

COCOA DYNAMIC FRAMEWORKS (ON IOS)

A yellow box with a silver latch and a white mug filled with brown chocolate soup topped with marshmallows are positioned behind the text.



COCOA FRAMEWORK

WHAT

A collection of resources

Compiled code + public headers

WHY

- Extensions
- Sharing code between applications
- Public distribution

WHY



DRY

DON'T REPEAT YOURSELF

DRY

IF THE CODE APPEARS MORE
THAN ONCE, IT BELONGS IN A
FRAMEWORK

DYNAMIC FRAMEWORKS

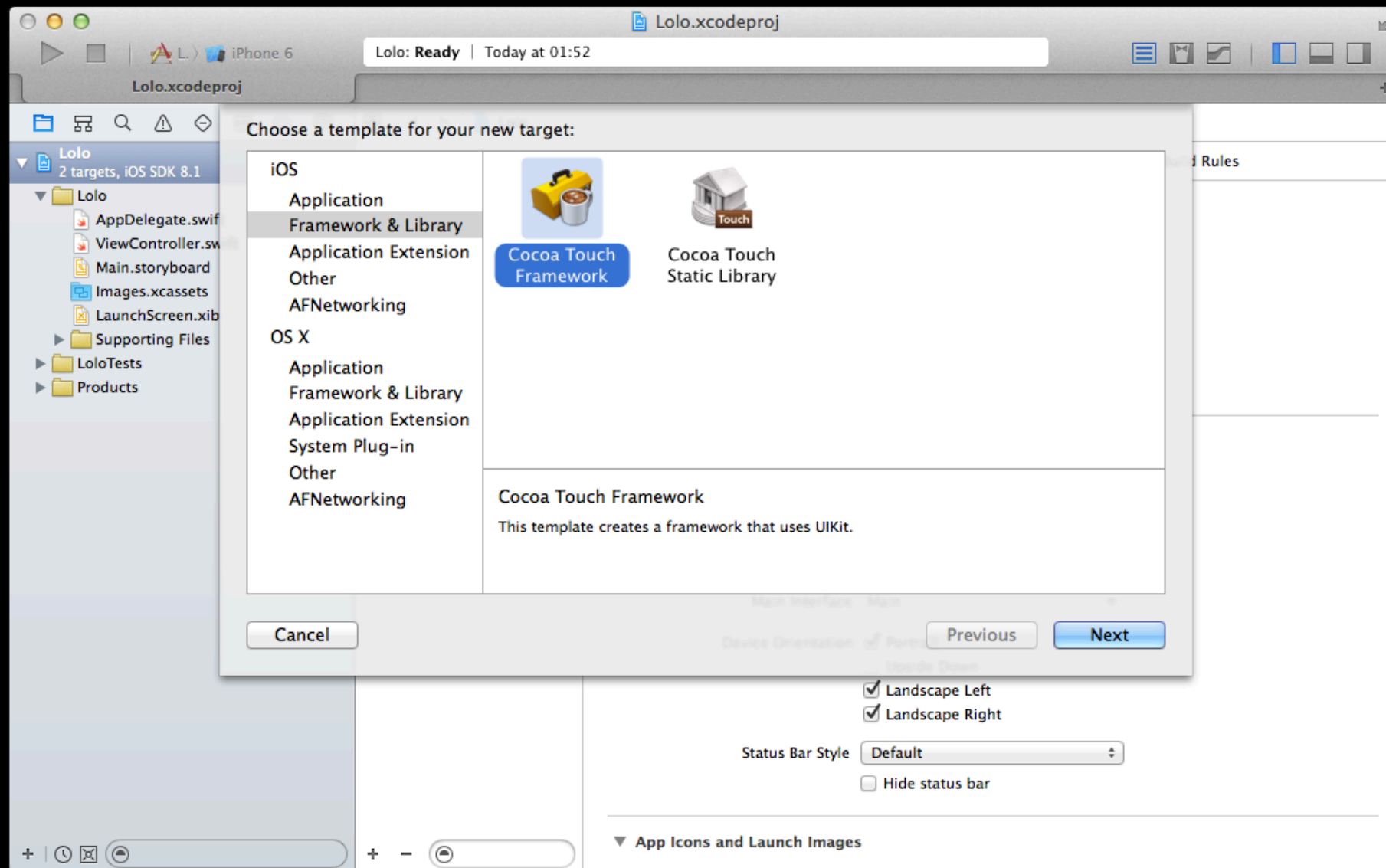
DYNAMIC FRAMEWORKS

**A dynamic framework, when updated,
does not require applications to relink
to them**

**OS X already supports Dynamically
Linked frameworks**

ios

HOW



AND...

COCOAPODS

COCOAPODS



mrackwitz commented 25 days ago

Collaborator

Please make support requests around Swift CocoaPods on this repo:

<https://github.com/CocoaPods/swift>

This Pull-Request makes first steps to bring the long-awaited Swift support by supporting Frameworks. Along the way it keeps full backwards-compatibility with CocoaPods 0.35 and tries to break as few stuff as needed.

This PR supersedes [#2222](#). See this for some of the original discussions.

Steps

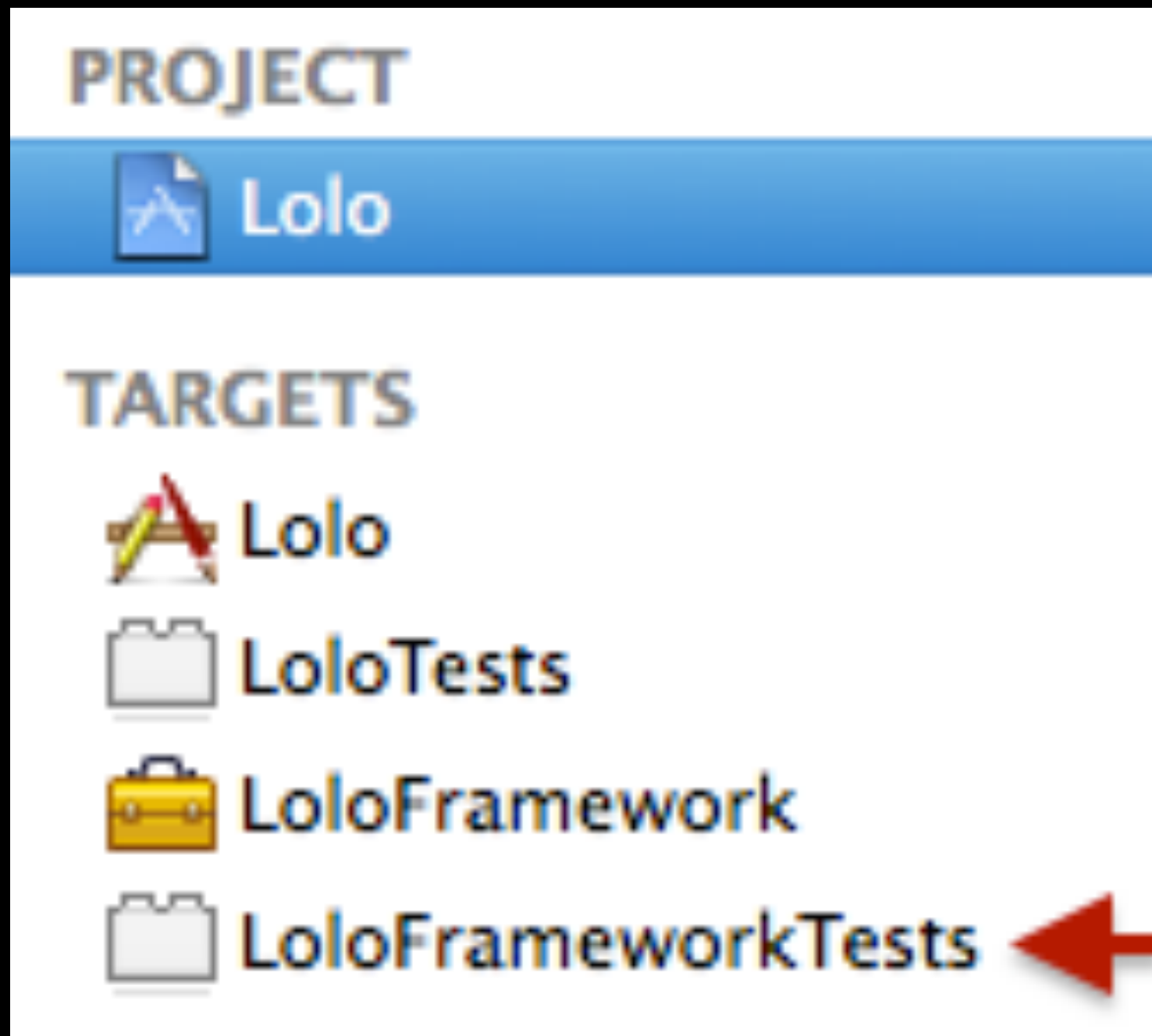
To make that all happen, the following path was taken, described here very roughly:

- Recognize Swift source files by file accessors
- Extend the analyzer to decide when it is required to build a framework
- Generate a framework product type if required
- Generate a custom umbrella header to import all public headers and avoid naming conflicts
- Generate a custom module map to inject the umbrella header
- Namespace frameworks in directories in build path, as their naming is constrained by underlying clang modules
- Migrate from a linked aggregate library to a linked framework (which is still just used instead of a target dependency to indirectly trigger the build of all pod targets)
- Link the built pod frameworks by build settings to the user target
- Set the build settings appropriate to reach all that
- Embed the built pod frameworks with a shell script build phase to the host application to allow configuration-dependent pods
- Auto opt-in OS X targets and iOS 8+ targets to dynamic frameworks (to be discussed in a separate issue, at the moment possibly more helpful for feedback)

COCOPODS

- Recognize Swift source files by file accessors
- Extend the analyzer to decide when it is required to build a framework
- Generate a framework product type if required

TESTABLE



EXTENSIONS

▼ Identity

Bundle Identifier

Version

Build


Team

▼ Deployment Info

Deployment Target

Devices

App Extensions ☒ Allow app extension API only



NS_EXTENSION_UNAVAILABLE_IOS

UIKitUI

UISheet

UIAlertView

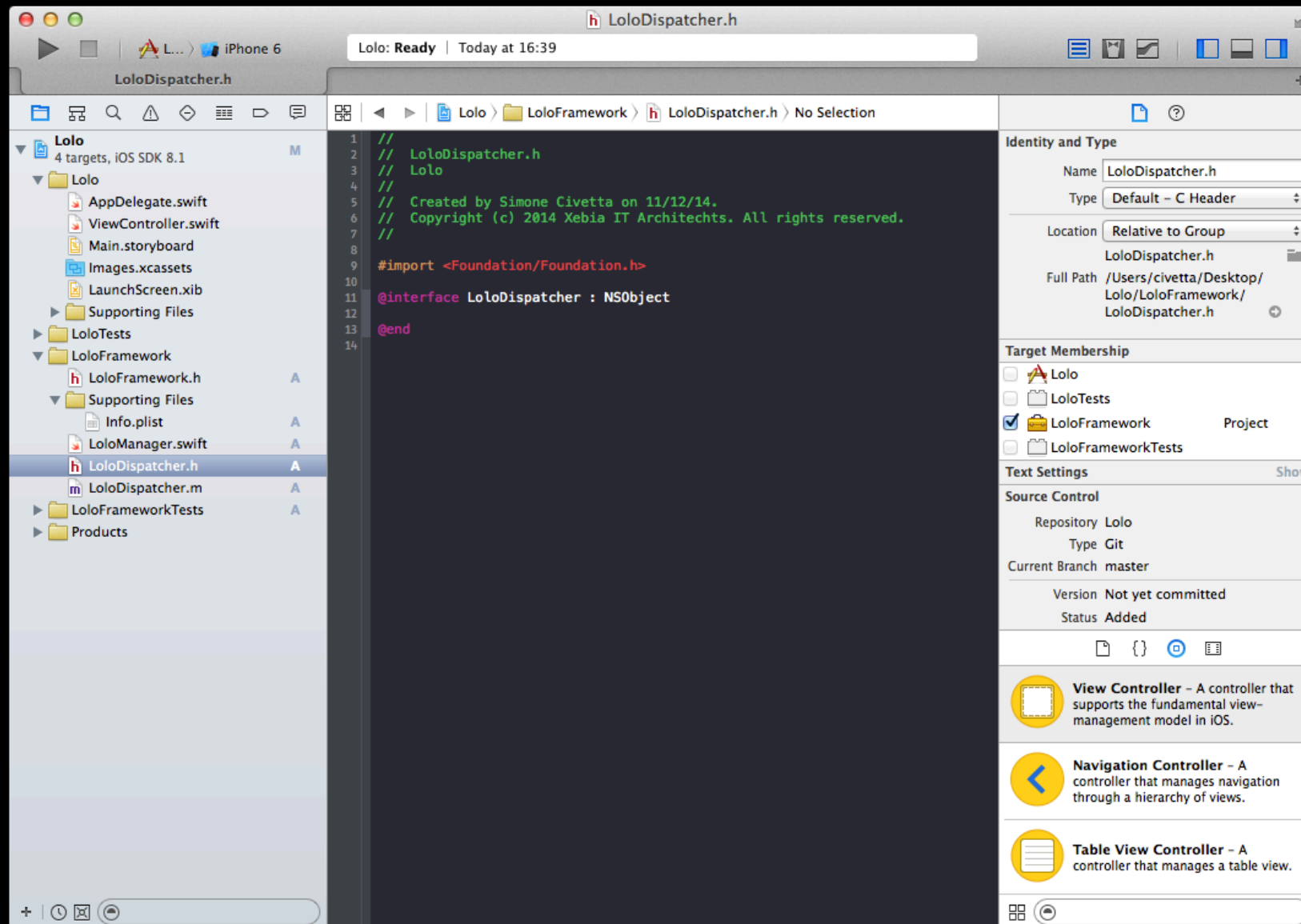
+ [UIApplication sharedApplication]

- [UIApplication beginIgnoringInteractionEvents]

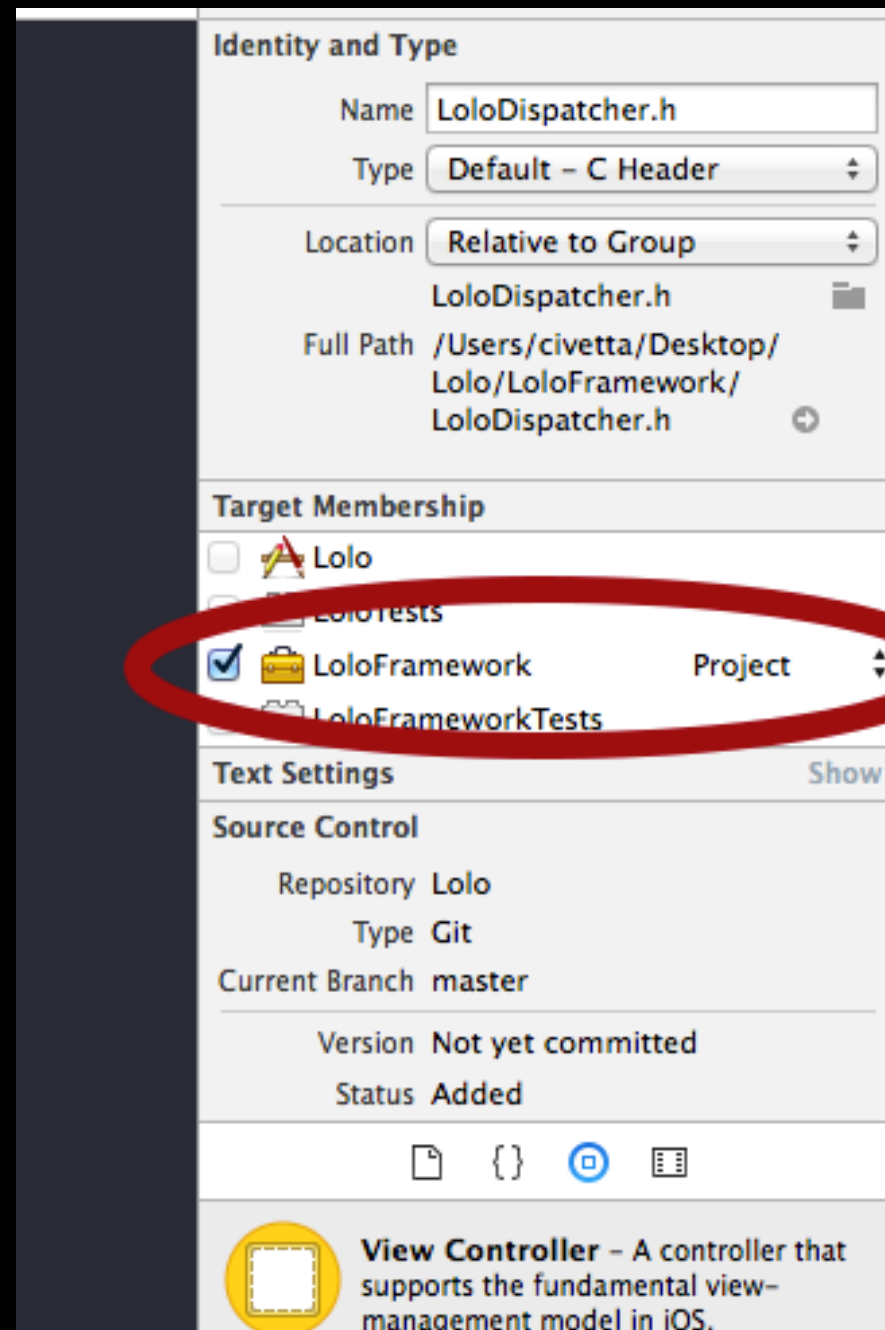
- [UIApplication endIgnoringInteractionEvents]

- [UIApplication openURL:]

VISIBILITY



VISIBILITY



VISIBILITY

- Project

- Public


- Private


VISIBILITY

▼ Headers (2 items)

×

▼ Public (2)

 LoloManager.h ...in LoloFramework

 LoloFramework.h ...in LoloFramework

▼ Private (0)

Add private header files here

▼ Project (0)

Add project header files here

+

—

REFERENCES :

WWDC 2014 Session 416



THANKS!!

 @viteinfinite

Xebia