

# MVC 2.0

NSTehran 4

# مشکلات اصلی MVC

- خلاف encapsulation بودن
- پیچیدگی و طولانی شدن هر VC
- وقت‌گیر بودن انجام تغییرات ساده
- سخت‌بودن استفاده دوباره از المنت‌ها
- لایه‌هایی زیادی که همزمان استفاده نمی‌شوند

# مشکل encapsulation

“هرچیزی کاملاً و واقعی از اطلاعات خودش اطلاع داشته باشد، و این اطلاعات نهایتاً تا یک مرحله بعد باشد”

```
func tableView(_ tableView: UITableView, cellForRowAt indexPath: IndexPath) -> UITableViewCell {  
    let cell = tableView.dequeueReusableCell(withIdentifier: "cell", for: indexPath)  
    cell.textLabel?.text = dataProvider.movies[indexPath.row].1  
    return cell  
}
```

```
override func prepare(for segue: UIStoryboardSegue, sender: Any?) {  
    guard segue.identifier == "showMovie" else { return }  
    guard let movieVC = segue.destination as? MovieViewController else { return }  
    movieVC.currentMovie = dataProvider.movies[tableView.indexPathForSelectedRow!.last!]  
    movieVC.dataProvider = dataProvider  
}
```

# تا چه زمانی VC کوتاه حساب میشه؟

• صفر تا ۵۰۰ = کوتاه

• ۵۰۰ تا ۳۰۰۰ = نسبتاً طولانی

• بیشتر از ۳۰۰۰ طولانی

# MVVM, React, MVP, FRP, VIPER ...

- عدم هماهنگی با لایبرری‌های اپل
- پیدا کردن هم‌تیمی آشنا با اون ساختار
- عدم سازگار بودن هدف لایبرری با نیاز شما
- لازم بودن به تغییرات و آپدیت نگه‌داشتن (کار غیرمفید)

# قانون اصلی

یک VC همزمان نباید هم الی رو مدیریت کنه و هم تغییرات کلی  
رو براساس محتوی

# رفع encapsulation

- برای اینکه همیشه هر VC از اطلاعات خودش کاملاً اطلاع داشته باشد، به مرحله‌ای نیاز هست تا این تصمیم رو بگیره که باتوجه به محتوی کدوم یکی از VC ها باید نمایش داده بشه "FlowViewController"

# Protocols

پروتوکل‌ها بهترین روش برای ارتباط و انتقال اطلاعات بین کلاس‌های مختلف هستند، هر جا می‌توانید از شون استفاده کنید



# Massive View Controller

Network callbacks •

Delegate methods •

DataModel changes •

Data-source methods •

IBActions •

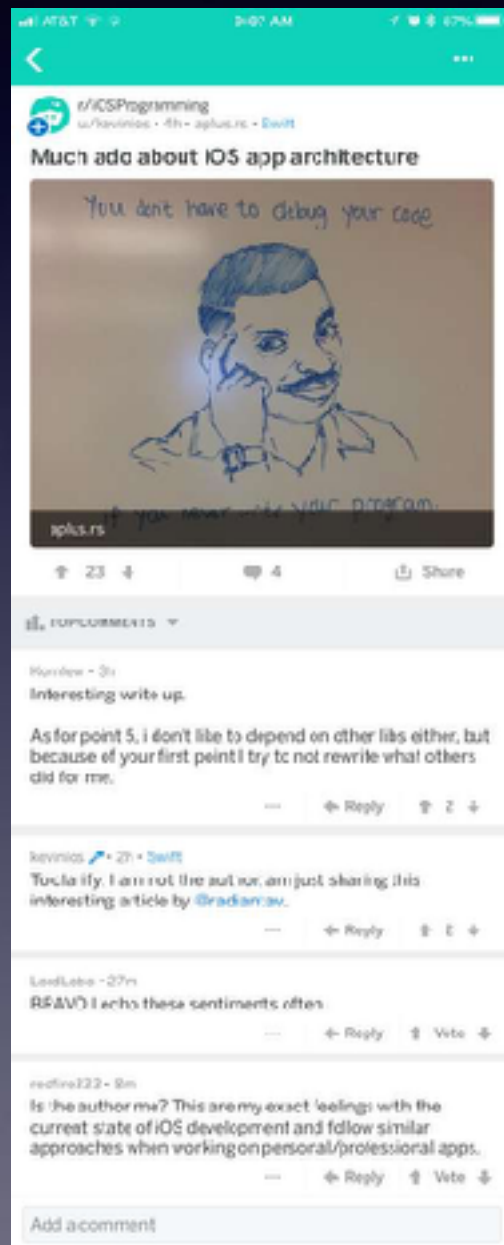
Layouts •

Logic •

# عقیده اشتباه

یک  $VC \neq$  یک صفحه کامل از اطلاعات

# عقیده اشتباه



# childVC

- مشخص بودن هدف هر VC
- امکان استفاده دوباره با کمترین دردسر
- Adaptive layout جدا برای هر VC
- انجام تست‌های هدف‌دار
- برای انجام تغییرات هر VC به جای دیگری وابسته نیست

كد

پرسش و پاسخ