



iOS Workshop

FHNW – Frühjahr 2017

iOS Entwicklungsressourcen

Referenz	Url	Thema
iOS API Reference	https://developer.apple.com/reference/	Offizielle Referenz von Apple zu allen Frameworks
Stanford iOS Kurs	https://itunes.apple.com/us/course/developing-ios-9-apps-swift/id1104579961	Jährlicher iOS-Kurs der Stanford-University in iTunesU
WWDC Videos	https://developer.apple.com/videos/	Aufzeichnungen der WWDC Sessions
Swift Playgrounds	http://www.apple.com/swift/playgrounds/	iPad App zum Lernen von Swift
StackOverflow	https://stackoverflow.com/questions/tagged/swift+ios	Sehr grosse Community im iOS-Bereich
App-Icons generieren	http://makeappicon.com	Alle Auflösungen automatisiert generieren lassen

Persistenz

```
class Dog: Object {  
    dynamic var name = ""  
    dynamic var age = 0  
}
```

```
let dog = Dog()  
dog.name = "Rex"  
dog.age = 1
```

```
let realm = try! Realm()  
try! realm.write {  
    realm.add(dog)  
}
```

```
let pups = realm.objects(Dog.self).filter("age < 2")
```



realm

→ [Link](#)

Dependencies



```
platform :ios, '10.0' use_frameworks!  
target 'MyApp' do pod 'AFNetworking',  
  '~> 2.6'  
  pod 'ORStackView', '~> 3.0'  
  pod 'SwiftyJSON', '~> 2.3'  
end
```

 [Link](#)

Networking

ALAMOFIRE

Elegant Networking in Swift

```
Alamofire.request("https://httpbin.org/get").responseData { response in
    debugPrint("All Response Info: \(response)")

    if let data = response.result.value, let utf8Text = String(data: data, encoding: .utf8) {
        print("Data: \(utf8Text)")
    }
}
```

→ [Link](#)

ObjectMapper

```
struct Temperature: Mappable {  
    var celsius: Double?  
    var fahrenheit: Double?  
  
    init?(map: Map) {  
  
    }  
  
    mutating func mapping(map: Map) {  
        celsius    <- map["celsius"]  
        fahrenheit  <- map["fahrenheit"]  
    }  
}
```

ALAMOFIRE

Elegant Networking in Swift

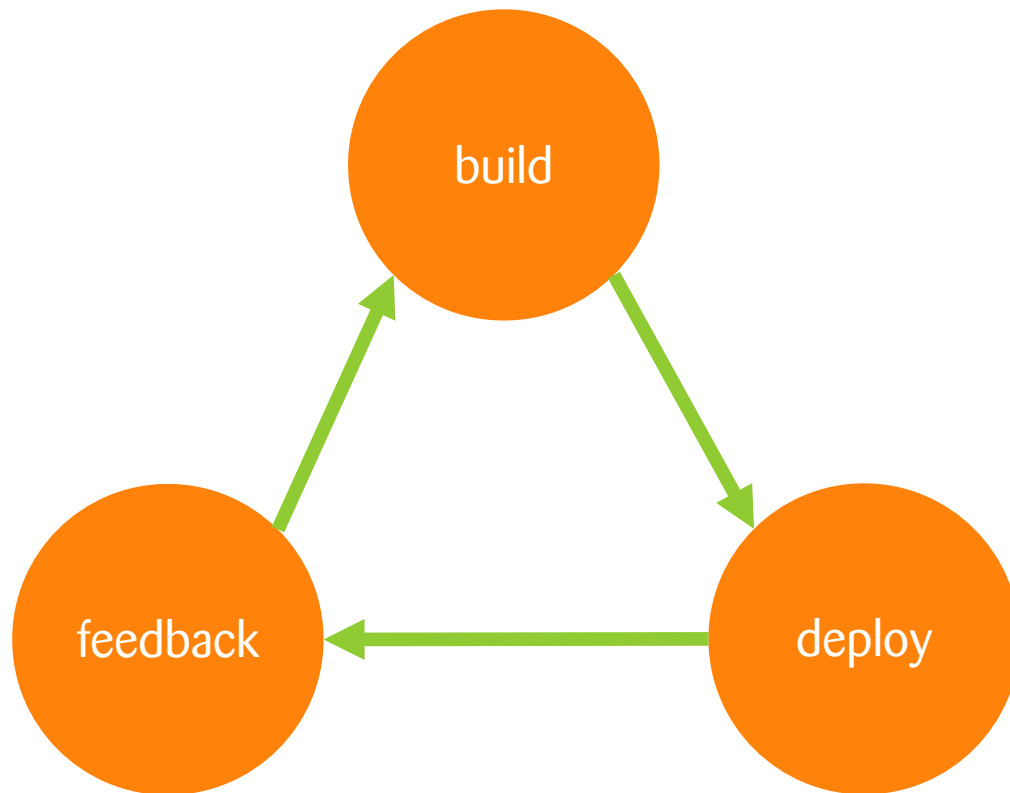


AlamofireObjectMapper

→ [Link](#)

BuddyBuild

buddybuild



→ [Link](#)