# **Objective C Coding Interview Challenge: URLPuller**

JAN 13TH, 2014

# **Programming Challenge**

The goal is to create an Objective-C class which takes an array of NSURL’s, downloads their contents asynchronously, and writes the contents of each URL to a unique file.

The class should be named URLPuller and should have the following instance methods:

* - (void) downloadUrlsAsync: (NSArray\*)urls
* - (void) waitUntilAllDownloadsFinish
* - (NSString\*) downloadedPathForURL: (NSURL\*)url

Here are the detailed descriptions of each method:

## downloadUrlsAsync

- (void) downloadUrlsAsync: (NSArray\*)urls

The urls array will contain an array of NSURL’s. The method should return immediately, eg, it’s an asynchronous API call that will run in the background.

The downloaded files can be stored in any directory, where the filename is {sha1 hash of url}.downloaded

## waitUntilAllDownloadsFinish

- (void) waitUntilAllDownloadsFinish

Block until all of the URLs have been downloaded.

If this is called while there is no download in progress, it should return immediately. Otherwise, it should block until all the downloads have finished.

## downloadedPathForURL

- (NSString\*) downloadedPathForURL: (NSURL\*)url

Return the path where the given url was downloaded to.

If the given url has not been downloaded yet, or was never requested to be downloaded, then it should return nil.

# **Objectives**

Demonstrate that:

* You can write a program that satisfies the API methods and described behavior.
* You can write code that is well factored, and easy to understand.
* You can clearly communicate questions (if any arise) regarding this specification.

# **Deliverable**

* Zipped Xcode project directory or a link to github repo that contains the Xcode project. It can be either an iOS or MacOS project.

# **Rules**

* You can use any available resource you can find on the Internet — eg, Stack Overflow, API docs, etc.
* You may not ask questions related to the problem on Stack Overflow, IRC, etc. (eg, the Internet is “read-only”)
* You cannot use any 3rd party libraries or products, you can only use classes and API’s in the standard iOS/Cocoa Objective-C runtime. If in doubt, just ask.
* You are welcome to go above and beyond the call of duty in any way that you see fit, however with the constraint that you must provide the above methods with signatures exactly matching with the specification