

Fundamentos iOS





Temario

- Camara
- Fotos de la galería
- Animaciones

Camara

- Manera sencilla UIImagePickerController
- Un ViewController que maneja la interfaz con el sistema para tomar fotografías (antes podía utilizarse para acceder a la galería del usuario, pero esto se deprecó en iOS14)
- Necesita *Privacy - Camera Usage Description* definido en Info.plist
- Pasos: Inicializar > set sourceType = .camera > presentar > llamada al delegate



PHPickerViewController

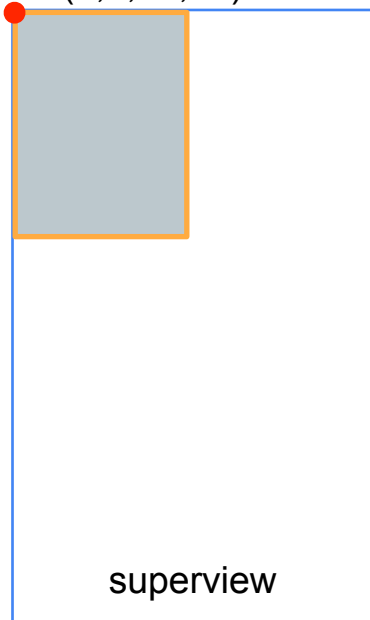
- Para importar imágenes de la galería a la app:
 `PHPickerViewController`
- Un ViewController que provee la interfaz con el sistema para el acceso a la galería del usuario
- iOS 14+



Animaciones - frame vs bounds

- frame: coordenadas del menor rectángulo que contiene a la vista, relativas a la vista contenedora
- bounds: coordenadas de la vista relativas a si misma

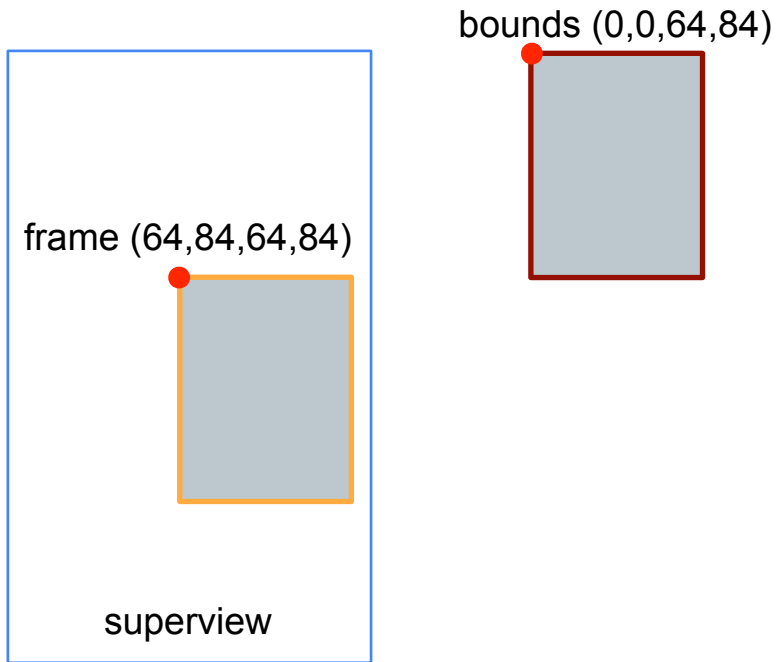
frame (0,0,64,84)



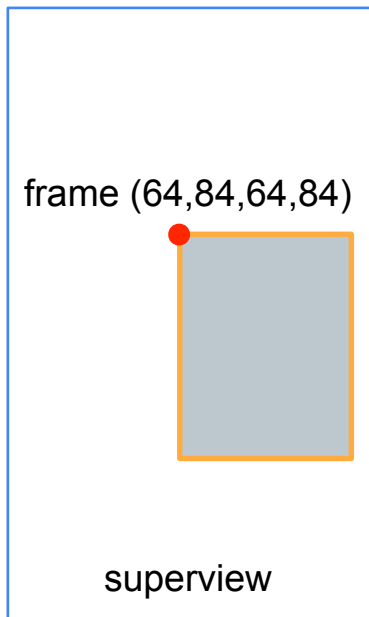
bounds (0,0,64,84)



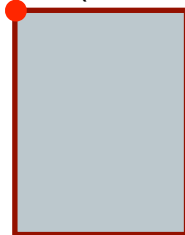
Animaciones - frame vs bounds



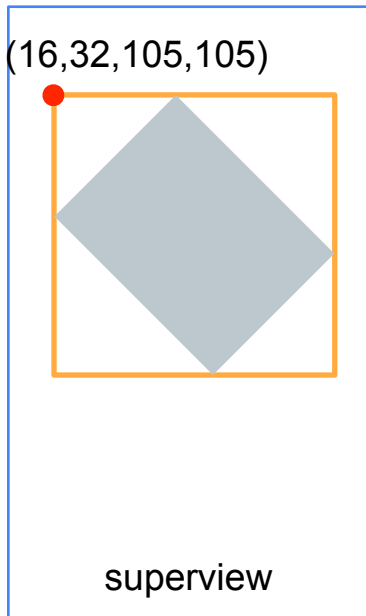
Animaciones - frame vs bounds



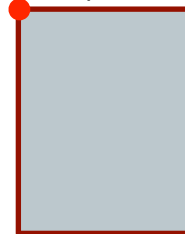
bounds (0,0,64,84)



frame (16,32,105,105)



bounds (0,0,64,84)



Animaciones

- `UIView.animate(withDuration duration: TimeInterval, animations: @escaping () -> Void, completion: ((Bool) -> Void)? = nil)`
- `UIView.animate(withDuration duration: TimeInterval, delay: TimeInterval, options: UIView.AnimationOptions = [], animations: @escaping () -> Void, completion: ((Bool) -> Void)? = nil)`



Animaciones - Options

- `.curveLinear`: por defecto, ningún cambio en el tiempo de la animación
- `.curveEaseIn`: la animación comienza despacio y acelera
- `.curveEaseOut`: la animación comienza normal y desacelera
- `.curveEaseInOut`: comienza despacio, se acelera y se desacelera sobre el final de la animación



Animaciones

- `UIView.animate(withDuration duration: TimeInterval, delay: TimeInterval, usingSpringWithDamping dampingRatio: CGFloat, initialSpringVelocity velocity: CGFloat, options: UIView.AnimationOptions = [], animations: @escaping () -> Void, completion: ((Bool) -> Void)? = nil)`



Animaciones - Options

- `usingSpringWithDamping`: acepta valores entre 0.0 y 1.0, donde 0.0 hace que tenga más rebote y 1.0 que tenga mayor “rigidez”
- `initialSpringVelocity`: controla la velocidad inicial de la animación, un valor de 1 equivale a decirle que toda la animación se realice en 1seg, igualmente, esta velocidad es luego sobrescrita por el calculo del Spring duration.



Animaciones - Constraints

- Hacer el cambio del constant del constraint fuera del bloque de `UIView.animate()` e incluir `self.view.layoutIfNeeded()` dentro del bloque.
- ```
UIView.animate(withDuration: 1.0): {
 self.view.layoutIfNeeded()
}
```





# KEEPCODING

## Tech School

Madrid | Barcelona | Bogotá

**Datos de contacto**