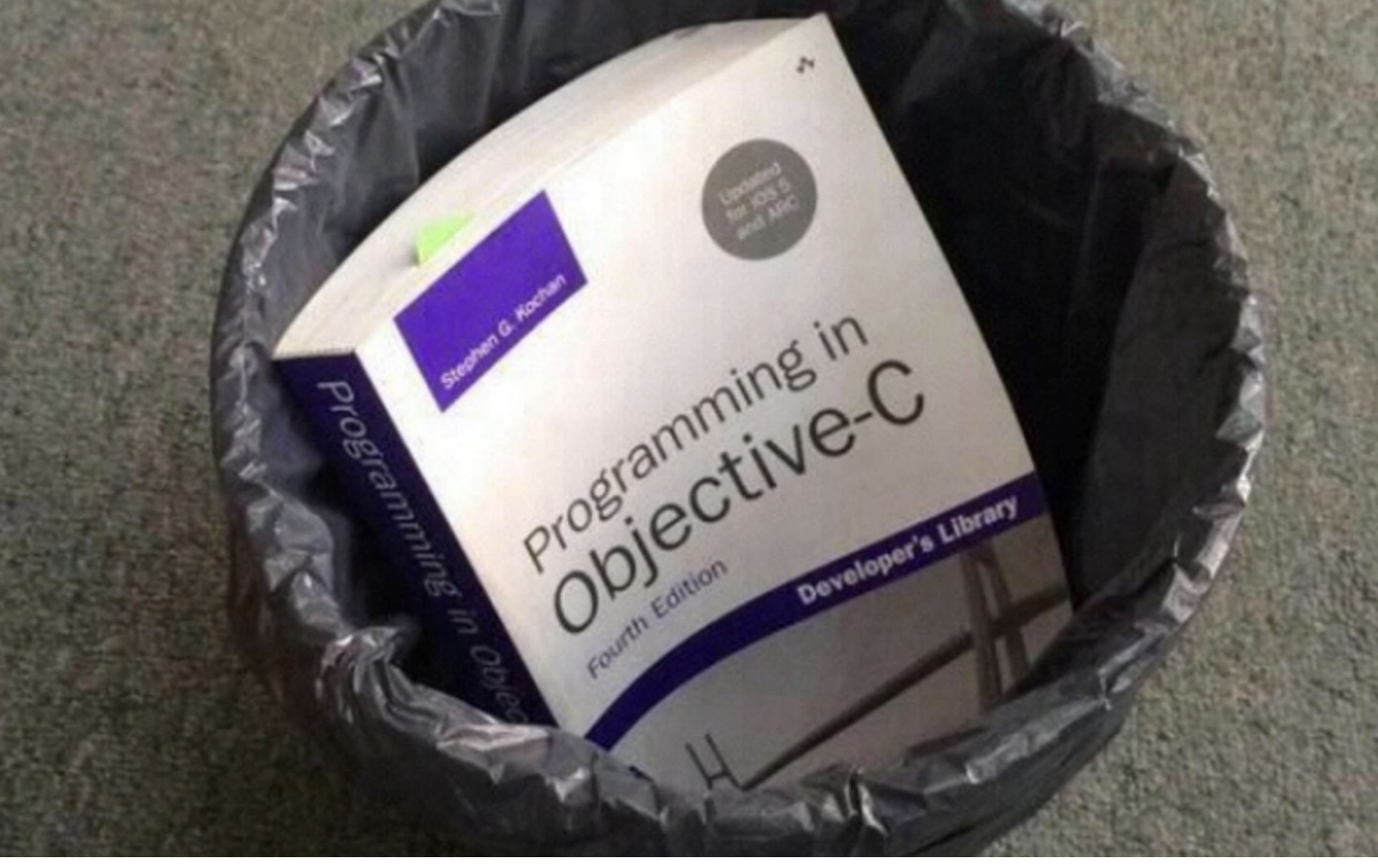
We ❤ Swift

@somkiat.cc



บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Swift look like ?

Adam Denenberg Become a fan
VP Engineering @ Huffington Post

Hello Swift, Meet Scala

Posted: 06/05/2014 1:22 pm EDT | Updated: 08/05/2014 5:59 am EDT

45 Like | 14 Share | 47 Tweet | 1 Share | 0 Comment

Yesterday, during a four and a half hour flight, I decided to read the Swift programming book (because....why not?). I was immediately blown away by Swift's similarities to Scala and think that there have been massive improvements from Objective-C.

UITextView *view = [[UITextView alloc] init] becomes something like

var view: UITextView

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Categories

Conference

Cooking

DSL

French

Gaelyk

Geek

Groovy

Groovy Weekly

Resources

Apple's Swift programming language inspired by Groovy

Posted on 02 June, 2014 (1 year ago)

During Apple's WWDC conference was announced a new programming language, called Swift, targeted at developing on iOS devices (and likely Mac OS X in the future as well).

When looking through the slides from the keynote, online documentation, and iTunes ebook, an ac
Groovy eye would immediately notice the inspiration the Swift designers took from Groovy.

In one of the slides covering Swift, Groovy was even mentioned as one of the languages clearly ma
developers much more productive, although I disagree with the placement of Groovy in terms of pe
as Groovy should clearly be leading the pack of the bottom right hand quadrant.

[redacted]

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

reddit PROGRAMMING comments related other discussions (4)

This is an archived post. You won't be able to vote or comment.

153 "Swift borrows extensively from C# and Rust" - Graydon Hoare (graydon2.dream
submitted 1 year ago by damian2000
198 comments share pocket

all 198 comments

sorted by: best

[–] glacialthinker 50 points 1 year ago

I'm glad to see ML influences spreading through programming languages and into popul

To me, Swift looks a lot like Haxe, but then Haxe is an ML-influenced (OCaml implement compatible with the common procedural/OO-languages (for re-targeting)... so, C# and F approximation too. ;)

permalink pocket

[–] OMouse 3 points 1 year ago

I'm still waiting for more Lisp features to be borrowed, I always find situations where amazing.

permalink parent pocket

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Jeremy Bae
@opt9

Follow

Hmm... Swift => Haskell, Ruby mix based on Rust?

FAVORITE
1

3:29 PM - 3 Jun 2014

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

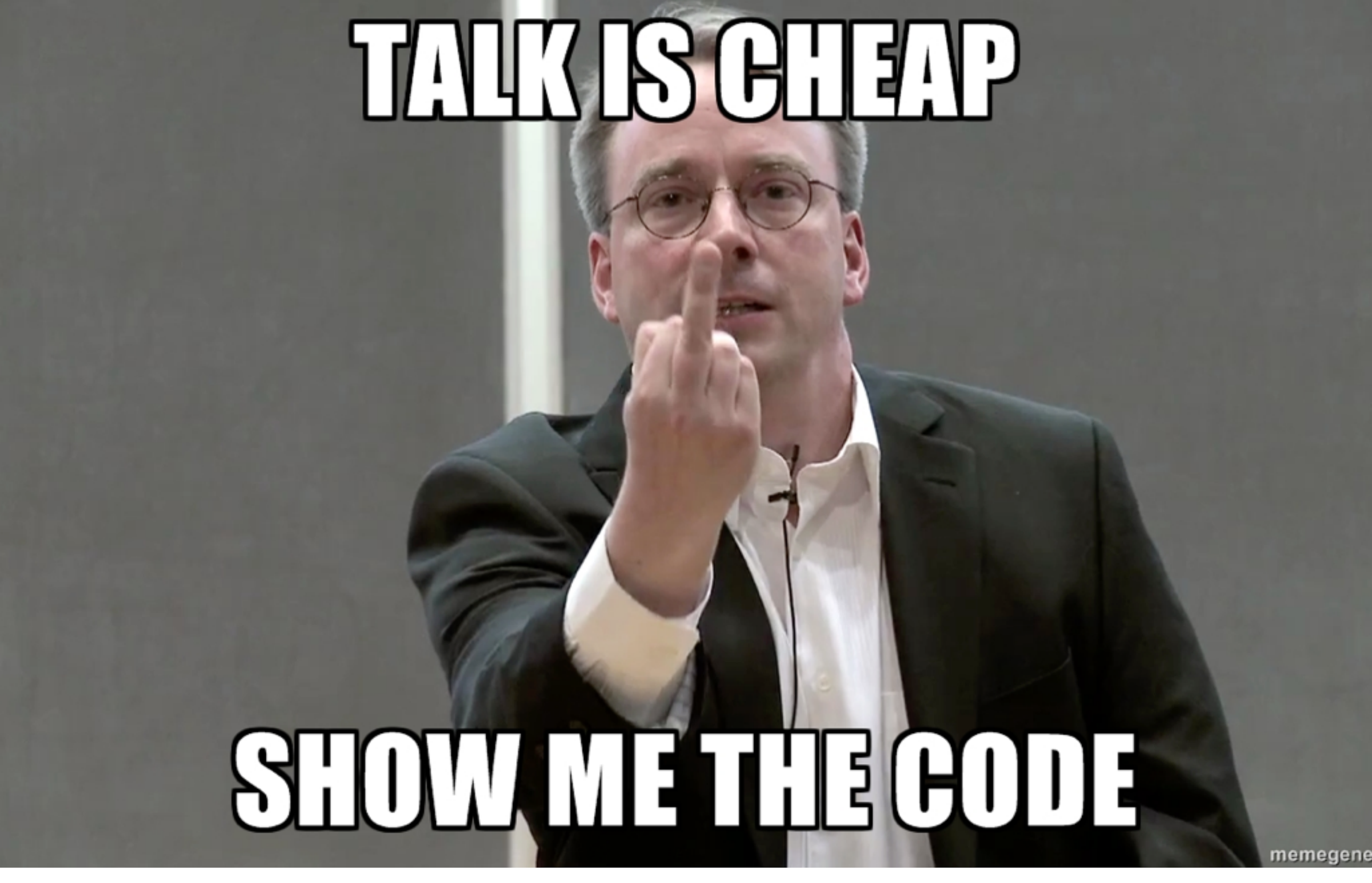
Scott O'Hara
@scottohara_

Follow

Swift looks as much like Haskell as it does JavaScript (None/Some, monad/monoid style types)

3:32 PM - 3 Jun 2014

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่


TALK IS CHEAP
SHOW ME THE CODE
memegene

```
let individualScores = [20, 50, 80, 100, 78]
var teamScore = 0
for score in individualScores {
    if score > 70 {
        teamScore += 5
    } else {
        teamScore += 1
    }
}
print(teamScore)
```

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Optional

```
var optionalName: String? = "Somkiat"

if optionalName != nil {

    print("Hello \(optionalName!)")

}
```

Optional

```
var optionalName: String? = "Somkiat"

if let name = optionalName {

    print("Hello \(name)")

}
```

Let’s start **Playground**

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Welcome to Xcode

Version 7.1 beta (7B75)

Get started with a playground
Explore new ideas quickly and easily.

Create a new Xcode project
Start building a new iPhone, iPad or Mac application.

Check out an existing project
Start working on something from an SCM repository.

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

MyPlayground.playground

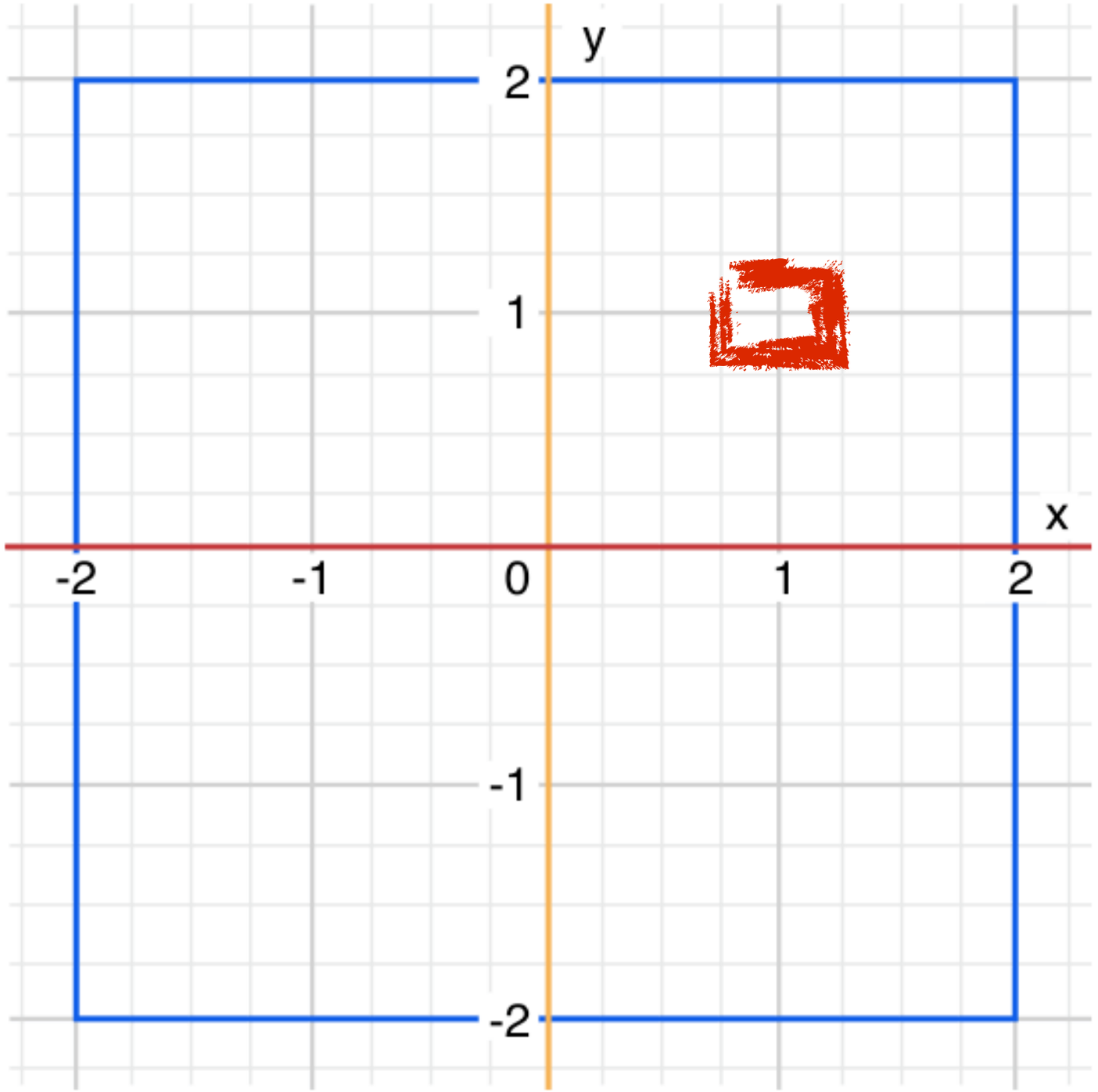
MyPlayground

```
1 let individualScores = [20,          [20, 50, 80, 100, 78]
      50, 80, 100, 78]
2 var teamScore = 0                    0
3 for score in
      individualScores {
4     if score > 70 {
5         teamScore += 5               (3 times)
6     } else {
7         teamScore += 1               (2 times)
8     }
9 }
```

```
true
true
Set new value id 10.0
Set new value success
100.0
600
1000
iPhone 6
Invalid direction
```

Switch-case !!

```
let animal = "hen"
switch animal {
case "duck" :
    let sound = "Quack Quack"
case "cow" :
    let sound = "Mooo Mooo”
case "chicken", "hen" :
    let sound = "Cock-a-doodle-doo"
default :
    let sound = "N/A"
}
```



บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Tuple

```
let somePoint = (1, 1)

switch somePoint {
case (0, 0) :
    print("0, 0”)
case (_, 0) :
    print("(\(somePoint.0), 0) in x-axis")
case (0, _) :
    print("(0, \(somePoint.1)) in y-axis")
case (-2...2, -2...2) :
    print("Inside the box")
default :
    print("Out of the box")
}
```

Named Tuple

let somePointNamed = (**x: 1, y: 1**)

somePointNamed**.x**

somePointNamed**.y**

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Function

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Named parameters

```
func sayHi( name: String, day: String ) -> String {
    return "Hi \(name), today is \(day)"
}
sayHi("somkiat", day: "Monday")
```

Named parameters optional

```
func sayHi( name: String, _ day: String ) -> String {
    return "Hi \(name), today is \(day)"
}
sayHi(“somkiat", "Monday")
```

Named parameters for external

```
func sayHi( fullName name: String,
            dayOfWeek day: String ) -> String {
    return "Hi \(name), today is \(day)"
}
```

```
sayHi(fullName: "Somkiat", dayOfWeek: "Monday")
```

Variadic parameters

```
func hiAll( names: String... ) -> String {
    var message:String = ""
    for name in names {
        message += "Hello \(name)" + "\n"
    }
    return message
}
print(hiAll("Somkiat", "Up1"))
```

In-Out parameters

```
func swap( inout first: Int, inout _ second: Int) {
    let tempFirst = first
    first = second
    second = tempFirst
}

var first = 1, second = 2
swap(&first, &second)
```

Return multiple result with tuple

```
func priceOf() -> (Double, Double, Double) {
    return (1.0, 2.0, 3.0)
}

print(priceOf().0)
print(priceOf().1)
print(priceOf().2)
```

Improve named

```
func priceOf() -> (min: Double, max: Double, sum: Double) {
    return (1.0, 2.0, 3.0)
}

print(priceOf().min)
print(priceOf().max)
print(priceOf().sum)
```

Functions are first-class type

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Function can be return value

```
func makeIncrement() -> (Int -> Int) {
    func addOne(number: Int) ->Int {
        return 1 + number
    }
    return addOne
}

var increment = makeIncrement()
increment(5)
```

Function parameter

```
func hasAnyMatch(numbers: [Int],
                 condition: Int -> Bool) -> Bool {
    for number in numbers {
        if condition(number) {
            return true
        }
    }
    return false
}

func lessThanTen(number: Int) -> Bool {
    return number < 10
}

var numbers = [30, 18, 1, 50]
hasAnyMatch(numbers, condition: lessThanTen)
```

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Anonymous function

```
hasAnyMatch(numbers,
        condition: { number in number < 3 })
```

```
hasAnyMatch(numbers,
        condition: { $0 < 3 })
```

Anonymous function

```
hasAnyMatch(numbers) {
    number in number < 3
}
```

Anonymous function

```
hasAnyMatch(numbers) {
    $0 < 3
}
```

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Where are Classes ?

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Hello class

```
class Employee {
    var id: Int = 0
    var firstName: String
    var lastName: String


    init(firstName: String, lastName: String) {
        self.firstName = firstName
        self.lastName = lastName
    }

    func simpleInfomation() -> String {
        return "Employee infomation \(firstName) \(lastName)"
    }
}

var employee = Employee(firstName: "somkiat", lastName: "pui")
employee.simpleInfomation()
```

Calculate properties

```
class Rectangle {
    var sideLength: Double = 5.0
    var area: Double {
        get {
            return sideLength * sideLength
        }

        set {
            sideLength = newValue
        }
    }
}

var shape = Rectangle()
shape.area = 10
print(shape.area)
```

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Observable properties

```
class Rectangle {
    var sideLength: Double = 5.0 {
        willSet {
            print("Set new value id \(newValue)")
        }

        didSet {
            print("Set new value success")
        }
}
```

Overriding

```
class Shape {
    func area() -> Double {
        return 0.0
    }
}
```

```
class Square: Shape {
    var width: Double = 0.0

    override func area() -> Double {
        return width * width
    }
}
```

```
var square = Square()
square.width = 5
square.area()
```

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Structure

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Like class

BUT pass by value

Struct used as Value Type

Data component manipulated by class

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Struct can not inheritance

```
struct Resolution {
    var width = 0
    var height = 0

    init() {}
}
class Video {
    var resolution = Resolution()
    var name: String?
    var frameRate = 0.0

    init() {}
}
```

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Struct as a Value Type

```
var normal = Resolution(width:600, height: 480)
var hd = normal

hd.width = 1000
```

```
print(normal.width) ?
print(hd.width) ?
```

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Good choice for struct

```
struct Point {
    var x = 0
    var y = 0
}
```

```
struct Point3D {
    var x = 0
    var y = 0
    var z = 0
}
```

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Enumeration

Named value type

```
enum iOSDeviceType {
    case iPhone
    case iPad
    case iWatch
}
```

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Dot syntax

```
enum iOSDeviceType {
    case iPhone
    case iPad
    case iWatch
}
```

```
var myDevice: iOSDeviceType = .iPhone

if myDevice == .iPhone {
    print("This is iPhone")
}
```

Associated values

```
enum iOSDeviceType {
    case iPhone(String)
    case iPad(String)
    case iWatch
}

var myDevice = iOSDeviceType.iPhone("6")
```

```
switch myDevice {
case .iPhone(let model):
    print("iPhone \(model)")
case .iPad(let model):
    print("iPad \(model)")
case .iWatch:
    print("iWatch")
}
```

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Raw values

```
enum Direction: Int {
    case Up = 1
    case Down
    case Right
    case Left
}

var direction = Direction(rawValue: 4)
print(direction?.rawValue)
```

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Raw values

```
if let direction = Direction(rawValue: 10) {
    switch direction {
    case .Up :
        print("Up direction")
    case .Down :
        print("Down direction")
    case .Right :
        print("Right direction")
    case .Left :
        print("Down direction")
    }
} else {
    print("Invalid direction")
}
```

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Emoji programming

```
var 😄 = "Smiley"
println(😄) // will print "Smiley"
let 🌎 = "🐶🐴🐱🐭"
var 🚢: String[] = []
for 💕 in 🌎 {
    🚢.append(💕+💕)
}
println(🚢) // will print [🐶🐶, 🐴🐴, 🐱🐱, 🐭🐭]
```

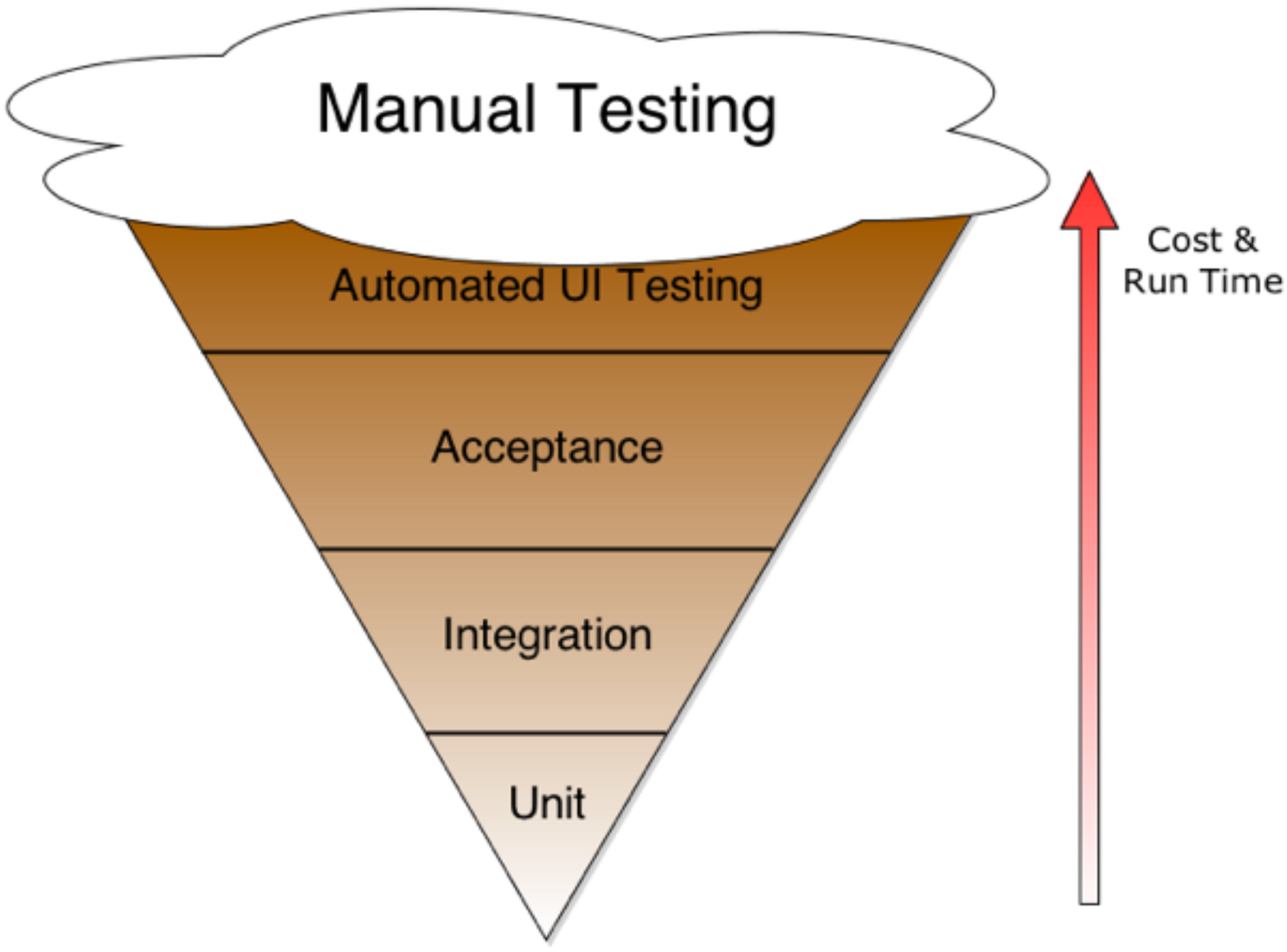
control + command + spacebar

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

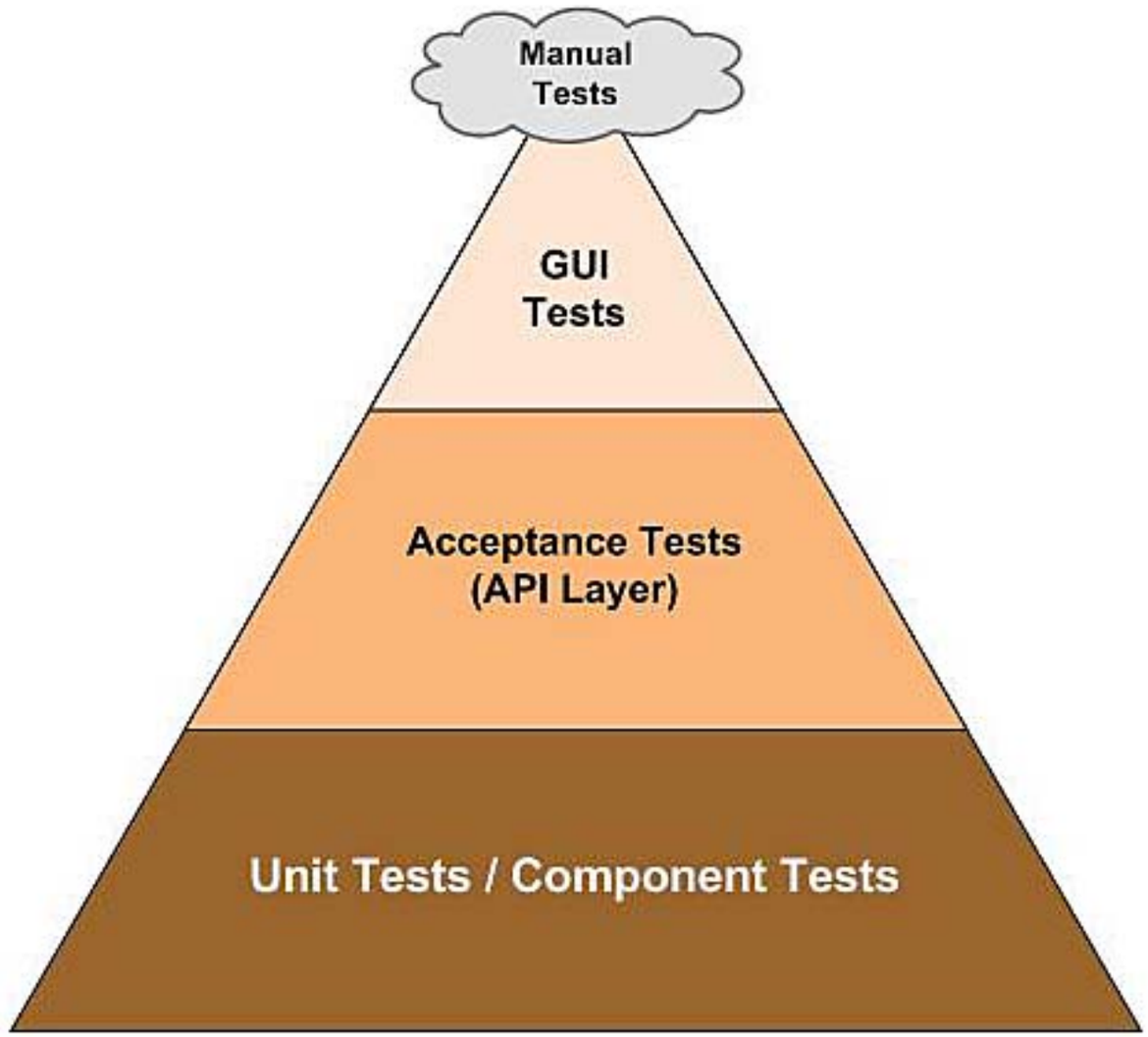
บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

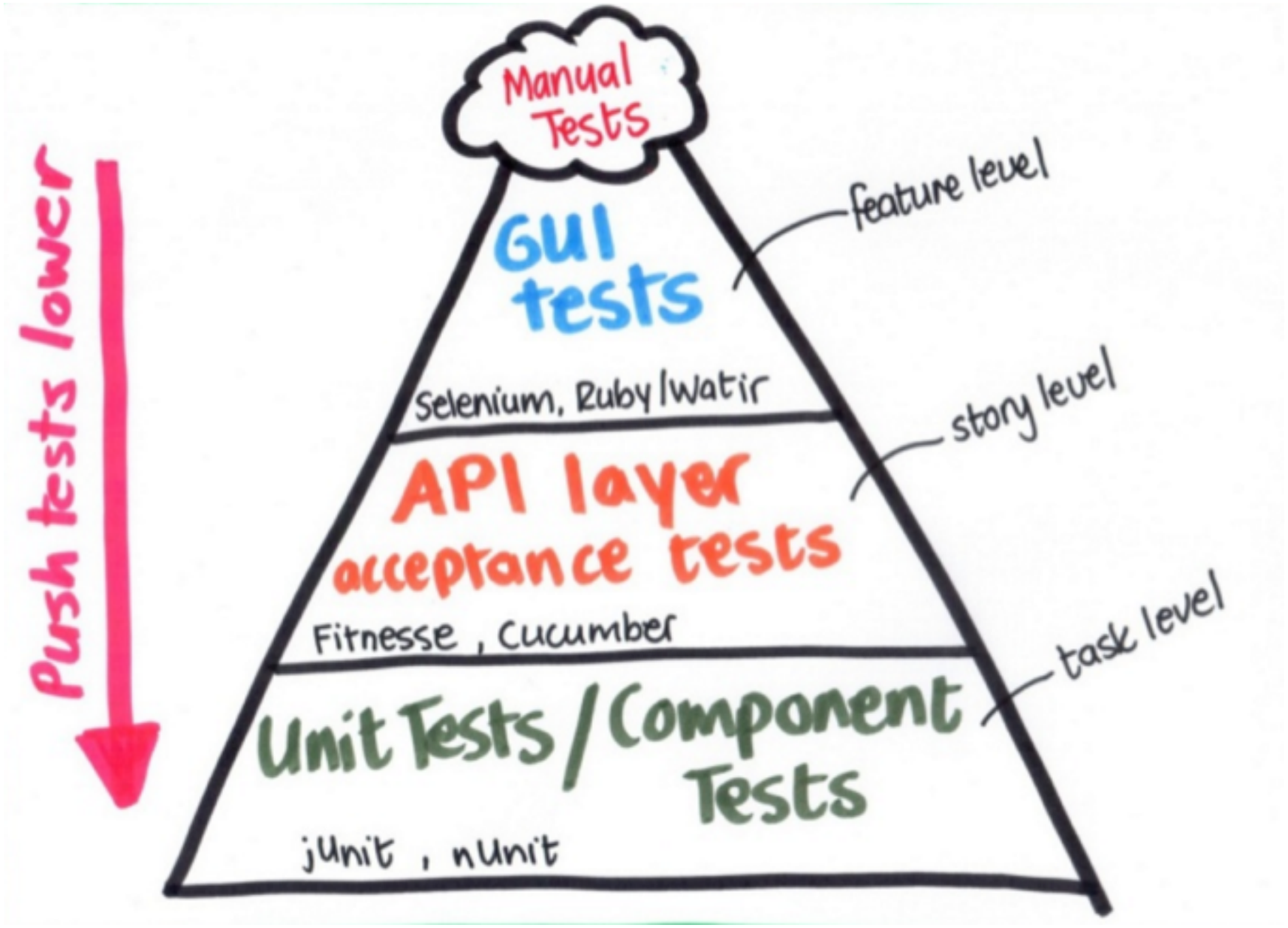
Mobile Testing !!



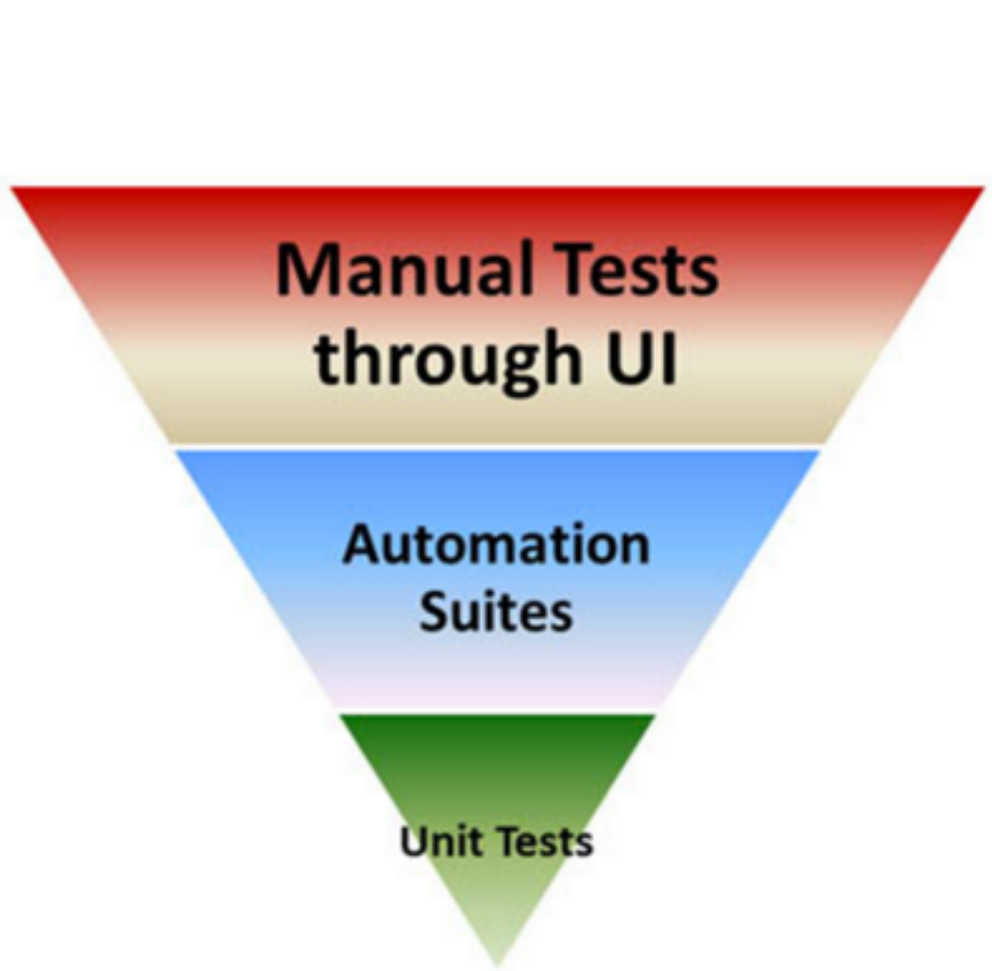
Mobile Testing



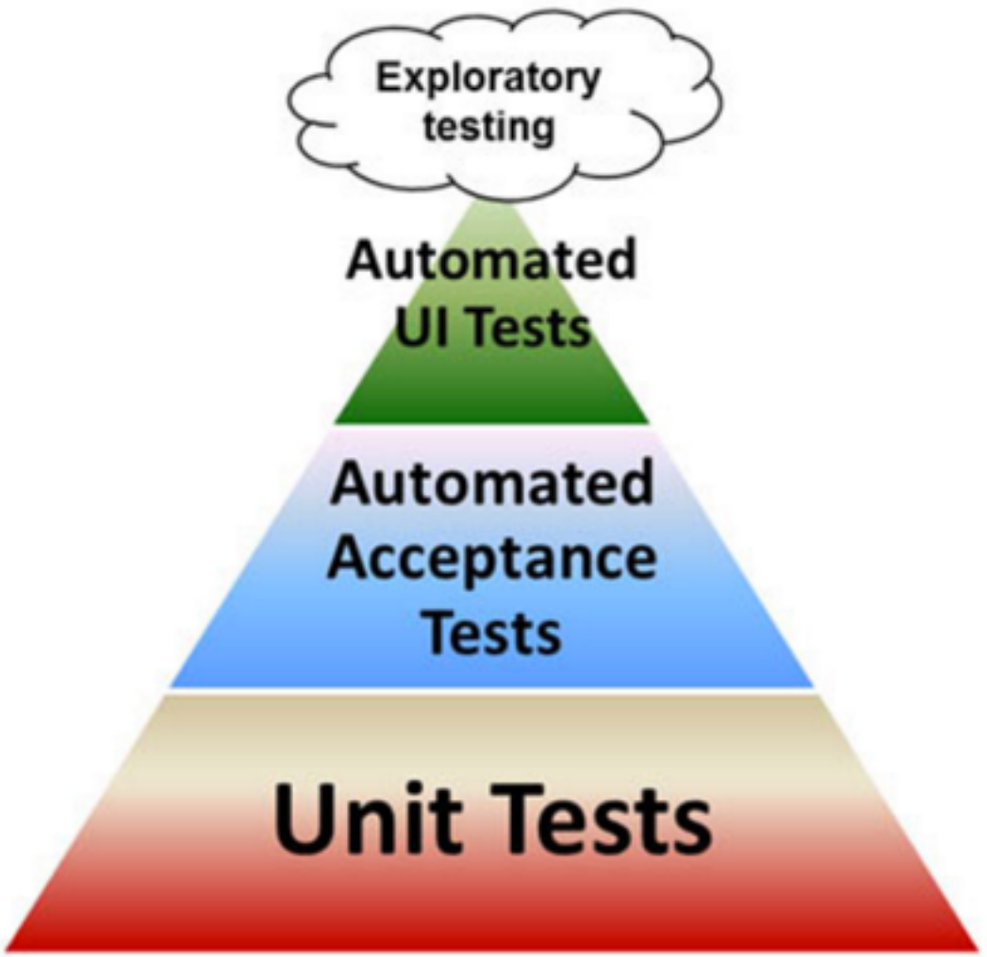
Mobile Testing



บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่



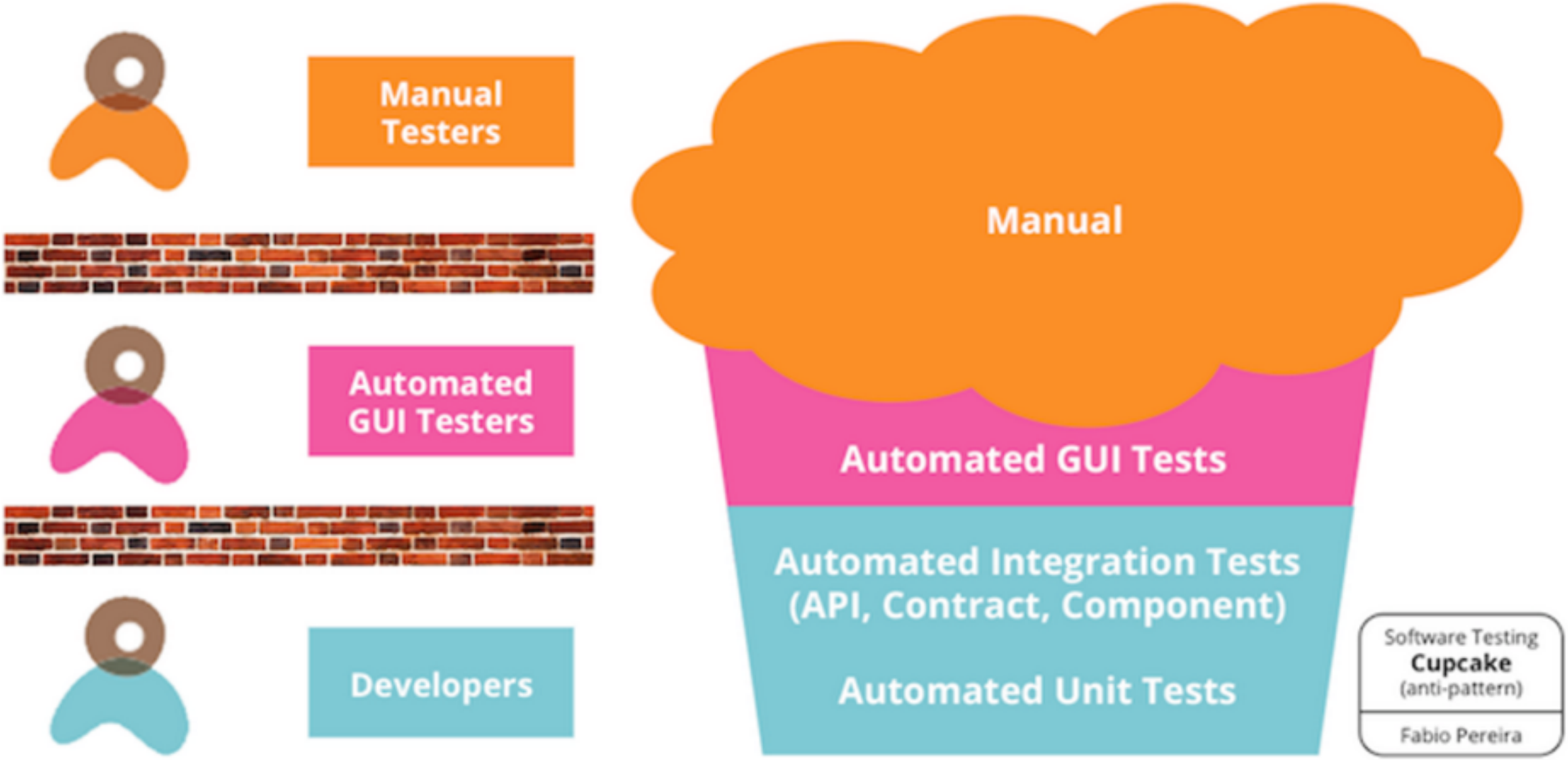
Traditional
(find bugs)



Agile
(prevent bugs)

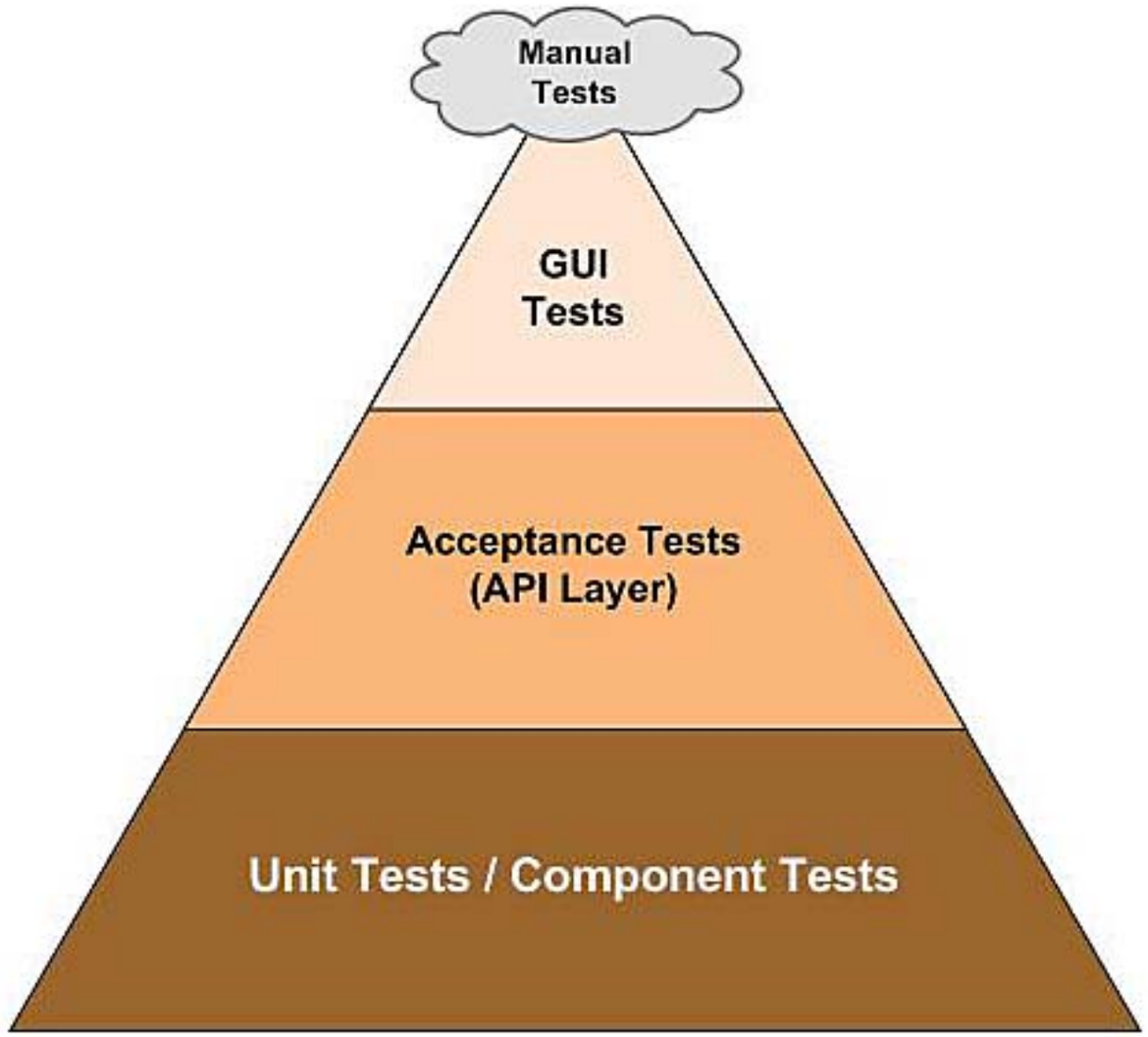
http://testingtweaks.com/what-is-angile-testing/

Mistake !!



https://www.thoughtworks.com/insights/blog/introducing-software-testing-cupcake-anti-pattern

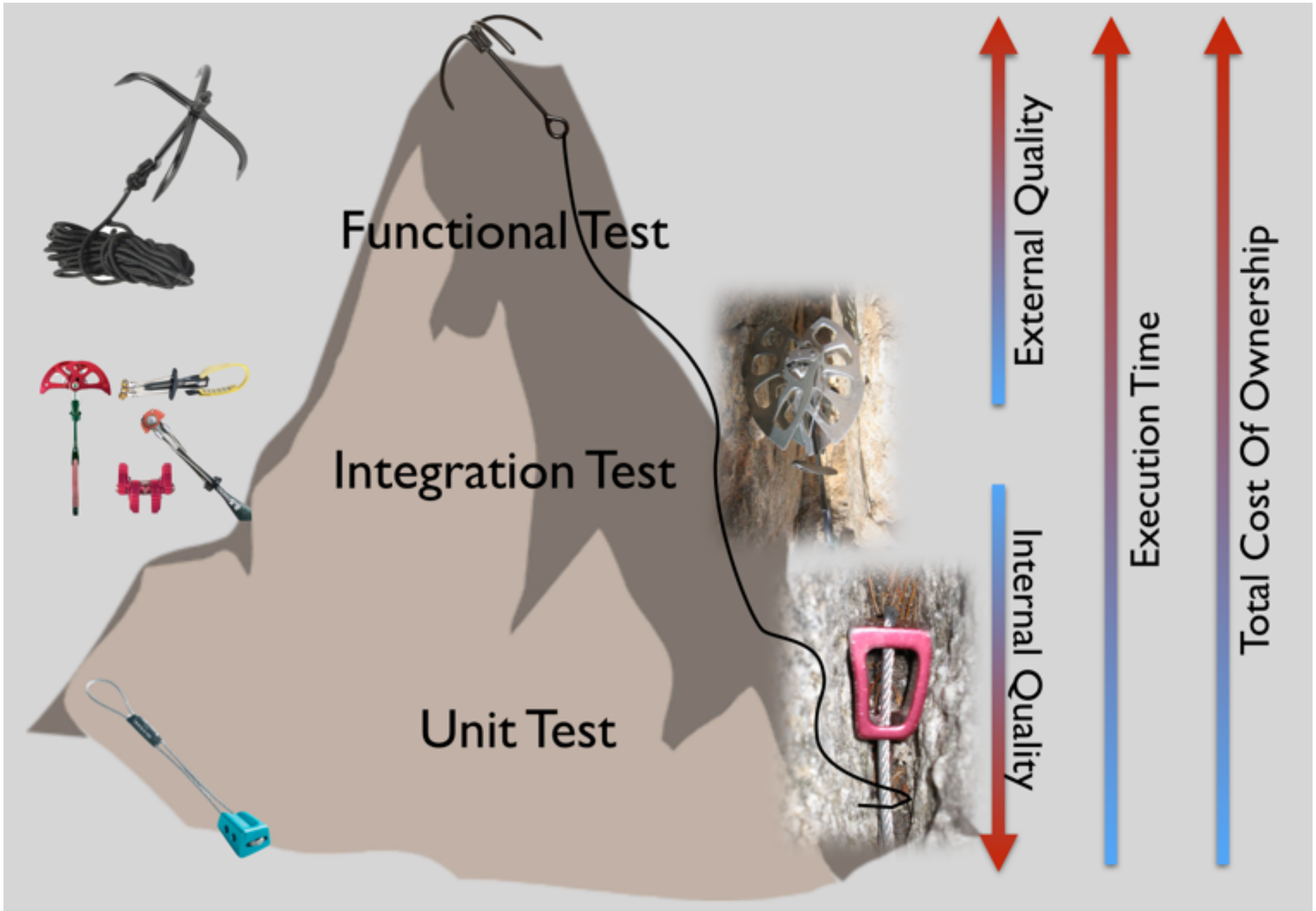
iOS Testing



UI Test

Unit Test

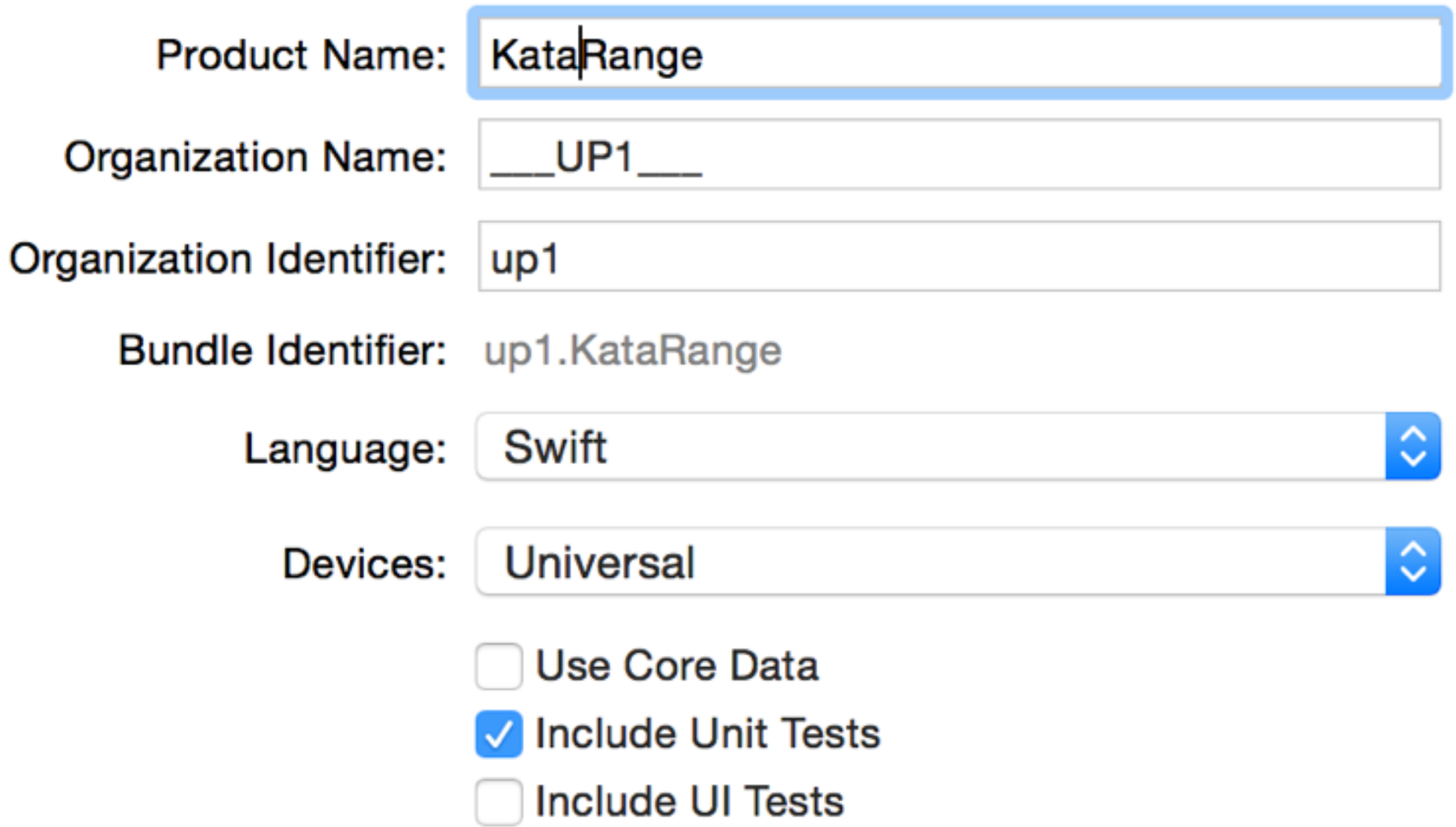
Testing


Functional Test
Integration Test
Unit Test
External Quality
Internal Quality
Execution Time
Total Cost Of Ownership

Getting start Unit testing with XCTest

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

New project with Unit test



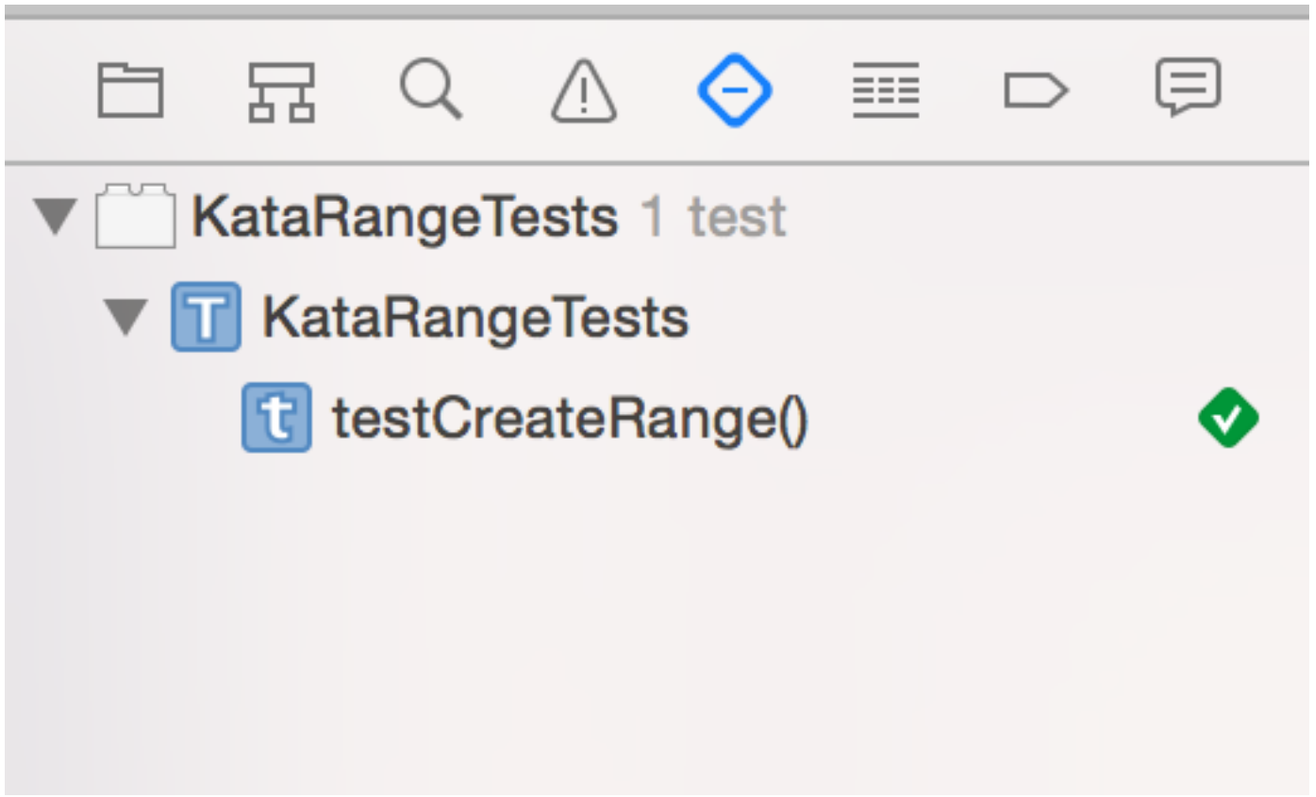
Structure of Test

```
import XCTest
@testable import KataRange

class KataRangeTests: XCTestCase {

    func testCreateRange() {
        //Arrange
        let range = MyRange()

        //Act
        range.setData("(1,5)")

        //Assert
        XCTAssertEqual("2,3,4", range.show())
    }

}
```

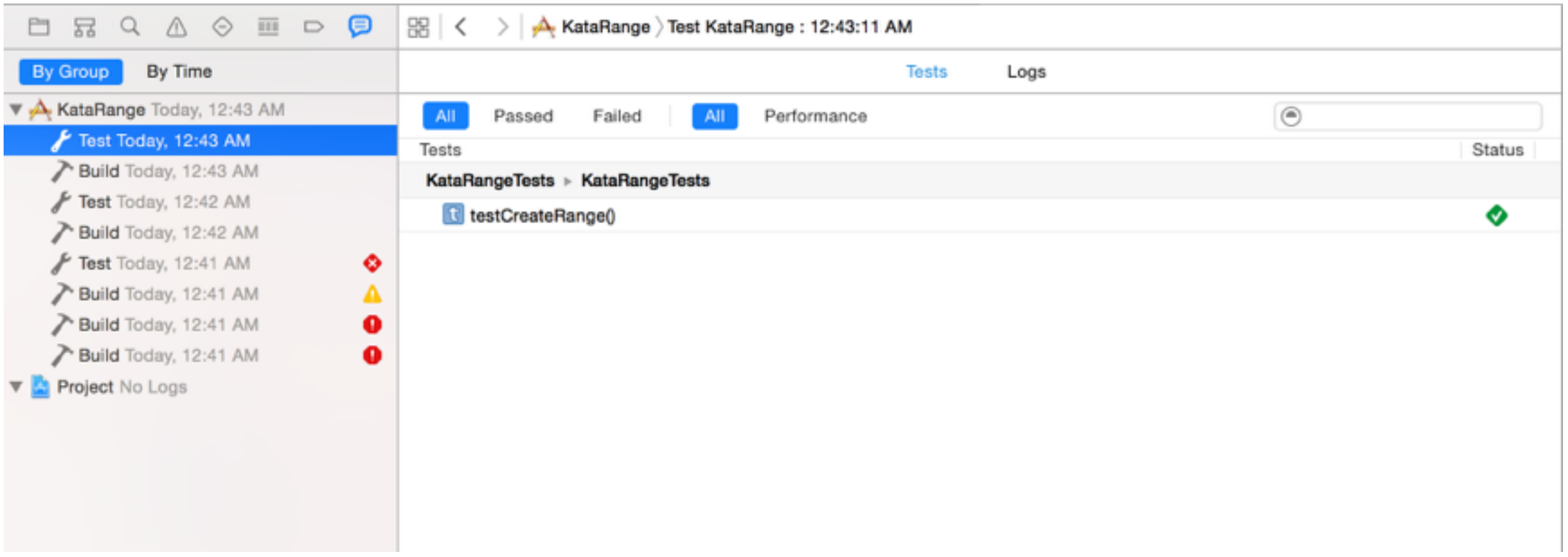
XCTest Assertion

- XCTFail (format…)
- XCTAssertNil (a1, format…)
- XCTAssertNotNil (a1, format…)
- XCTAssert (a1, format…)
- XCTAssertTrue (a1, format…)
- XCTAssertFalse (a1, format…)
- XCTAssertEqualObjects (a1, a2, format…)
- XCTAssertEquals (a1, a2, format…)
- XCTAssertEqualsWithAccuracy (a1, a2, accuracy, format…)
- XCTAssertThrows (expression, format…)
- XCTAssertThrowsSpecific (expression, specificException, format…)
- XCTAssertThrowsSpecificNamed (expression, specificException, exceptionName, format…)
- XCTAssertNoThrow (expression, format…)
- XCTAssertNoThrowSpecific (expression, specificException, format…)
- XCTAssertNoThrowSpecificNamed (expression, specificExcepton, exceptionName, format…)

Test result



Test panel active

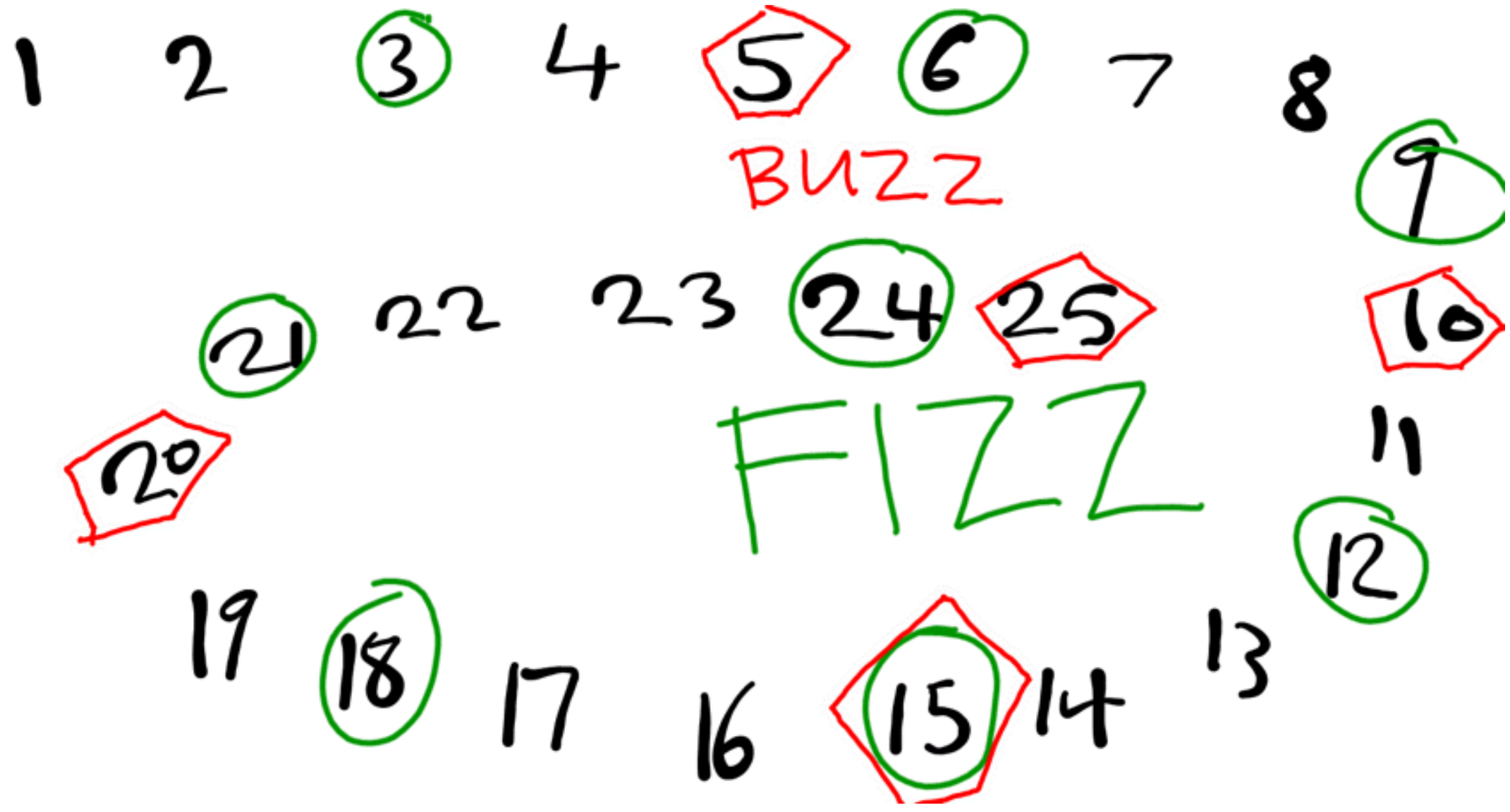


บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Write first test

FizzBuzz



บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Test ที่ดีต้อง FIRST

Fast
Independent
Repeatable
Self-validating
Timely

สิ่งที่ไม่ใช่ Unit test

External service

- File system
- Database
- WebService
- API calls

สิ่งที่ไม่ใช่ Unit test

Over specification

Compare screen image

Compare HTML

Test Naming

“What ’s in a name ?”

That which we call a rose

by any other name would smell

as sweet

Romeo and Juliet

Guide to Test Writing

Don’t say “**test**”, say “**should**”

SPRINT3R

Siam Chamnankit Co., Ltd., Odd-e (Thailand) Co., Ltd. and Alliance

Don’t use the word “test”

Use the word “should”

```
tranferShouldDeductSumFromSourceAccountBalance()

transferShouldAddSumLessFeesToDestinationAccountBalance()

depositShouldAddAmountToAccountBalance()
```

Guide to Test Writing

Don’t test your class, test **behaviour**

SPRINT3R

Siam Chamnankit Co., Ltd., Odd-e (Thailand) Co., Ltd. and Alliance

Guide to Test Writing

Test class names are important too

Structure your test well

Tests are **deliverable** too

SPRINT3R

Siam Chamnankit Co., Ltd., Odd-e (Thailand) Co., Ltd. and Alliance

Test Structure

The ratio of time spent (code)

versus writing is

over 10 to 1

Robert C. Martin, Clean Code

Workshop TDD

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Code แบบไหนที่ชอบ ?

Code แบบไหนที่ไม่ชอบ ?

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

PROBLEM

Circular Buffer

http://en.wikipedia.org/wiki/Circular_buffer

PROBLEM

Circular Buffer

http://en.wikipedia.org/wiki/Circular_buffer

5 นาที

ทำความเข้าใจกับปัญหา

สลับคู่ทุกๆ 5 นาที

แก้ไข code ต่อไป

Retrospective 5 นาที

5 นาที

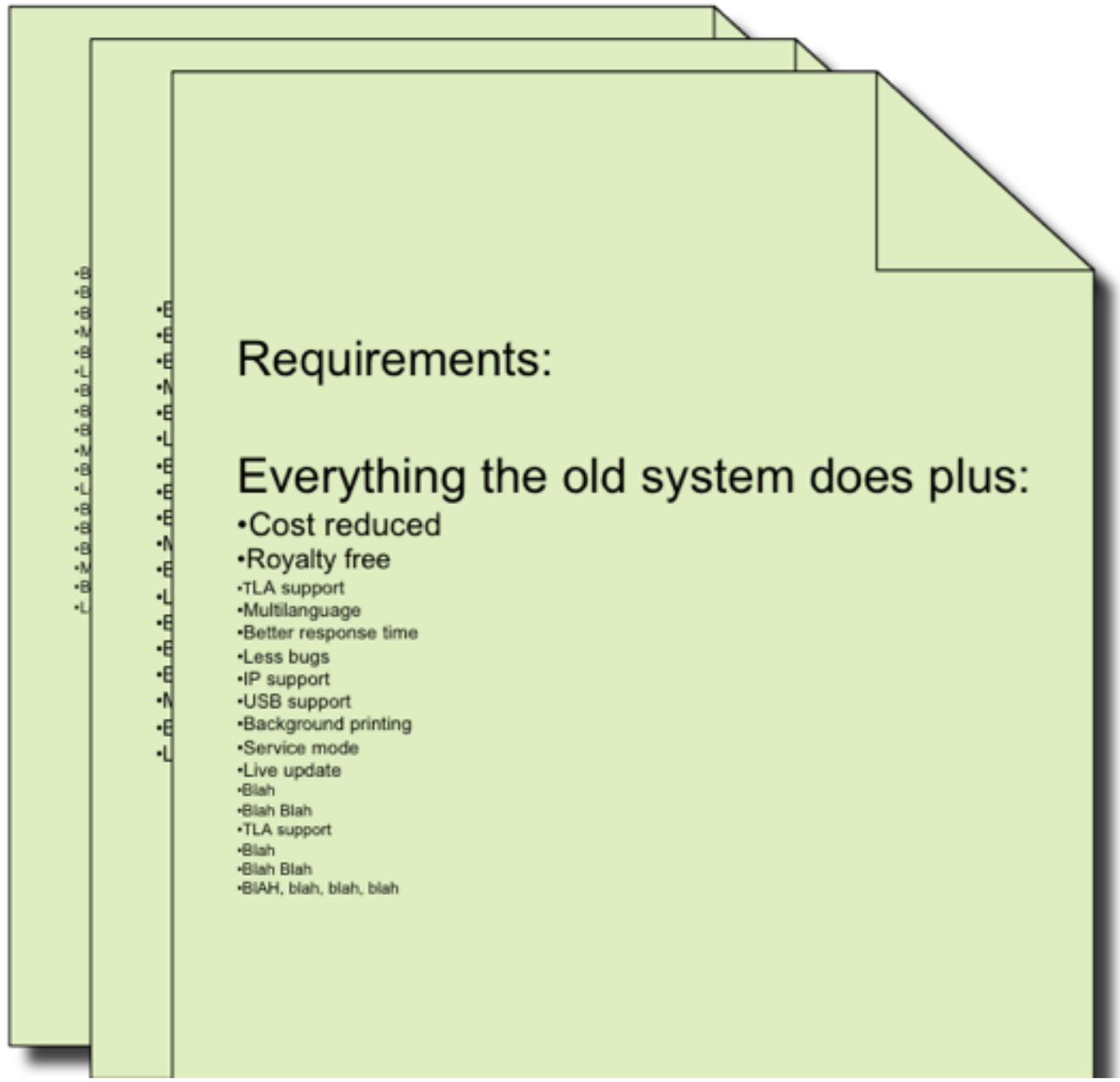
เขียน code ต่อไป

Retrospective 5 นาที

Is this how Software Developers view their Customer?

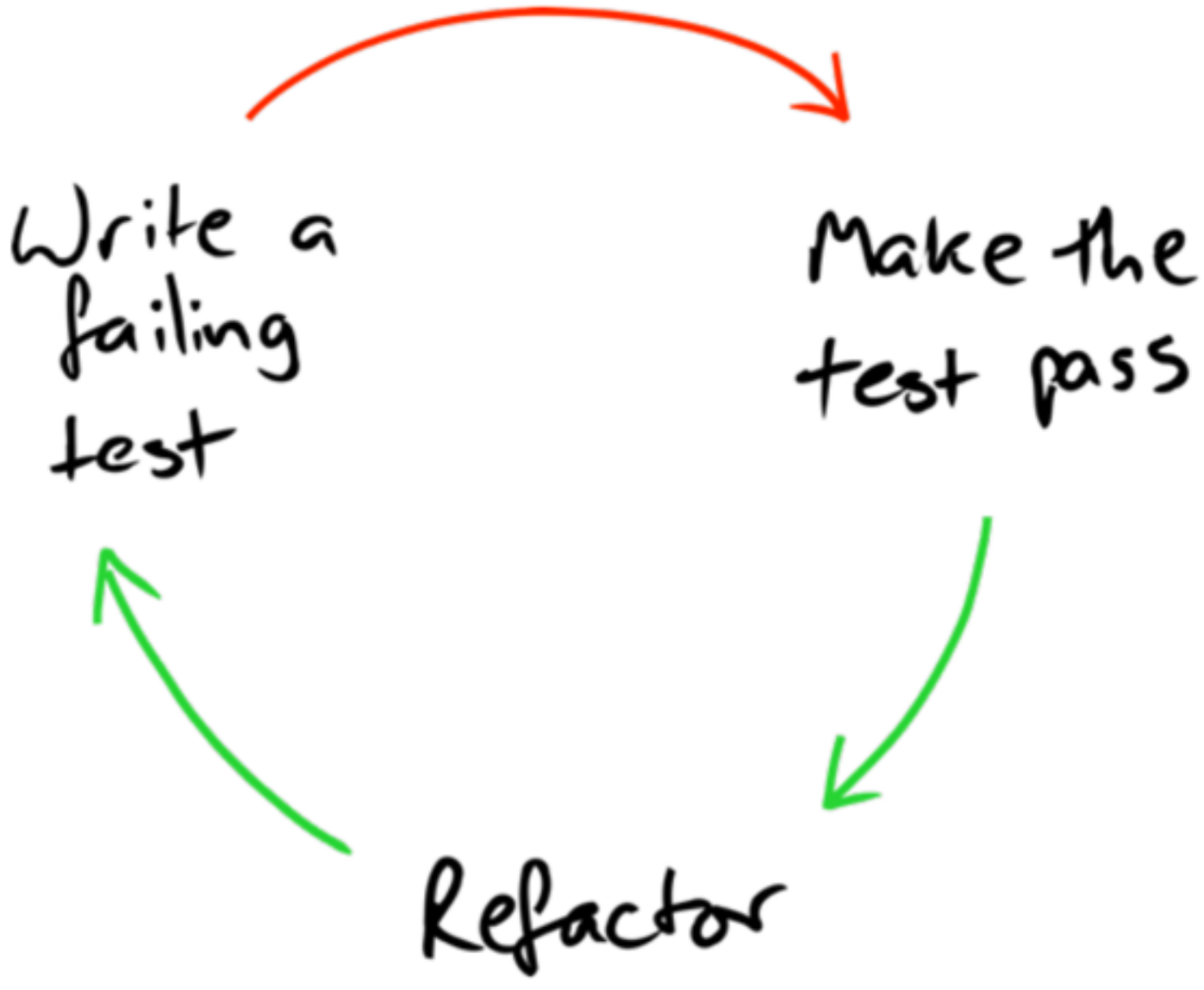
Requirements Change

- A sign of a successful system
- Customers want new features
- Marketing wants backwards compatibility
- Old behavior must be preserved

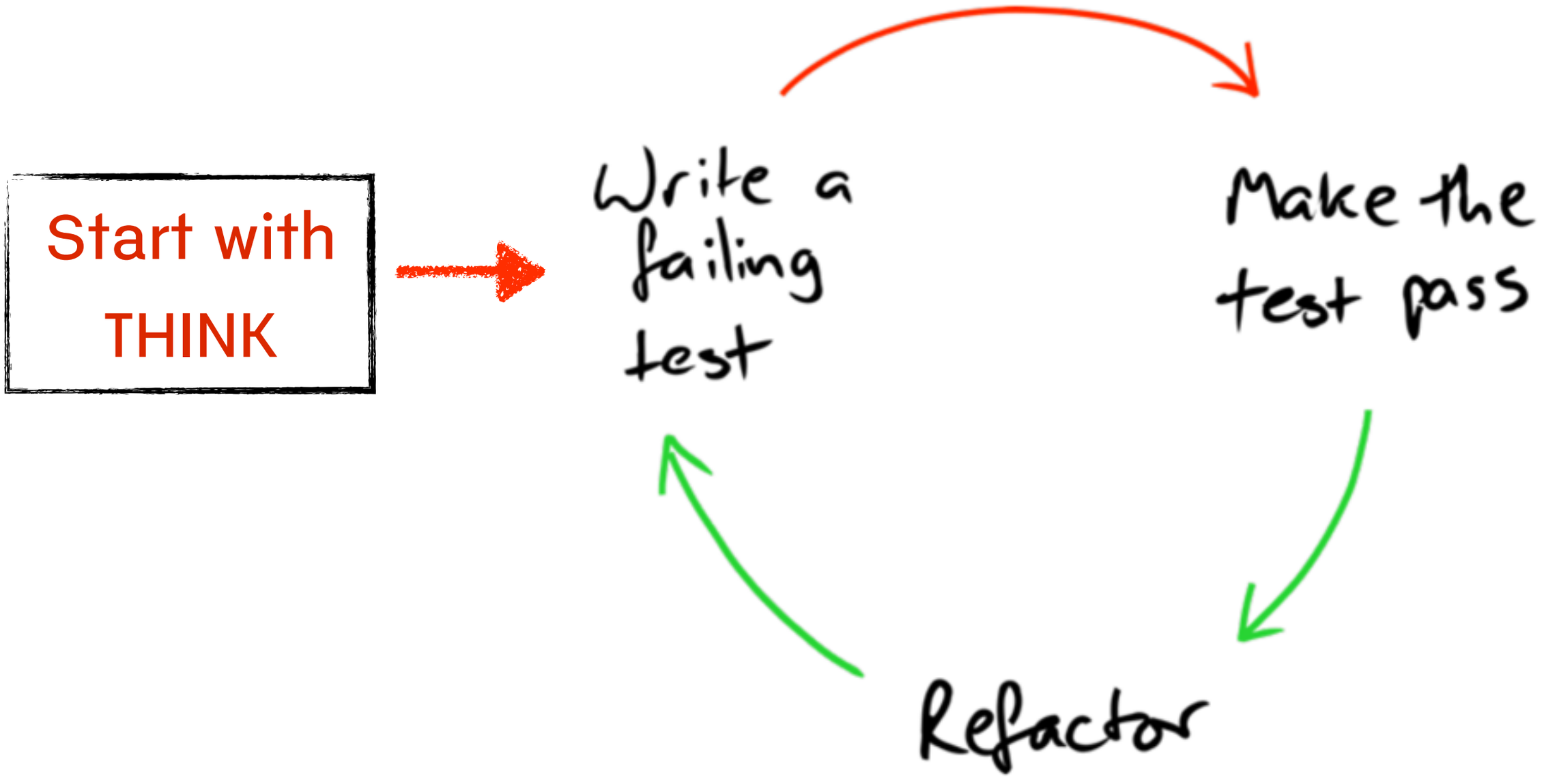


เราต้องการวิธีการใหม่ ?

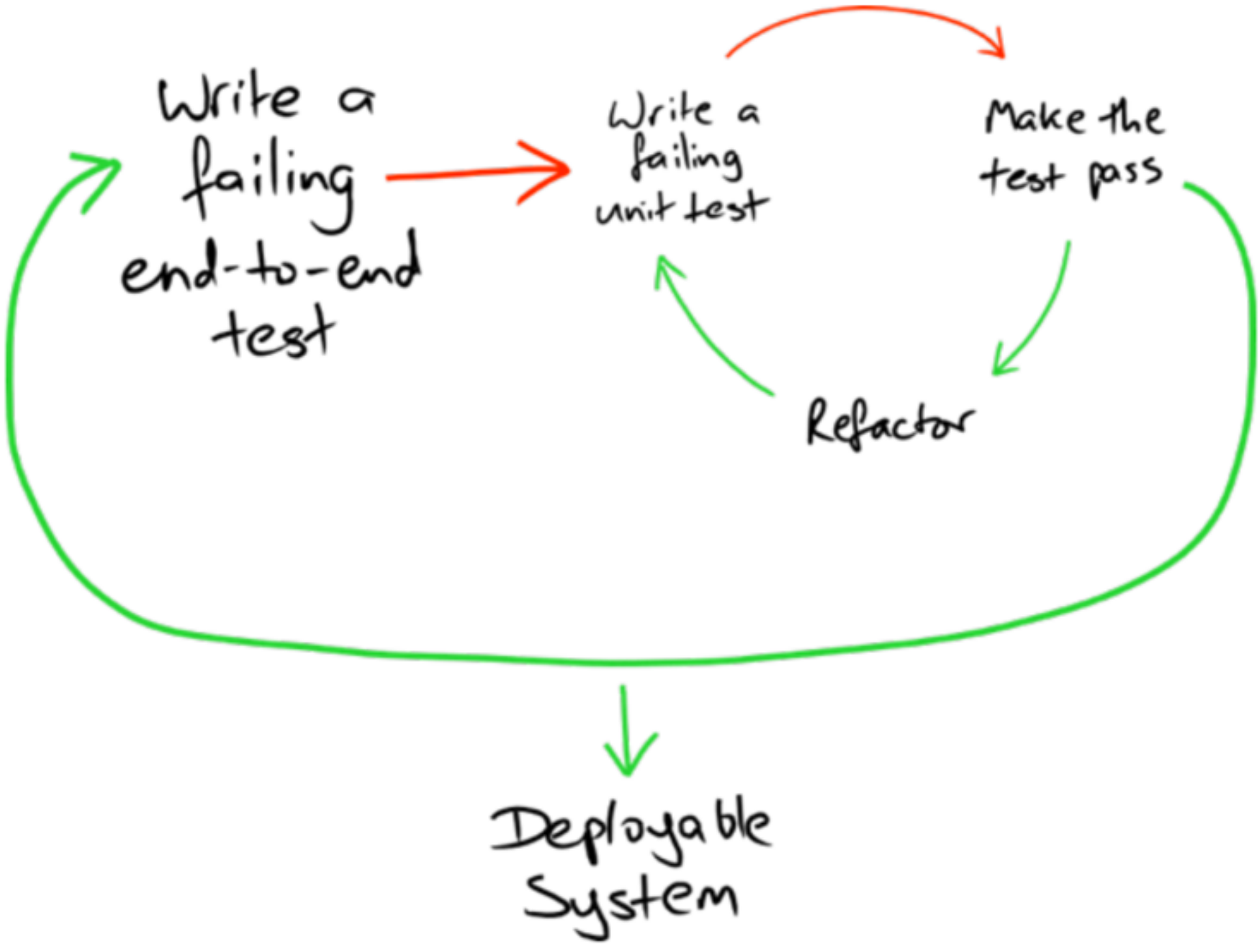
TEST DRIVEN DEVELOPMENT



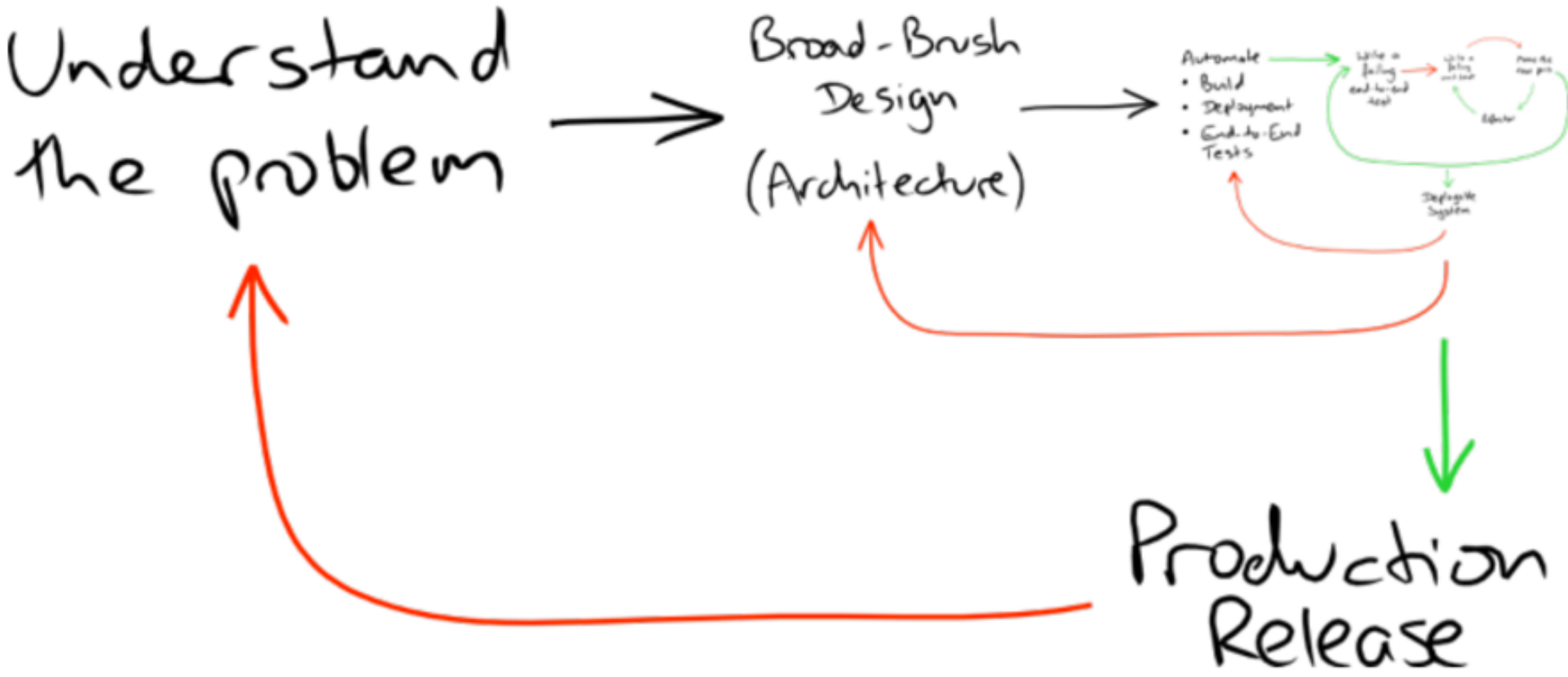
Improve TDD Cycle



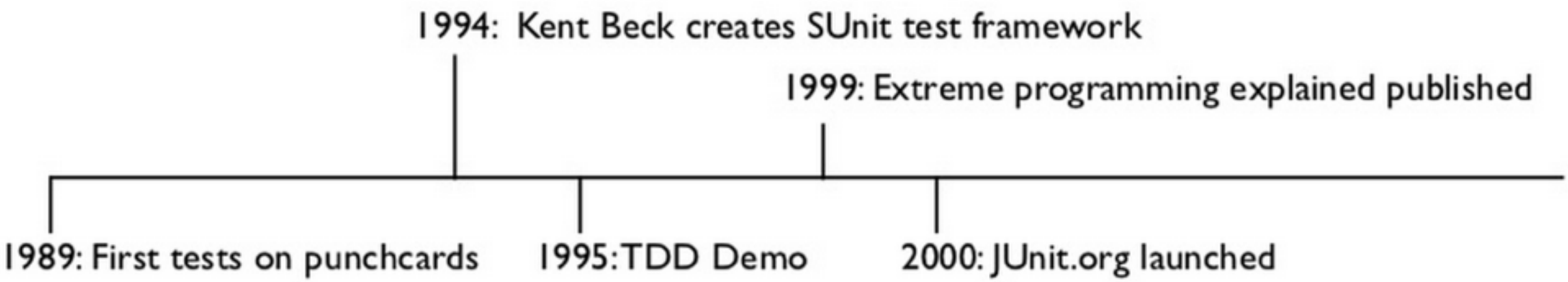
ACCEPTANCE TEST DRIVEN DEVELOPMENT



LARGER FEEDBACK LOOP



ประวัติของ TDD



Proactive vs Reactive

หาปัญหาก่อน

ทดสอบทีหลัง

ตรวจสอบ req ก่อนทำ

แก้ไขตาม defect ที่แจ้ง

ตรวจสอบ code

debug code

ทำไมต้อง TDD

Software **เปลี่ยน**อยู่เสมอ

คนเรามัก **มโน** และ ทำ**ผิดพลาด**อยู่เสมอ

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

ทำไมต้อง TDD

บอกคุณว่า ผิดตรงไหน

ไม่ต้องเสียเวลา **debug**

ทำไมต้อง TDD

Test นั้นช่วยคุณสร้างสิ่งที่ **นโม**

และสร้างความคาดหวังขึ้นมา

ทำไมต้อง TDD

ผลที่ได้คือ Testable design

Loosely coupled, High codesion

STEVE FREEMAN & NAT PRYCE

Never write **new functionality**

without a **failing test**

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

ROBERT C. MARTIN

Testing is about **trust**

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

SIMPLE DESIGN RULES

All the tests pass

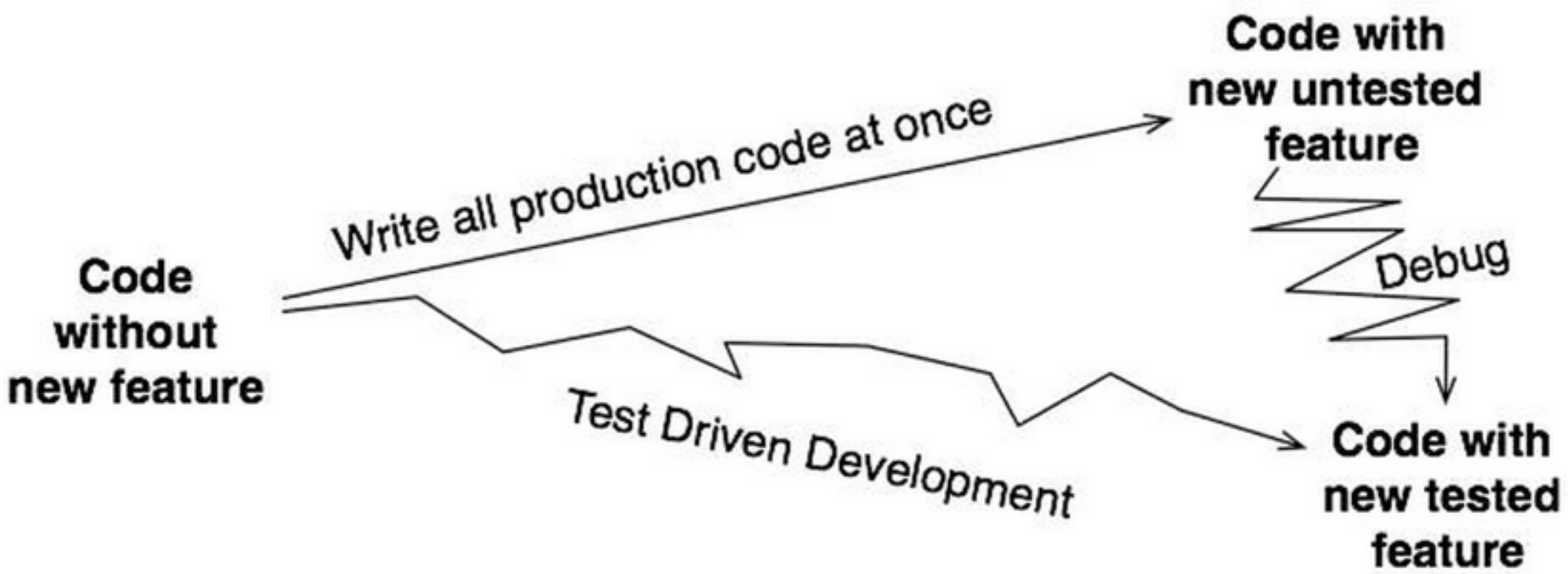
There is no duplication

Focus your intent

Classes and methods are minimized

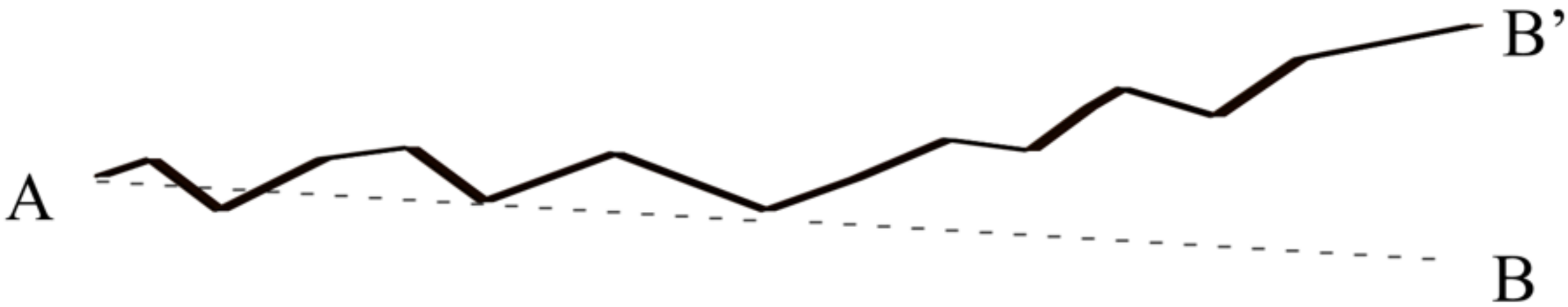
TDD vs DLP

DLP = Debug Later Programming



รูปจาก http://www.renaissancesoftware.net/

Small Step



ก่อนจะเริ่มเขียน Test

ทำการออกแบบก่อน

5 นาที

1. มี API หรือ method ชื่ออะไรบ้าง

2. Data structure ง่ายที่สุดที่จะใช้งาน

3. การทำงานภายในของระบบ ที่จะเกิดขึ้น

Test แรกที่จะเขียนคือ อะไร ?

Empty Buffer ?

สร้าง Buffer ครั้งแรกจะต้องว่าง ?

Empty Buffer ?

ทำการเพิ่มข้อมูล

Empty Buffer ?

ทำการเพิ่มข้อมูล และ ดึงข้อมูล ออกมาใช้

Full Buffer ?

สร้าง Buffer ครั้งแรกจะต้องเต็ม ?

อะไรต่อไป ?

เพิ่มข้อมูล

เพิ่มครั้งละหนึ่ง

เพิ่มครั้งละสอง

เพิ่มครั้งละสาม

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

กำหนดขนาดของ Buffer

ค่า default ของ Buffer

กำหนดค่าขนาดของ Buffer ได้เอง

อะไรต่อไป ?

Full Buffer

เพิ่มข้อมูลจนเต็ม

เพิ่มข้อมลจนเต็ม และ ดึงจนว่าง

อะไรต่อไป ?

ตรวจสอบพฤติกรรม

เพิ่มข้อมูล

เพิ่มข้อมูล

ดึงข้อมูล

เพิ่มข้อมูล

ดึงข้อมูล

ดึงข้อมูล

Overflow ?

Underflow ?

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Kata Range

Input	Result
[1,5]	1,2,3,4,5
[1,5)	1,2,3,4
(1,5]	2,3,4,5
(1,5)	2,3,4

TDD in Real world

Design Principle

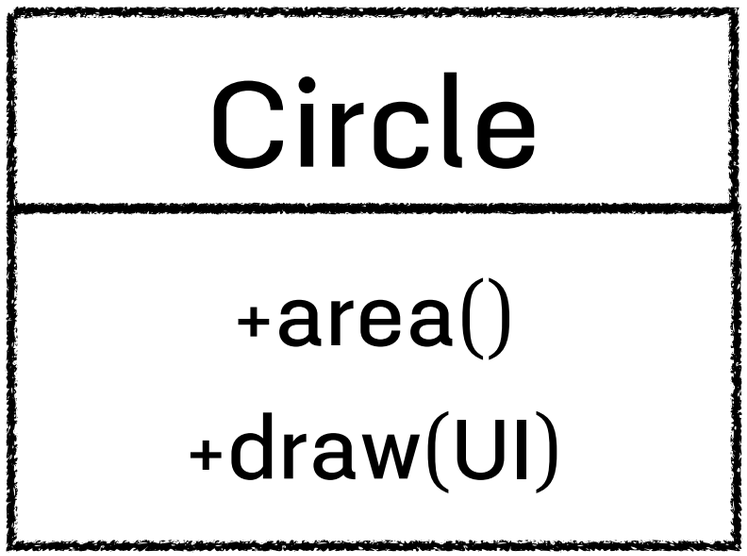
SPRINT3R

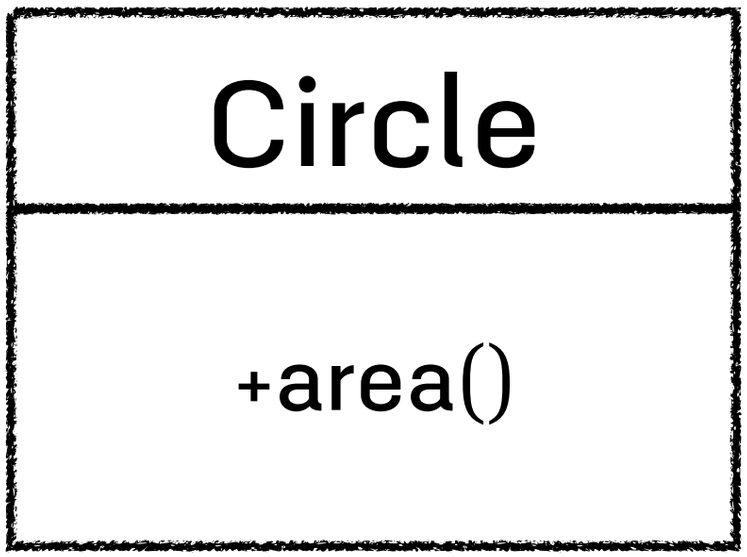
Siam Chamnankit Co., Ltd., Odd-e (Thailand) Co., Ltd. and Alliance

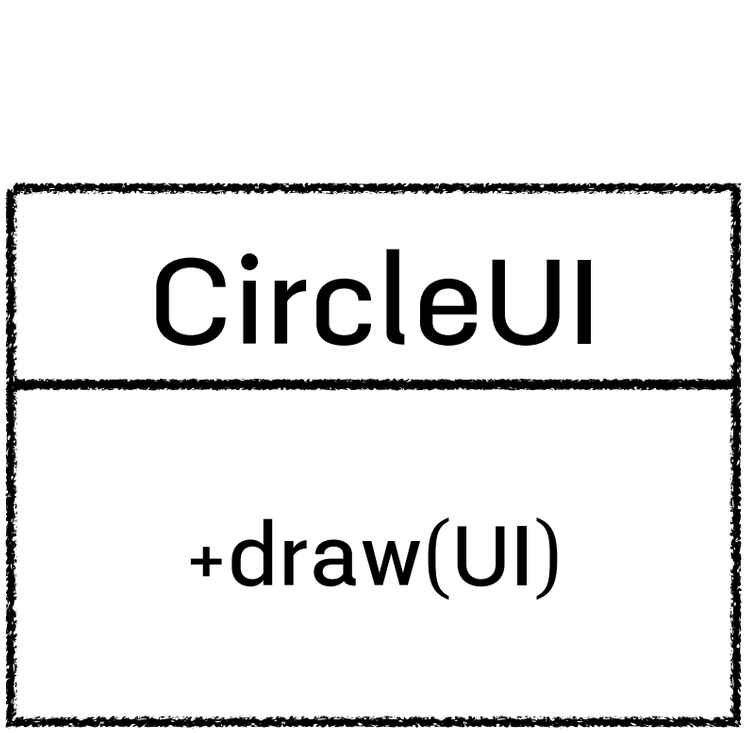

SOLID
Software development is not a Jenga game.


Single Responsibility Principle
Just because you *can* doesn't mean you *should*.

Single Responsibility Principle




Circle
+area()


CircleUI
+draw(UI)

Workshop

Single Responsibility Principle

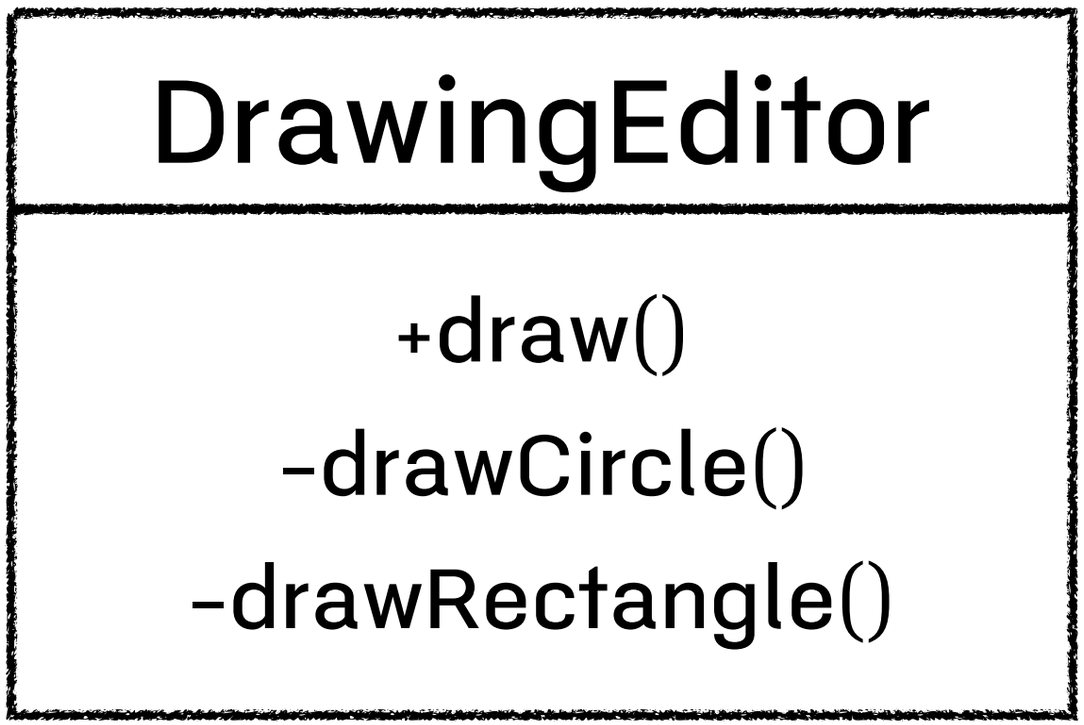
SPRINT3R

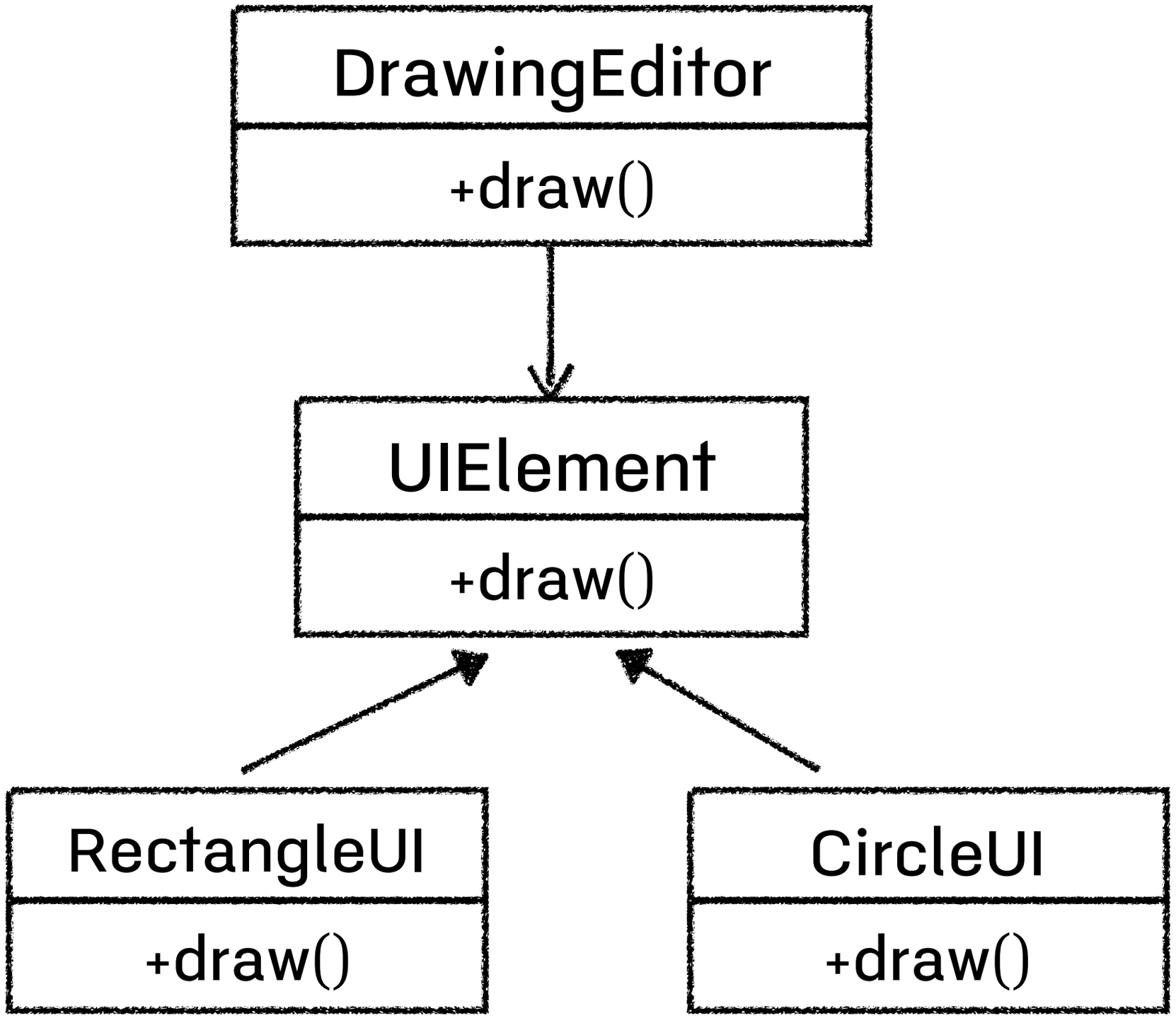
Siam Chamnankit Co., Ltd., Odd-e (Thailand) Co., Ltd. and Alliance

SPRINT3R

Siam Chamnankit Co., Ltd., Odd-e (Thailand) Co., Ltd. and Alliance

Open/Closed Principle



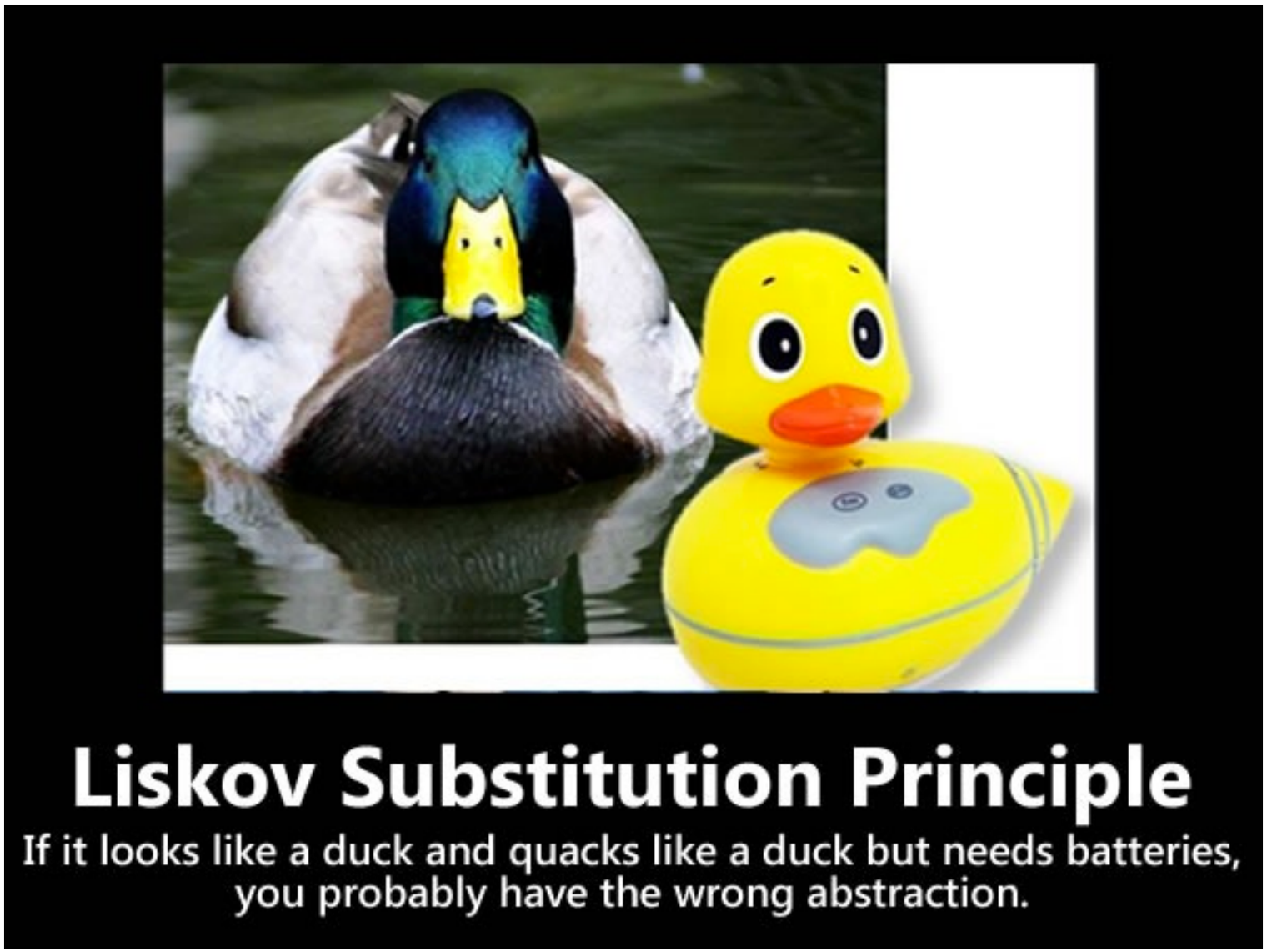

DrawingEditor
+draw()
UIElement
+draw()
RectangleUI
+draw()
CircleUI
+draw()

Workshop

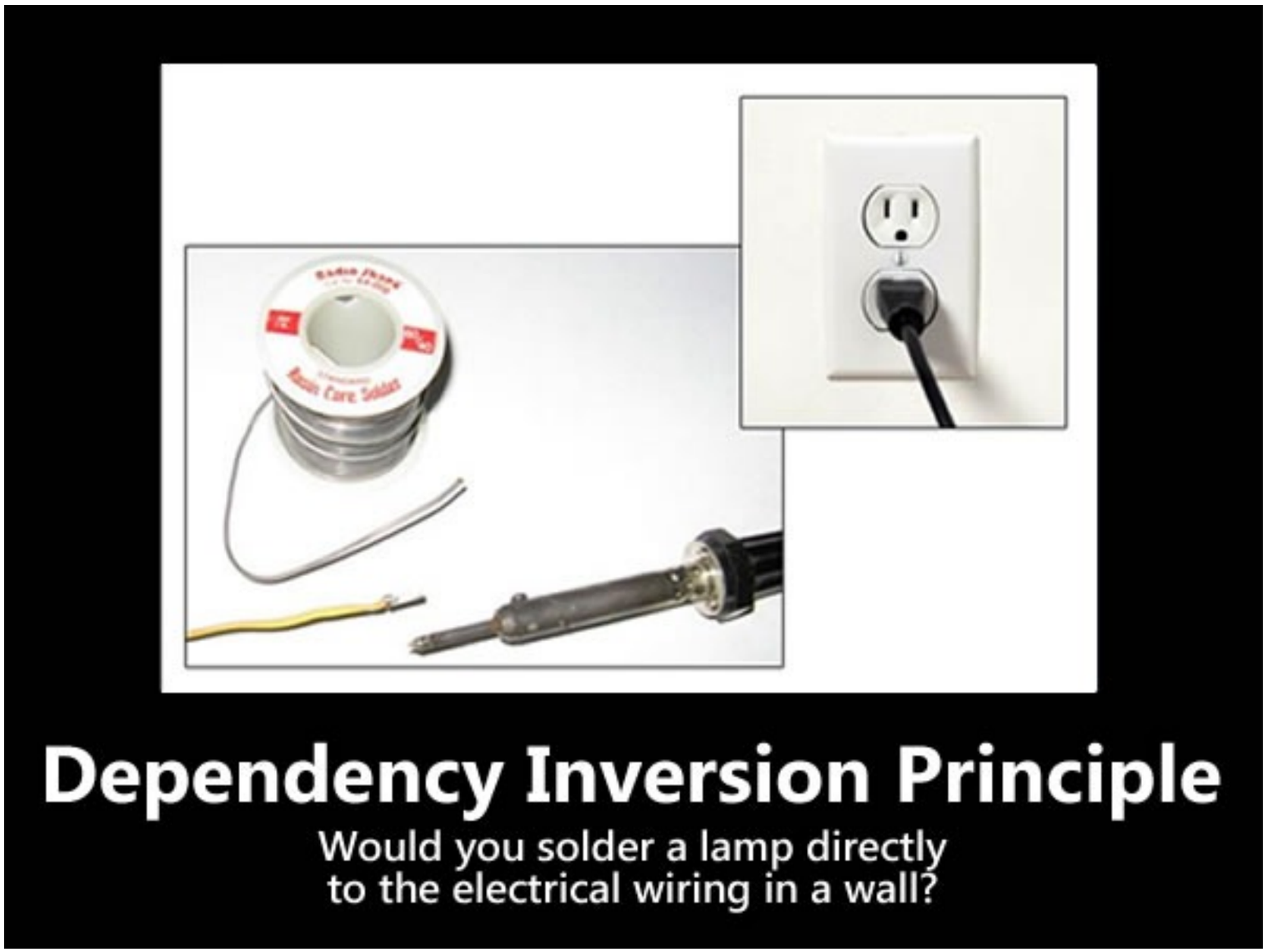
Open/Closed Principle

SPRINT3R

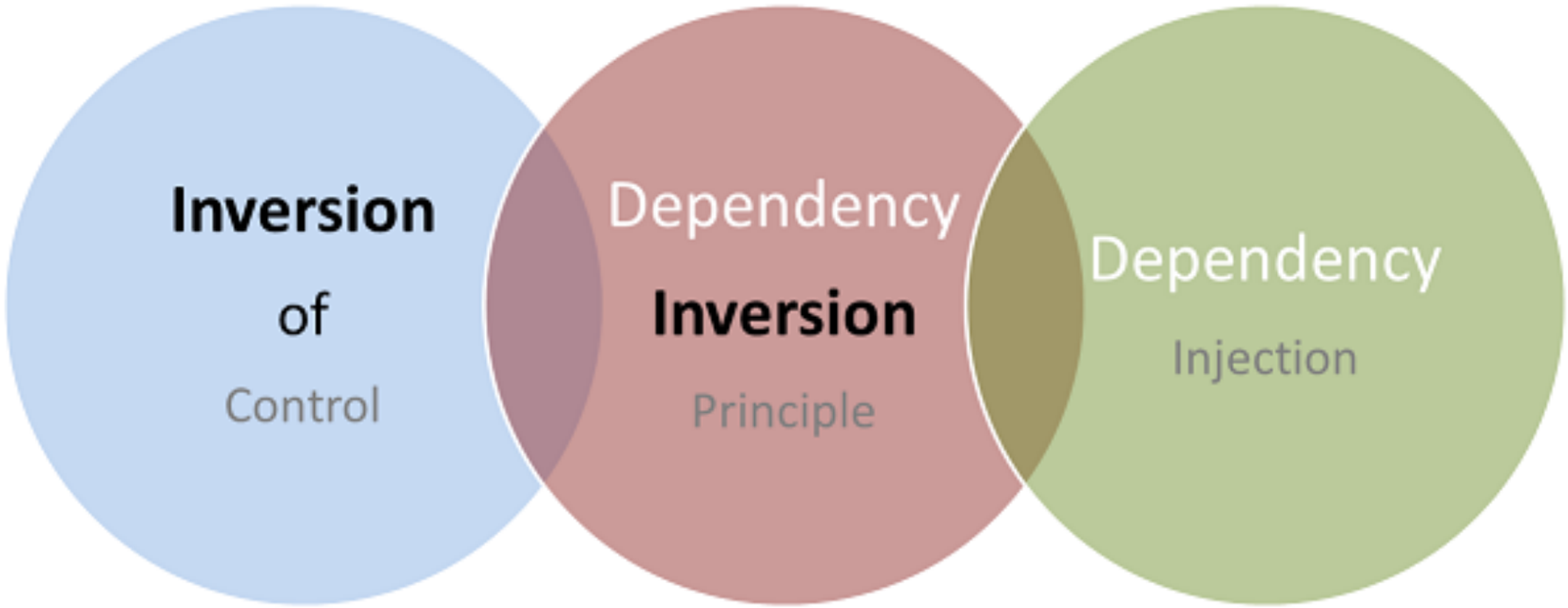
Siam Chamnankit Co., Ltd., Odd-e (Thailand) Co., Ltd. and Alliance


Liskov Substitution Principle
If it looks like a duck and quacks like a duck but needs batteries,
you probably have the wrong abstraction.


Interface Segregation Principle
You want me to plug this in *where?*

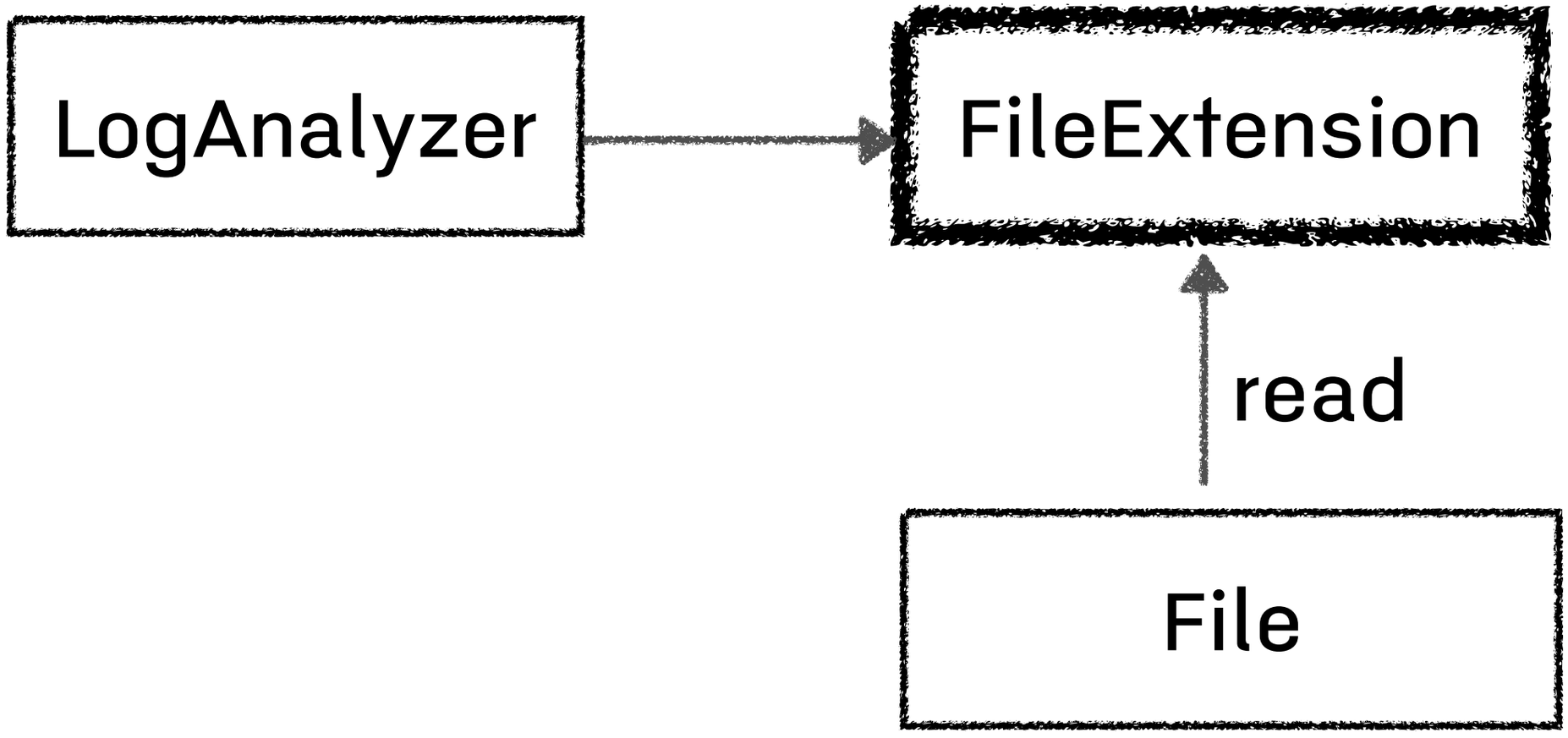

Dependency Inversion Principle
Would you solder a lamp directly
to the electrical wiring in a wall?

Dependency Inversion Principle



Dependency Inversion Principle

Move to new class



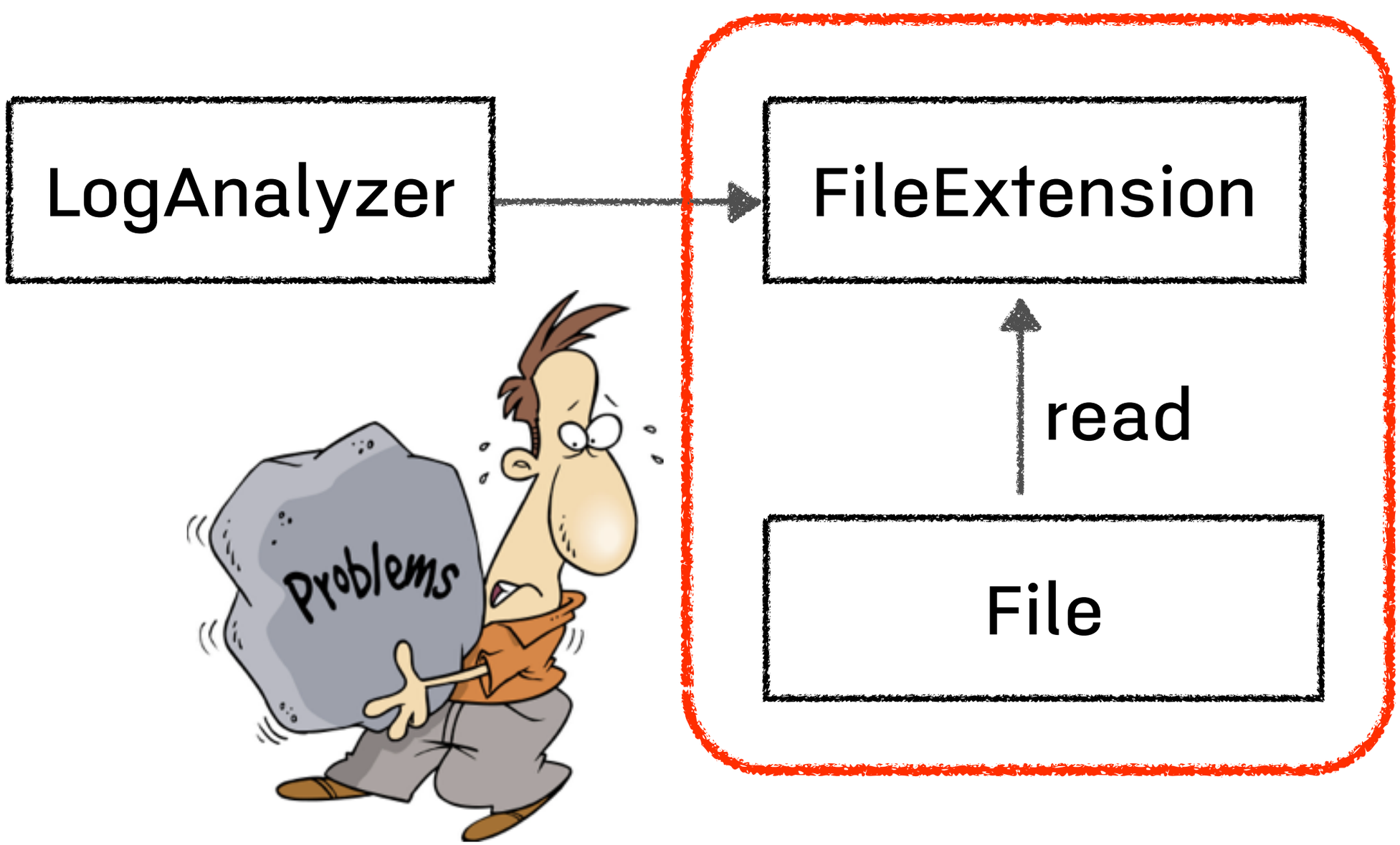
Workshop

Discuss about FileExtension.java

Problem ?

Fast
Independent
Repeatable
Self-validating
Timely

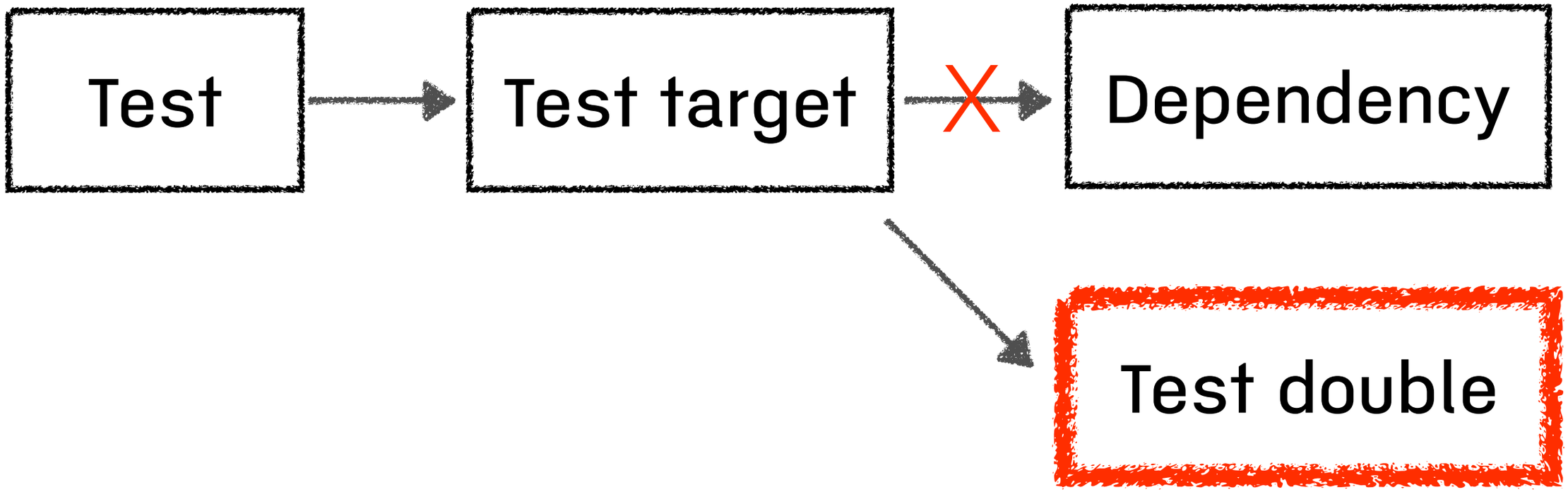
Dependency with File



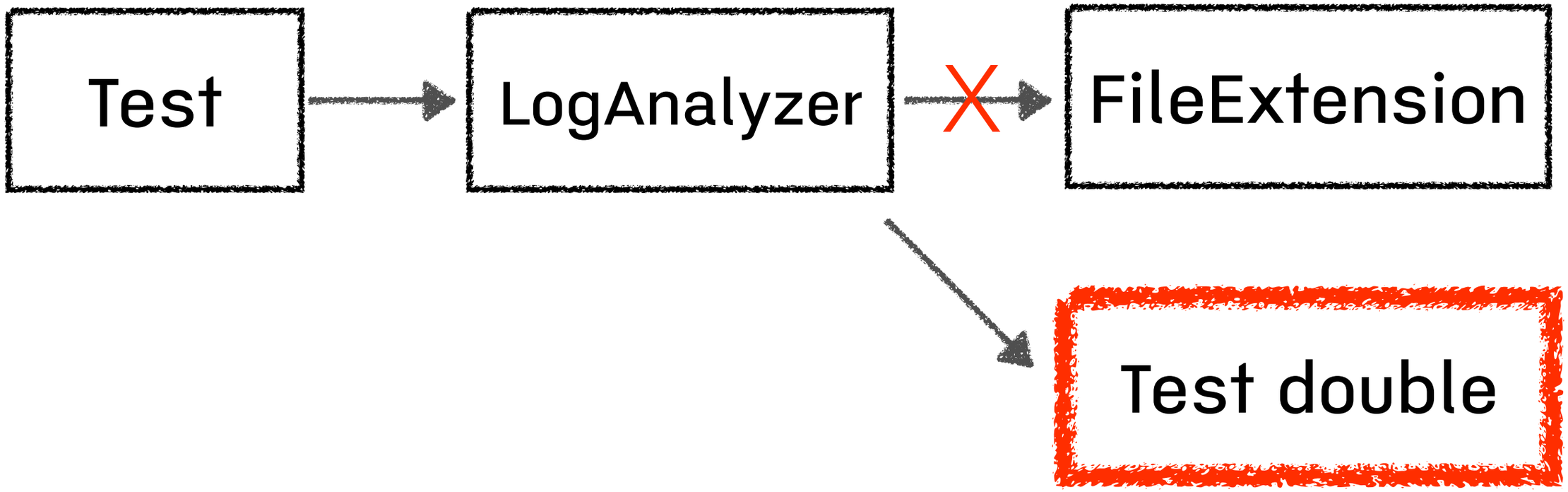
Discuss

How to Break dependency ?

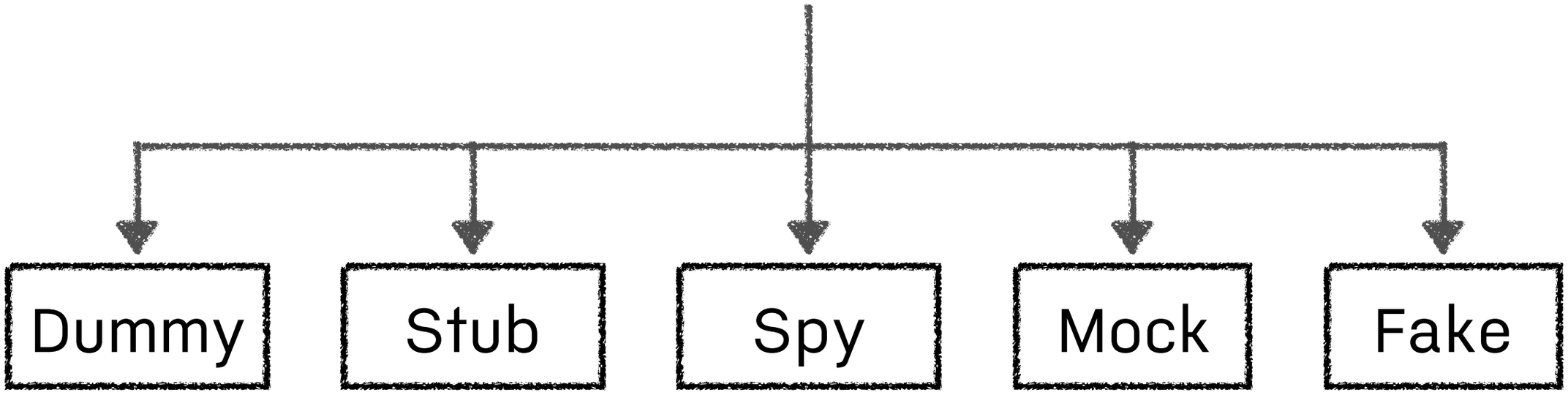
Isolate Dependency



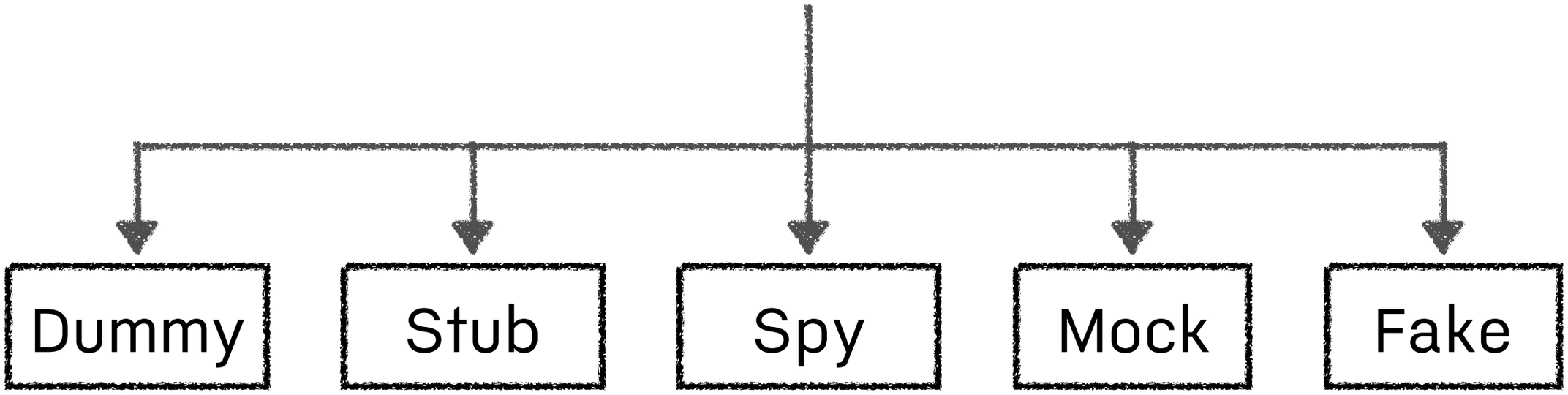
Isolate Dependency



Test Double

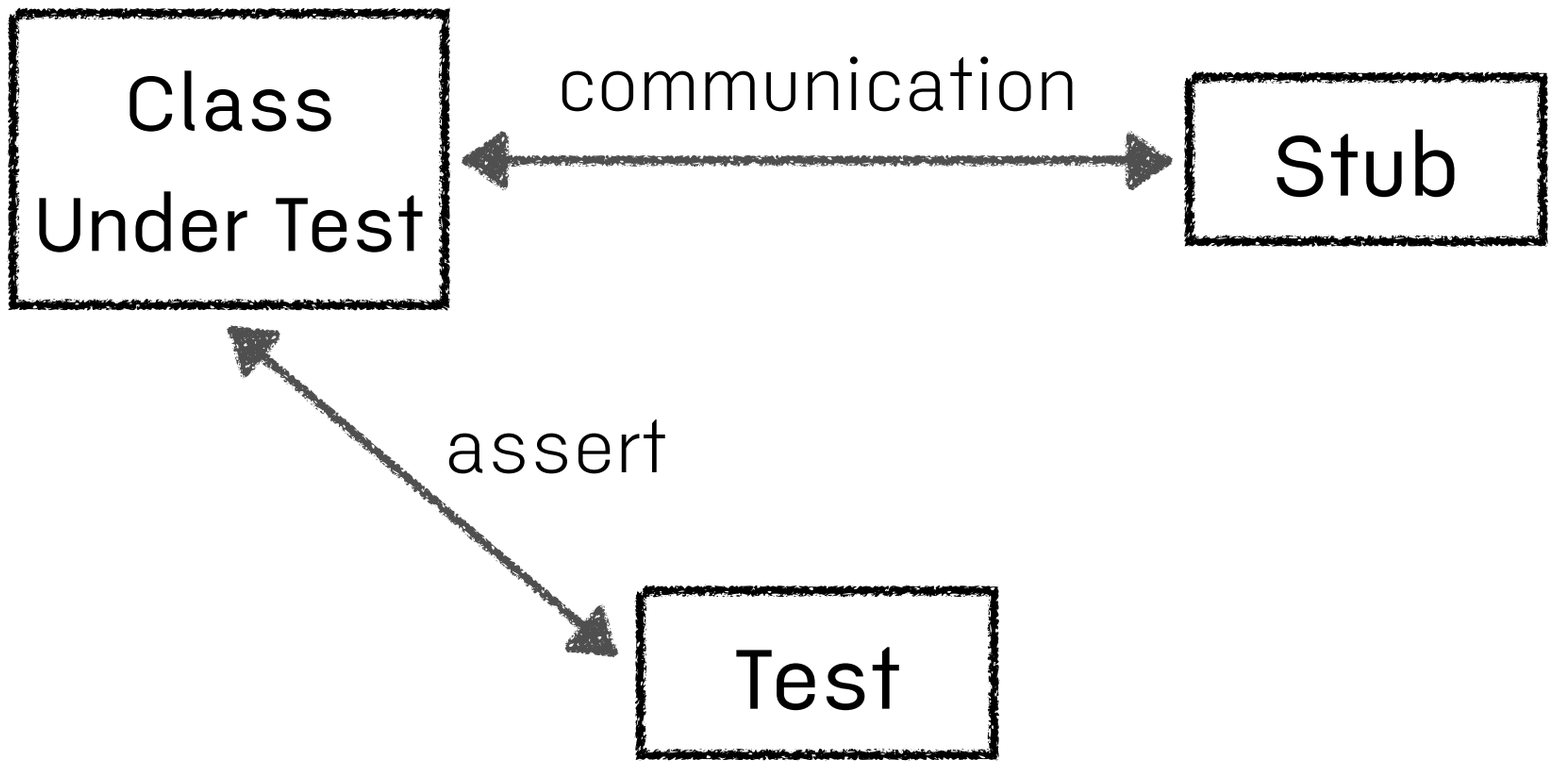


Test Double



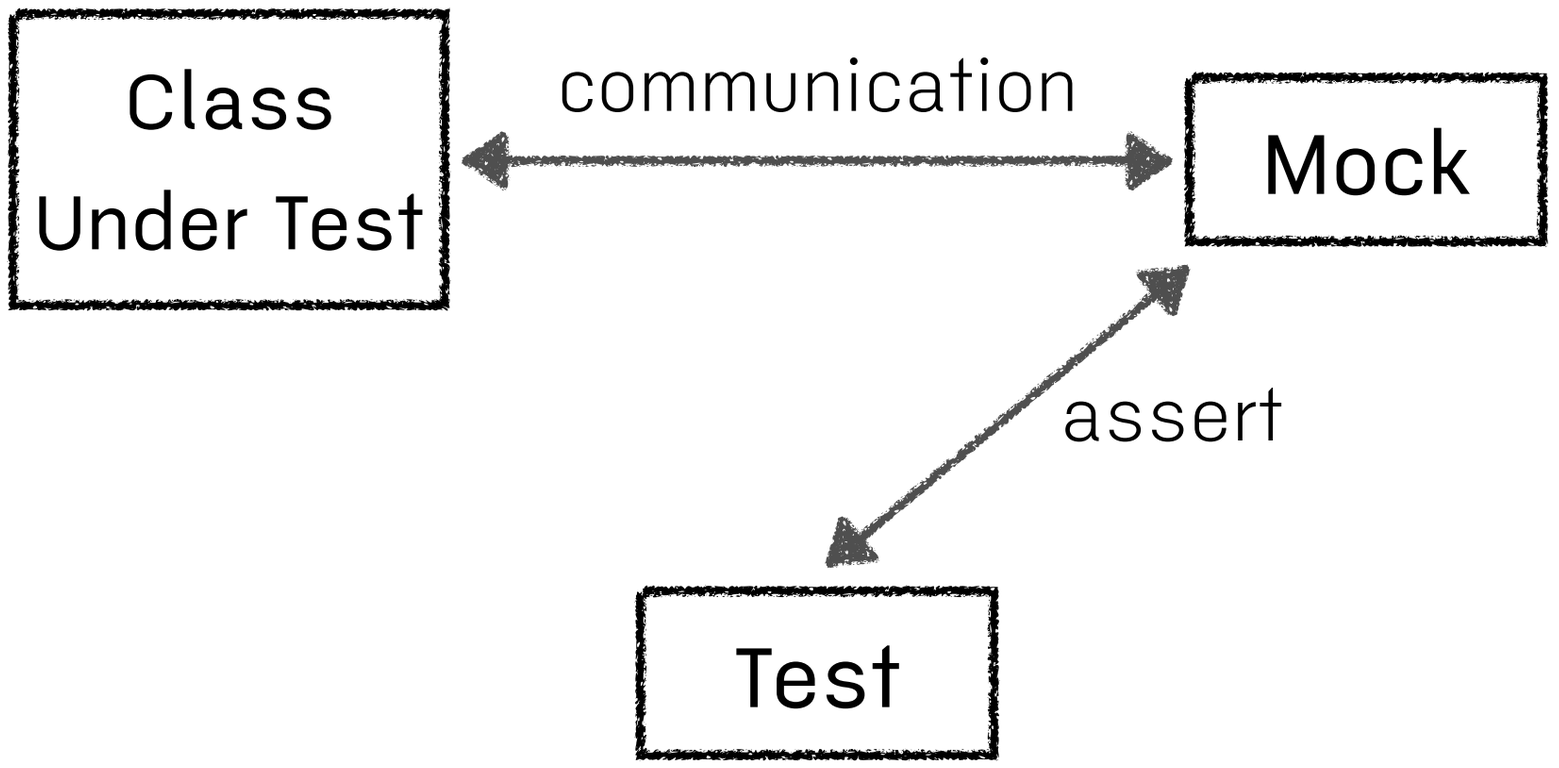
Configuration & **Hard code**

Stub



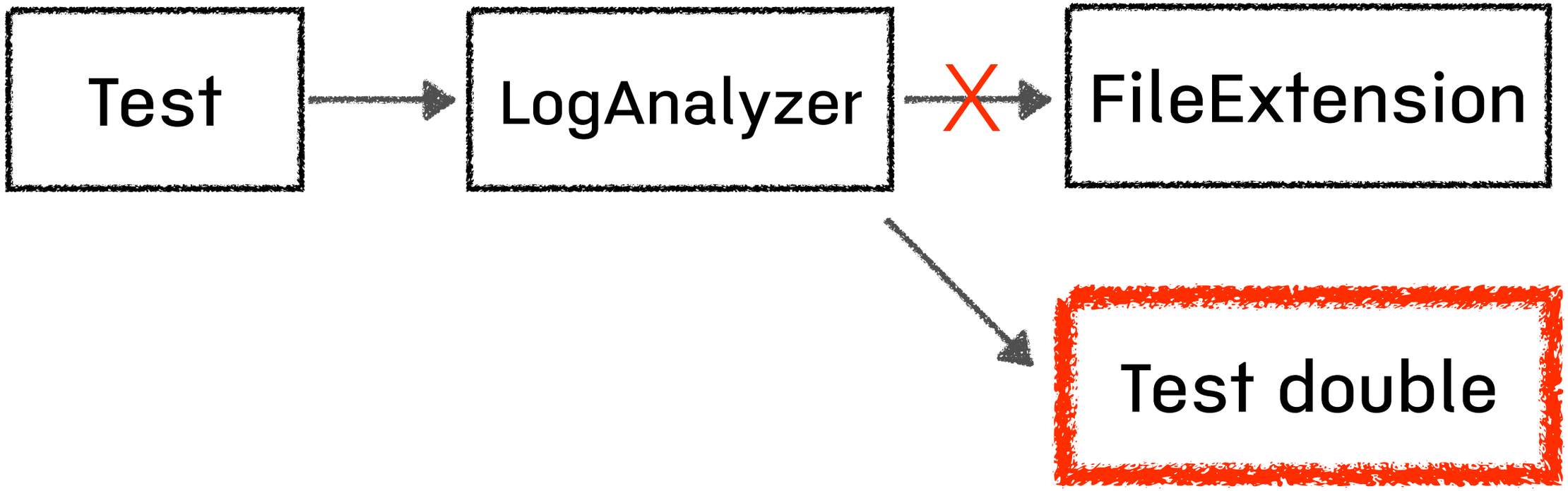
Make sure the test run smoothly

Mock object



To verify that the test pass

How to use Test double ?



Demo & Discuss

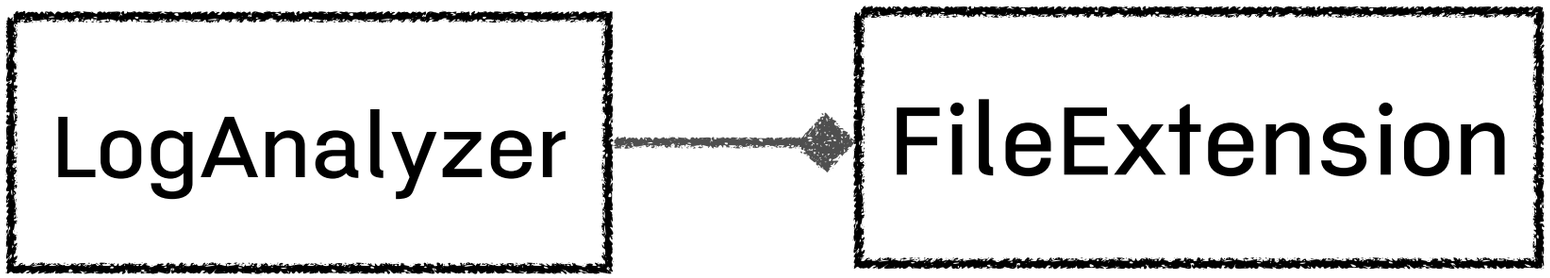
Break dependency with File System

Discuss

How to solve this problem ?

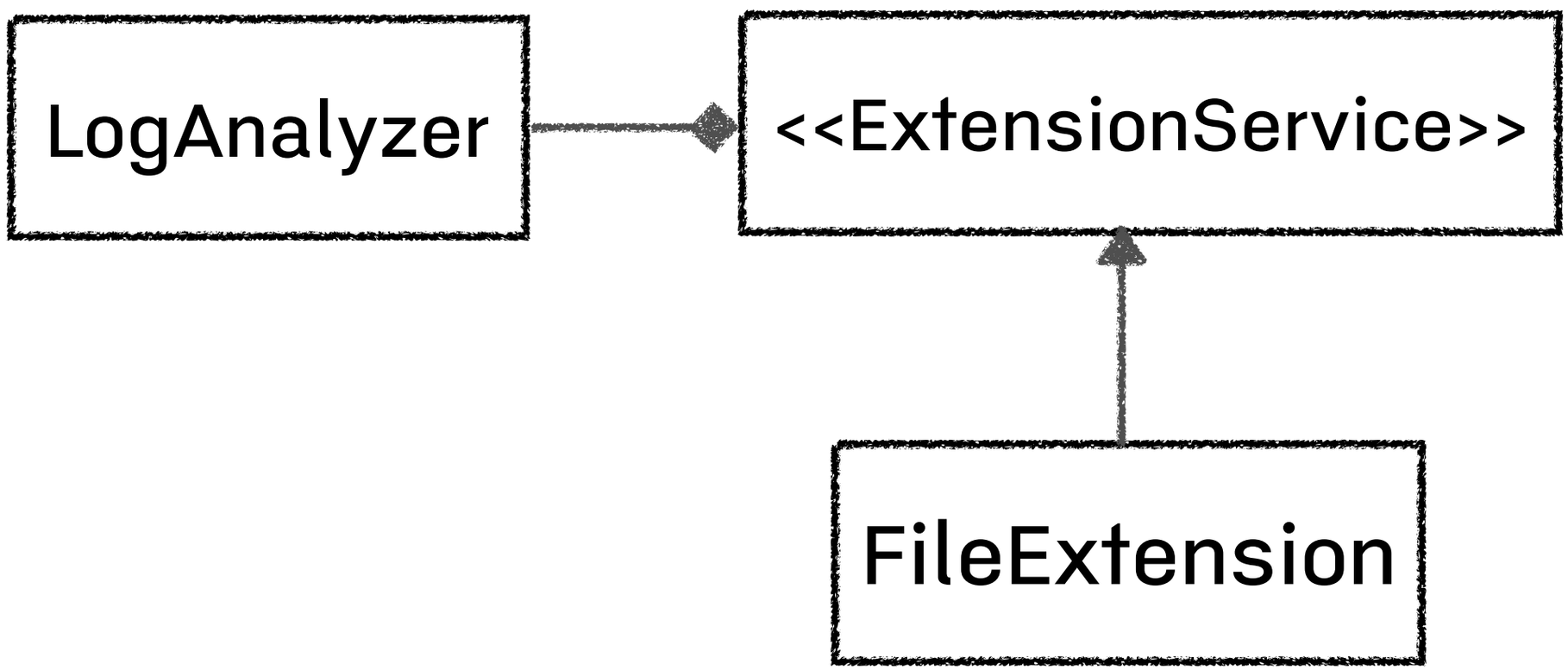
All about Abstraction

Tight coupling !!



All about Abstraction

Loose coupling !!

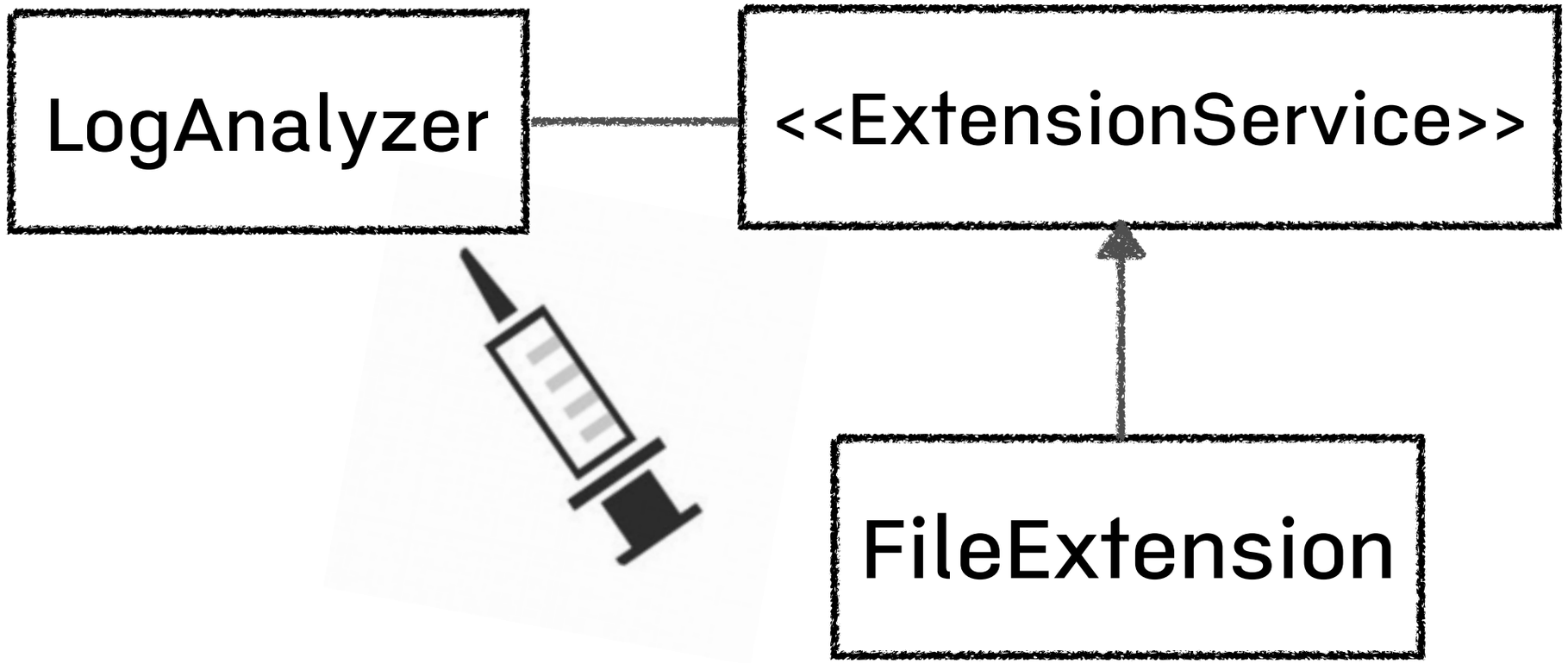


Demo & Discuss

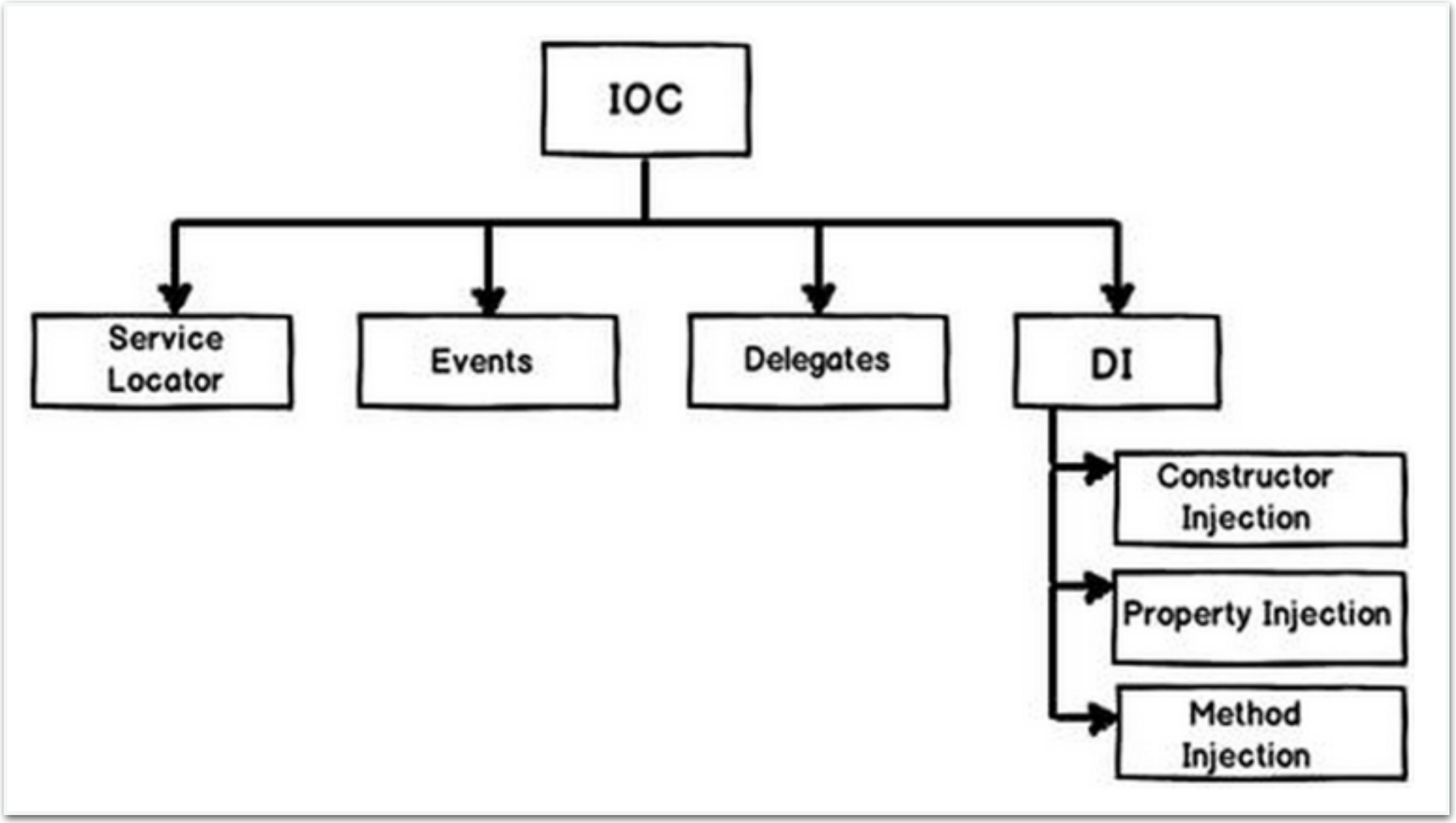
Dependency Inversion Principle (DIP)

How to use FileExtension ?

Loose coupling



Dependency Injection



Dependency Injection

Constructor Injection

Property Injection

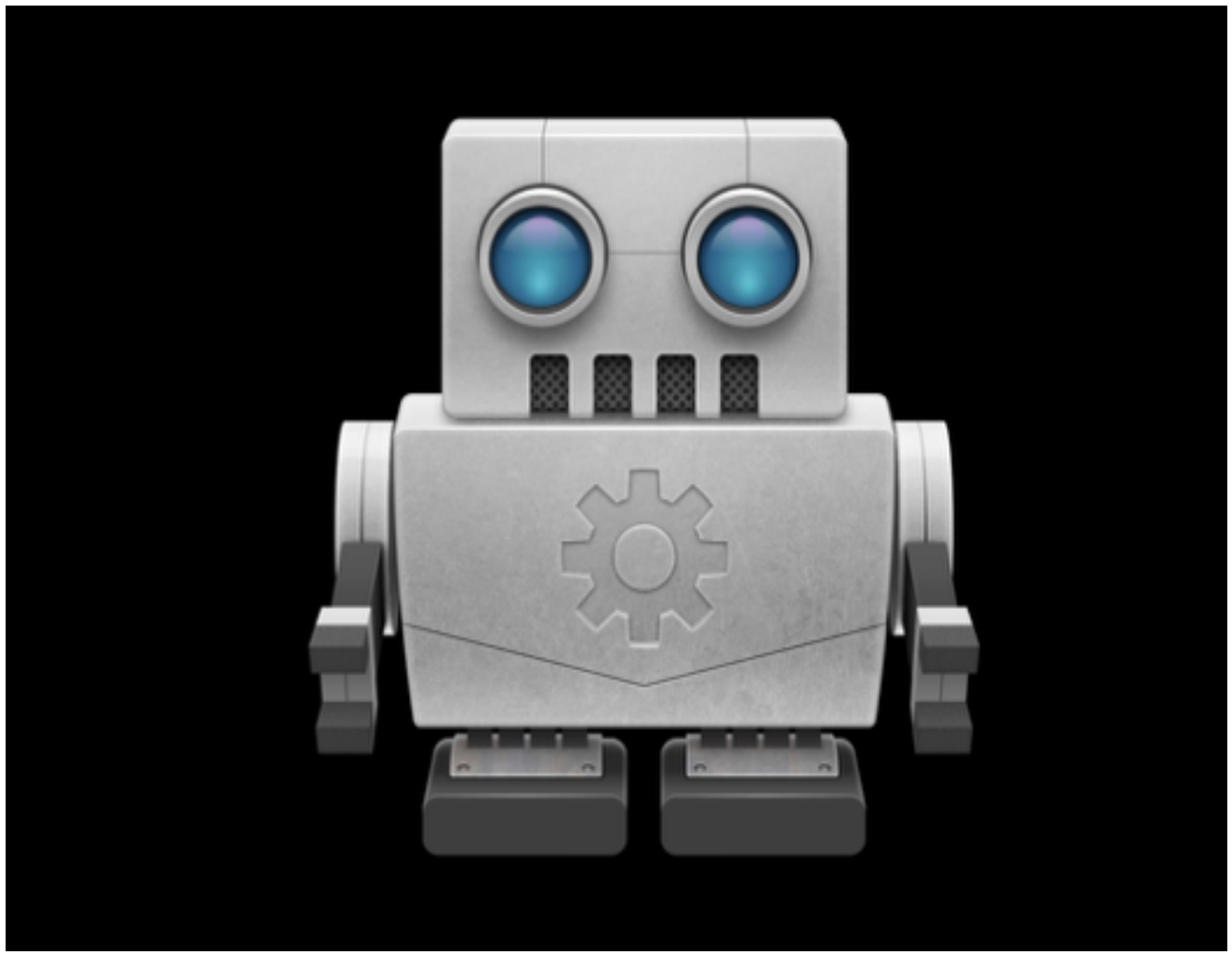
Method Injection

Demo & Discuss

Dependency Injection of LogAnalyzer

Getting start with UI Test

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่



บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Required

iOS 9+ , OS X 10.11+

iOS device must enable for development

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

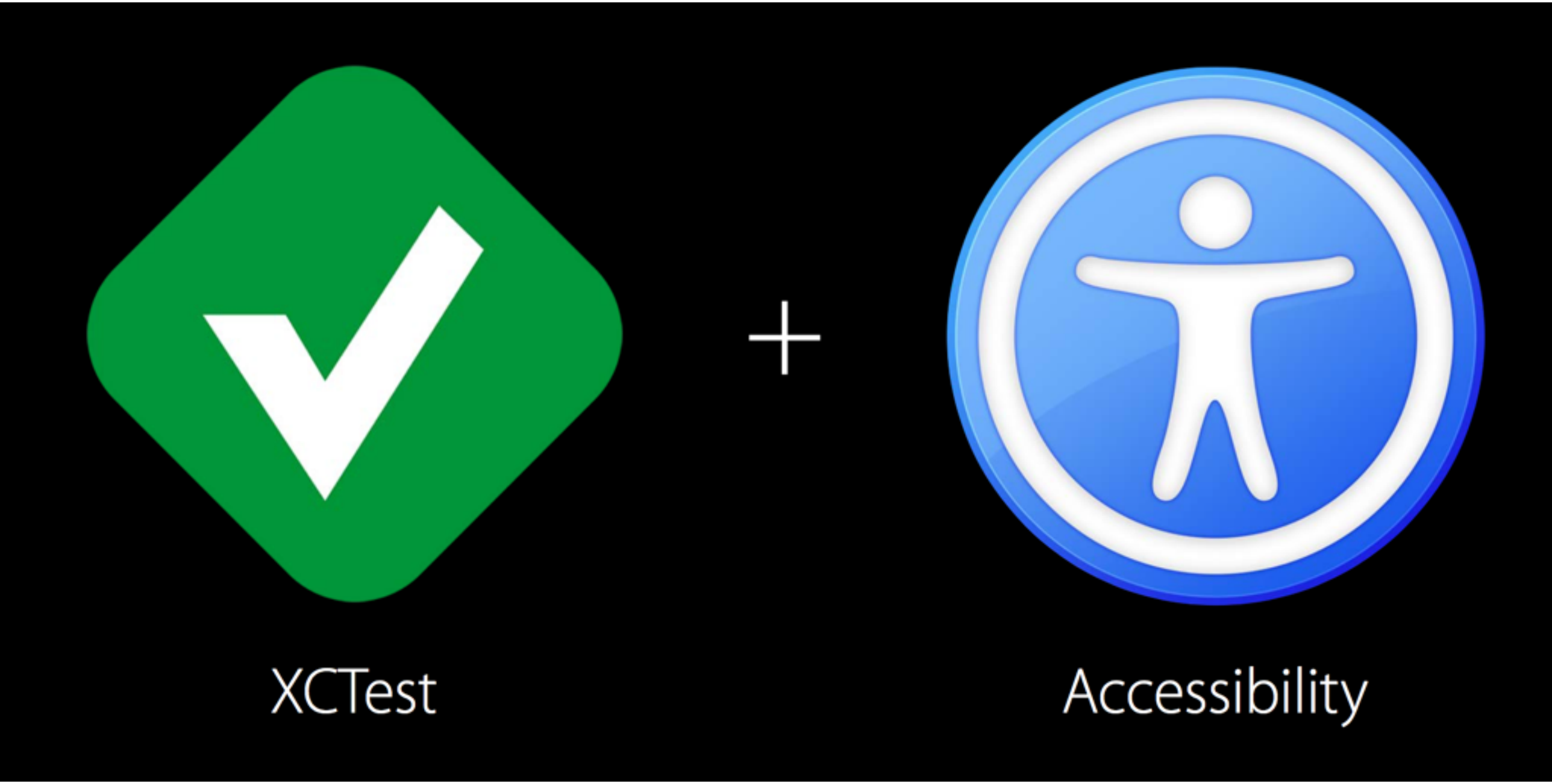
Main components

UI Testing
UI Recording
Accessibility

UI Testing

Find and interact with element in UI
Validate UI property
Validate UI state

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่



บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Let’s start **UI Testing**

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Product Name: WorkingWithUI

Organization Name: ___UP1___

Organization Identifier: up1

Bundle Identifier: up1.WorkingWithUI

Language: Swift

Devices: Universal

- [ ] Use Core Data
- [ ] Include Unit Tests
- [x] Include UI Tests

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

```
import XCTest

class WorkingWithUIUITests: XCTestCase {

    override func setUp() {
        super.setUp()
        continueAfterFailure = false
        XCUIApplication().launch()
    }

    override func tearDown() {
        super.tearDown()
    }

    func testExample() {
    }

}
```

UI Recording

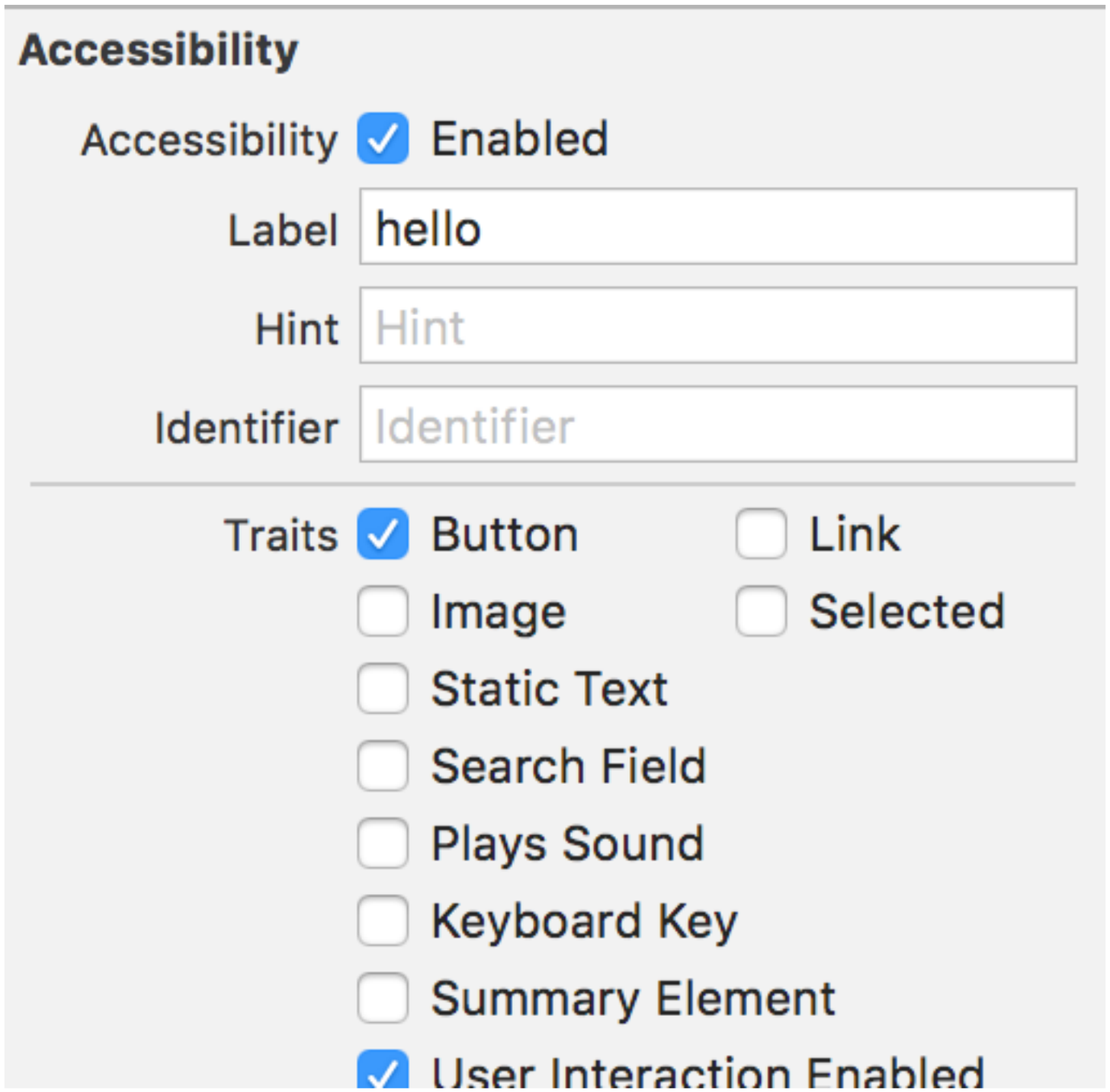
Interact with application
Generate the code

```
15
16  func testExample() {
17
18  }
19
20  }
21
22
```

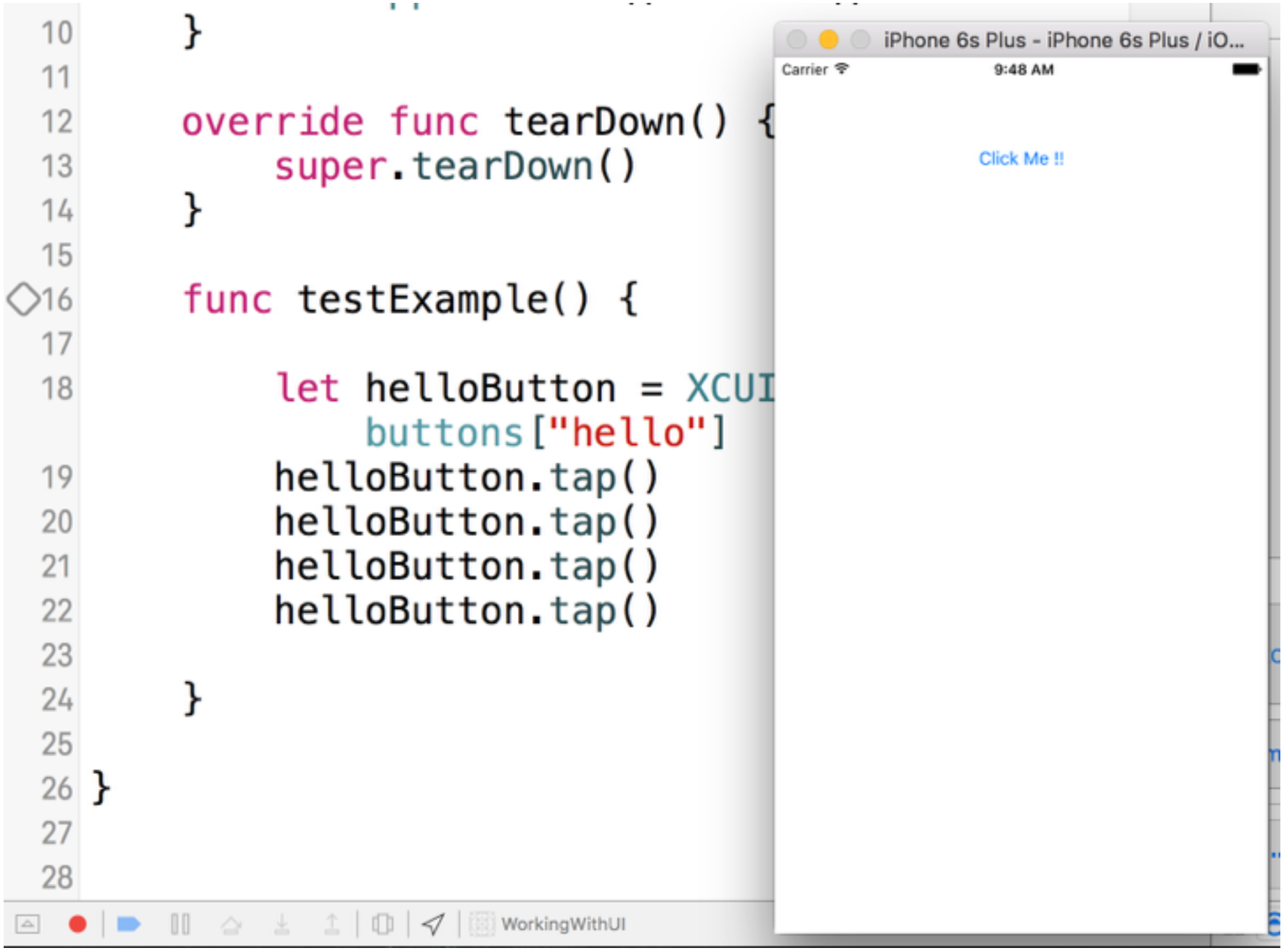
1. Put cursor

2. Click to start recording

Enable accessibility

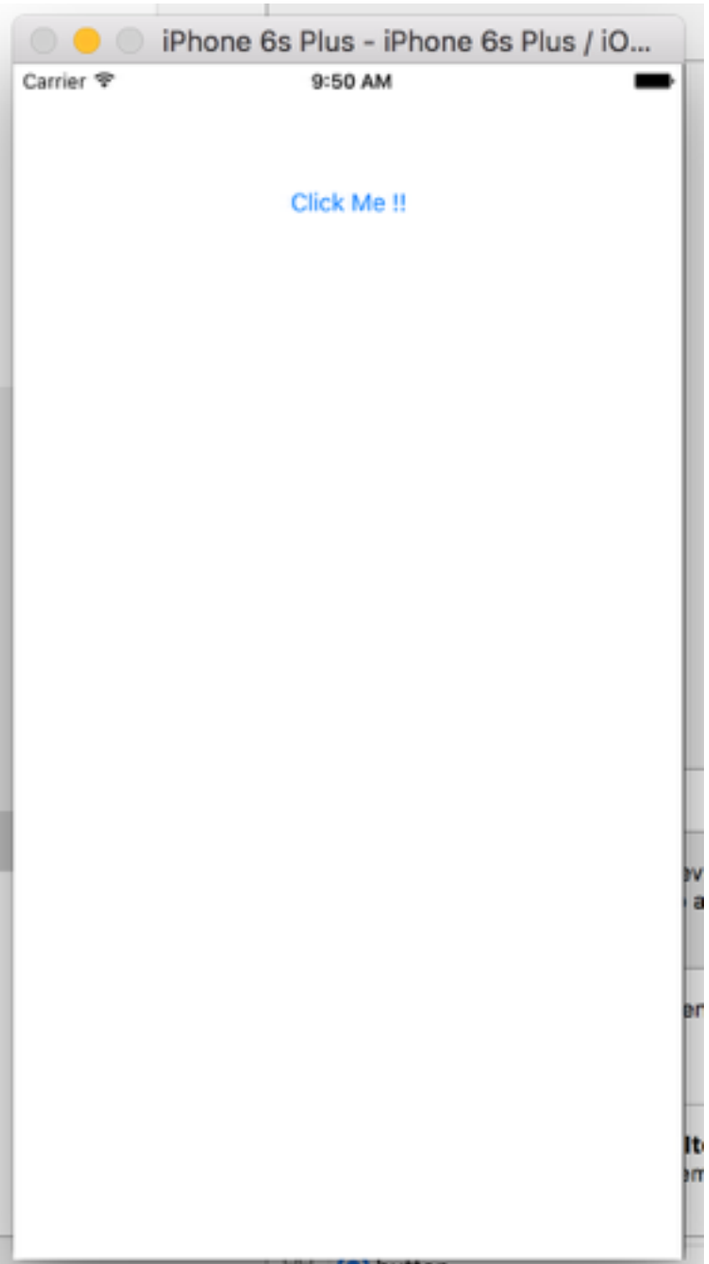


Enable accessibility



Disable accessibility

```
11
12      override func tearDown() {
13          super.tearDown()
14      }
15
16      func testExample() {
17
18          let clickMeStaticText =
                XCUIApplication()
                .staticTexts["Click Me !!"]
19          clickMeStaticText.tap()
20          clickMeStaticText.tap()
21          clickMeStaticText.tap()
22
23
24      }
25
26  }
27
28
```



Add assertion

```
func testExample() {

    let app = XCUIApplication()
    let helloButton = app.buttons["hello"]
    helloButton.tap()

    let counterStaticText = app.staticTexts["counter"]
    counterStaticText.tap()

    XCTAssertEqual("0", counterStaticText.label)

}
```

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

UI Testing APIs

1. XCUIApplication
2. XCUIElement
3. XCUIElementQuery

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Element

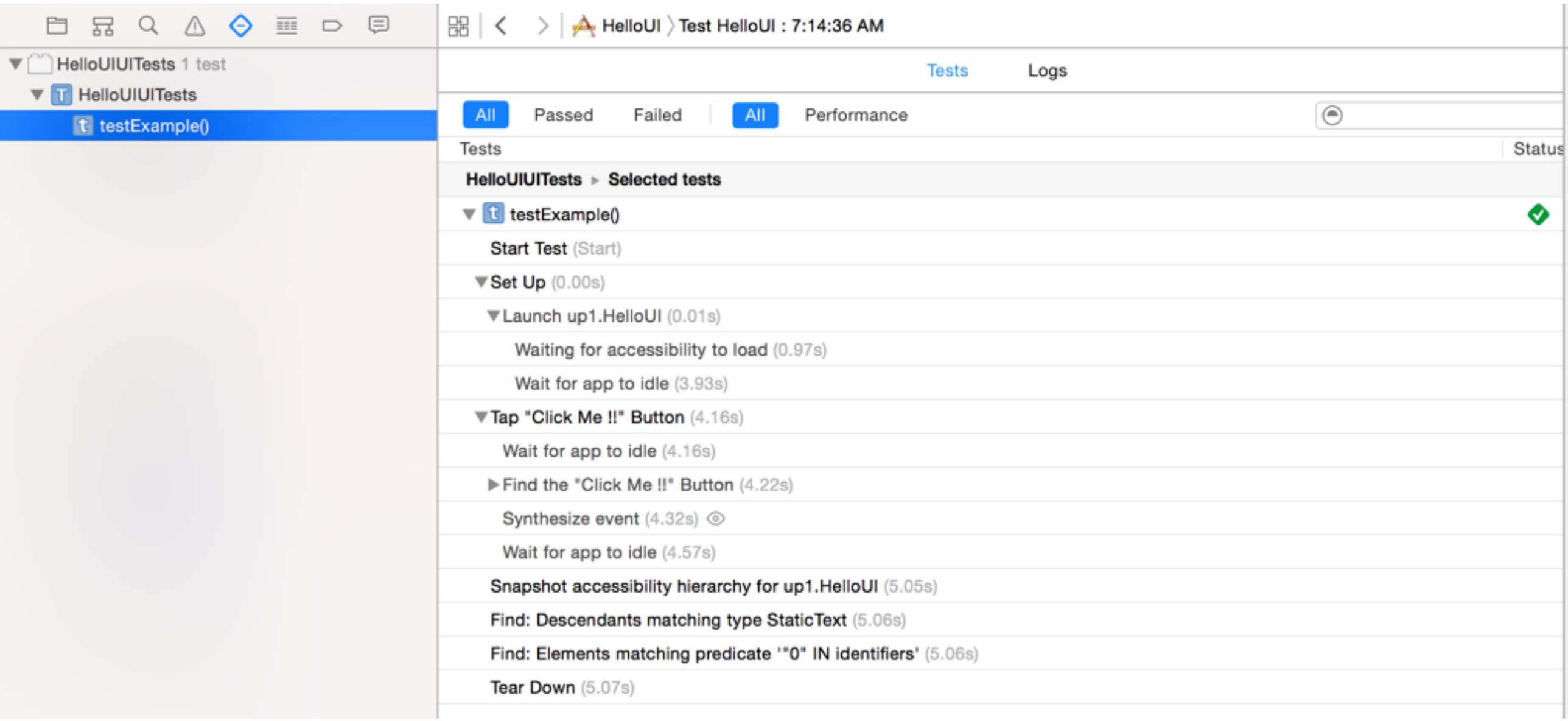
Application

View

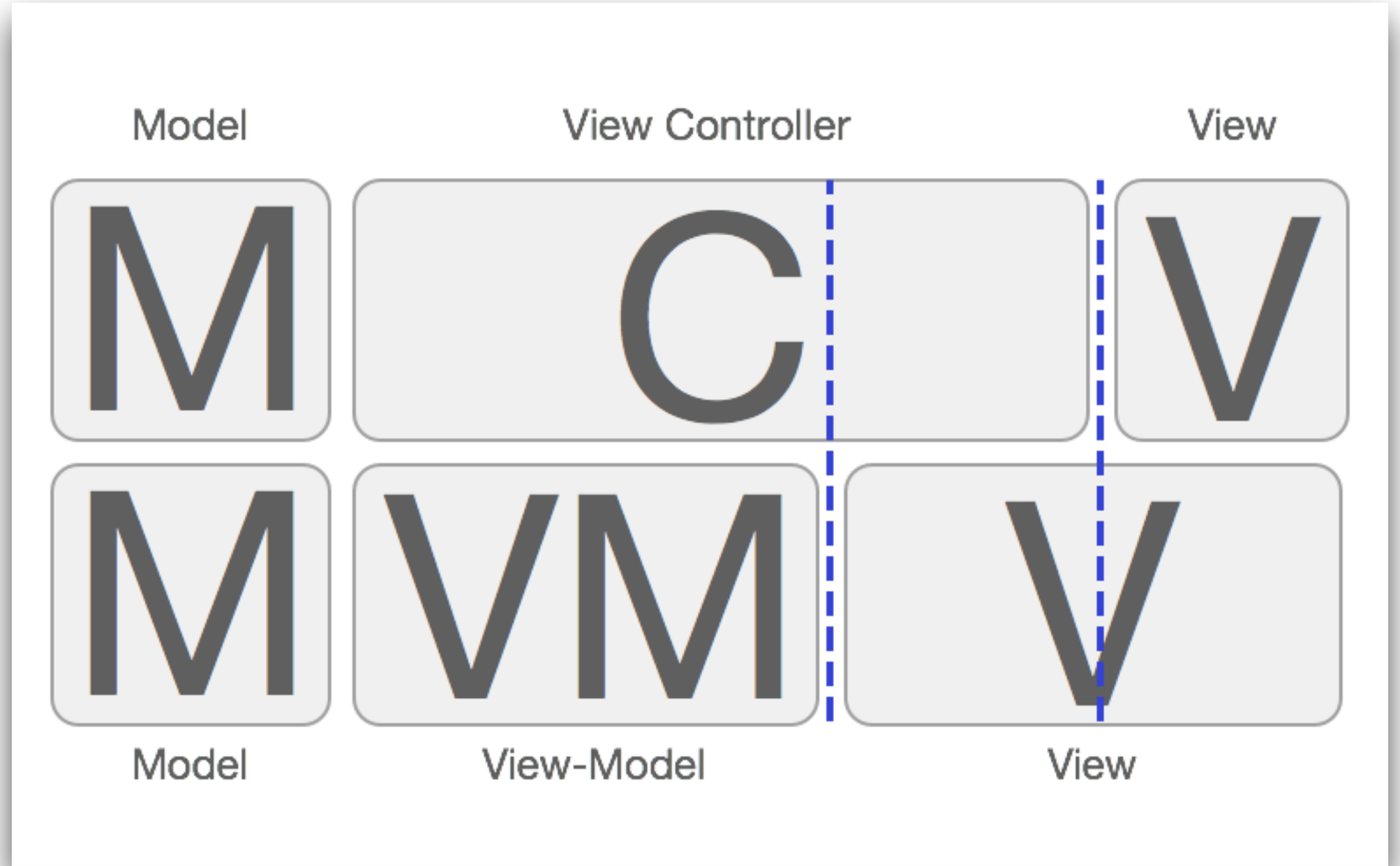
“Click Me” button

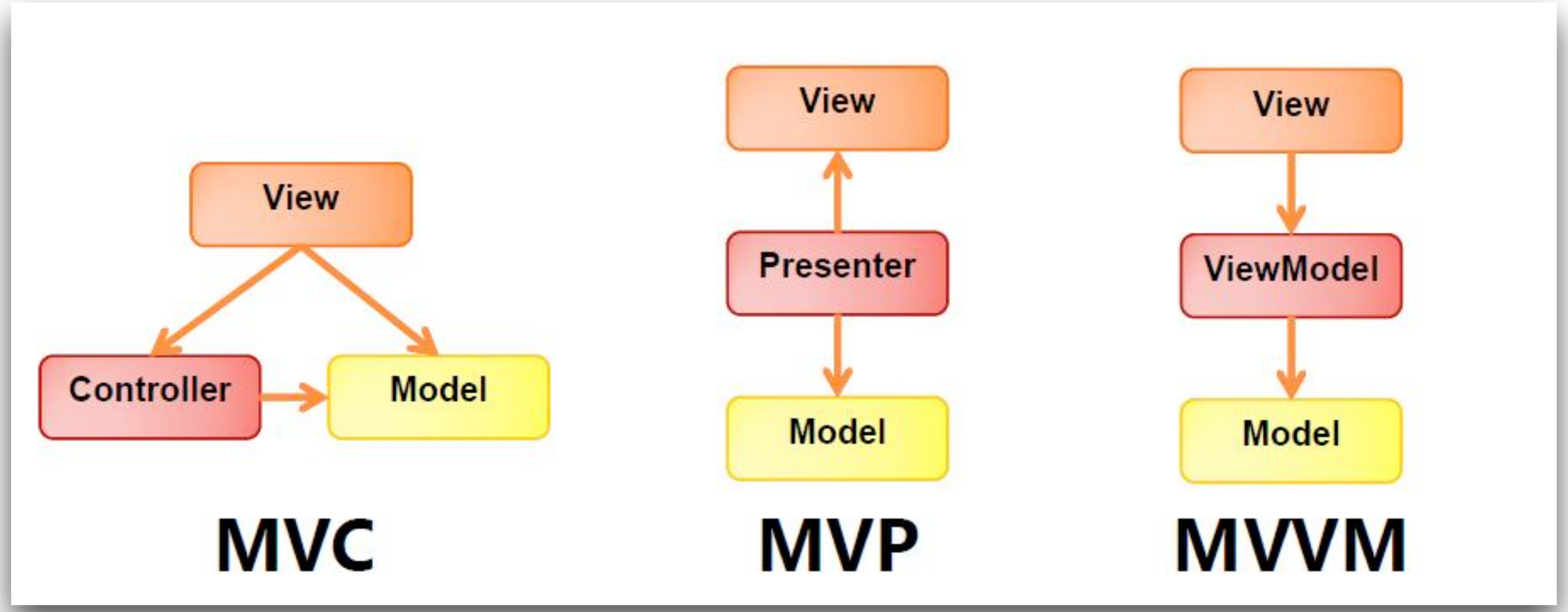
“1” static text

Test report


HelloUI 〉Test HelloUI : 7:14:36 AM
HelloUIUITests 1 test
HelloUIUITests
testExample()
Tests
Logs
All
Passed
Failed
All
Performance
Tests
Status
HelloUIUITests ▸ Selected tests
testExample()
Start Test (Start)
Set Up (0.00s)
Launch up1.HelloUI (0.01s)
Waiting for accessibility to load (0.97s)
Wait for app to idle (3.93s)
Tap "Click Me !!" Button (4.16s)
Wait for app to idle (4.16s)
Find the "Click Me !!" Button (4.22s)
Synthesize event (4.32s)
Wait for app to idle (4.57s)
Snapshot accessibility hierarchy for up1.HelloUI (5.05s)
Find: Descendants matching type StaticText (5.06s)
Find: Elements matching predicate '"0" IN identifiers' (5.06s)
Tear Down (5.07s)

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่


Model
View Controller
View
M
C
V
M
VM
V
Model
View-Model
View



บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

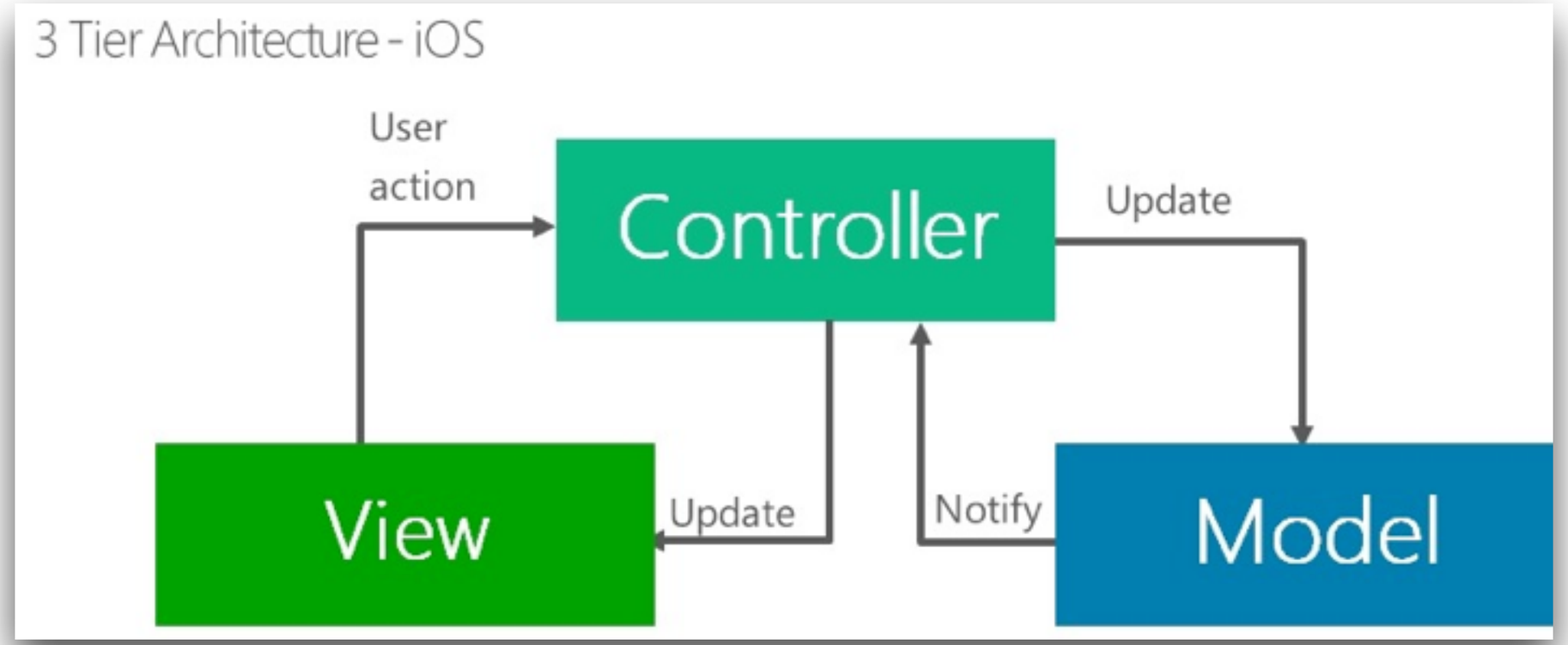
Model View ViewModel

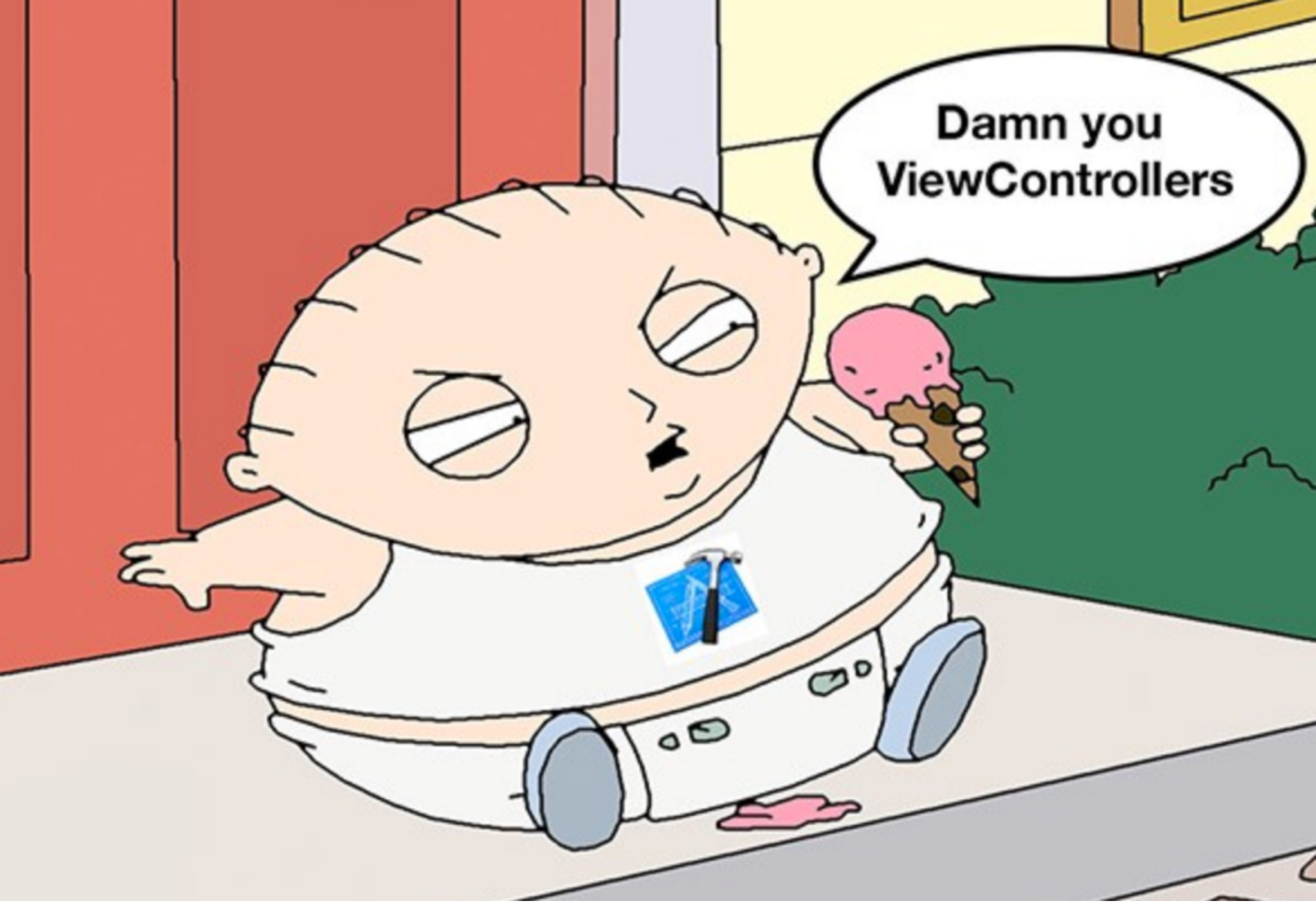
Data

Presentation

Behavior

What problem of MVC ?

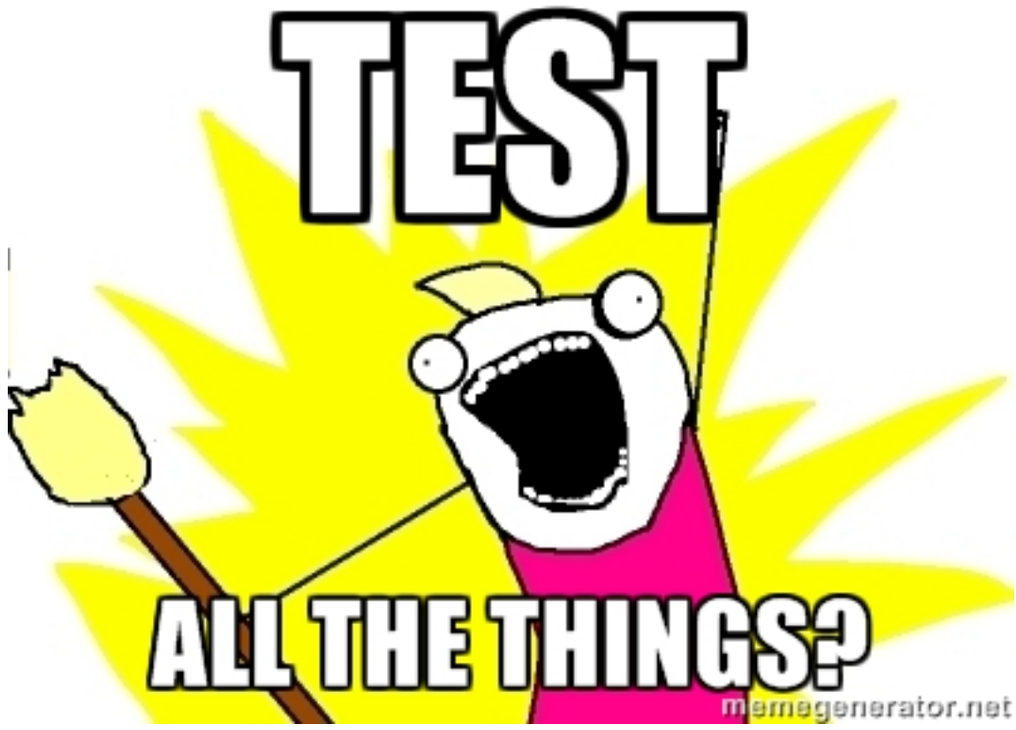



Damn you
ViewControllers

Massive View Controller

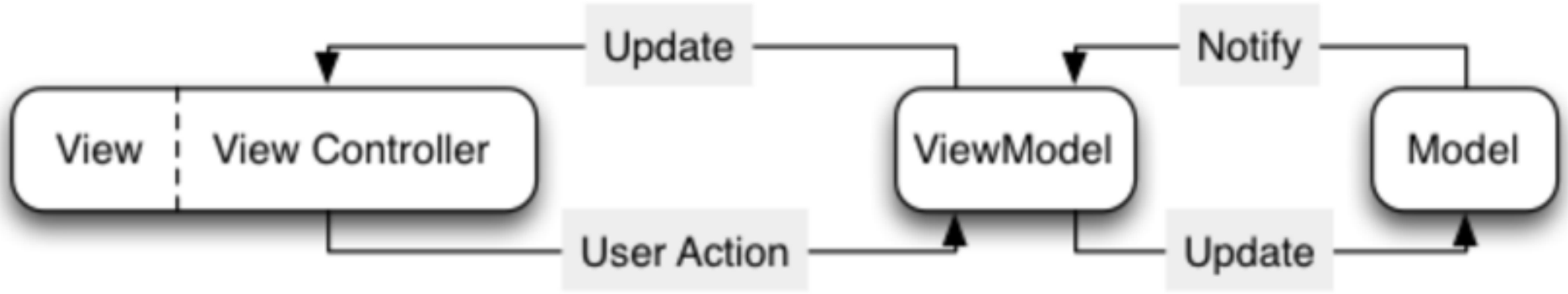
บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Hard to unit test



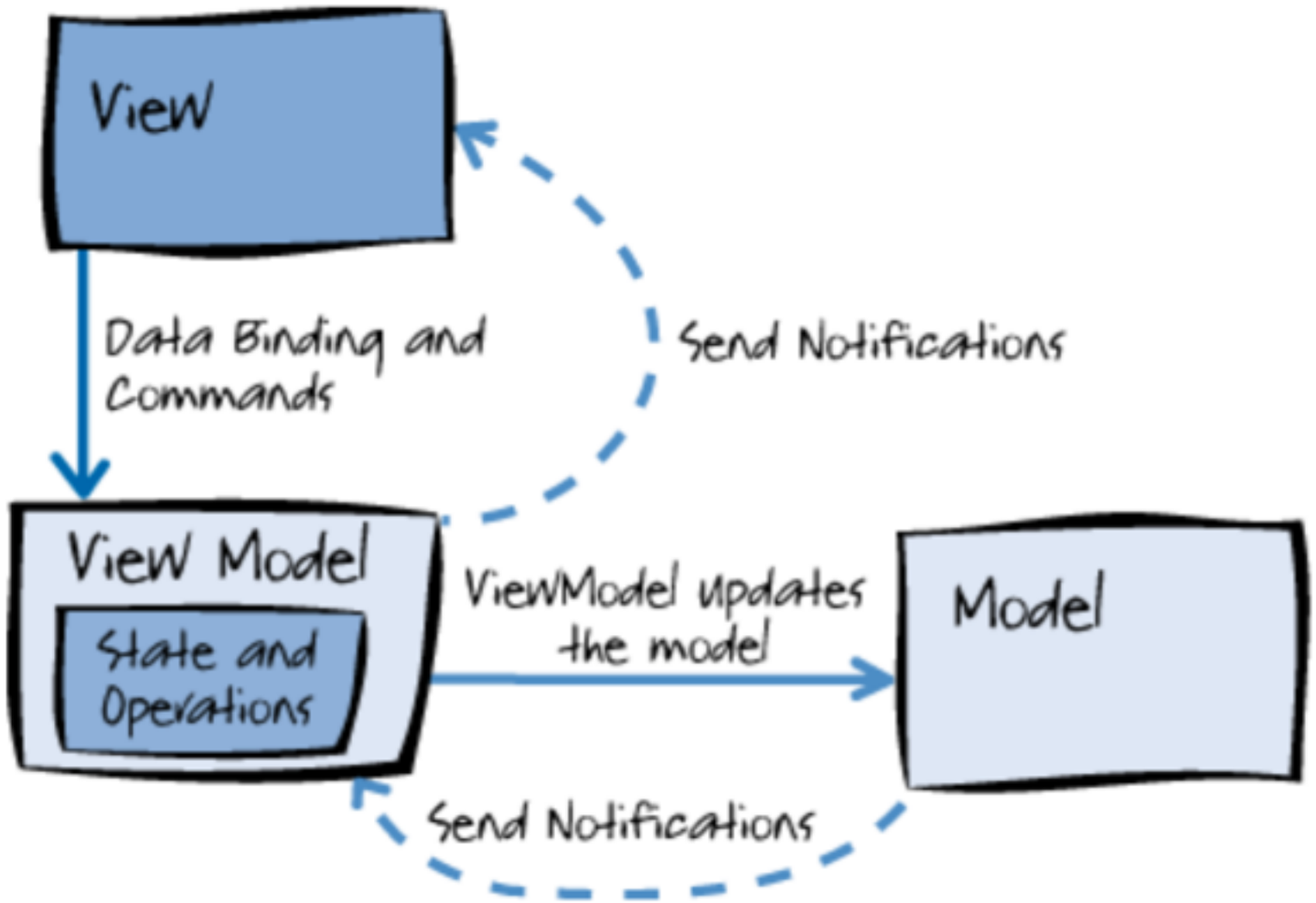
Hard to reuse code

Platform/framework specific



Model View ViewModel

Data
Presentation
Behavior



บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Model View ViewModel

Data layer

Entity
Persistence
Network

User Model

```
class User {
    var id: Int
    var firstname: String
    var lastname: String
    var address: String
    var birthday: NSDate?
    var pictureUrl: String?
}
```

Model View **ViewModel**

Behavior layer

Value transformation

View state

Input validation

Operation and action

User ViewModel

```
class UserViewModel {
    var name: String
    var birthday: String
    var picture: Picture

    enum Picture {
        case Network(NSURL)
        case Placeholder(String, Int)
    }
}
```

Model **View** ViewModel

Presentation layer

User interface
Data update
Events
Operation and action

User View

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Benefit ?

Separation of Concern

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Easy to unit test

Reuse code

Other ways ?

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

MVP

MVC

VIPER

MVVM

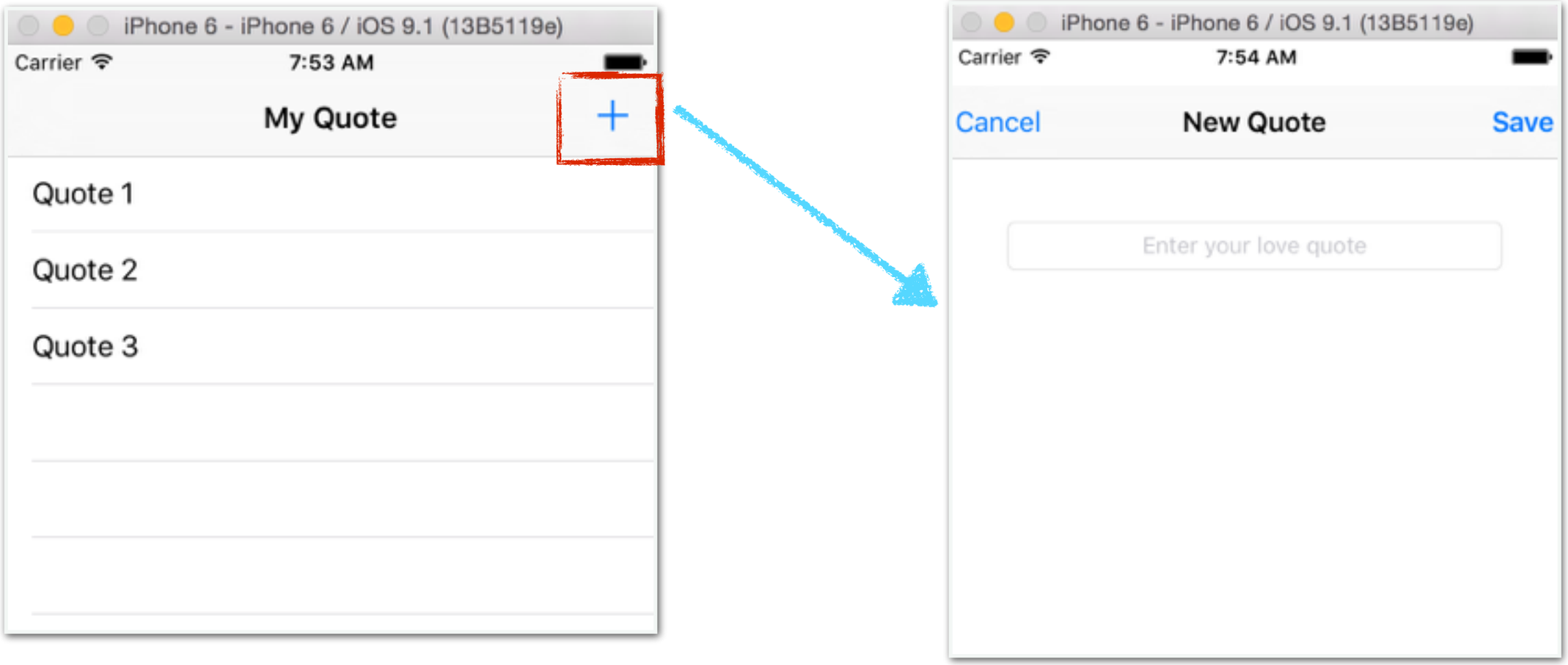
MVA ??

MVI

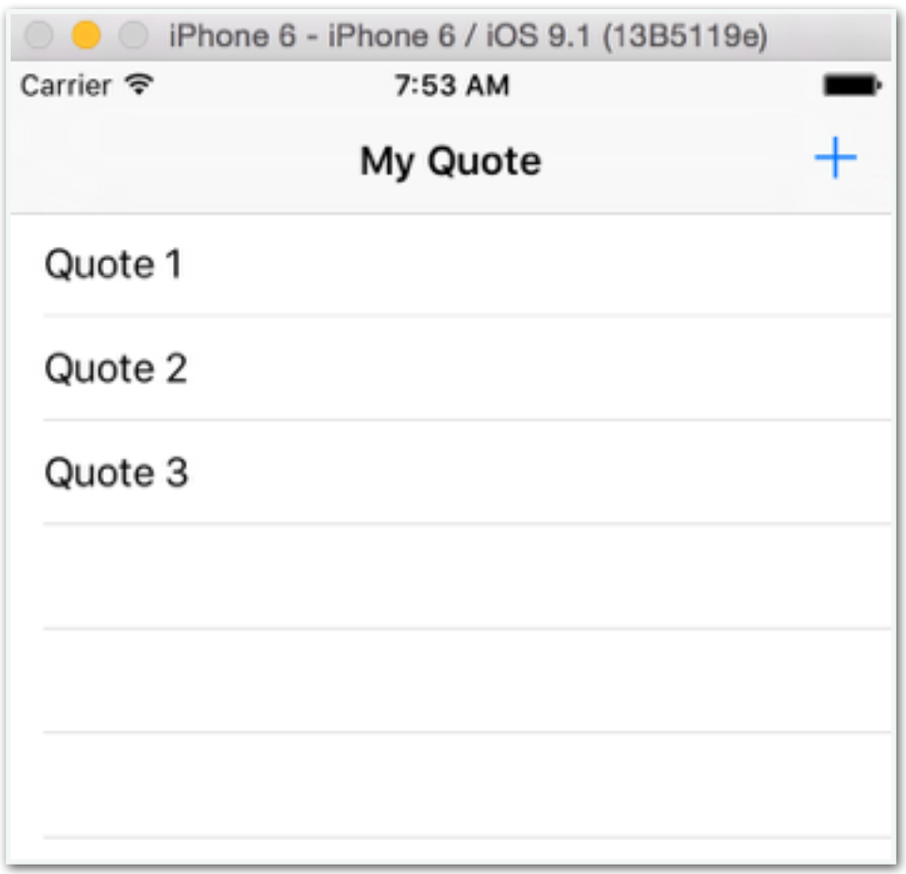
บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

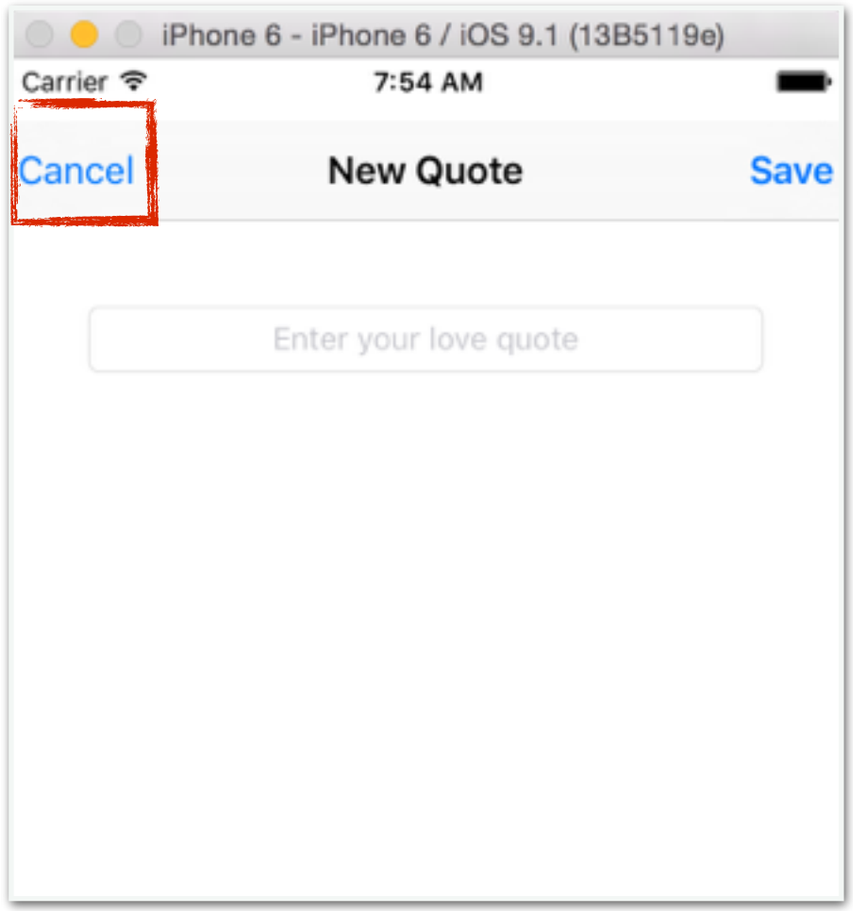
บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Develop Quote app

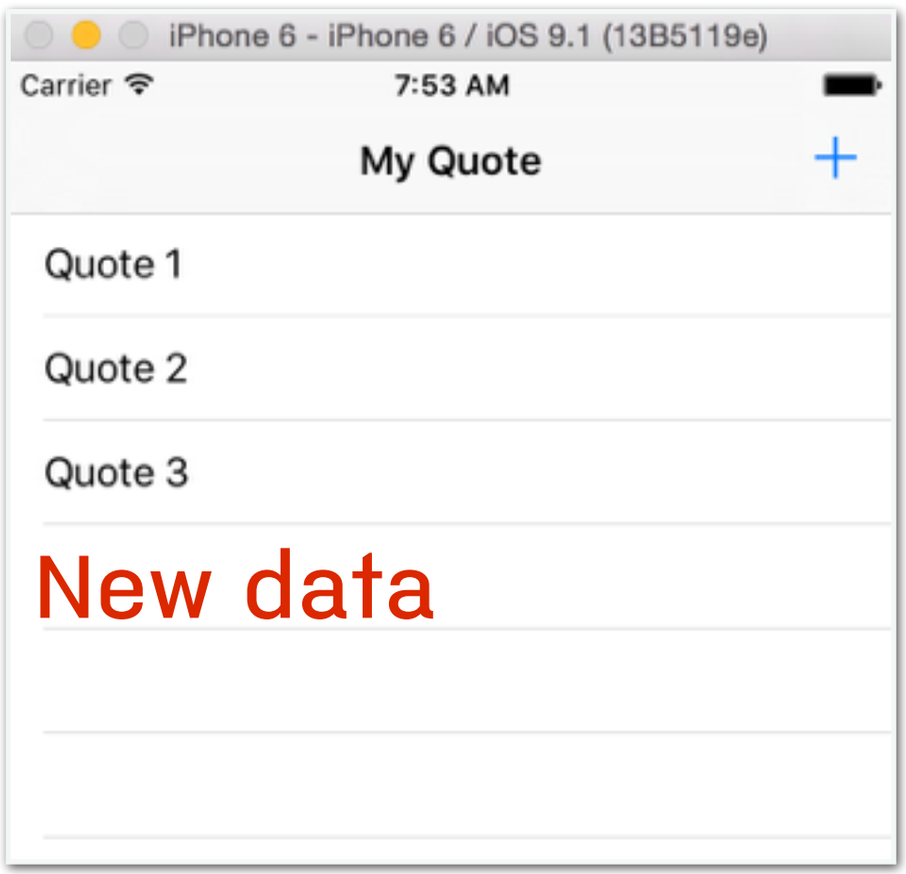


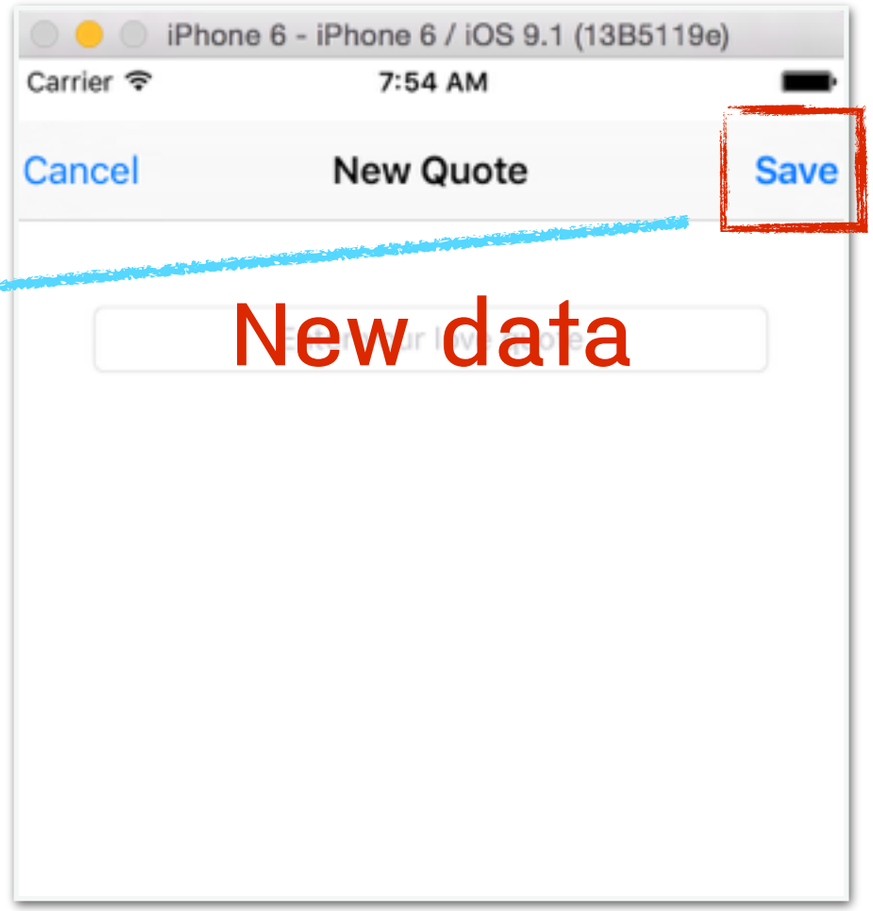
Develop Quote app



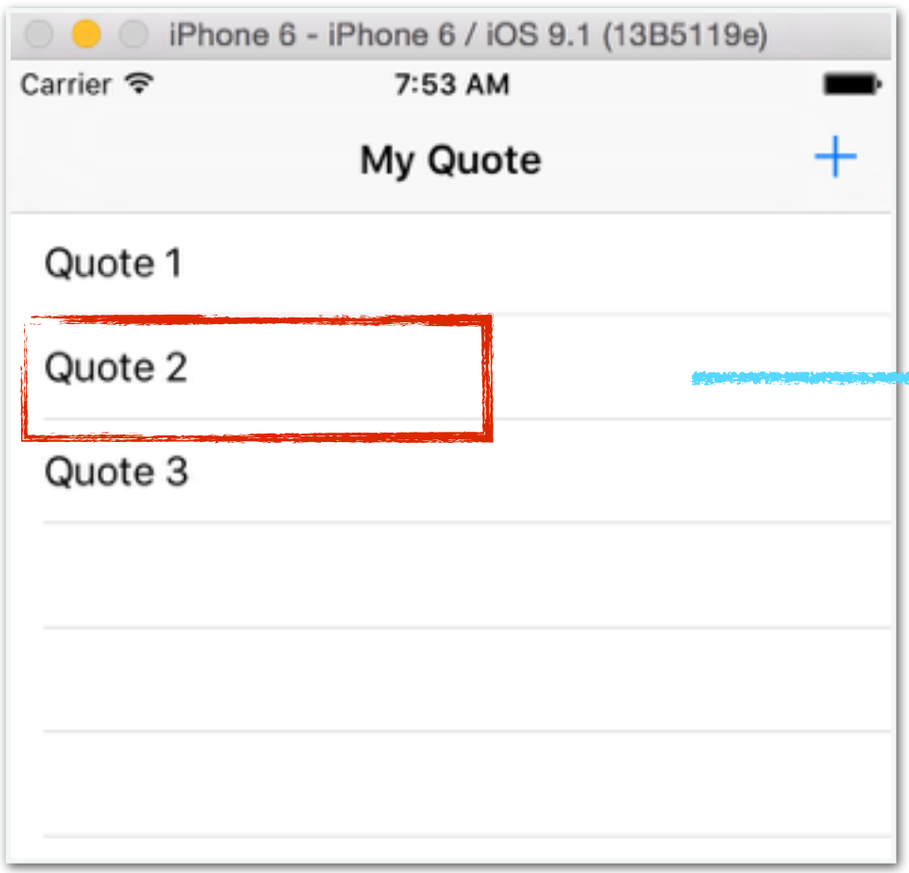


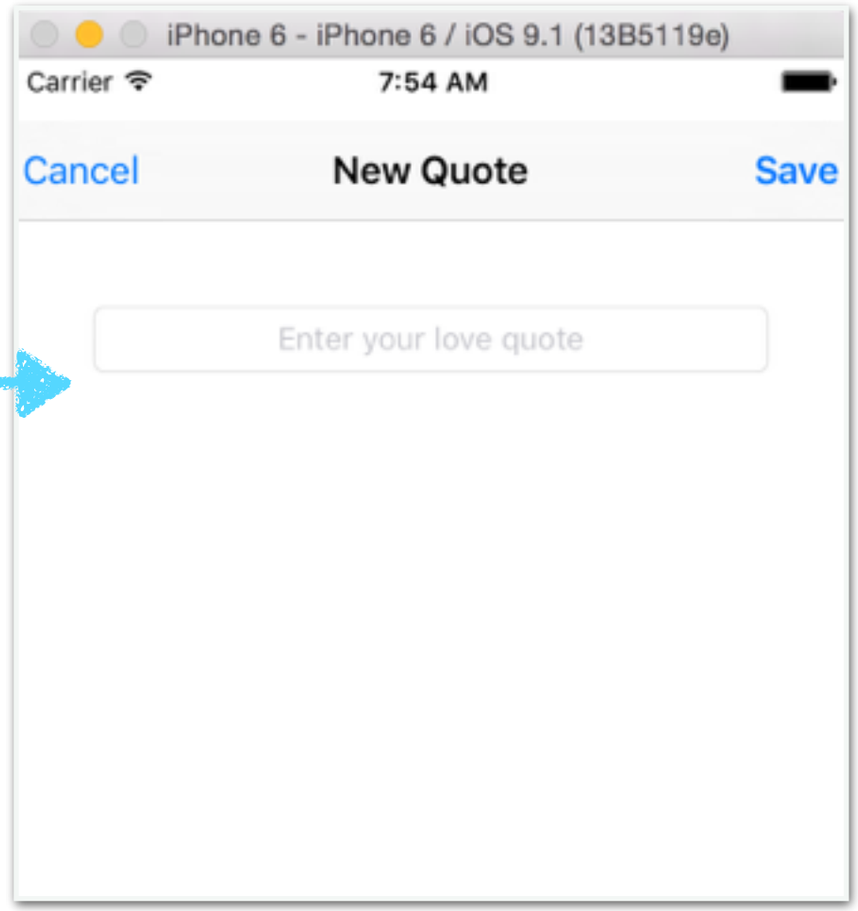
Develop Quote app



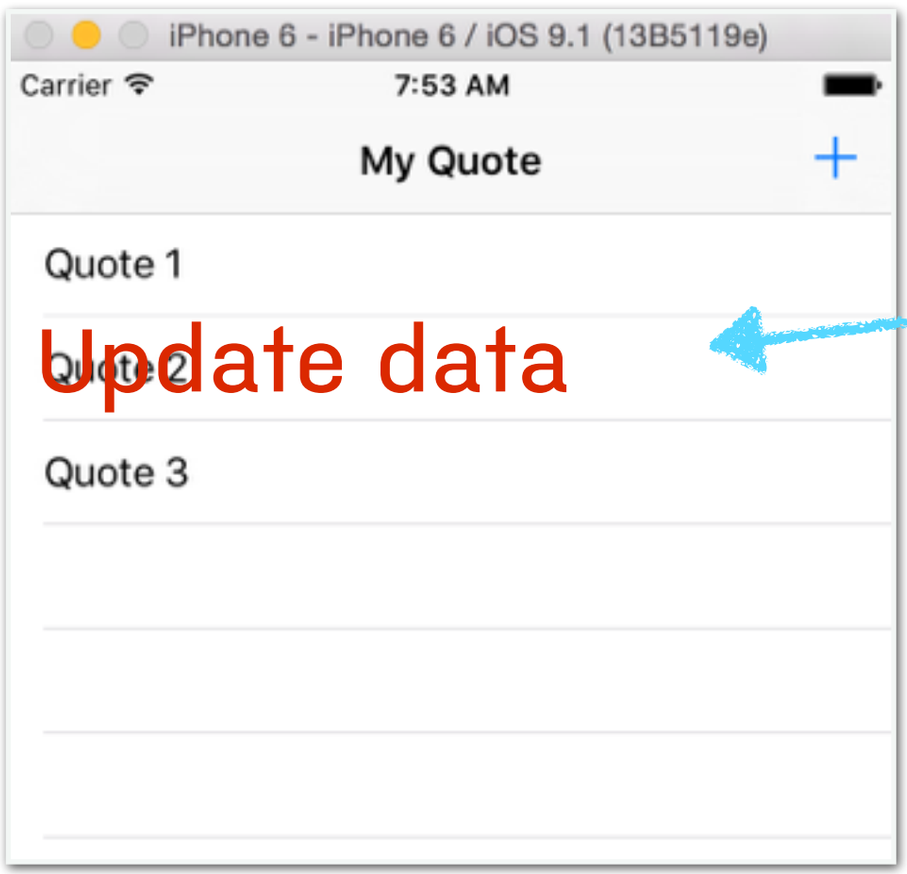


Update quote !!

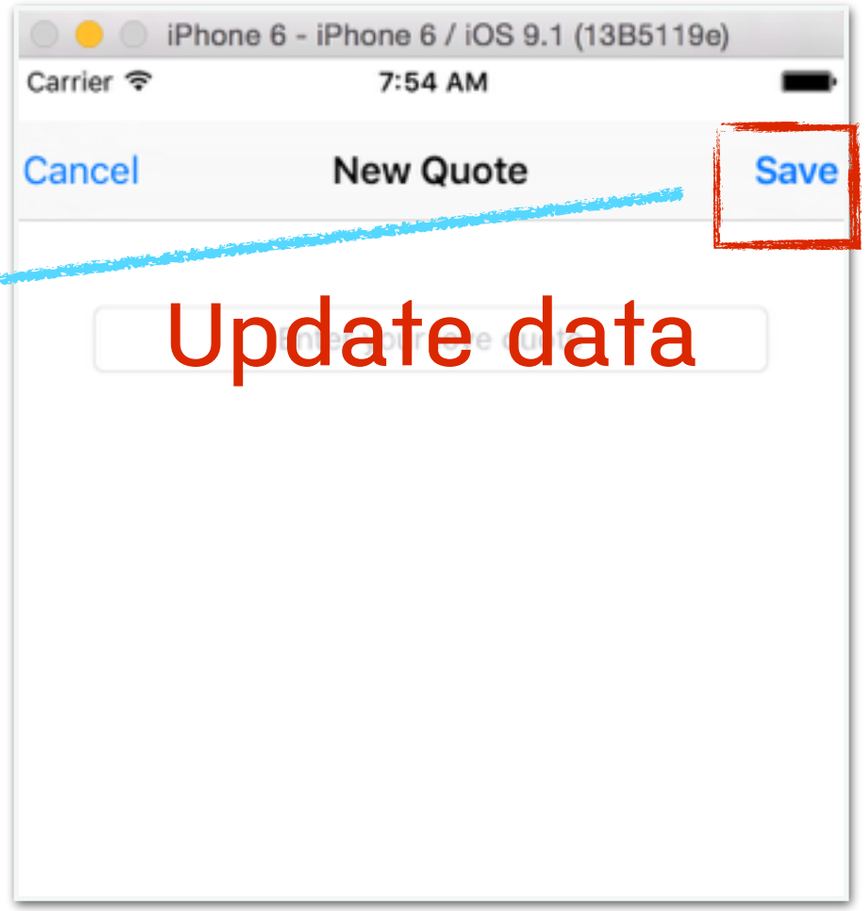




Update quote !!



Update data



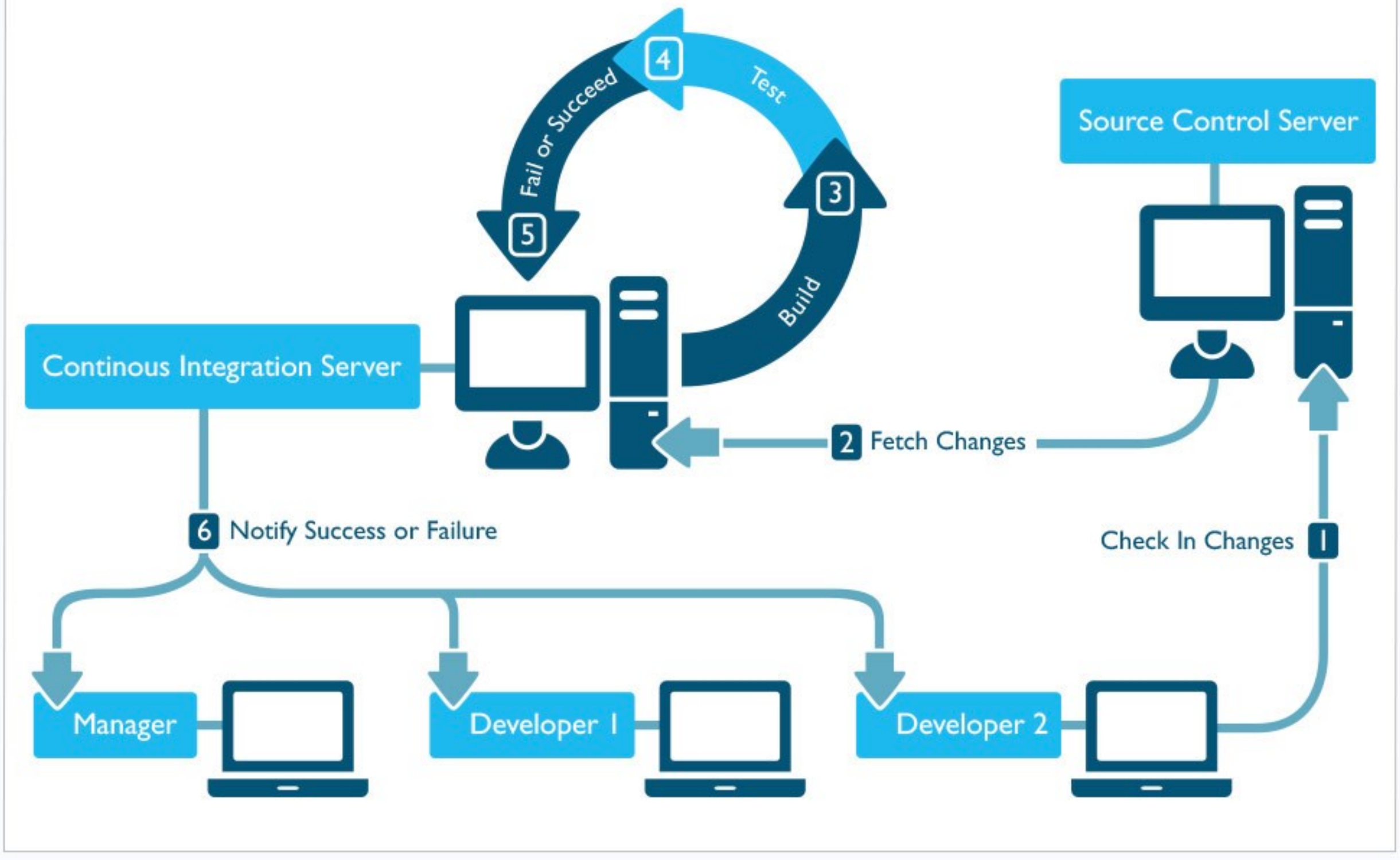
Update data

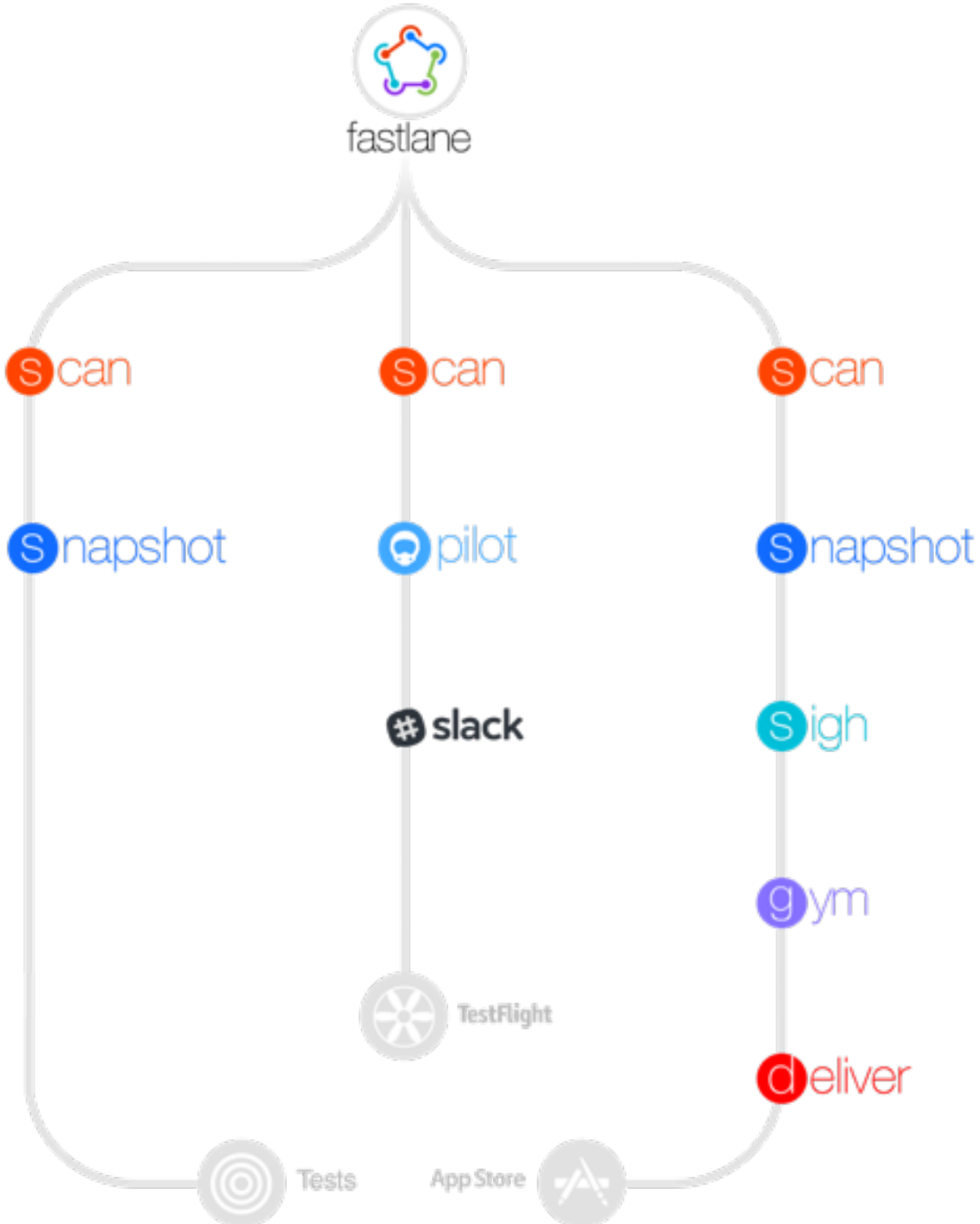
บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Continuous Integration
Continuous Delivery

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่


Source Code
Version Control System
Development Team
Source Code Build
Publish Reports
Staric Code Analysis
Run Automates Functional Tests
Run Automated Unit Tests
Provision and Deploy to Test Environment
Code Coverage Analysis
Set-up Test 'Fixtures'
Built Artifact
Continuous Integration
1
2
3
4
5
6
7
8
9
10
11
12


Source Control Server
Continous Integration Server
Test
Build
Fail or Succeed
4
3
5
2 Fetch Changes
6 Notify Success or Failure
Check In Changes
1
Manager
Developer 1
Developer 2



https://fastlane.tools/

Software requirement

- Ruby 2.0+
- Xcode CLI
- Paid Apple Developer Account

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

Installation

$sudo gem install fastlane

Let’s start

$fastlane init

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่

We ❤ Swift

บริษัท สยามชำนาญกิจ จำกัด และเพื่อนพ้องน้องพี่