

Fundamentos iOS





Temario



- Ciclo de Vida de un UIViewController
- MVC
- UITableViewController
- UICollectionViewController
- Delegados
- Observadores

UIViewController - Ciclo de vida

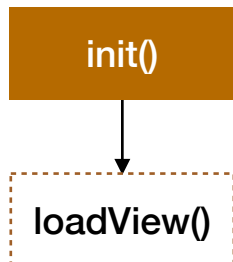
Como todos los type en Swift, los UIViewControllers comienzan con un `init()`

`init()`

- `init(coder:)` si el ViewController es creado utilizando storyboards
- `init(nibName:bundle:)` si es creado programáticamente o por medio de xibs

Es llamado **solo una vez** durante todo el ciclo de vida del controlador

UIViewController - Ciclo de vida

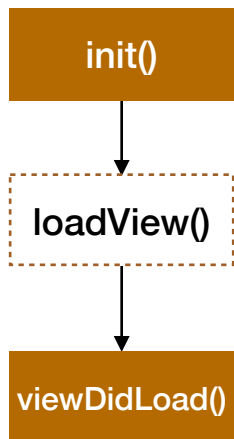


Se llama al momento de cargar la vista en memoria

- Es el método que carga la vista para el ViewController
- Útil cuando el ViewController es creado de manera programática
- En la implementación base de UIViewController, es donde carga la interfaz y se conectan los outlets y actions automáticamente, cuando se utilizan storyboards o xibs

Es llamado **solo una vez** durante todo el ciclo de vida del controlador

UIViewController - Ciclo de vida

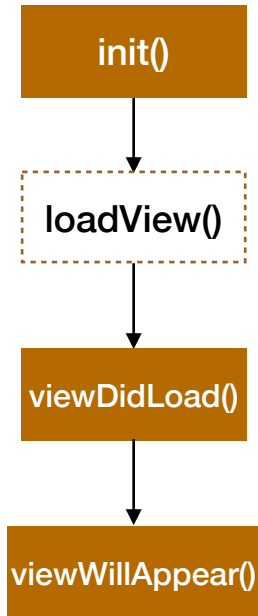


Es llamado cuando la vista es creada

- Los bounds de la vista no son finales
- Los elementos de la vista ya están creados y accesibles

Es llamado **solo una vez** durante todo el ciclo de vida del controlador

UIViewController - Ciclo de vida

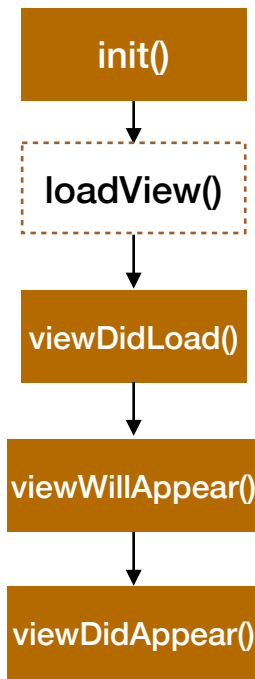


Se llama cada vez que la vista está por agregarse en la jerarquía de vistas

- Útil para realizar tareas que son necesarias cada vez que se esté por mostrar una pantalla

Es llamado **más de una vez** durante el ciclo de vida del controlador

UIViewController - Ciclo de vida

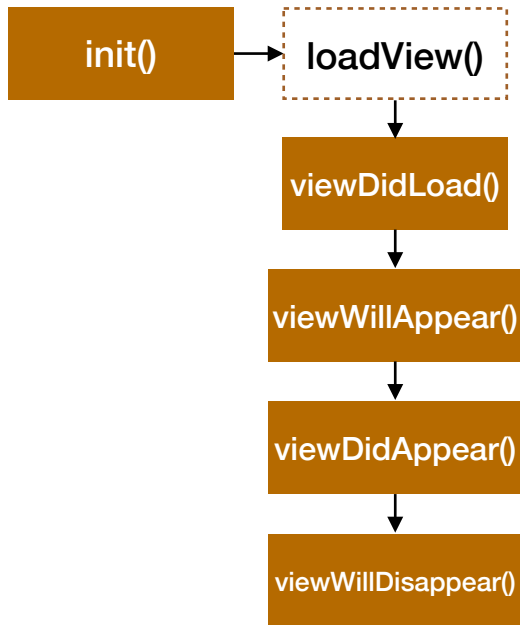


Se llama cada vez que la vista es agregada a la jerarquía de vistas

- Útil para realizar tareas que son necesarias cada vez que se muestra una pantalla, como comenzar animaciones o reproducir un video automáticamente

Es llamado **más de una vez** durante el ciclo de vida del controlador

UIViewController - Ciclo de vida

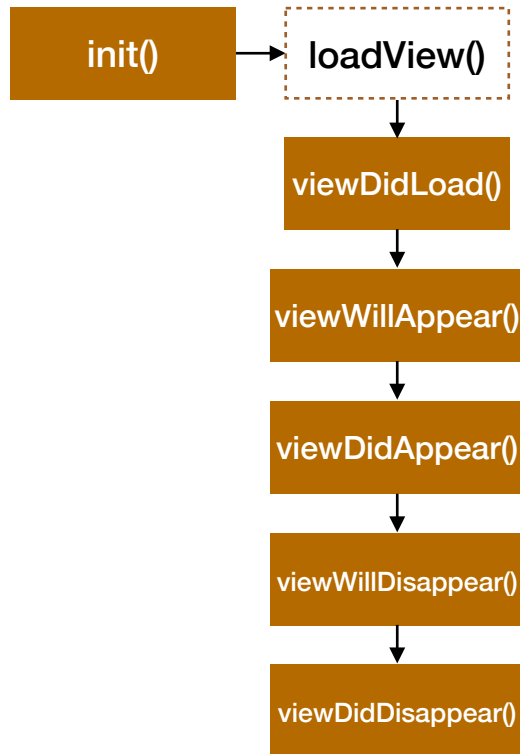


Contraparte de viewWillAppear()

- Ocurre antes de que el próximo ViewController se muestre en pantalla
- Sobre escrito en muy raras ocasiones

Es llamado **más de una vez** durante el ciclo de vida del controlador

UIViewController - Ciclo de vida

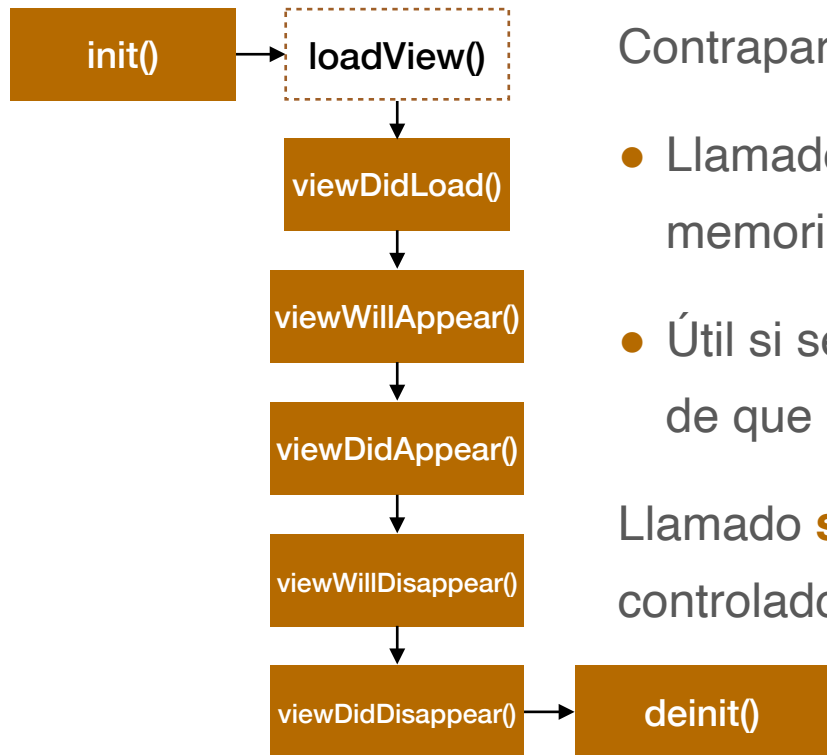


Contraparte de viewDidLoad()

- Ocurre cuando la vista ya no está presente en pantalla
- Sobre escrito para casos en los que se quieran finalizar tareas que no deberían ejecutarse cuando no se muestra la vista en pantalla, como una llamada a red o dejar de observar propiedades

Es llamado **más de una vez** durante el ciclo de vida del controlador

UIViewController - Ciclo de vida



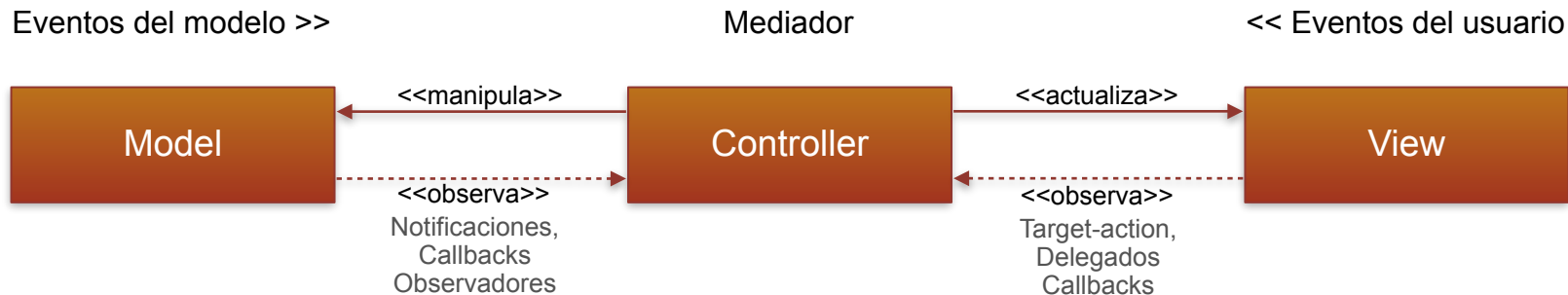
Contraparte de init()

- Llamado cuando el ViewController deja de estar en memoria
- Útil si se requiere algún tipo extra de cleanup antes de que se elimine el ViewController

Llamado **solo una vez** durante el ciclo de vida del controlador



MVC (Apple)

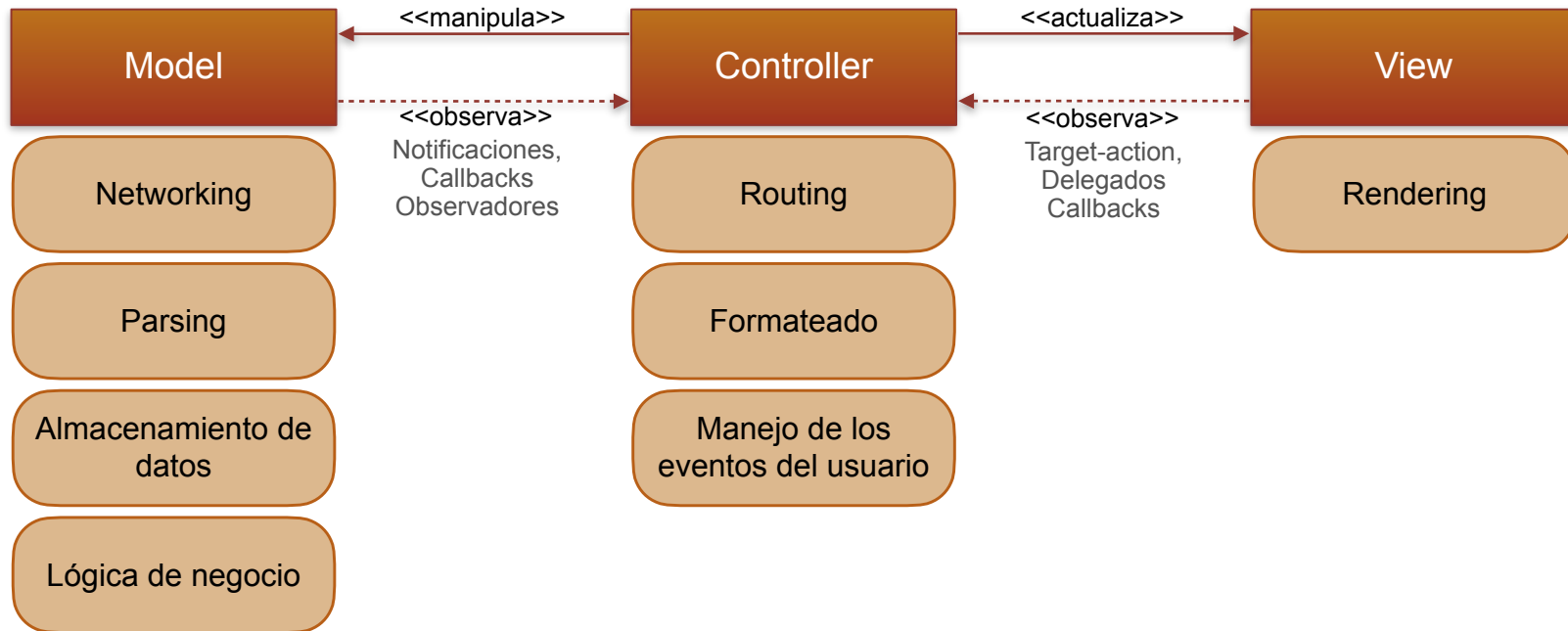


MVC (Apple)

Eventos del modelo >>

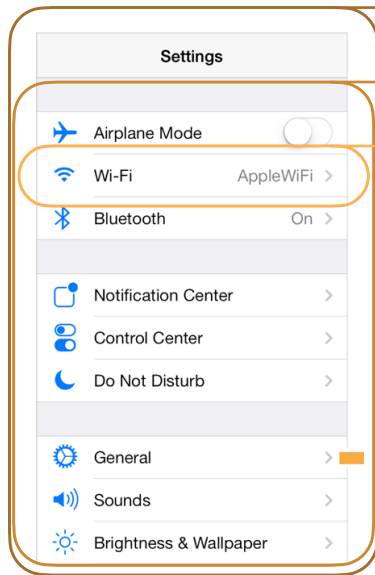
Mediador

<< Eventos del usuario



UITableViewController

- Un UIViewController especializado en el manejo de tablas
- Necesita de un datasource y de un delegate para funcionar como uno lo esperaríamos



UITableViewController

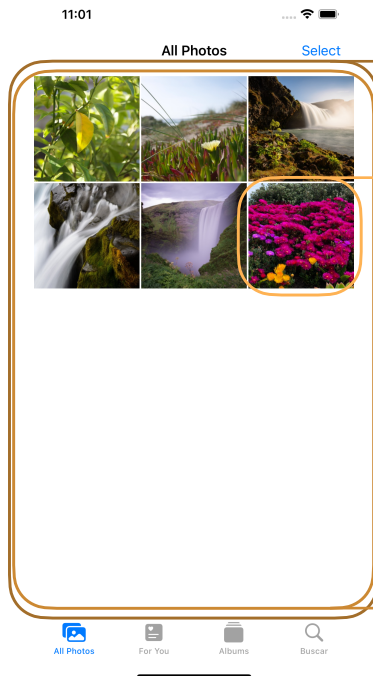
UITableView

UITableViewCell



UICollectionViewController

- Un UIViewController especializado en el manejo de colecciones
- Necesita de un datasource y de un delegate para funcionar como uno lo esperaría



UICollectionViewController

UICollectionViewCell

UICollectionView



Delegados

- Un delegado es un objeto que actúa en nombre de o en coordinación con otro, cuando ese otro se encuentra con un evento en el programa.
- Un delegado es cualquier objeto que recibe una llamada cuando sucede algo interesante. El “algo interesante” depende del contexto, por ejemplo, en un tableView puede ser cuando se hace click sobre una celda, en un navigationController puede ser cuando el usuario se mueve entre pantallas
- Relación 1:1 del objeto con su delegado



Observadores

- Un objeto que está al pendiente de eventos de otro objeto
- Útil cuando más de un objeto necesita reaccionar a un mismo evento
- Relación 1:Varios del objeto con sus observadores
- Caso más común en UIKit -> NotificationCenter

Observadores - Notification Center

- Tablón de anuncios
- Un objeto publica una notificación
- Hay otros objetos “atentos” a dicha notificación
- Cuando este objeto observador ve lo que le interesa, reacciona





KEEPCODING

Tech School

Madrid | Barcelona | Bogotá

Datos de contacto