

Eureka!

Speed up iOS forms development

Martin Barreto

@mtnBarreto



Dropbox - Kindle - Foursquare - Chase Mobile
AOL Radio - Smile - MLB.com - American Airlines

15K+ apps using it!

Topics

- **Key concepts**
- **Row types**
- **Hide/Show row/sections**
- **Validations**
- **OSS learned lessons and tips**



Key Concepts

Let's start from the beginning.

Key Concepts – DSL

- **Form:** Represents the form which has a collection of sections.
- **Section:** Represents a section of the UITableView. Contains a collection of rows.
- **Row:** Contains information about the form fields such as title, callbacks, associated table view cell and row value.
- **FormViewController:** Eureka form controller, subclass of UIViewController.

Key Concepts – DSL

```
class ViewController : FormViewController {  
  
    override func viewDidLoad() {  
        super.viewDidLoad()  
  
        form +++  
        Section(header: "Mobile Day Example", footer: "Just a very simple example")  
            <<< NameRow() {  
                $0.title = "Name"  
            }  
            <<< ButtonRow() {  
                $0.title = "Submit"  
            }.onCellSelection { cell, row in  
                print("submitting form")  
            }  
    }  
}
```

MOBILE DAY EXAMPLE

Name

Submit

Just a very simple example

Key Concepts – Operators

+++ Adds a section

<<< Inserts a row

```
form
  +++ Section("First Section")
    <<< TextRow() { $0.placeholder = "Your Name.." }
    <<< DateRow()
  +++ Section("Second Section")
    <<< TextAreaRow()
    <<< DateRow(){ $0.value = Date() }
    <<< ButtonRow() { $0.title = "Submit" }
```

Key Concepts – Tags

- **Tags**
 - **Identify rows and sections**
 - **They are optional**
 - **Mainly used to:**
 - **Obtain specific row or section**
 - **Get values out of the form**

Key Concepts - Tags

```
form +++
  Section(header: "Mobile Day Example", footer: "Just a very simple example")
    <<< NameRow("name_row") { ←
      $0.title = "Name"
    }
    <<< ButtonRow() {
      $0.title = "Submit"
    }.onCellSelection { [weak self] cell, row in
      print("submitting form")
      self?.submitAction()
    }
  }
```

```
let formValues = form.values()
print(formValues)
// ["name_row": Optional("Martin")]
```

```
let row: NameRow = form.rowBy(tag: "name_row")!
print("Name: \(row.value)")
// Name: Optional("Martin")
```

Key Concepts – Callbacks

Allow us to change appearance and behavior of a row.

- **onChange**
- **cellSetup**
- **cellUpdate**
- **onCellSelection**
- **onCellHighlightChanged**
- **onRowValidationChanged**
- **onExpandInlineRow, onCollapseInlineRow**
- **onPresent**

Key Concepts - Callbacks

```
let row = SwitchRow() { row in
    row.title = "The title"
    → }.onChange { row in
        row.title = (row.value ?? false) ? "The title expands when on" : "The title"
        row.updateCell()
    → }.cellSetup { cell, row in
        cell.backgroundColor = .lightGray
    → }.cellUpdate { cell, row in
        cell.textLabel?.font = .italicSystemFont(ofSize: 18.0)
    }
```

The title







The background of the slide is a red gradient with faint, semi-transparent code snippets in a monospace font. The code includes JavaScript-like syntax such as 'for (var b =', 'modified text', 'UNIQUE:', 'func', 'return', 'c = 0;', 'c =', 'c.length;', 'c.unique =', 'use_arr', and '#User_log'.

Row Types

There are many available rows and you can also create custom ones

Row Types

Label Row LabelRow	Button Row Button Row	Check Row CheckRow 
Switch Row SwitchRow 	Slider Row SliderRow 5.0 	Stepper Row StepperRow 1.0 
Text Area Row TextAreaRow		

Row Types

TextRow	Abc def	TextRow
NameRow	Abc Def	NameRow
URLRow	abc.com	URLRow
IntRow	123	IntRow
PhoneRow	+123	PhoneRow
PasswordRow	●●●●●●	PasswordRow
EmailRow	john@	EmailRow
DecimalRow	12,344.00	DecimalRow
TwitterRow	😊😊😊	TwitterRow
AccountRow	abc	AccountRow
ZipCodeRow	1123	ZipCodeRow

Row Types

Date Row

DateRow

May 26, 2002

<	>	Done
February	25	2000
March	26	2000
April	26	2001
May	27	2002
June	28	2003
July	29	2004
August	30	2005

Picker shown in the keyboard.

Date Row (Inline)

DateInlineRow

February	27	2003
March	28	2014
April	29	2015
May	30	2016
June	31	2017
July	1	2018
August	2	2019

The row expands.

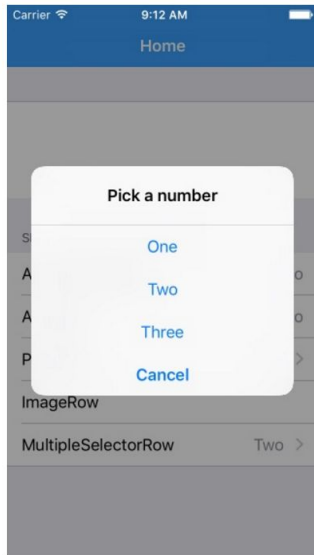
Date Row (Picker)

June	30	2004
July	31	2005
August	1	2006
September	2	2007
October	3	2008
November	4	2009
December	5	2010

The picker is always visible.

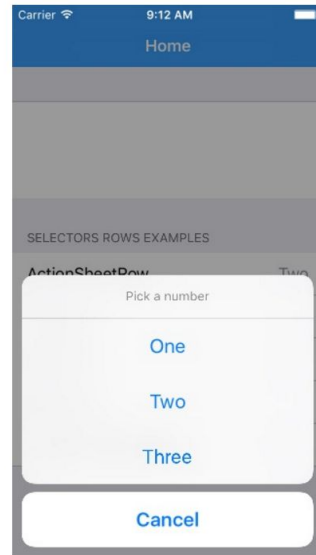
Row Types

Alert Row



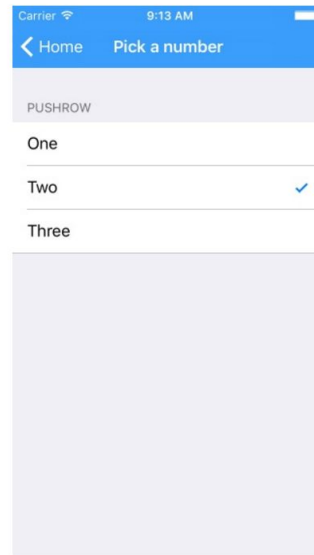
Will show an alert with the options to choose from.

ActionSheet Row



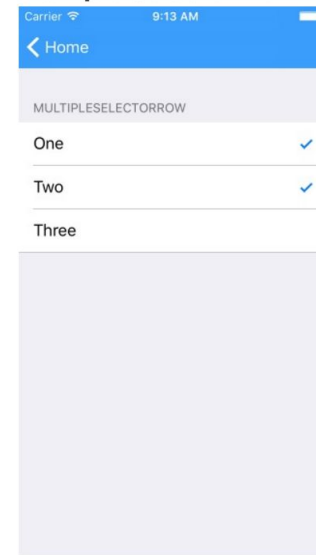
Will show an action sheet with the options to choose from.

Push Row



Will push to a new controller from where to choose options listed using Check rows.

Multiple Selector Row



Like PushRow but allows the selection of multiple options.

Row Types

Segmented Row



Segmented Row (w/Title)

Pick a number



Picker Row

Option 1
Option 2
Option 3

Presents options of a generic type through a picker view

(There is also Picker Inline Row)



Hide/Show rows and sections

Nothing is static nowadays

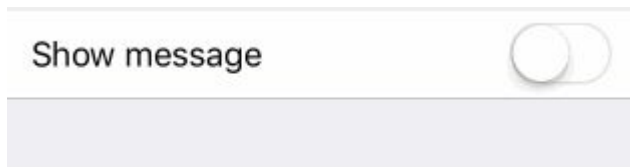
Hide/Show rows and sections

- **Eureka automatically updates the table view whenever a row or section is inserted/removed/replaced**
- **Hidden condition: Hide/Show a specific row or section depending on the value of another row**

Hide/Show rows and sections

```
form +++ Section()  
  <<< SwitchRow("switch_row_tag"){  
    $0.title = "Show message"  
  }  
  <<< LabelRow(){  
    $0.title = "Switch is on!"  
    → $0.hidden = Condition.function(["switch_row_tag"]) { form in  
      return !((form.rowBy(tag: "switch_row_tag") as? SwitchRow)?.value ?? false)  
    }  
  }  
}
```

```
form +++ Section()  
  <<< SwitchRow("switch_row_tag"){  
    $0.title = "Show message"  
  }  
  <<< LabelRow(){  
    $0.title = "Switch is on!"  
    → $0.hidden = Condition.predicate(NSPredicate(format: "$switch_row_tag == false"))  
  }  
}
```





Validations

Form validation has never been so easy!

Validations

- **Each row has a collection of validation rules.**
- **Rules provided by default:**
 - **Required**
 - **Email**
 - **URL**
 - **GreaterThan, SmallerThan**
 - **MinLength, MaxLength**
 - **Closure**
- **You can create your own rules if needed.**

Validations

- **row.validationOptions** property lets us specify when validation rules should be evaluated.
 - **validatesOnDemand**
 - **validatesOnChange**
 - **validatesOnBlur**
 - **validatesOnChangeAfterBlurred**
- **row.isValid** boolean property indicates if the row is valid or not.
- **row.validate()**, **form.validate()** allows us to manually perform validations. Both methods return a list of errors.

Validations

- **row.validationErrors** returns validation errors list.
- **row.onRowValidationChanged** can be used to get notified when validations change.

```
<<< TextRow() {  
    $0.title = "Required Rule"  
    $0.add(rule: RuleRequired())  
    $0.validationOptions = .validatesOnChange  
}
```


Validations

- **row.validationErrors** returns validation errors list.
- **row.onRowValidationChanged** can be used to get notified when validations change.

```
<<< TextRow() {
    $0.title = "Email Rule"
    $0.add(rule: RuleRequired())
    var ruleSet = RuleSet<String>()
    ruleSet.add(rule: RuleRequired())
    ruleSet.add(rule: RuleEmail())
    $0.add(ruleSet: ruleSet)
    $0.validationOptions = .validatesOnChangeAfterBlurred
}
```

Validations

- **row.validationErrors** returns validation errors list.
- **row.onRowValidationChanged** can be used to get notified when validations change.

```
<<< URLRow() {
    $0.title = "URL Rule"
    $0.add(rule: RuleURL())
    $0.validationOptions = .validatesOnChange
}
.cellUpdate { cell, row in
    if !row.isValid {
        cell.titleLabel?.textColor = .red
    }
}
```

Validations

```
+++ Section(header: "More sophisticated validations UX using callbacks", footer: "")

  <<< TextRow() {
    $0.title = "Required Rule"
    $0.add(rule: RuleRequired())
    $0.validationOptions = .validatesOnChange
  }
  .cellUpdate { cell, row in
    if !row.isValid {
      cell.titleLabel?.textColor = .red
    }
  }
  .onRowValidationChanged { cell, row in
    let rowIndex = row.indexPath!.row
    while row.section!.count > rowIndex + 1 && row.section?[rowIndex + 1] is LabelRow {
      row.section?.remove(at: rowIndex + 1)
    }
    if !row.isValid {
      for (index, validationMsg) in row.validationErrors.map({ $0.msg }).enumerated() {
        let labelRow = LabelRow() {
          $0.title = validationMsg
          $0.cell.height = { 30 }
        }
        row.section?.insert(labelRow, at: row.indexPath!.row + index + 1)
      }
    }
  }
}
```

Validations

MORE SOPHISTICATED VALIDATIONS UX USING
CALLBACKS

Required Rule



Email Rule

URL Rule

Password

Confirm Password

Range Rule

Tap to force form validation



OSS learned lessons and tips

There is no magic! Just hard work!

**Giving back to the
community is what
really matters!!**

Passion

- Use OSS everyday
- Awesome solution
- Perfectionist

Open source is all about discussion!

- Open Mindset
- Incrementally
- Community Driven

- **Creating an Issue**
 - **Reporting an Issue.**
 - **Asking for a feature.**
 - **Discussing a feature or whatever.**
 - **Answering community questions.**

Contributing to a project - 2

- **Making a pull request**
 - **Fixing an issue.**
 - **Implementing a new feature.**
 - **Documenting. Not everything is code.**

Contributing - My first experience

Ability to know the download progress of an image. #1274

Closed mtnbarreto wants to merge 3 commits into AFNetworking:master from smartlabs:download_progress_uimageView

Fork 8,733

Conversation 1 Commits 3 Files changed 2



mtnbarreto commented on Sep 2, 2013

This change allow us to easily show a progress indicator on a UIImageView.

mtnbarreto added some commits on Aug 27, 2013

- Ability to know the download progress of an image. 44d65f6
- Merge pull request #1 from mtnbarreto/downloadProgressUIImageCategory ... da42eda
- Merge branch 'master' of github.com:AFNetworking/AFNetworking into do... ... e3e05b0



matt commented on Sep 2, 2013

AFNetworking contributor

Thanks for your contribution, @mtnbarreto. However, this feature has been previously proposed or discussed in #1106, #765, #506, #748, #704, #681, and #506.

matt closed this on Sep 2, 2013

New pull request

1

Tips - OSS library from Scratch - 1

- **Great Documentation!**
 - **Very basic library usage.**
 - **Gif / Benchmarks / Code snippet to show elegance / Compare with the old way.**
 - **Installation: Support as many dependency managers as possible.**
 - **PR template, New issue template.**

Tips - OSS library from Scratch - 2

- **Tests & CI.**
- **Tag issues.**
- **Whenever possible refer to another issue to answer/solve an issue.**
- **Make public project goals and direction.**

Tips - OSS library from Scratch - 2

- **Release Notes.**
- **Migration Guide.**
- **Create a community around the library.**

Make it popular

Make it popular - 1

- **Blog post introducing the library.**
- **Community influencers.**
- **Publish library on Awesome Swift, Awesome iOS.**
- **Submit it to CocoaControls.**
- **PR to iOS Goodies.**
- **Post on Hacker News.**

Make it popular - 2

- **Readme makes clear that your library is different than the other ones?**
- **Make sure to have a cool logo**
- **Promote it from your twitter account, mention influencers.**

My first experience and examples

 **Martin Barreto | Xmartlabs** <martin@xmartlabs.com> 10/04/2014 ☆  

to m ▾

Hey Matt,

I have written a library to easily create table-view forms on iOS.
It would be amazing if you have a chance to take a look at the library and let me know your thought!

Here is the library repository:
<https://github.com/xmartlabs/XLForm>

Thanks in advance!
Regards,
Martin

...

 **Matt Thompson** <m@mattt.me> 11/04/2014 ☆  

to me ▾

Hi Martin,

This is interesting, but I think I prefer the abstractions used by FXForms (<https://github.com/nicklockwood/FXForms>).
Have you seen that project?

...

 **Martin Barreto** <martin@xmartlabs.com> 11/04/2014 ☆  

to Matt ▾

Matt,
Thanks for the reply! I really appreciate your feedback!
I have seen FXForms and it's really a great library. It seems to me that the simplicity is its main strength. I tried to put more focus on flexibility and also addressed the requirements of projects i have been doing.

I will be improving it based on many feedbacks.



Show HN Daily @ShowHNDaily · 9 Apr 2014

Show HN: **XLForm** â Powerful and flexible forms on iOS.
[github.com/xmartlabs/XLForm...](https://github.com/xmartlabs/XLForm)



CocoaPods New Pods @CocoaPodsFeed · 29 Sep 2015

[Eureka by @xmartlabs] Elegant iOS Forms in pure Swift 2



Eureka

Elegant iOS Forms in pure Swift 2

cocoapods.org



1




5





Cocoa Controls @cocoacontrols · 16 Apr 2014

XLForm for iOS: **XLForm** is the most flexible and powerful iOS library to create dynamic table-view... -

●●●●○ Antel 

5:16 PM

29% 

Cancel

Add Event

Done

Xmartlabs Event

Location

XLForm

XLForm is the most flexible and powerful iOS library to create dynamic table-view forms. The goal of the library is to get the same power of han...

cocoacontrols.com





XMARTLABS @xmartlabs · 29 Sep 2015

Introducing **Eureka!** - Elegant iOS forms in Pure Swift 2 |

blog.xmartlabs.com/2015/09/29/Int...



6



10



13



INTRODUCING EUREKA! - ELEGANT IOS FORMS IN PURE SWIFT 2



by **Martin Barreto**

Sep 29, 2015

Share on:



We are incredibly excited to announce **Eureka!**, our first Swift open source project designed to easily build dynamic table-view forms.

Why we built Eureka!

**What we achieved by
following that**



XMARTLABS @xmartlabs · 16 Apr 2014

#XLForm and @xmartlabs #1 in @github today!! #trending #iOS #daydream
github.com/xmartlabs/XLForm... cc @objcio @objectivec

GitHub Search or type a command Explore Feat

Trending Objective-C repositories on GitHub today

Repositories Developers

- xmartlabs / XLForm** Objective-C
XLForm is the most flexible and powerful iOS library to create dynamic table-view forms.
★ 67 📄 6 built by
- FuzzyAutocomplete / FuzzyAutocompletePlugin** Objective-C
A Xcode 5 plugin that adds more flexible autocompletion rather than just prefix-matching.
★ 53 📄 4 built by
- coolbeet / CBHazeTransitionViewController** Objective-C
A small clone of Haze's beautiful transition view.
★ 48 📄 6 built by
- shu223 / AnimatedTransitionGallery** Objective-C
A gallery app of custom animated transitions for iOS 7.
★ 39 built by
- TimOliver / TOWebViewController** Objective-C
A modal view controller for iOS that allows users to quickly view web sites from within a

GitHub Search or type a command Explore Feat

Trending Objective-C developers on GitHub today

Repositories Developers

- xmartlabs (xmartlabs)**
 XLRemoteImageView
UIImageView that shows a progress indicator while the image is loading from server. It r
- facebook (Facebook)**
 three20
Three20 is an Objective-C library for iPhone developers
- FuzzyAutocomplete**
 FuzzyAutocompletePlugin
A Xcode 5 plugin that adds more flexible autocompletion rather than just prefix-matching
- github (GitHub)**
 Rebel
Cocoa framework for improving AppKit
- apache (The Apache Software Foundation)**
 cordova-ios
Mirror of Apache Cordova iOS





Pablo Martí @startpablo · 20 May 2014

AFNetworking + JSONModel + FXForms = Cool! #iosdev (but... another cool option, **XLForm**: [github.com/xmartlabs/XLForm...](https://github.com/xmartlabs/XLForm))



1



Carl Hunter Roach @CarlHunterRoach · 18 Aug 2014

@mtnBarreto thanks for **XLForm** - looks fabulous. do you know if anyone is tweaking it for Xcode 6?



1



In reply to Alexander Akers 🐼



Brian Gesiak @modocache · 10 May 2015

@a2 Oh, so Carthage builds the tests, too? And you wanted to use **XLForm** but not have your project depend on Expecta? 🤔 140 chars is hard



1





Diego Mazzone @diego_mazzone · 13 Aug 2015

Installing **XLForm** 3.0.1 (was 3.0.0) 😎 @mtnBarreto @xmartlabs



Andrew Yates @ay8s · 3 Apr 2015

Anyone else made use of **XLForm** to rapidly develop/prototype iOS apps?
Really impressed with how quick it is.



Adrian Kosmaczewski @akosma · Apr 12

wow 😲 👏

xmartlabs/Eureka: Elegant iOS form builder in Swift 2
[github.com/xmartlabs/Eure...](https://github.com/xmartlabs/Eureka)





Tim Oliver @TimOliverAU · Feb 7

Oh my god. How did I not know about **XLForm** sooner?? This thing trivialises making settings menus in iOS!

Switch

Check

Step counter 50

Fruits

Slider

Image

← 1 ↻ 1 ❤ 5 ⋮



MEHUL SRIVASTAVA @SrivastavaMehul · Feb 24

@xmartlabs Hey thanks for **Eureka!** Have easily cut days out of my production time!!



Steve Tibbett @stevex · Sep 22

I think I would use **XLForm** over Eureka just to protest the gratuitous operator overloading.



xmartlabs/Eureka

Elegant iOS form builder in Swift. Contribute to Eureka development by creating an account on GitHub.

github.com



CODE

Eureka

This looks to be an extremely comprehensive library if you want to implement a

- [Redbird](#) - Pure-Swift implementation of a Redis client from the original protocol spec. OS X + Linux compatible.
- [blackfish](#) - A fast HTTP web server based on Node.js and Express written in Swift
- [RAMReel](#) - RAMReel is a controller that allows you to choose options from a list
- [Scale](#) - Unit converter in Swift. [Blog post here](#).
- [XLPagerTabStrip](#) - Android PagerTabStrip for iOS.
- [ValentinesSwift](#) - You love Swift & Swift loves you! [More fun here!](#) And a [fun write-up here!](#) 💖

Top 10 iOS Swift libraries every iOS developer should know about

Published in [Tools](#), [Swift](#), [iOS development](#), [Learning](#) • December 10th, 2015

• [9 Comments](#)

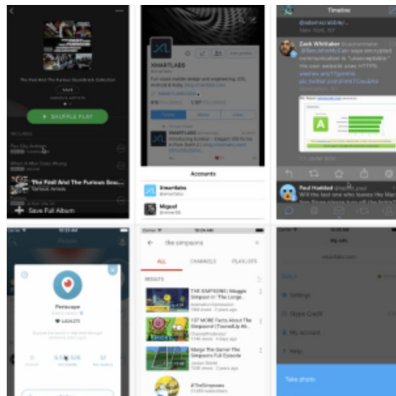
39 Open Source Swift UI Libraries For iOS App Development

This is “amazing” series of open source projects.

Developed by Apple Inc, **Swift** is currently the most popular programming language on [Github](#) and it has one of the most active communities that kindly contribute their open source projects.

No 13

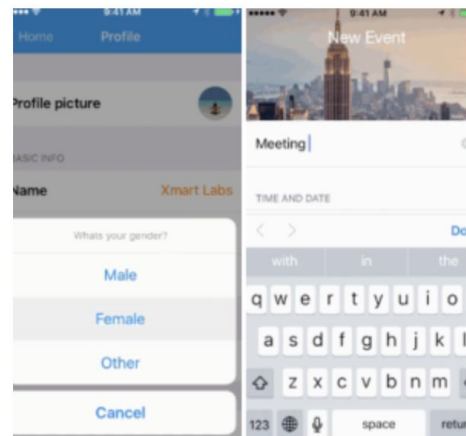
XLActionController: Fully customizable and extensible action sheet controller written in Swift 3 [1346 stars on Github].



<Form UI>

No 36

Eureka: Elegant iOS form builder in Swift [4117 stars on Github].



Swift

15 repositories <> 2 languages Last updated 14 days ago



apple / swift

The Swift Programming Language

C++ ★ 35,094 🍴 5,037 Updated 19 minutes ago Starred by 8 people you know



Alamofire / Alamofire

Elegant HTTP Networking in Swift

Swift ★ 20,335 🍴 3,493 Updated 3 days ago Starred by 5 people you know



Carthage / Carthage

A simple, decentralized dependency manager for Cocoa

Swift ★ 8,191 🍴 677 Updated an hour ago Starred by 2 people you know



realm / SwiftLint

A tool to enforce Swift style and conventions.

Swift ★ 5,023 🍴 363 Updated 8 hours ago Starred by 3 people you know



xmartlabs / Eureka

Elegant iOS form builder in Swift

Swift ★ 4,280 🍴 395 Updated 6 days ago Starred by 10 people you know



Related showcases

Package managers

Across programming languages and platforms, these popular package manage...

Programming languages

A list of programming languages that are actively developed on GitHub.

[View all](#)

New showcases

Serverless Architecture

Build powerful, event-driven, serverless architectures with these open-s...

Fabric Mobile Developer Tools

Tools to help mobile app development teams build, grow, and monetize the...



Infrastructure

User Engagement

Customer Support

Authentication

Deep Linking

A/B Testing

Testing

Push

SDK Wrapper

App Platform

Ad Attribution

Messaging

IoT/Beacons

documentation

🕒 **HPGrowingTextView** 12 Apps (6% of Top 200)
Multi-line/Autoresizing UITextView similar to SMS-app.



🕒 **SVPullToRefresh** 11 Apps (6% of Top 200)
Give pull-to-refresh to any UIScrollView with 1 line of code.



🕒 **NJKWebViewProgress** 10 Apps (5% of Top 200)
UIWebView progress interface.



🕒 **DACircularProgress** 9 Apps (5% of Top 200)
DACircularProgress is a UIView subclass with circular UIProgressView properties.



📊 **Eureka** 9 Apps (5% of Top 200)
Elegant iOS Forms in pure Swift 2



🕒 **CHTCollectionViewWaterfallLayout** 9 Apps (5% of Top 200)
The waterfall (i.e., Pinterest-like) layout for UICollectionView.



🕒 **MK** 8 Apps (4% of Top 200)
A Material Design Framework In Swift



🕒 **FXBlurView** 8 Apps (4% of Top 200)
UIView subclass that replicates the iOS 7 realtime background blur effect, but works on iOS 5 and above.





swift ranking

Montevideo	1 / 7 🏆
Uruguay	1 / 8 🏆
Worldwide	18 / 30 093 🏆
Repos :	12 📖
Stars :	9098 ★



Montevideo

Uruguay

Tweet your ranking 🐦

Refresh your profile

objective-c ranking

Montevideo	1 / 31 🏆
Uruguay	1 / 35 🏆
Worldwide	118 / 114 812 🏆
Repos :	16 📖
Stars :	4886 ★

Help Github Awards stay alive !

Thank you!

<https://github.com/xmartlabs>



@mtnBarreto

We are hiring!
careers@xmartlabs.com