

Testing con Xcode 7

NSConfArg 2016

Pablo Bendersky
@pbendersky
quadiontech.com

Hoja de ruta



- Por qué no?
- Cobertura
- UI Testing
- Mejorando el código

¿Por qué no?

“Vamos a perder tiempo.”

“Estoy yo solo. No me hace falta.”

“Igual lo agarra QA.”

Hoja de ruta



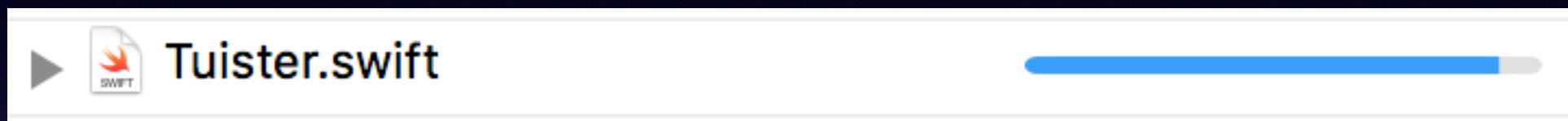
- Por qué no?✓
- Cobertura
- UI Testing
- Mejorando el código

Demo

Cobertura - Resumen

Info	Arguments	Diagnostics
Build Configuration	<div>Debug</div>	
Code Coverage	<input checked="" type="checkbox"/> Gather coverage data	
Debugger	<input checked="" type="checkbox"/> Debug executable	
Debug Process As	<div><input checked="" type="radio"/> Me (Pablo)</div> <div><input type="radio"/> root</div>	

Cobertura - Resumen



```
func search(account: ACAccount, term: String, completion: ([Tweet]) -> Void) {  
    let url = NSURL(string: "https://api.twitter.com/1.1/search/tweets.json")  
    let request = SLRequest(forServiceType: SLServiceTypeTwitter, requestMethod: .GET, URL: url,  
        parameters: ["q": term])  
    request.account = account  
    request.performRequestWithHandler { (data: NSData!, urlResponse: NSHTTPURLResponse!, error:  
        NSError!) -> Void in  
        if error == nil {  
            let json = JSON(data: data)  
            var tweets = [ Tweet ]()  
            for status in json["statuses"] {  
                let tweet = Tweet(json: status.1)  
                tweets.append(tweet)  
            }  
            completion(tweets)  
        } else {  
            print("error: \(error)")  
        }  
    }  
}
```

Cobertura - Resumen

 Tweet.swift

```
public class Tweet: CustomStringConvertible {
```

```
    public var text: String!  
    public var user: User!
```

```
    init(json: JSON) {  
        if let text = json["text"].string {  
            self.text = text  
        }  
        self.user = User(json: json["user"])  
    }
```

```
// MARK: - CustomStringConvertible Methods
```

```
    public var description: String {  
        return "\(self.dynamicType) text: \(self.text)"  
    }
```




```
}
```

30

0

0

Cobertura - Resumen

▶		AppDelegate.swift	<div><div></div></div>
▶		DetailViewController.swift	<div><div></div></div>
▶		MasterViewController.swift	<div><div></div></div>

Cobertura - Resumen

- Simple e integrado
- Métrica
- Buscar el target
- Resultados espurios

Hoja de ruta



- Por qué no?✓
- Cobertura✓
- UI Testing
- Mejorando el código

Demo

UI Testing - Resumen

- Fácil de adoptar
- Grabación desde Simulador
- Fomenta Accessibility



Hoja de ruta



- Por qué no?✓
- Cobertura✓
- UI Testing✓
- Mejorando el código

Mejorando el código

```
class Tuister {  
    func ensureAccessToAccounts(completion: (Bool) -> Void)  
  
    func accounts() -> [ACAccount]  
  
    func firstAccount() -> ACAccount?  
  
    func search(account: ACAccount, term: String, completion: ([Tweet]) -> Void)  
}
```


Mejorando el código

```
override func viewDidLoad(animated: Bool) {  
    let tuister = Tuister()  
    tuister.ensureAccessToAccounts { [weak self] (granted: Bool) -> Void in  
        tuister.search(tuister.firstAccount()!, term: "nsconfarg",  
            completion: { (tweets: [Tweet]) -> Void in  
                self?.tweets = tweets  
                dispatch_async(dispatch_get_main_queue(), { () -> Void in  
                    self?.tableView.reloadData()  
                })  
            })  
        })  
    }  
    super.viewDidLoad(animated)  
}
```


Mejorando el código

```
protocol Tuister {  
    func ensureAccessToAccounts(completion: (Bool) -> Void)  
  
    func accounts() -> [ACAccount]  
  
    func firstAccount() -> ACAccount?  
  
    func search(account: ACAccount, term: String, completion: ([Tweet]) -> Void)  
}
```


Demo

Conclusiones

- Métricas
- UI Testing
- Mejoras!

¿Preguntas?

Material:

<http://pablin.org/nsconfarg16>

¡Gracias!

Material:

<http://pablin.org/nsconfarg16>