

ios



Media Library

Sisoft Technologies Pvt Ltd

SRC E7, Shipra Riviera Bazar, Gyan Khand-3, Indirapuram, Ghaziabad

Website: www.sisoft.in Email: info@sisoft.in

Phone: +91-9999-283-283

WORKING WITH IMAGES AND THE CAMERA

- Camera may or may not be available
 - but the Photo Library is
- Video may or may not be available
 - only on iPhone 3GS
- Don't reinvent the wheel
 - Use the provided picker control
- The picker supports different sources
 - Photo Library, Saved Photos or Camera
- Always check to see if your desired source is available
 - you can restrict when publishing if necessary

USING THE IMAGEPICKER

- Create an instance of the `UIImagePickerController` and set yourself as the delegate for it
- When needed, choose your source type (Photo Library, Saved Photos or Camera) and animate the picker
- Respond to the delegate events for the user either picking an image or canceling

view controller

Some Implementation

```
#import <UIKit/UIKit.h>

@interface ImagePickerController : UIViewController
<UIImagePickerControllerDelegate, UINavigationControllerDelegate>
{
    UIImagePickerController *ipc;
    IBOutlet UIImageView *bgImage;
}
@property (nonatomic, retain) UIImageView *bgImage;

- (IBAction) buttonClicked;

@end
```

```
3
6 -(IBAction) buttonClicked {
7     ipc = [[UIImagePickerController alloc] init];
8     ipc.delegate = self;
9     ipc.sourceType = UIImagePickerControllerSourceTypePhotoLibrary;
10
11     [self presentModalViewController:ipc animated:YES];
12 }
13
```

Methods Of UIImagePickerControllerDelegate



```
16 - (void)imagePickerControllerDidCancel:(UIImagePickerController *)picker {
17     // release picker
18     [[picker parentViewController] dismissModalViewControllerAnimated:YES];
19     [picker release];
20 }
21
22
23 - (void)imagePickerController:(UIImagePickerController *)picker
24 didFinishPickingMediaWithInfo:(NSDictionary *)info {
25     // set image
26     bgImage.image = [info objectForKey:UIImagePickerControllerOriginalImage];
27     // release picker
28     [[picker parentViewController] dismissModalViewControllerAnimated:YES];
29     [picker release];
30 }
31 }
```

Checking For Source

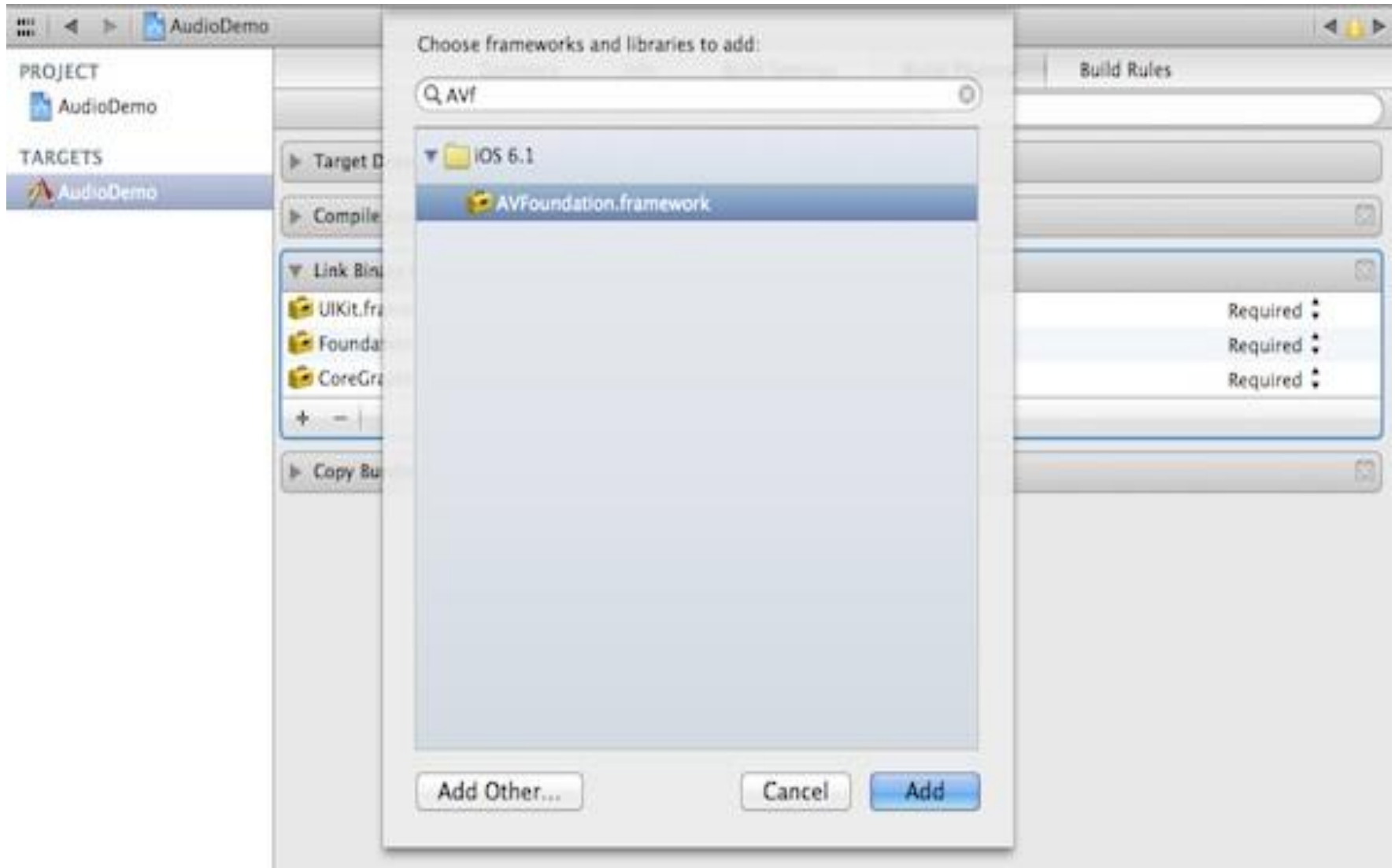
```
6 -(IBAction) buttonClicked {
7     ipc = [[UIImagePickerController alloc] init];
8     ipc.delegate = self;
9
10    if ([UIImagePickerController isSourceTypeAvailable:UIImagePickerControllerSourceTypeCamera]
11        {
12        ipc.sourceType = UIImagePickerControllerSourceTypeCamera;
13    } else
14    {
15        ipc.sourceType = UIImagePickerControllerSourceTypePhotoLibrary;
16    }
17
18    [self presentViewController:ipc animated:YES];
19 }
```

Recording And Playing Audio



- iOS provides various framework to let you work with sound in your app such as System Sound Services, AVFoundation Framework, Media Player Framework, OpenAL etc..
- One of the frameworks that you can use to play and record audio file is the AV Foundation Framework
- The AVFoundation provides easy ways to deal with audio

Adding Framework



Used classes of AVFoundation



- AVAudioPlayer
 - think of it as an audio player for playing sound files
 - By using the player, you can play sounds of any duration and in any audio format available in iOS
- AVAudioRecorder
 - an audio recorder for recording audio



Audio Session

- An audio session is the intermediary between your application and iOS for configuring audio behavior
- Initializing your audio session
 - `AVAudioSession *session = [AVAudioSession sharedInstance];`
- Activating and Deactivating your audio session
 - `[session setActive:YES error:nil];`
 - `[session setActive:NO error:nil];`



Audio Session Category

- An audio session category is a key that identifies a set of audio behaviors for your application
- By setting a category, you indicate your audio intentions to the system
- For example
 - [session setCategory:
 - AVAudioSessionCategoryPlayAndRecord error:nil];
 - It may be only play or record etc..



Delegate Methods

- AVAudioRecorderDelegate Methods
 - audioRecorderDidFinishRecording
 - audioRecorderEncodeErrorDidOccur
- AVAudioPlayerDelegate
 - audioPlayerDidFinishPlaying
 - audioPlayerEncodeErrorDidOccur
- For example on Recording and Playing audio

Visit link:

- <http://www.appcoda.com/ios-avfoundation-framework-tutorial/>

Recording And Playing Video



- For recording video we will use UIImagePickerController controller to which we have used in capturing photo/choosing photo
- Defining source typeCamer, which open camera by default
- To capture video we define media type to
 - *kUTTypeMovie*
 - picker.mediaTypes = `[[NSArray alloc] initWithObjects: (NSString *) kUTTypeMovie, nil];`

Adding Media Frameworks



- MediaPlayer framework brings us two main classes to display videos or movies
- MPMoviePlayerController
 - playback occurs in a view owned by the movie player
- MPMoviePlayerViewControllerclass
 - is designed to present a simple view controller for displaying full-screen movies



MPMoviePlayerController

- Configure video view for video playback is simply allocating MPMPC and defining its size
- Sending play message to Play
- Using MPMoviePlayerController notification i.e MPMPLPlaybackDidFinishNotification to close media player view when playback finished
- For example on video recording and plying visit link:<http://www.appcoda.com/video-recording-playback-ios-programming/>