**OVERVIEW AND ARCHITECTURE OF IOS**

**INTRODUCTION TO IOS:**

IOS is a OS developed and distributed by application mainly for I phone, I pad, Touch and Apple TV.

IOS mainly derived from OS.

**ARCHITECTURE OF IOS:**

Architecture of [IOS](https://intellipaat.com/blog/tutorial/ios-tutorial/ios-graphics/) is a layered architecture.

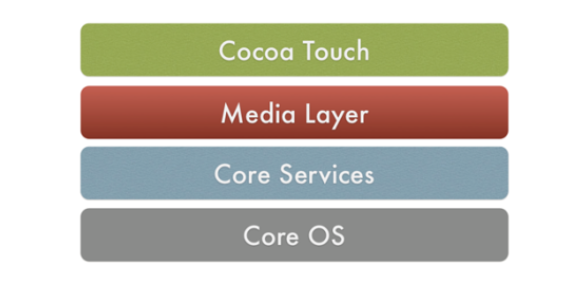
At the **Uppermost level iOS** works as an intermediary between the underlying hardware and the apps you make. Apps do not communicate to the underlying hardware directly.

Apps talk with the hardware through a collection of well defined system interfaces. These interfaces make it simple to write apps that work constantly on devices having various hardware abilities.

**Lower layers** gives the basic services which all application relies on and higher level layer gives sophisticated graphics and interface related services.

Apple provides most of its system interfaces in special packages called **frameworks.**

A framework is a directory that holds a dynamic shared library that is .a files, related resources like as header files, images, and helper apps required to support that library.



**Fig: Layers of IOS**

Every layer have a set of Framework which the developer use to construct the applications.

**1. Core OS Layer:**

The Core OS layer holds the low level features that most other technologies are built upon.

* Core Bluetooth Framework.
* Accelerate Framework.
* Local Authentication framework.

**2. Core Services Layer:**

Some of the Important Frameworks available in the core services layers are detailed :

* Address book frame work :
* Cloud Kit frame work
* Core Foundation framework:

**3. Media Layer:**

Graphics, Audio and Video technology is enabled using the Media Layer.

Graphics Framework:

* UI Kit Graphics
* Core Graphics framework
* Core Animation
* Core Images

**4. Cocoa Touch Layer**

* Event Kit framework
* Game Kit Framework
* I Ad Framework

**Objective C:**

Objective C is used in iOS development. It is built on top of the C programming language and which offers object oriented capabilities and a dynamic runtime. It includes all familiar elements like as primitive types i.e. int, float, char etc. structures, pointers, functions, and control flow constructs.

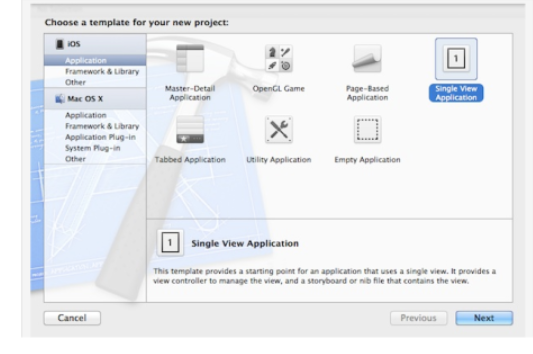
**FIRST IOS APPLICATION:**

**How to create first iOS App in Xcode?**

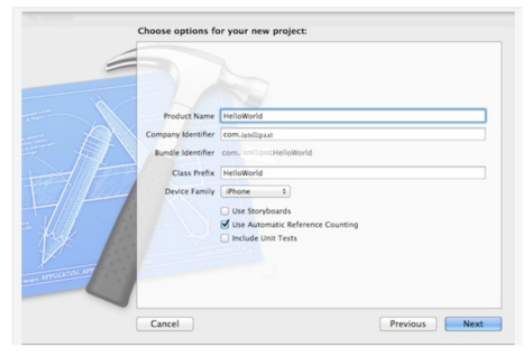
**Step 1**: Open Xcode and choose Create a new Xcode project to start a new project.



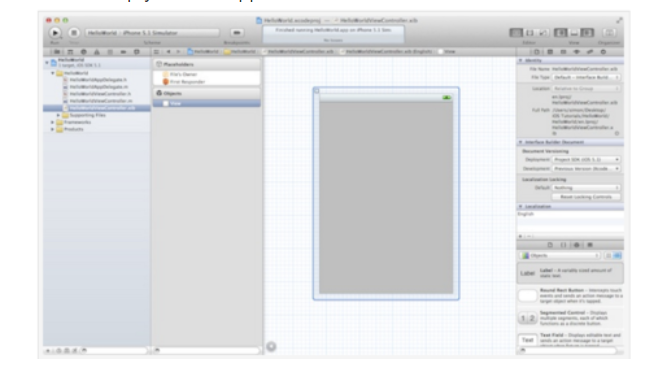
**Step 2** − Select Single View Application and click “Next”.



**Step 3:**Fill specified information



* **Product Name: HelloWorld**–  It indicate the name of app.
* **Company Identifier: intellipaat**–It specifies a domain name. If you don’t have domain name you can use “edu.self”.
* **Class Prefix: HelloWorld**– Xcode uses the class prefix to name the class automatically.
* **Device Family: iPhone**–  use iPhone
* **Use Automatic Reference Counting: [checked]**– By default, this should be enabled. It is chosen to automatically release the resources assigned once it goes beyond scope.
* **Step 4:**Choose HelloWorld View Controller.xib to complete your app. After choosing the file the editor changes to an interface builder and shows an empty view of app like as follows:



**Step 5:** To alter the label of the button double press it and name it “Hello World” and then run the app again then it looks like as follows:

