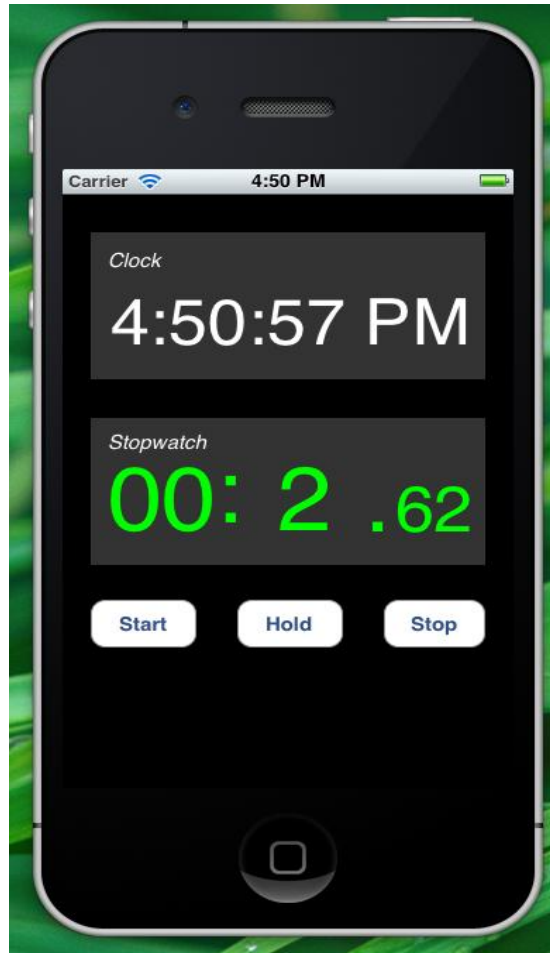


## BÀI TẬP ỨNG DỤNG NSTIMER

**Yêu cầu:** Tạo ra một cái đồng hồ bấm giờ với các chức năng cơ bản xem giờ và bấm giờ.



### Phân tích :

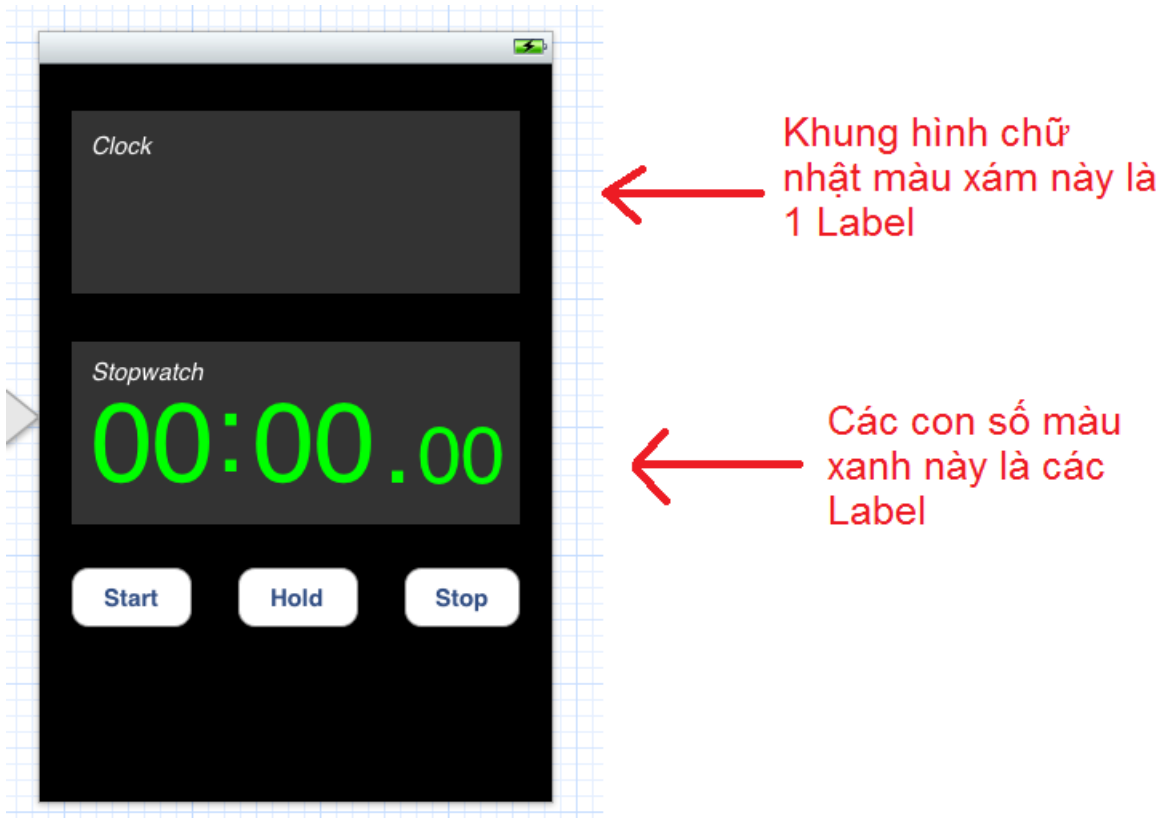
- Để có được giao diện trên ta cần các Label và các Button để trình chiếu và các nút button để điều khiển.

### Chức năng :

- Khi người dùng click vào nút Start thì đồng hồ bắt đầu chạy.
- Khi người dùng click vào nút Hold thì đồng hồ tạm dừng.
- Khi click nút Stop thì đồng hồ trở về ban đầu. Sẵn sàng chạy tiếp khi click Start.

## 1. Thiết kế giao diện

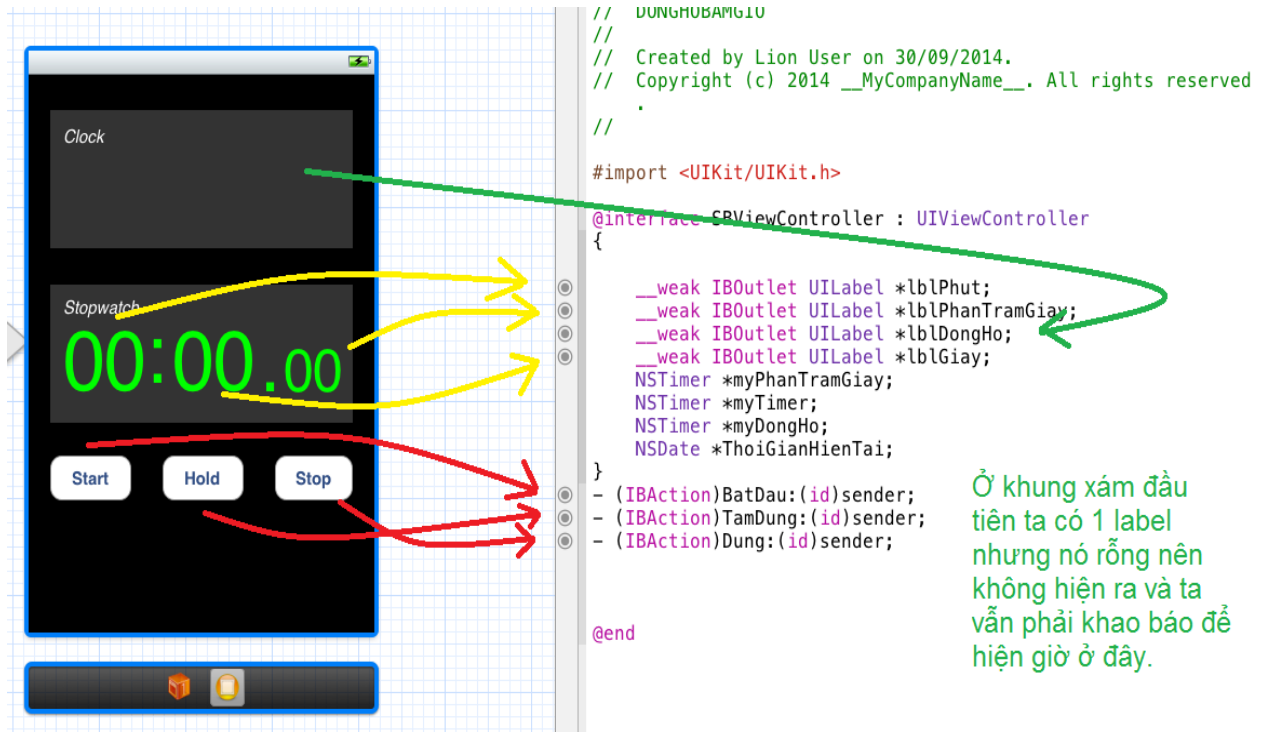
- Mô tả ứng dụng: Chúng ta sẽ tạo ra một giao diện với **Round Rec Button** và các **Label**.
- Ta tạo giao diện như hình dưới. Kéo **Round Rec Button** và **Label** vào storyboard như hình dưới:



- Chỉnh màu nền là màu đen , kéo các Label vào và chỉnh màu xám. Ta kéo chồng các **Label** đè lên nhau để thiết kế.

## 2. Khai Báo

- Ta khai báo các UI trong file .h:



```
// DUNGHUBAMG10
//
// Created by Lion User on 30/09/2014.
// Copyright (c) 2014 __MyCompanyName__. All rights reserved
//
#import <UIKit/UIKit.h>

@interface CBViewController : UIViewController
{
    __weak IBOutlet UILabel *lblPhut;
    __weak IBOutlet UILabel *lblPhanTramGiay;
    __weak IBOutlet UILabel *lblDongHo;
    __weak IBOutlet UILabel *lblGiay;
    NSTimer *myPhanTramGiay;
    NSTimer *myTimer;
    NSTimer *myDongHo;
    NSDate *ThoiGianHienTai;
}
- (IBAction)BatDau:(id)sender;
- (IBAction)TamDung:(id)sender;
- (IBAction)Dung:(id)sender;

@end
```

Ở khung xám đầu tiên ta có 1 label nhưng nó rỗng nên không hiện ra và ta vẫn phải khai báo để hiện giờ ở đây.

## 3. Viết code xử lý

- Ta qua file .m, viết code vào hàm **ViewDidLoad** và các hàm vừa tạo như sau:

```
- (void)viewDidLoad
{
    [super viewDidLoad];
    // Do any additional setup after loading the view,
    typically from a nib.
    [self ThoiGianCuaDongHo];
}
```

- Sau đó viết các đoạn code sau:
- Đoạn code này giúp ta định dạng thời gian và hiện thông tin về thời gian ra Label

```

-(void)ThoiGianCuaDongHo
{
    [myDongHo invalidate];
    myDongHo = nil;
    ThoiGianHienTai = [NSDate date];
    NSDateFormatter *DinhDangThoiGian = [[NSDateFormatter alloc] init];
    [DinhDangThoiGian setTimeStyle:NSDateFormatterMediumStyle];
    lblDongHo.text = [DinhDangThoiGian stringFromDate:ThoiGianHienTai];
    myDongHo = [NSTimer scheduledTimerWithTimeInterval:0.01 target:self selector:@selector
        (ThoiGianCuaDongHo) userInfo:nil repeats:YES];
}

```

- Đoạn code này tạo hàm Kích hoạt, khi hàm Kích hoạt chạy sẽ chạy đoạn code NSTimer dẫn đến gọi hàm Xử Lý chạy
- Theo khoảng thời gian 0.01 giây sẽ gọi hàm Xử Lý.

```

-(void)KichHoat
{
    myPhanTramGiay = [NSTimer scheduledTimerWithTimeInterval:0.01 target:self selector:@selector(XuLy)
        userInfo:nil repeats:YES];
}

```

- Đoạn code này thực hiện khi ta nhấn nút Start sẽ gọi hàm Kích hoạt.

```

- (IBAction)BatDau:(id)sender {
    [self KichHoat];
}

```

- Đoạn code này sẽ thực hiện khi ta nhấn nút Hold hay Stop

```

- (IBAction)TamDung:(id)sender {
    [myPhanTramGiay invalidate];
}

- (IBAction)Dung:(id)sender {
    [myPhanTramGiay invalidate];
    lblPhanTramGiay.text = @"00";
    lblGiay.text = @"00";
    lblPhut.text = @"00";
}

-(void)XuLy
{
    int phantramgiay = [lblPhanTramGiay.text intValue];
    int phantramgiaymoi = phantramgiay + 1 ;
    lblPhanTramGiay.text = [NSString stringWithFormat:@"%i",
        phantramgiaymoi];
    if ([lblPhanTramGiay.text intValue] == 99) {
        int ThoiGian = [lblGiay.text intValue];
        int ThoiGianChay = ThoiGian + 1 ;
        lblGiay.text = [NSString stringWithFormat:@"%i",ThoiGianChay];
        if ([lblGiay.text intValue] == 59) {
            int Phut = [lblPhut.text intValue];
            int PhutChay = Phut + 1;
            lblPhut.text = [NSString stringWithFormat:@"%i",PhutChay];
            [myPhanTramGiay invalidate];
            lblPhanTramGiay.text =@"00";
            lblGiay.text = @"00";
        }
        [myPhanTramGiay invalidate];
        lblPhanTramGiay.text =@"00";
        [self KichHoat];
    }
}

```

Sau đó tiến hành chạy thử và kiểm tra.

