



Lập trình ứng dụng trên iOS

Multitasking – Background – Thread

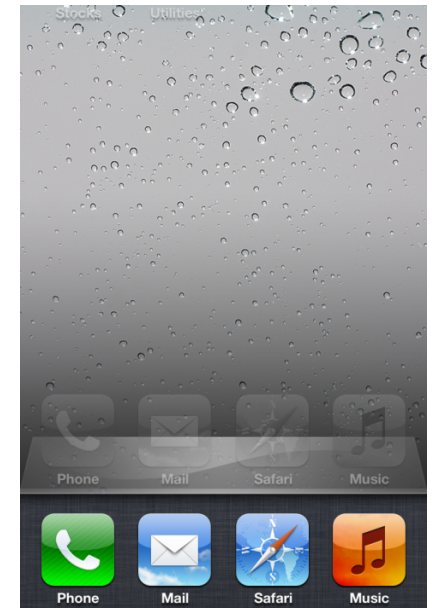


Nội dung

- Mutitasking
- Background
- Thread

Multitasking

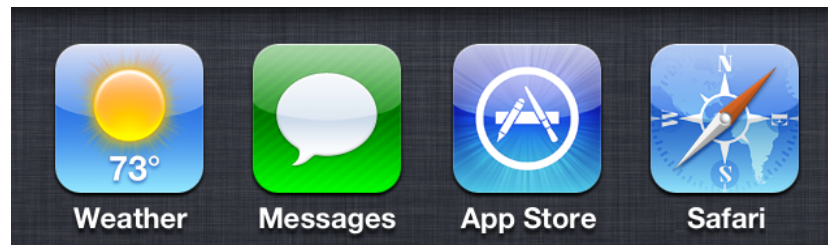
- Multitasking là một chức năng của iOS cho phép các ứng dụng thực hiện các tác vụ trong chế độ background (trong khi người dùng đang sử dụng một ứng dụng khác).
- Multitasking được hỗ trợ trên:
 - iPhone 3GS +
 - iPod touch (3rd generation +)
 - iPad (iOS 4+)





Multitasking

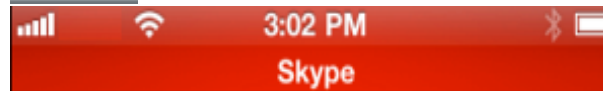
- Các bước để chuyển nhanh giữa các ứng dụng
 - Double-click nút Home để hiện thị các ứng dụng hiện tại đang sử dụng



- Swipe sang trái hoặc phải để xác định vị trí ứng dụng bạn muốn
- Tap vào ứng dụng để chuyển nó sang foreground

Multitasking

- Các ứng dụng multitasking trong iOS 4
 - Chuyển nhanh giữa các ứng dụng
 - Push và Local notifications
 - Background audio
 - Location tracking
 - Voice over IP





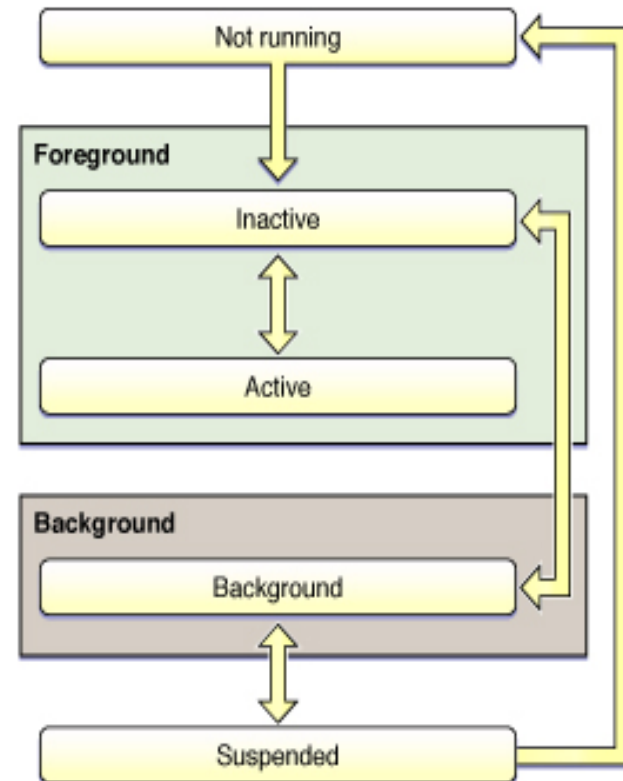
Multitasking

- Kiểm tra hỗ trợ multitasking

```
19 -(BOOL)isMultitaskingOS
20 {
21     BOOL bgSupport=NO;
22     if ([[UIDevice currentDevice] respondsToSelector:@selector(isMultitaskingSupported)])
23         bgSupport=[UIDevice currentDevice].multitaskingSupported;
24     return bgSupport;
```

Background

- iPhone OS không hỗ trợ ứng dụng background
- iOS 4+ hỗ trợ ứng dụng background





Background

- Kiểm tra trạng thái của ứng dụng lúc runtime

```
28 -(BOOL)isForeground
29 {
30     if (![self isMultitaskingOS])
31         return YES;
32     UIApplicationState state = [UIApplication sharedApplication].applicationState;
33     //return (state==UIApplicationStateActive || state==UIApplicationStateInactive );
34     return (state==UIApplicationStateActive);
35 }
```




Thread

- Thread Costs

| item | Approximate cost |
|------------------------|-------------------------------|
| Kernel data structures | Approximately 1 KB |
| Stack space | 1 MB (iOS main thread) |
| | 512 KB (secondary threads) |
| Creation time | Approximately 90 microseconds |



Thread

- Tạo thread
 - Sử dụng NSThread
 - Sử dụng POSIX Thread
 - Sử dụng NSObject để spawn Thread



Thread

- Tạo thread

- Sử dụng NSThread

- + (void)detachNewThreadSelector:(SEL)aSelector toTarget:(id)aTarget withObject:(id)anArgument

- + start

- Ví dụ 1:

- ```
[NSThread detachNewThreadSelector:@selector(myThreadMainMethod:) toTarget:self withObject:nil];
```

- Ví dụ 2:

- ```
NSThread* myThread = [[NSThread alloc] initWithTarget:self selector:  
    @selector(myThreadMainMethod:) object:nil];  
[myThread start]; // Actually create the thread
```

Thread

- Tạo thread
 - Sử dụng POSIX Thread

```
void* PosixThreadMainRoutine(void* data)
{
    // Do some work here.
    return NULL;
}
```

```
void LaunchThread()
{
    // Create the thread using POSIX routines.
    pthread_attr_t attr;
    pthread_t      posixThreadID;
    int            returnVal;
    returnVal = pthread_attr_init(&attr);
    assert(!returnVal);
    returnVal = pthread_attr_setdetachstate(&attr, PTHREAD_CREATE_DETACHED)
        ;
    assert(!returnVal);
    int threadError = pthread_create(&posixThreadID, &attr, &
        PosixThreadMainRoutine, NULL);
    returnVal = pthread_attr_destroy(&attr);
    assert(!returnVal);
    if (threadError != 0)
    {
        // Report an error.
    }
}
```



Thread

- Tạo thread
 - Sử dụng NSObject để spawn Thread
 - (void)performSelector:(SEL)aSelector withObject:
(id)anArgument afterDelay:(NSTimeInterval)delay inModes:
(NSArray *)modes
- ```
[myObj performSelectorInBackground:@selector(doSomething) withObject:nil];
```



# Thread

- Thiết lập thuộc tính của Thread
  - Stack size
    - (void)setStackSize:(NSUInteger)s  
pthread\_attr\_setstacksize → **POSIX**
  - Storage
    - (NSMutableDictionary \*)threadDictionary  
pthread\_setspecific  
pthread\_getspecific } **POSIX**
  - Detached State
    - int pthread\_detach(pthread\_t thread) → **POSIX**
  - Priority
    - (void)setThreadPriority:(double)priority  
pthread\_setschedparam → **POSIX**



# Demo