

Lesson 3: Frameworks

INTRODUCTION

Frameworks are the libraries that represent best practices for development on the iOS platform. Frameworks provide the interfaces needed to incorporate technologies, services, and features into the iOS app which makes app development much easier since a lot of the work is done.

LESSON OBJECTIVES

By the end of this lesson, the student will be able to:

1. Define framework.
2. Discuss how to add a framework to a project.
3. Identify Core Location and Map Kit frameworks.
4. Identify the different ways to get a location on an iOS device.
5. Define longitude, latitude and altitude.
6. Explain the pros and cons of different location methods.
7. Identify the effect of using a lesser degree of accuracy in designing iOS apps.
8. Add annotations to a map app.

LEARNING SEQUENCE

Required Reading	Read the following: <ul style="list-style-type: none">• Lesson 3: Frameworks
Resources	View the following: <ul style="list-style-type: none">• Xcode 5 iOS 7 Map Kit Framework Get User Location – Tutorial (8:44)• Xcode 5 iOS7 Map Kit Framework – Location Search Tutorial (13:20) Other resources: <ul style="list-style-type: none">• Developer website > iOS Apps icon > iOS Developer Library link• iOS 6 Core Location Tutorial• Getting Location Information using the iOS 7 Core Location Framework
Assignments	Complete the following: <ol style="list-style-type: none">1. Practice Exercise – Map It2. Practice Exercise – Where is This?3. Quiz 3



This work by the National Information Security and Geospatial Technologies Consortium (NISGTC), and except where otherwise noted, is licensed under the [Creative Commons Attribution 3.0 Unported License](#).

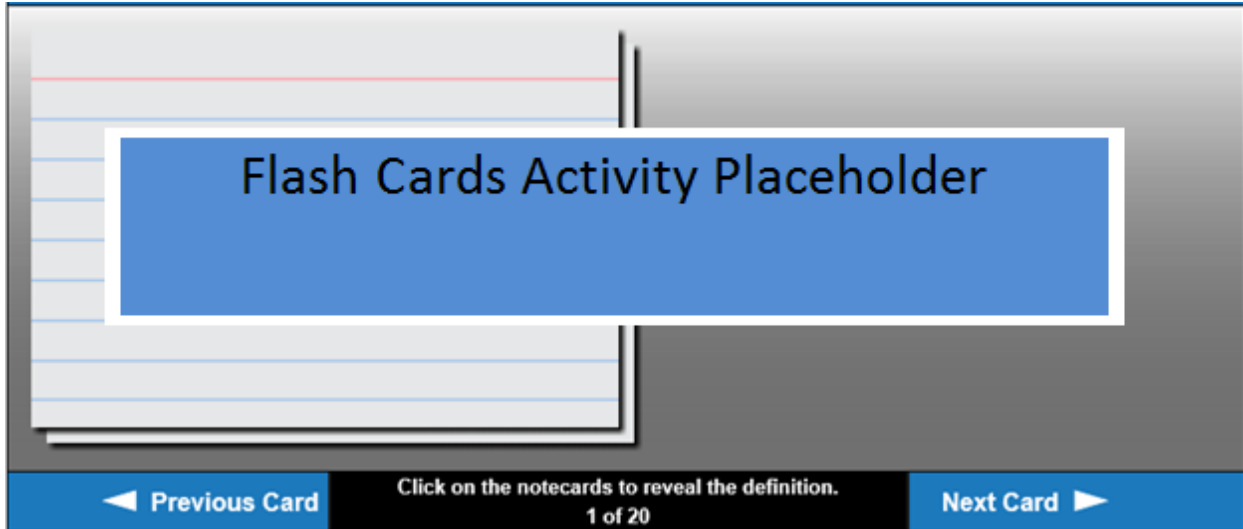
Authoring Organization: Collin College

Written by: Original Author, Elizabeth Pannell; Edited Version, Susan Sands

Copyright: © National Information Security, Geospatial Technologies Consortium (NISGTC)

KEY TERMS

As you read your lesson, pay close attention to the [key terms and phrases](#) listed throughout the lesson. These terms and concepts are important to your understanding of the information provided in the lesson.



INSTRUCTION

What is a Framework?

A **framework** is a hierarchical directory containing shared resources including a shared library, nib files, image files, localized strings, header files, and reference documentation. When developing an app, the project links to one or more frameworks. For example, an iPhone project links by default to the Foundation, UIKit, and Core Graphics frameworks.

A list of available frameworks can be found on [Apple's Developer website](#):

1. Click the iOS Apps icon.
2. Click the iOS Developer Library link.
3. In the column on the left, the frameworks are listed alphabetically by layer.

Adding a Framework

Frameworks are added to a project in Xcode.

1. Select the target in the pane on the left.
2. Click on Build Phases.
3. Click the triangle to expand the **Link Binary With Libraries** section as shown in Figure 1.
4. Click the Plus sign (+).



This work by the National Information Security and Geospatial Technologies Consortium (NISGTC), and except where otherwise noted, is licensed under the [Creative Commons Attribution 3.0 Unported License](#).

Authoring Organization: Collin College

Written by: Original Author, Elizabeth Pannell; Edited Version, Susan Sands

Copyright: © National Information Security, Geospatial Technologies Consortium (NISGTC)

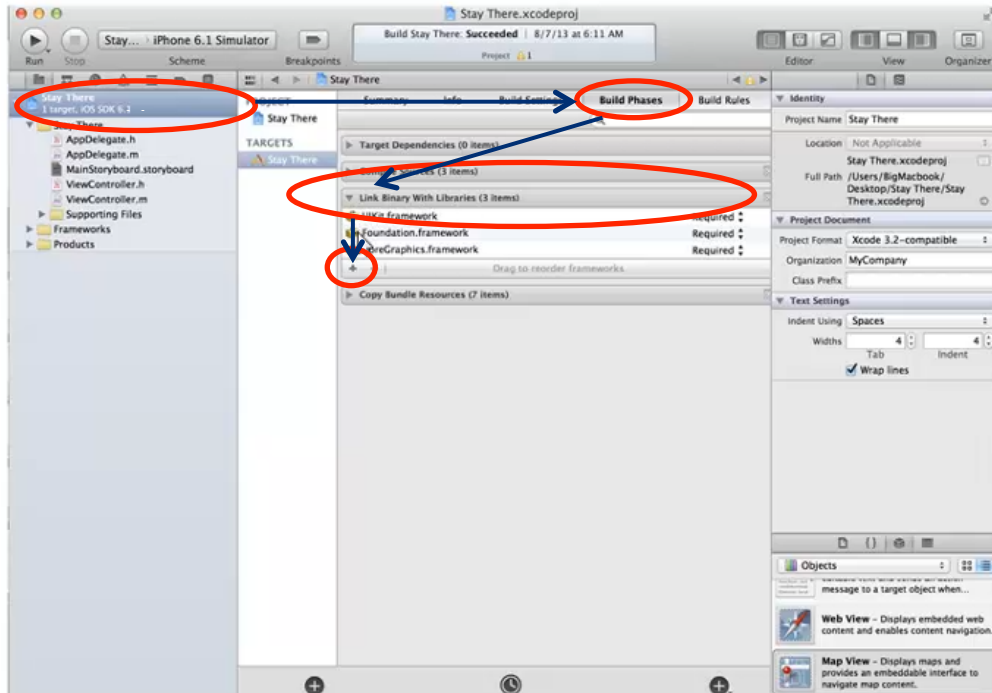


Figure 1: Adding a Framework

A list of libraries and frameworks will appear, as shown in Figure 2.

5. Scroll down and select the **CoreLocation.framework**
6. Click the Add button.



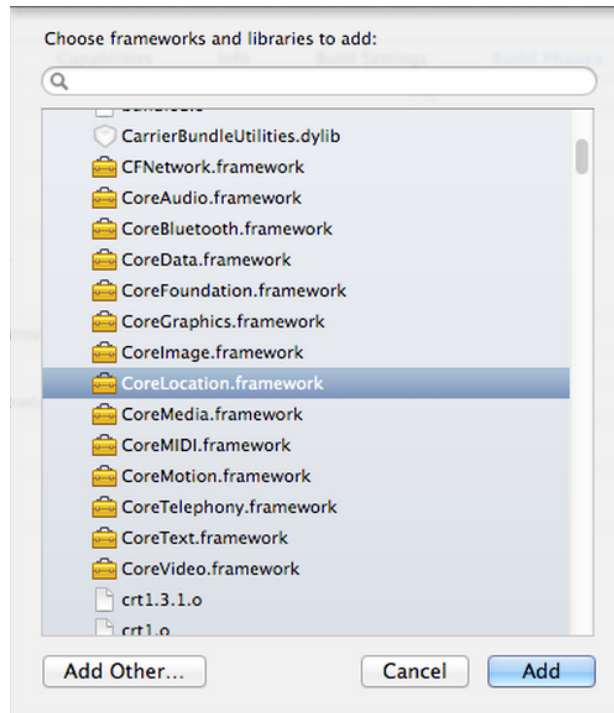


Figure 2: Libraries and frameworks

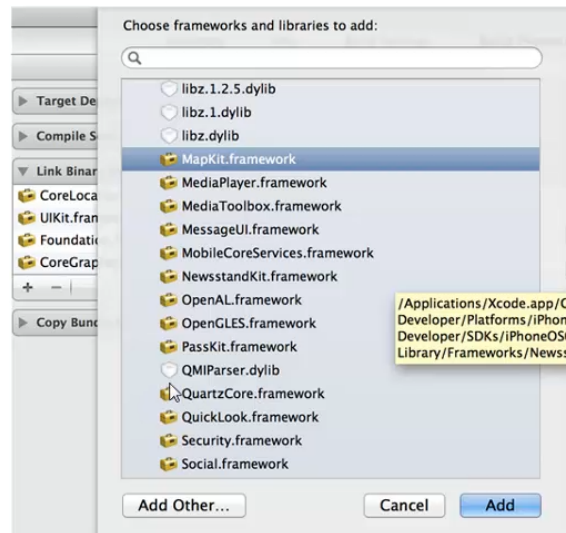


Figure 3: Available frameworks and libraries

7. Click the plus sign in the **Link Binaries With Library** section again.
8. Scroll down and find MapKit.framework and click the Add button.

Core Location Framework

The Core Location Framework allows a developer to:

- Get the user's location.



This work by the National Information Security and Geospatial Technologies Consortium (NISGTC), and except where otherwise noted, is licensed under the [Creative Commons Attribution 3.0 Unported License](https://creativecommons.org/licenses/by/3.0/).

Authoring Organization: Collin College

Written by: Original Author, Elizabeth Pannell; Edited Version, Susan Sands

Copyright: © National Information Security, Geospatial Technologies Consortium (NISGTC)

- Get the latitude and longitude coordinates by **geocoding** (finding the geographical coordinates from other data such as an address) location data.
- Take a location and reverse it to an address.
- Determine the direction or heading.
- Determine the distance traveled.

The Core Location framework is usually used along with the MapKit framework.

MapKit Framework

The MapKit framework embeds maps directly into a view. The MKMapView class provides a map to display map or satellite data. The MapKit framework also includes the ability to:

- Add annotations such as pins, titles, and subtitles.
- Add overlays which are used to customize map views.
- Reverse geocode (providing **latitude**, a north-south position, and **longitude**, an east-west position, in looking up an address).

Core Location Services

There are three services associated with Core Location:

1. Significant-change location service only triggers when a certain distance has been covered. This option takes less power.
2. Standard location service uses GPS, cell and Wi-Fi to determine location. This option has more configuration options. This is the default service, and it will use more power and resources because it will look for the location more often.
3. Region monitoring is a boundary crossing. A region is defined with a unique identifier string. An event is triggered only when the boundary line is crossed. This type of service is often used in notification or reminder apps.

Device Location

Depending on the device's capabilities, a device can determine its location using the following:

- GPS (Assisted GPS) is used when the device reads microwave signals from satellites to get its location.
- **Cell tower triangulation** uses multiple cell towers to track a device's location by measuring the time delay that a signal takes to return back to the towers from the device. This technology requires a cell phone connection.
- **Wi-Fi Protected Setup (WPS)** is designed to connect a Wi-Fi enabled device (like a smart phone) to a wireless network easily and securely. WPS uses the MAC address from a nearby Wi-Fi access point to "guess" the location. This method works better in higher density locations like urban areas.

Problems in determining location can be caused by several factors:

- An Internet or cell connection is required.



This work by the National Information Security and Geospatial Technologies Consortium (NISGTC), and except where otherwise noted, is licensed under the [Creative Commons Attribution 3.0 Unported License](https://creativecommons.org/licenses/by/3.0/).

Authoring Organization: Collin College

Written by: Original Author, Elizabeth Pannell; Edited Version, Susan Sands

Copyright: © National Information Security, Geospatial Technologies Consortium (NISGTC)

- A user has disabled location services for a specific app.
- A user has disabled location services for all apps.
- A user has disabled location services in the parental control settings.
- The device does not have a particular capability or that capability is turned off.

All three methods of determining location will drain the device's battery. If minimum accuracy is used in the app code, this will save on battery drain. For example, if ten kilometers can be traveled before an update, use that distance rather than having an update done every ten feet. It is also a best practice to turn off location services within the app code when it is not required anymore.

Location Manager

The **CLLocationManager** class conveys location-related events to the application. Use an instance of this class to establish the parameters for determining when location events should be delivered. To set the accuracy, use the following code:

```
locationManager.desiredAccuracy=kCLLocationAccuracyBest;
```

This statement will use the most accurate device location method currently available. For example, if only Wi-Fi is available, then Wi-Fi will be used as the means for obtaining accuracy even though GPS would be the more accurate choice.

The following example uses a different option for this property. Notice that a distance is specified as part of code:

```
locationManager.desiredAccuracy=kCLLocationAccuracyNearestTenMeters;
```

To stop sending updates, use the following code:

```
[locationManager stopUpdatingLocation];
```

Read the section titled Configuring CLLocationManager in the [iOS 6 Core Location Tutorial](#) for additional examples using the desiredAccuracy property.

By default, location manager reports updates whenever there are any changes detected in the location of the device. Use the distanceFilter property to notify when the location changes by a specific amount to prevent unnecessary polling and to save on battery power.

```
locationManager.desiredAccuracy=kCLLocationAccuracyKilometer;
locationMagager.distanceFilter=400;
```

The distanceFilter property specifies the minimum distance measured in meters that a device must move horizontally before an update is generated.

Browse to the document, [Getting Location Information using the iOS 7 Core Location Framework](#). Select Configuring the Distance Filter to see example of using this property.

Click on Longitude and Latitude to see examples of the following:

- Longitude and latitude values



This work by the National Information Security and Geospatial Technologies Consortium (NISGTC), and except where otherwise noted, is licensed under the [Creative Commons Attribution 3.0 Unported License](#).

Authoring Organization: Collin College

Written by: Original Author, Elizabeth Pannell; Edited Version, Susan Sands

Copyright: © National Information Security, Geospatial Technologies Consortium (NISGTC)

- Horizontal and vertical accuracy value
- **Altitude** value (a vertical distance measurement)

Handling Direction

There are two ways of getting direction information:

1. A **magnetometer** can determine direction. This is also known as a heading.
2. A GPS can determine the direction that a device is moving. This is also known as a course.

A magnetometer is a compass direction (i.e., north, south, east, and west) whereas GPS determines the direction a device is moving (not the direction it is facing).

Using MapKit

Prior to iOS 6, Apple used Google Maps. Starting with iOS 6, **TomTom maps**, a map source owned by TomTom International, are used (along with some other sources purchased by Apple). Therefore, the MapKit framework and Annotations delegate are now required.

Also added in iOS 6 is the MKMapItem which provides built-in functionality for easily creating turn-by-turn directions.

A map view can be added in one of two ways:

- Drag a Map view object to a view in Interface Builder.
- Program with the MKMapView class and initialize with initWithFrame as a subview.

Set the following:

- The mapType property.
- The zoomEnabled and scrollEnabled properties.
- The region to be displayed.

View the video, *Xcode 5 iOS 7 Map Kit Framework Get User Location – Tutorial* (8:44) to learn how to get the user location using the Map Kit framework and map view.

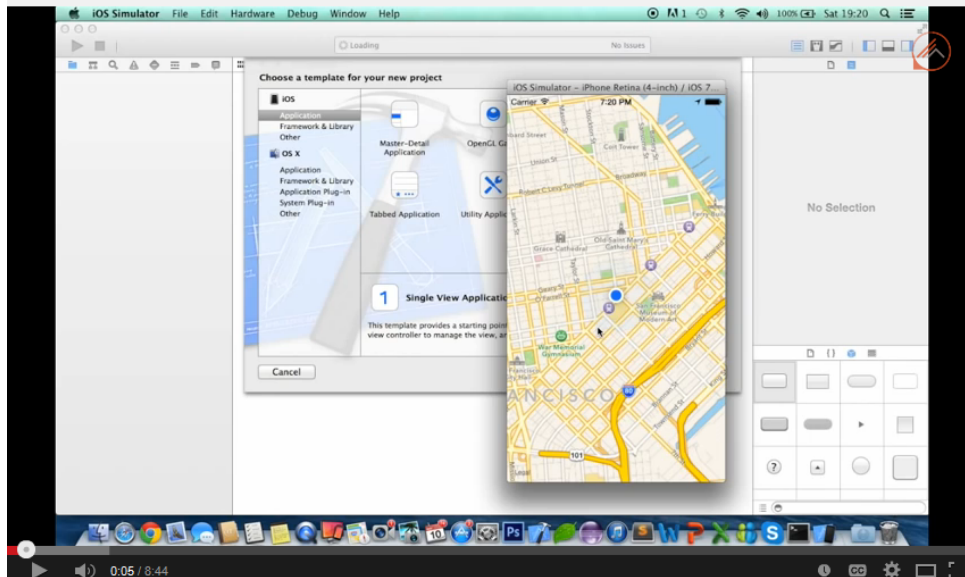


This work by the National Information Security and Geospatial Technologies Consortium (NISGTC), and except where otherwise noted, is licensed under the [Creative Commons Attribution 3.0 Unported License](https://creativecommons.org/licenses/by/3.0/).

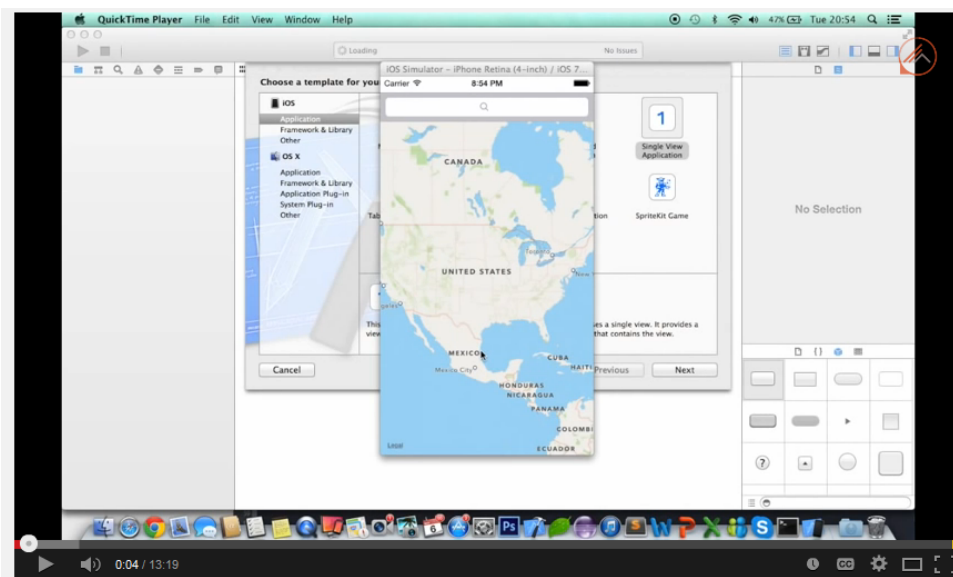
Authoring Organization: Collin College

Written by: Original Author, Elizabeth Pannell; Edited Version, Susan Sands

Copyright: © National Information Security, Geospatial Technologies Consortium (NISGTC)



The second video, *Xcode 5 iOS7 Map Kit Framework – Location Search Tutorial (13:20)*, will teach the viewer how to use the Map Kit framework to search for a location specific location.



SUMMARY

This lesson defined frameworks and looked at adding additional frameworks to a project beyond the three default frameworks (Foundation, UIKit, and Core Data). Specifically, this lesson looked at incorporating the Core Location and MapKit frameworks into an app.



This work by the National Information Security and Geospatial Technologies Consortium (NISGTC), and except where otherwise noted, is licensed under the [Creative Commons Attribution 3.0 Unported License](https://creativecommons.org/licenses/by/3.0/).
 Authoring Organization: Collin College
 Written by: Original Author, Elizabeth Pannell; Edited Version, Susan Sands
 Copyright: © National Information Security, Geospatial Technologies Consortium (NISGTC)

ASSIGNMENTS

1. Practice Exercise – Map It
2. Practice Exercise – Where is this?
3. Quiz 3



This work by the National Information Security and Geospatial Technologies Consortium (NISGTC), and except where otherwise noted, is licensed under the [Creative Commons Attribution 3.0 Unported License](https://creativecommons.org/licenses/by/3.0/).

Authoring Organization: Collin College

Written by: Original Author, Elizabeth Pannell; Edited Version, Susan Sands

Copyright: © National Information Security, Geospatial Technologies Consortium (NISGTC)