

Разработка приложений для iOS

Лекция II

Расширенные возможности устройств

Глеб Тарасов
gleb34@gmail.com

Вспомним прошлое занятие

Какие UINavigationController от Apple
работают только на iPad?

Какие UINavigationController от Apple
работают только на iPad?

UIPopoverController, UISplitViewController

**Какие есть два вида
автоматического растягивания
интерфейсов в iOS?**

Какие есть два вида автоматического растягивания интерфейсов в iOS?

старый и более простой Autosizing,
новый и мудреный AutoLayout

Как отключить AutoLayout в StoryBoard?

Как отключить AutoLayout в StoryBoard?

снять галку «use auto layout»
в первой вкладке справа

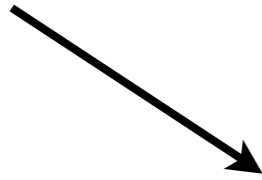
Акселерометр, гироскоп

Ориентация

```
UIInterfaceOrientation orientation =  
    [UIApplication sharedApplication].statusBarOrientation;  
  
if (UIInterfaceOrientationIsPortrait(orientation))  
    NSLog(@"portrait");  
else  
    NSLog(@"landscape");  
  
typedef enum {  
    UIInterfaceOrientationPortrait,  
    UIInterfaceOrientationPortraitUpsideDown,  
    UIInterfaceOrientationLandscapeLeft,  
    UIInterfaceOrientationLandscapeRight  
} UIInterfaceOrientation;
```

Акселерометр

1.0 по оси



ускорение $+1.0g$ вдоль этой оси



Когда телефон лежит на столе:

$$x = 0$$

$$y = 0$$

$$z = -1$$

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

```
- (void)viewDidLoad
```

```
{
```

```
    [super viewDidLoad];
```

```
    self.manager = [[CMMotionManager alloc] init];
```

```
    [self.manager startAccelerometerUpdates];
```

```
    self.timer = [NSTimer scheduledTimerWithTimeInterval:1/30.0
```

```
                target:self
```

```
                selector:@selector(doAccUpdate)
```

```
                userInfo:nil
```

```
                repeats:YES];
```

```
}
```

```
- (void)doAccUpdate
```

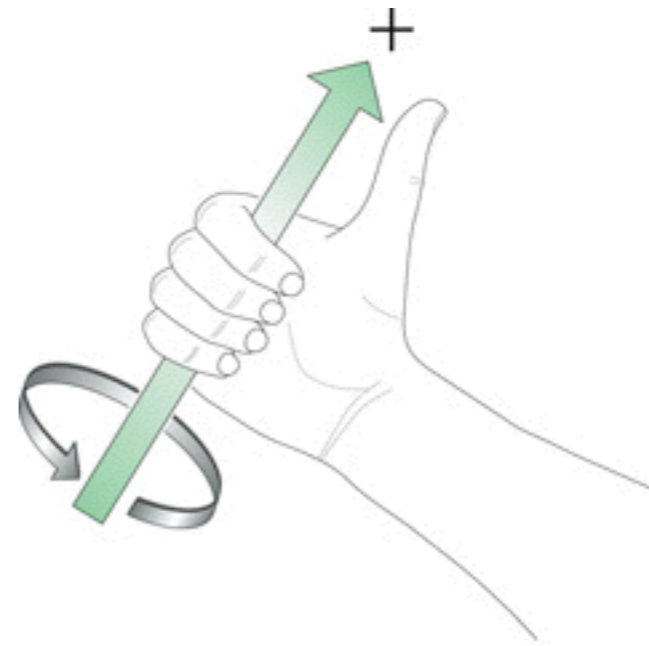
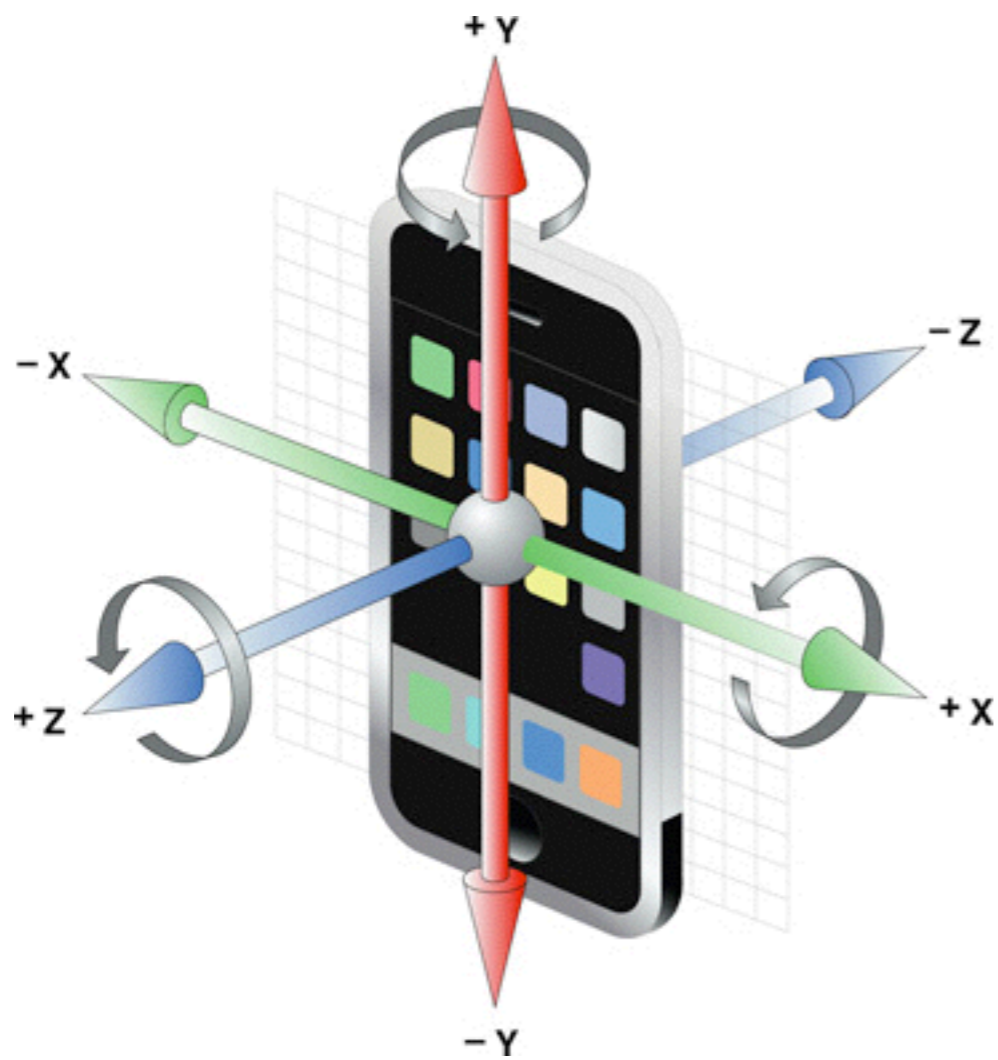
```
{
```

```
    CMAcceleration acc = self.manager.accelerometerData.acceleration
```

```
    NSLog(@"%g %g %g", acc.x, acc.y, acc.z);
```

```
}
```

Гироскоп



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

```
- (void)viewDidLoad
```

```
{
```

```
    [super viewDidLoad];
```

```
    self.motionManager = [[CMMotionManager alloc] init];
```

```
    [self.manager startGyroUpdates];
```

```
    self.timer = [NSTimer scheduledTimerWithTimeInterval:1/30.0
```

```
                target:self
```

```
                selector:@selector(doGyroUpdate)
```

```
                userInfo:nil
```

```
                repeats:YES];
```

```
}
```

```
- (void)doGyroUpdate
```

```
{
```

```
    CMRotationRate rate = self.motionManager.gyroData.rotationRate;
```

```
    NSLog(@"%g %g %g", rate.x, rate.y, rate.z);
```

```
}
```

радианы в секунду

```
- (void) viewDidLoad
{
    [super viewDidLoad];
    self.manager = [[CMMotionManager alloc] init];
}

- (void) viewWillAppear:(BOOL)animated
{
    [super viewWillAppear:animated];

    [self.manager startGyroUpdates];
    self.timer = ...
}

- (void) viewWillDisappear:(BOOL)animated
{
    [super viewWillDisappear:animated];

    [self.timer invalidate];
    [self.manager stopGyroUpdates];
}
```

Фильтр нижних частот

http://ru.wikipedia.org/wiki/Фильтр_нижних_частот

<http://developer.apple.com/library/ios/#samplecode/AccelerometerGraph/Introduction/Intro.html>

Геолокация

Координаты

```
@interface MyViewController : UIViewController<CLLocationManagerDelegate>
```

```
self.manager = [[CLLocationManager alloc] init];  
self.manager.delegate = self;  
self.manager.desiredAccuracy = kCLLocationAccuracyHundredMeters;  
[self.manager startUpdatingLocation];
```

```
- (void)locationManager:(CLLocationManager *)m  
    didUpdateToLocation:(CLLocation *)newLocation  
    fromLocation:(CLLocation *)oldLocation  
{  
  
    CLLocationCoordinate2D coord = newLocation.coordinate;  
    NSLog(@"%g %g", coord.latitude, coord.longitude);  
}
```

Центр Москвы: 55.747598, 37.626801

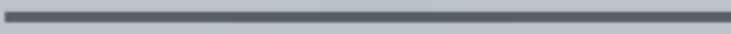
Компас

```
self.locationManager = [[CLLocationManager alloc] init];
self.locationManager.desiredAccuracy = kCLLocationAccuracyBest;
[self.locationManager setDelegate:self];
[self.locationManager startUpdatingHeading];
```

```
- (void)locationManager:(CLLocationManager *)manager
    didUpdateHeading:(CLHeading *)newHeading
{
    NSLog(@"%g", newHeading.magneticHeading);
}
```

0 - север, 90 - восток, 180 - юг, 270 - запад

Воспроизведение аудио



AirDrop



AirPlay



```
- (void) viewDidAppear:(BOOL)animated
{
    [[UIApplication sharedApplication] beginReceivingRemoteControlEvents];
    [self becomeFirstResponder];
}

- (void) remoteControlReceivedWithEvent:(UIEvent *)receivedEvent
{
    if (receivedEvent.type == UIEventTypeRemoteControl)
    {
        switch (receivedEvent.subtype)
        {
            case UIEventSubtypeRemoteControlPlay:
                [self.player play];
                break;
            case UIEventSubtypeRemoteControlPause:
                [self.player pause];
                break;
            case UIEventSubtypeRemoteControlPreviousTrack:
                [self prevTrack];
                break;
            case UIEventSubtypeRemoteControlNextTrack:
                [self nextTrack];
                break;
            default:
                break;
        }
    }
}
```

44:21

-1:23:12

Радио-Т 386

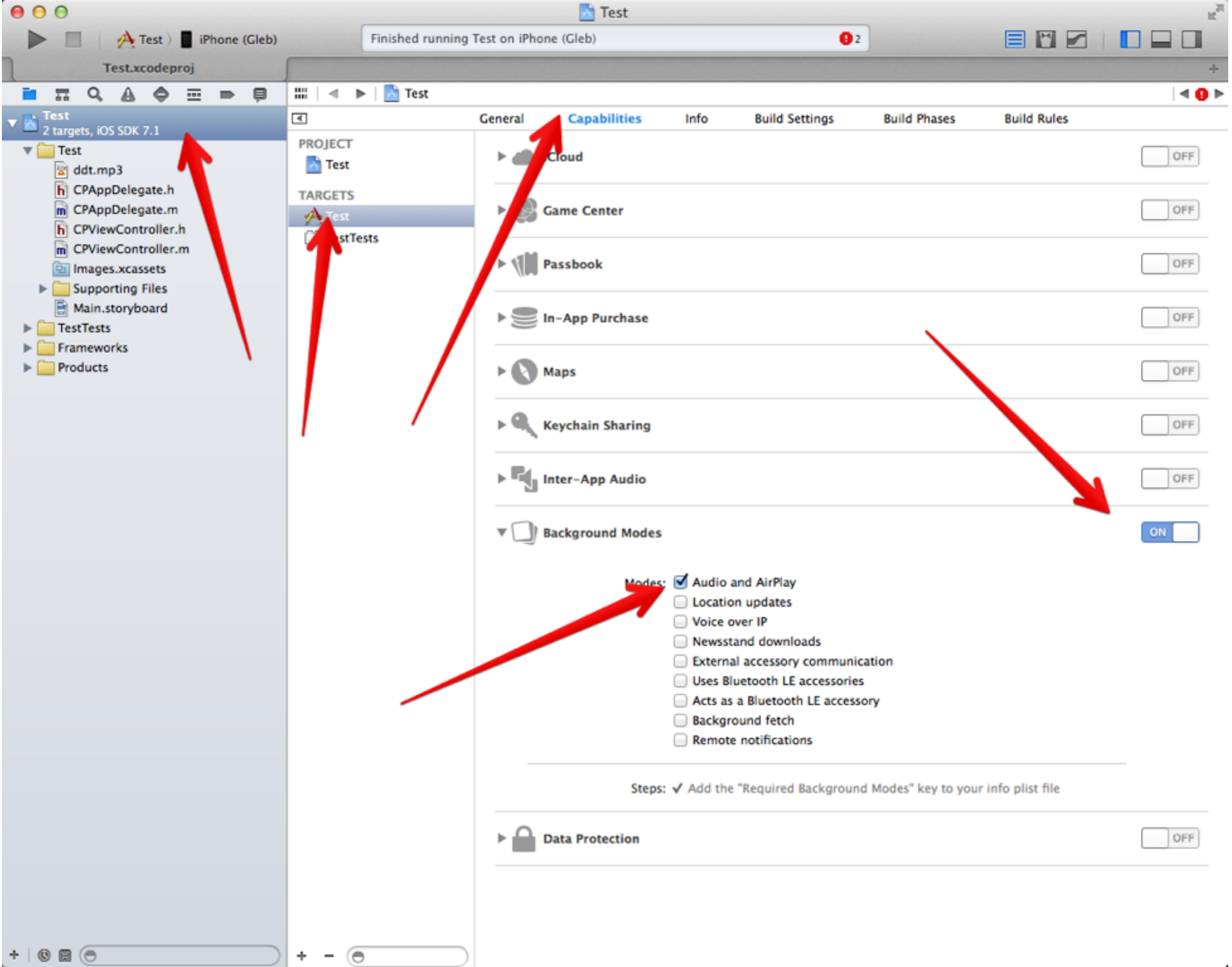
Umputun, Bobuk, Gray, Ksenks — Радио-Т

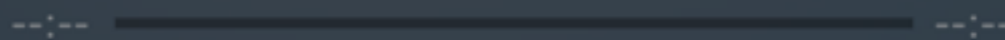


еженедельный hi-tech
подкаст

> Разблокируйте







> Разблокируйте



```
UIImage *img = [UIImage imageNamed:@"ddt"];
MPMediaItemArtwork *art = [[MPMediaItemArtwork alloc] initWithImage:img];

NSDictionary *dict = @{ MPMediaItemPropertyTitle : @"Свобода",
                        MPMediaItemPropertyAlbumTitle : @"Иначе",
                        MPMediaItemPropertyAlbumTrackNumber : @15,
                        MPMediaItemPropertyAlbumTrackCount : @16,
                        MPMediaItemPropertyArtist : @"ДДТ",
                        MPMediaItemPropertyPlaybackDuration : @125,
                        MPNowPlayingInfoPropertyPlaybackRate : @1,
                        MPMediaItemPropertyGenre : @"Рок",
                        MPMediaItemPropertyArtwork : art };

[MPNowPlayingInfoCenter defaultCenter].nowPlayingInfo = dict;
```

0:04 -2:00

Свобода

ДДТ — Иначе



> Разблокируйте



AVQueuePlayer

```
NSURL *url1 = [[NSBundle mainBundle] URLForResource:@"song1"  
withExtension:@"mp3"];
```

```
NSURL *url2 = [[NSBundle mainBundle] URLForResource:@"song2"  
withExtension:@"mp3"];
```

```
NSURL *url3 = [[NSBundle mainBundle] URLForResource:@"song3"  
withExtension:@"mp3"];
```

```
AVPlayerItem *item1 = [[AVPlayerItem alloc] initWithURL:url1];
```

```
AVPlayerItem *item2 = [[AVPlayerItem alloc] initWithURL:url2];
```

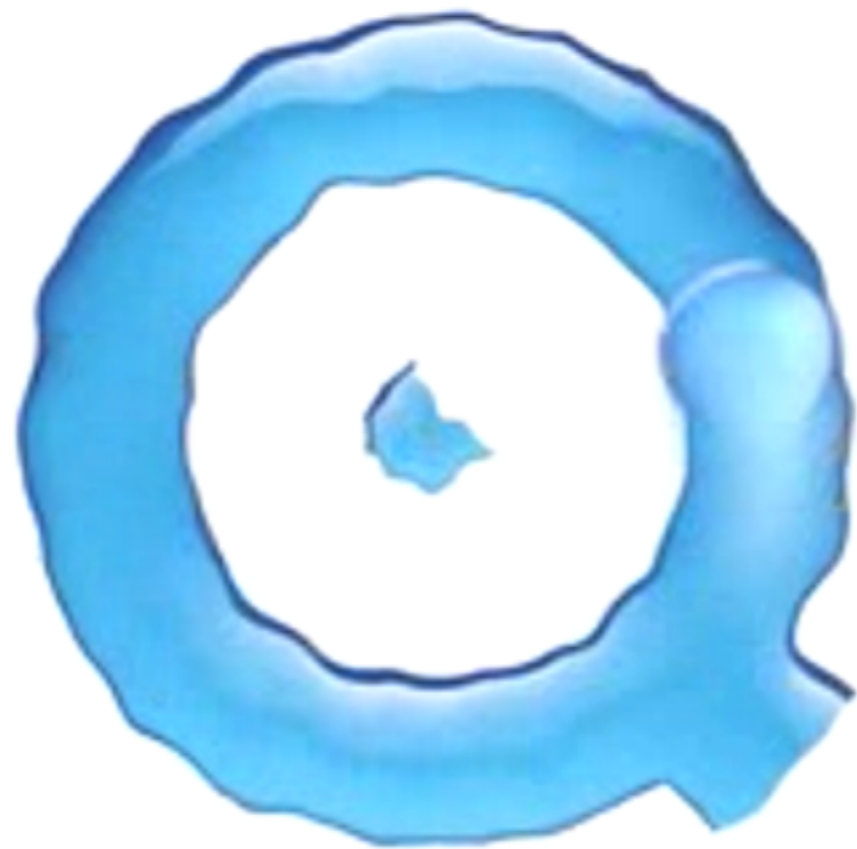
```
AVPlayerItem *item3 = [[AVPlayerItem alloc] initWithURL:url3];
```

```
NSArray *items = @[ item1, item2, item3 ];
```

```
self.player = [[AVQueuePlayer alloc] initWithItems:items];
```

```
[self.player play];
```

Воспроизведение видео



QuickTime



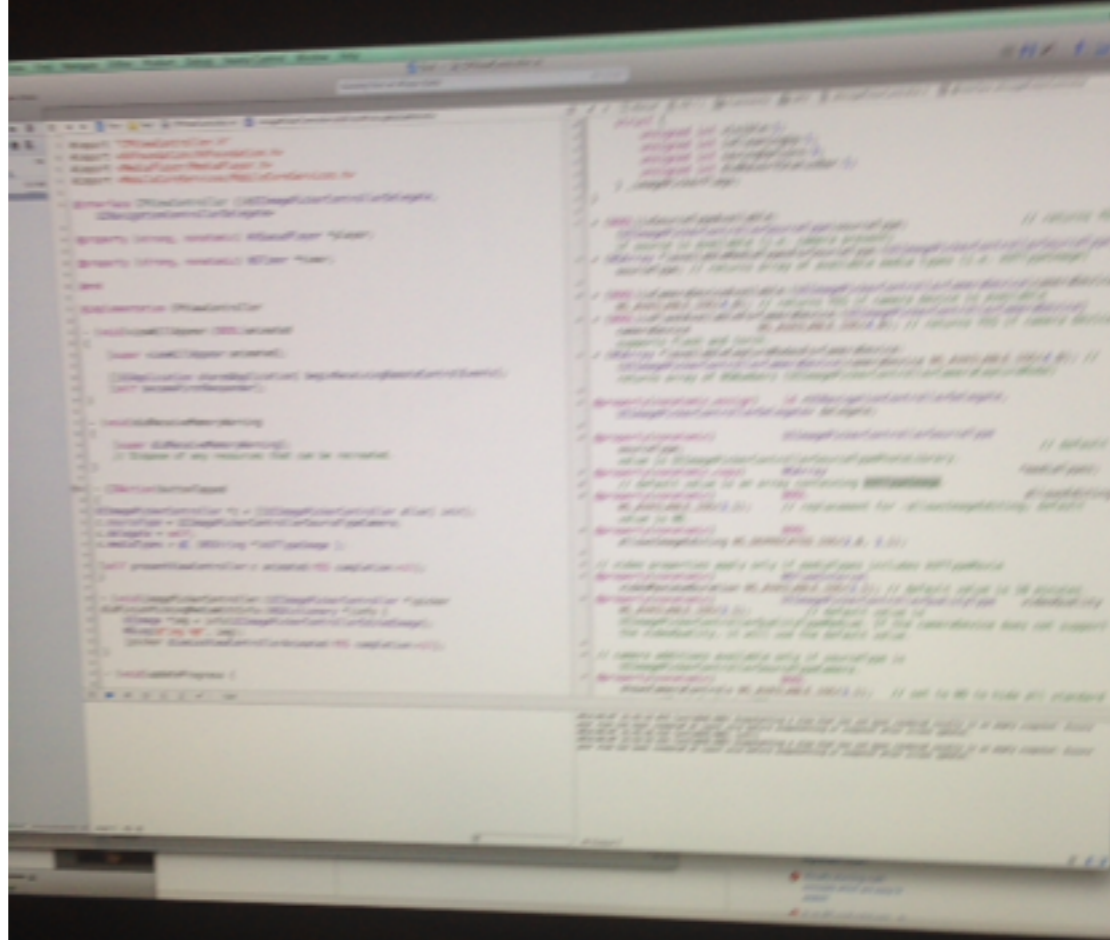

```
- (IBAction)buttonTapped
{
    NSURL *url = [[NSBundle mainBundle] URLForResource:@"sample"
                                                    withExtension:@"mp4"];

    MPMoviePlayerViewController *vc = [[MPMoviePlayerViewController alloc]
                                        initWithContentURL:url];

    [self presentViewControllerAnimated:vc];
}
```

ФОТО И ВИДЕО

⚡ Auto



Cancel



ФОТО С КАМЕРЫ

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

```
UIImagePickerController *c = [[UIImagePickerController alloc] init];  
c.sourceType = UIImagePickerControllerSourceTypeCamera;  
c.delegate = self;  
c.mediaTypes = @[ (NSString *)kUTTypeImage ];
```

```
[self presentViewController:c animated:YES completion:nil];
```

```
- (void)imagePickerController:(UIImagePickerController *)picker  
didFinishPickingMediaWithInfo:(NSDictionary *)info {  
    UIImage *img = info[UIImagePickerControllerEditedImage];  
    NSLog(@"img %@", img);  
    [picker dismissViewControllerAnimated:YES completion:nil];  
}
```

Фото из библиотеки

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

```
UIImagePickerController *c = [[UIImagePickerController alloc] init];  
c.delegate = self;  
c.mediaTypes = @[ (NSString *)kUTTypeImage ];  
  
[self presentViewController:c animated:YES completion:nil];
```

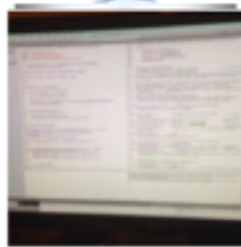
Photos

Cancel



Camera Roll

1 504



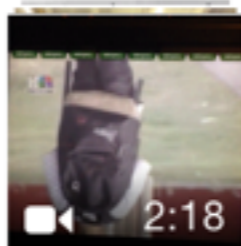
My Photo Stream

1 000



Panoramas

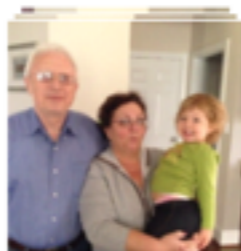
9



Shared by You



Shared by Оля Кондратьева



Америка

Shared by You



Видео с камеры

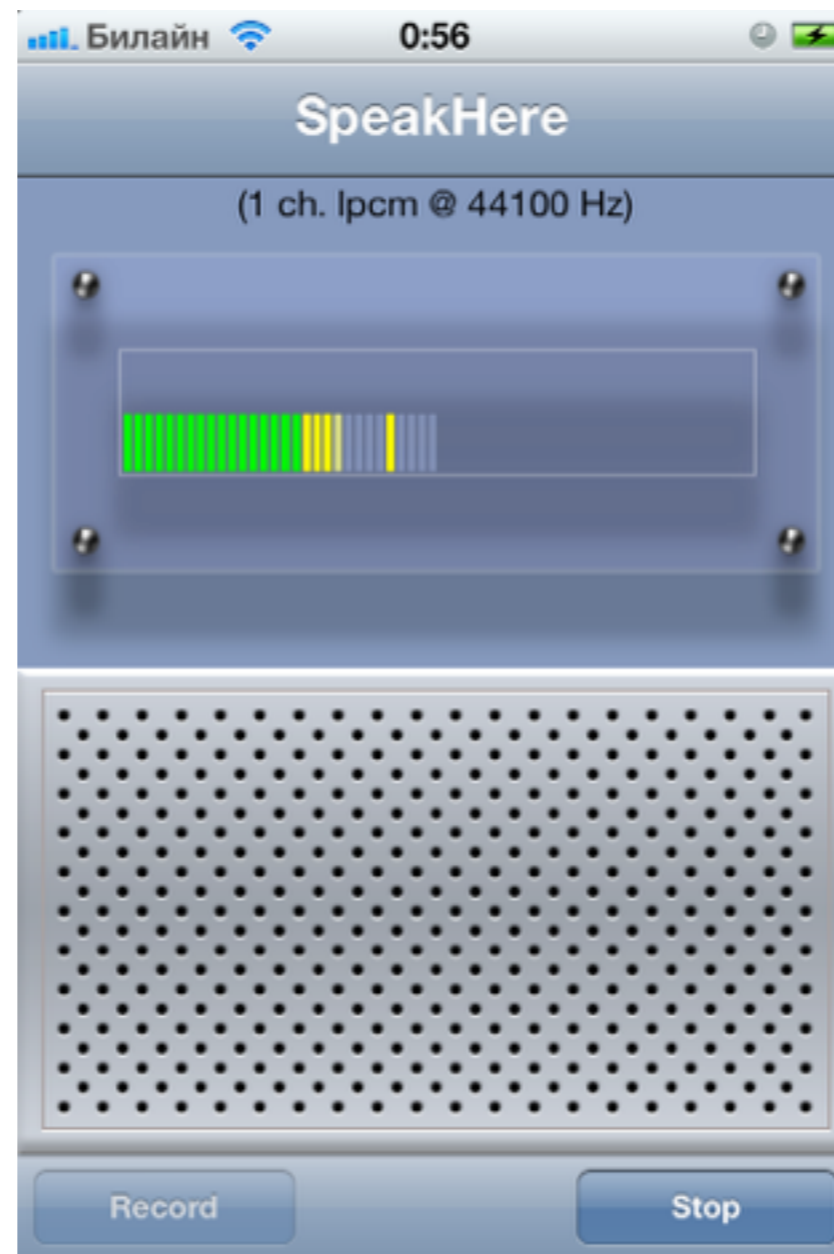
```
UIImagePickerController *c = [[UIImagePickerController alloc] init];  
c.sourceType = UIImagePickerControllerSourceTypeCamera;  
c.delegate = self;  
c.mediaTypes = @[ (NSString *)kUTTypeMovie ];
```

```
- (void)imagePickerController:(UIImagePickerController *)picker  
didFinishPickingMediaWithInfo:(NSDictionary *)info {  
    NSURL *videoUrl = [info objectForKey:UIImagePickerControllerMediaURL];  
    NSLog(@"%@", videoUrl);  
    [picker dismissViewControllerAnimated:YES completion:nil];  
}
```

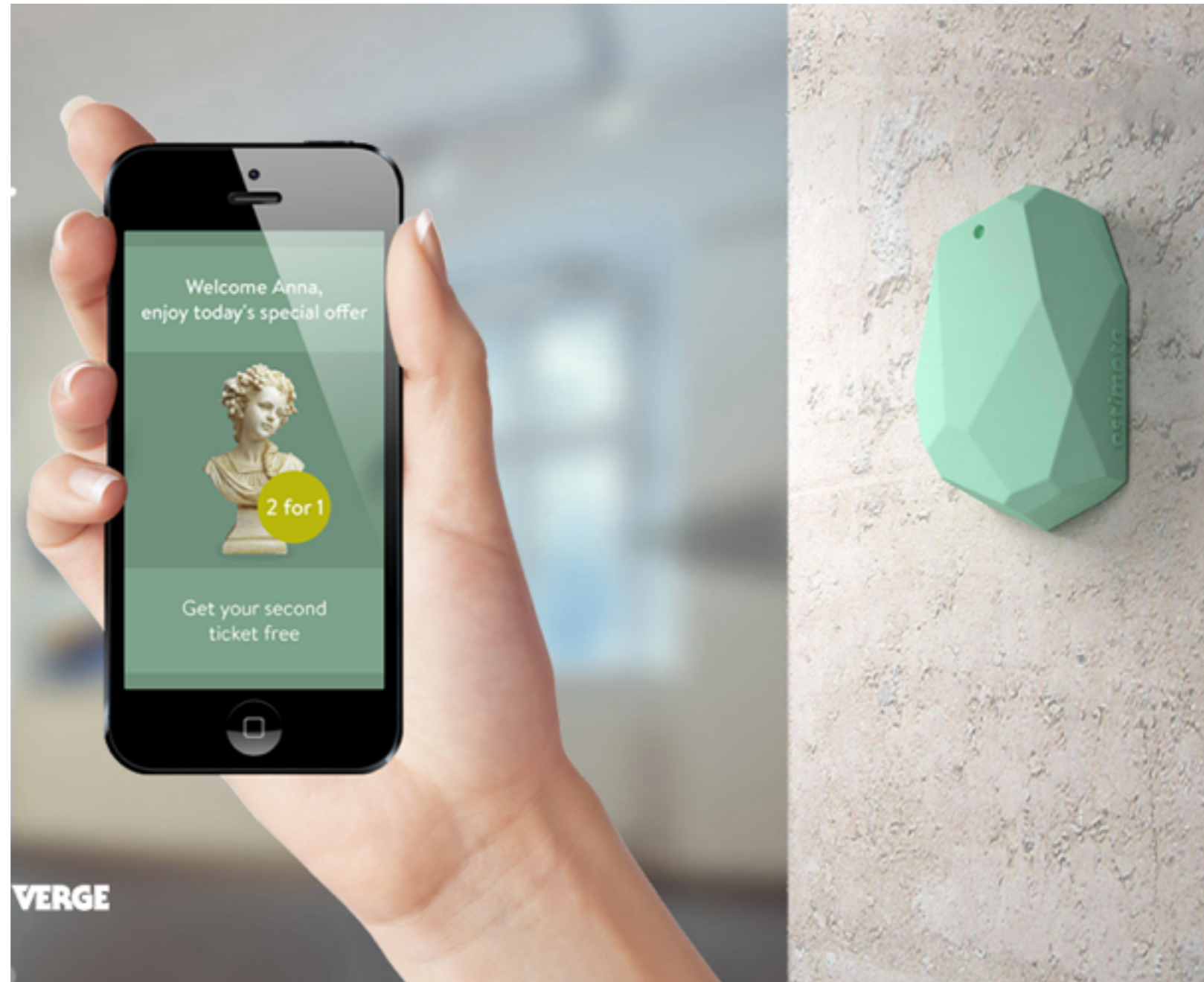
Работа с микрофоном

SpeakHere

<http://developer.apple.com/library/ios/#samplecode/SpeakHere/Introduction/Intro.html>



iBeacon

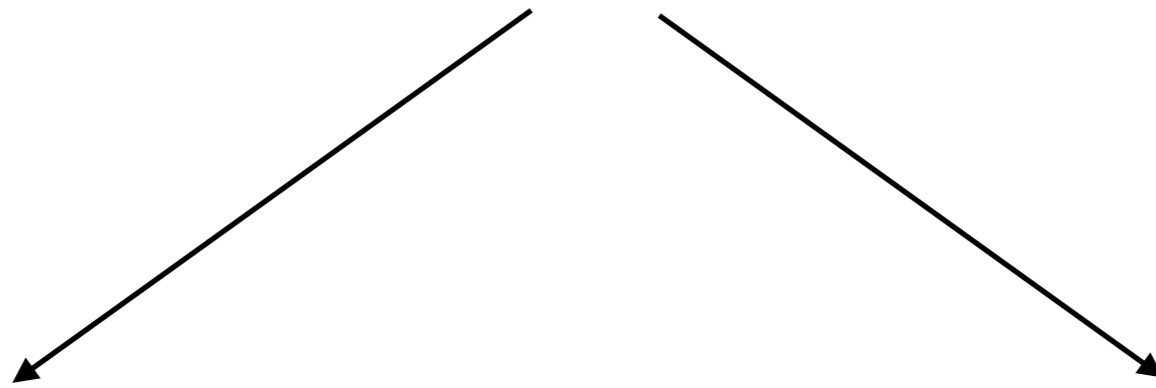


<http://habrahabr.ru/company/touchinstinct/blog/195104/>

Домашнее задание

- продумать, что из пройденного можно использовать в вашем приложении
- добавить поддержку
- если ничего — поиграться в отдельном приложении

Уведомления



local
notifications

remote
notifications

Local

```
+ (void)addNotificationForDate:(NSDate *)date
{
    UILocalNotification *n = [[UILocalNotification alloc] init];
    n.fireDate = date;

    n.timeZone = [NSTimeZone defaultTimeZone];
    n.alertBody = @"Уведомление";
    n.alertAction = @"Открыть";
    n.soundName = UILocalNotificationDefaultSoundName;
    n.applicationIconBadgeNumber = 1;

    n.userInfo = @{ @"custom_id" : @12 };

    [[UIApplication sharedApplication] scheduleLocalNotification:n];
}
```

B AppDelegate:

```
- (BOOL)application:(UIApplication *)application
    didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
{
    ...

    UILocalNotification *n =
        launchOptions[UIApplicationLaunchOptionsLocalNotificationKey];
    if (n)
    {
        // реагируем на уведомление
    }
    ...
}

- (void)application:(UIApplication *)application
    didReceiveLocalNotification:(UILocalNotification *)notification
{
    if (application.applicationState == UIApplicationStateActive)
        return;

    NSNumber *num = n.userInfo[@"custom_id"];
    // реагируем на уведомление
}
```

Remote

apns-php

<https://code.google.com/p/apns-php/>

ruby apns

<https://github.com/jpoz/APNS>

B AppDelegate:

```
- (BOOL)application:(UIApplication *)application
    didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
{
    [UIApplication.sharedApplication registerForRemoteNotificationTypes:
        (UIRemoteNotificationTypeAlert
         | UIRemoteNotificationTypeSound
         | UIRemoteNotificationTypeBadge)];

    ...
}

- (void)application:(UIApplication *)application
    didRegisterForRemoteNotificationsWithDeviceToken:(NSData *)deviceToken
{
    // отправляем токен на сервер
}
```



```
- (BOOL)application:(UIApplication *)application
    didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
{
    ...

    NSDictionary *userInfo =
        launchOptions[UIApplicationLaunchOptionsRemoteNotificationKey];
    if (userInfo)
    {
        // тут реагируем на уведомление
    }

    ...
}
```

```
- (void)application:(UIApplication *)application
    didReceiveRemoteNotification:(NSDictionary *)userInfo
{
    // если было активно - ничего не делаем
    if (application.applicationState == UIApplicationStateActive)
    {
        return;
    }

    // и тут реагируем на уведомление
}
```

Распознавание жестов

UIGestureRecognizer

UIView:

- (void)addGestureRecognizer:([UIGestureRecognizer](#) *)gestureRecognizer

UITapGestureRecognizer

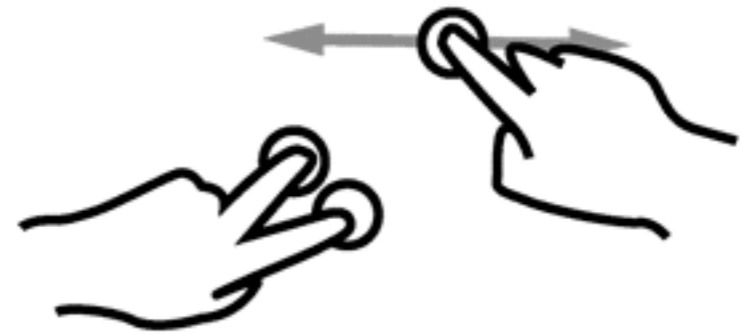
```
UITapGestureRecognizer *t = [[UITapGestureRecognizer alloc]
                               initWithTarget:self
                               action:@selector(tapped)];

t.numberOfTapsRequired = 1;
t.numberOfTouchesRequired = 1;
[self.view addGestureRecognizer:t];
```



UIPanGestureRecognizer

```
UIPanGestureRecognizer *p = [[UIPanGestureRecognizer alloc]
                               initWithTarget:self
                               action:@selector(pan:)];
[self.view addGestureRecognizer:p];
```



```
- (void)pan:(UIPanGestureRecognizer *)sender
{
    CGPoint t = [sender translationInView:self.view];
    CGPoint v = [sender velocityInView:self.view];
    NSLog(@"%@ ", NSStringFromCGPoint(t));
    NSLog(@"%@ ", NSStringFromCGPoint(v));
}
```

UIPinchGestureRecognizer

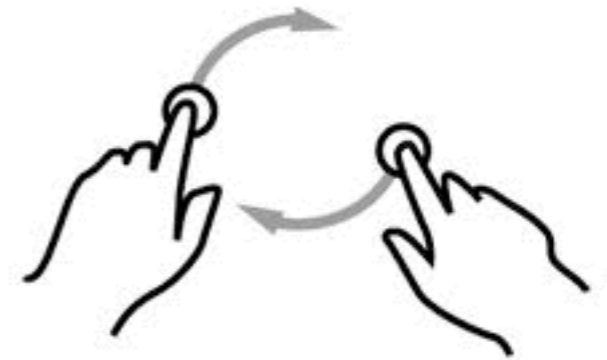
```
UIPinchGestureRecognizer *pinch = [[UIPinchGestureRecognizer alloc]
                                     initWithTarget:self
                                     action:@selector(pinch:)];
[self.view addGestureRecognizer:pinch];
```



```
- (void)pinch:(UIPinchGestureRecognizer *)sender
{
    CGFloat scale = sender.scale;
    NSLog(@"%g", scale);
}
```

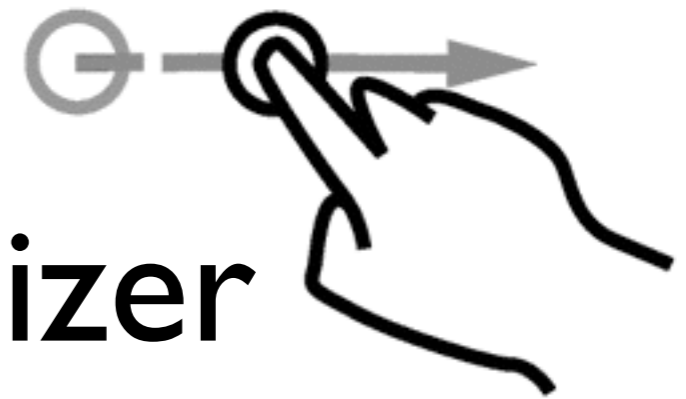
UIRotationGestureRecognizer

```
UIRotationGestureRecognizer *r = [[UIRotationGestureRecognizer alloc]
                                   initWithTarget:self
                                   action:@selector(rotate:)];
[self.view addGestureRecognizer:r];
```



```
- (void)rotate:(UIRotationGestureRecognizer *)sender
{
    CGFloat r = sender.rotation;
    NSLog(@"%g", r);
}
```

```
UISwipeGestureRecognizer *s = [[UISwipeGestureRecognizer alloc]
                               initWithTarget:self
                               action:@selector(pan:)];
s.direction = UISwipeGestureRecognizerDirectionLeft;
[self.view addGestureRecognizer:s];
```



- UISwipeGestureRecognizer
- UILongPressGestureRecognizer

```
UILongPressGestureRecognizer *p = [[UISwipeGestureRecognizer alloc]
                                    initWithTarget:self
                                    action:@selector(pan:)];
p.minimumPressDuration = 0.5;
[self.view addGestureRecognizer:p];
```



UIGestureRecognizerDelegate

- (BOOL)gestureRecognizerShouldBegin:(UIGestureRecognizer *)gestureRecognizer;
- (BOOL)gestureRecognizer:(UIGestureRecognizer *)gestureRecognizer
shouldRecognizeSimultaneouslyWithGestureRecognizer:
(UIGestureRecognizer *)otherGestureRecognizer;
- (BOOL)gestureRecognizer:(UIGestureRecognizer *)gestureRecognizer
shouldReceiveTouch:(UITouch *)touch;

UIGestureRecognizerState

```
typedef enum {  
    UIGestureRecognizerStatePossible,  
  
    UIGestureRecognizerStateBegan,  
  
    UIGestureRecognizerStateChanged,  
  
    UIGestureRecognizerStateEnded,  
  
    UIGestureRecognizerStateCancelled,  
  
    UIGestureRecognizerStateFailed,  
  
    UIGestureRecognizerStateRecognized = UIGestureRecognizerStateEnded  
} UIGestureRecognizerState;
```

Всё!

Глеб Тарасов

gleb34@gmail.com

twitter.com/pilot34