que no lo tengan:

`Menu File > New > File > Resource > Asset Catalog`

- Para añadir al catálogo imágenes que aun no están en el proyecto
 - Pulsar el botón **+** y luego seleccionar **Import**.
- Para mover imágenes que ya están en el proyecto al catálogo:
 - Pulsar el botón **+** y seleccionar **Import from project**.
 - Se mueven al catálogo.
 - O arrastrar los ficheros de imágenes al navegador de ficheros, sobre la carpeta del catálogo.
- Las imágenes se cargan en la aplicación igual independientemente de si están en un catálogo o fuera de él:

```
let img: UIImage? = UIImage(named: "nombre")
```

- primero se buscan en los catálogos.



Troceado de Imágenes (Slicing)

Sliced Images

- Permiter usar imágenes pequeñas en zonas donde se necesitan imágenes grandes.
 - indicando que zona de la imagen pequeña se debe estirar o repetir para crear la imagen grande.
- Las imágenes se editan gráficamente en Xcode para delimitar cuatro zonas:
 - Se sitúan unos delimitadores horizontales y/o verticales para separar las zonas
 - Principio de la imagen
 - Esta zona se pinta igual en la imagen grande generada.
 - Zona con el relleno de la imagen.
 - Se repite (tiled) o estira (stretched) para pintar la imagen grande.
 - Zona a eliminar de la imagen al rellenar.
 - No se pinta en la imagen estirada.
 - Final de la imagen
 - Esta zona se pinta igual en la imagen grande generada.
- Usar el **inspector** de atributos, sección **Slicing** para configurar las direcciones sobre las que se aplicará el troceado, las zonas, la forma de pintar usar la zona central, ...

