

📧 Join our exclusive newsletter for Recruiters and Sourcers

Launching on Oct 20th, 2023.
Become tech-savvy – sign up today!

📧 Your work email

iOS

Swinject

A lightweight dependency injection framework for Swift apps. It allows you to split an app into loosely-coupled components, which can then be maintained and tested more easily.

NSOperation

Represents a single unit of work. It's an abstract class that offers a useful, thread-safe structure for modeling state, priority, dependencies, and management.

GHUnit

GHUnit is a test framework for [Mac OS X](#) and [iOS](#). It can be used standalone or with other testing frameworks like SenTestingKit or GTM.

3D Touch

A pressure sensitive technology Apple introduced in iPhone 6s and 6s Plus devices running iOS 9. With 3D Touch, an iPhone app is now able to not only tell that the user is touching the device's screen, but it is able to sense how deeply users press the display, letting do the things they do most often, faster and in fewer steps as well as preview all kinds of content and even act on it without having to actually open it.

Accelerate framework

Available in both [Swift](#) and [Objective-C](#), the Accelerate Framework is used to make large-scale mathematical and image calculations much easier for developers and optimized for high performance tasks. As a result it is used extensively in machine learning programs. The framework contains a variety of [C APIs](#) for vector and matrix math, digital signal processing, large number handling, and image processing.

AFNetworking

A delightful networking library for [iOS](#) and Mac OS X. It is extending the powerful high-level networking abstractions built into [Cocoa](#). It has a modular architecture with well-designed, feature-rich [APIs](#) that are a joy to use.

Alamofire

A [Swift](#)-based [HTTP](#) networking library for [iOS](#) and Mac OS X. It provides an elegant interface on top of Apple's Foundation networking stack that simplifies a number of common networking tasks.

APNs

Apple Push Notification service (APNs) is the centerpiece of the remote notifications feature. It is a robust, secure, and highly efficient service for app developers to propagate information to [iOS](#) (and, indirectly, watchOS), tvOS, and macOS devices. The notification information sent can include badges, sounds, newsstand updates, or custom text alerts.

AppCode

An [IDE](#) used for developing iOS/macOS apps. AppCode natively supports [Objective-C](#), [Swift](#), [C](#) and [C++](#), as well as [JavaScript](#), [XML](#), [HTML](#), [CSS](#) and [XPath](#).

Auto Layout

A constraint-based layout system that allows developers to create an adaptive [UI](#) that responds appropriately to changes in screen size and device orientation. Auto Layout dynamically calculates the size and position of all the views in your view hierarchy. What's more, it is compatible with many of Apple's most exciting [APIs](#), including animations, motion effects, and sprites.

ARC

Automatic Reference Counting, ARC for short, is a technology used to track and manage your app's memory usage.

ARKit

Apple's AR development platform for [iOS](#). Enables developers to build high-detail augmented-reality apps.

AVFoundation

A full featured framework that provides an [Objective-C](#) interface to play and create time-based audiovisual media on iOS, macOS, watchOS and tvOS.

Bond

A binding framework for [Swift](#). It is built on top of [ReactiveKit](#). Provides AppKit and UIKit bindings, reactive delegates and datasources to ReactiveKit.

Carthage

A ruthlessly simple dependency manager for macOS and iOS. It exclusively uses dynamic frameworks instead of static libraries – this is the only way to distribute Swift binaries that are supported by iOS 8 and up.

Clean Architecture (VIPER)

An application of Clean Architecture to [iOS](#) projects. Divides an app's logical structure into distinct layers of responsibility, thereby making it easier to isolate dependencies (e.g. your [Database](#)) and test the interactions at the boundaries between layers.

Cocoa

An Apple's object-oriented [API](#) for macOS. Cocoa apps are developed using specifically [Xcode](#). It consists of the Foundation Kit, Application Kit, [Core Data](#) frameworks, etc.

Cocoa Touch

A user interface framework provided by Apple for building software applications on [iOS](#), main features are: Core Animation; Core Audio; [Core Data](#). It is primarily written in [Objective-C](#) language.

CocoaPods

An application level dependency manager for the cocoa projects [Swift](#) and [Objective-C](#). There are more than ten thousand libraries that will help you in creating fast and elegant applications.

Core Animation

A graphics compositing framework used by macOS, [iOS](#), watchOS, and tvOS. Used to animate the views and other visual elements of apps.

This website uses cookies



We use cookies to continuously improve your experience on our site. [More info.](#)

Got it!

Core
Graphics

A vector drawing framework in MacOS and [iOS](#), based on the Quartz advanced drawing engine. It provides 2D drawing, path-based drawing, color management, offscreen rendering, display, parsing etc.

Core
Location

An iOS framework that provides services used to get and monitor the current location data, allows determining a device's geographic location, altitude, orientation. This library uses the available equipment to determine the position and direction of the user.

CoreFounda
tion

A cross-platform [C](#)-based [API](#) that provides reference-counted data structures, also provides some crucial [Objective-C](#) classes, allows to customize how memory is managed and objects are identified.