

Delegates and Web Requests

IAP 2010 ❄️

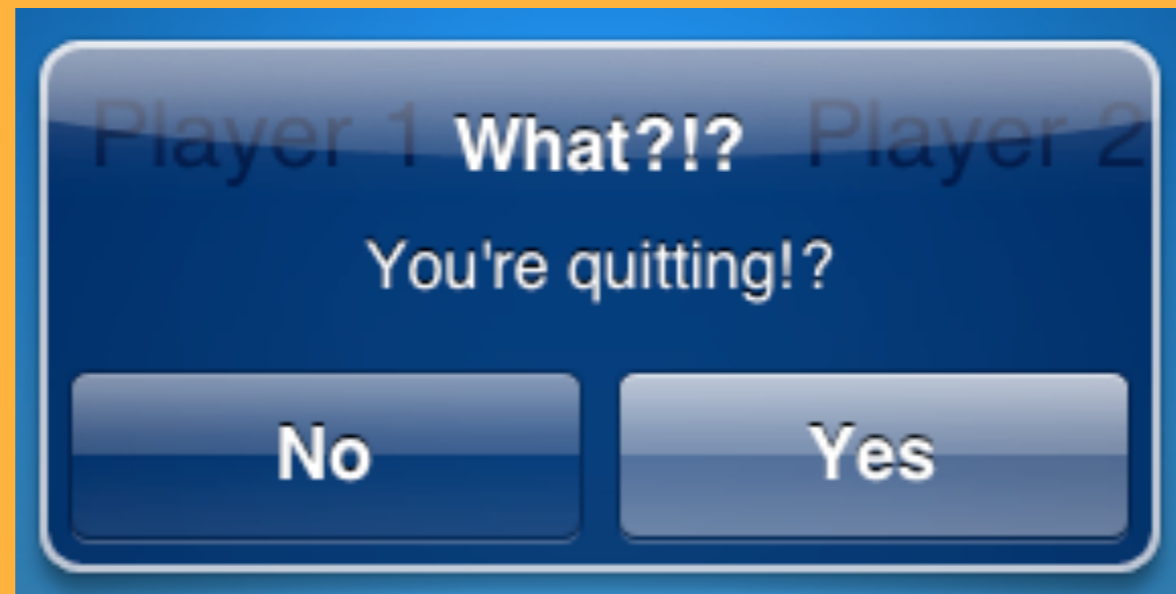
iphonedev.csail.mit.edu

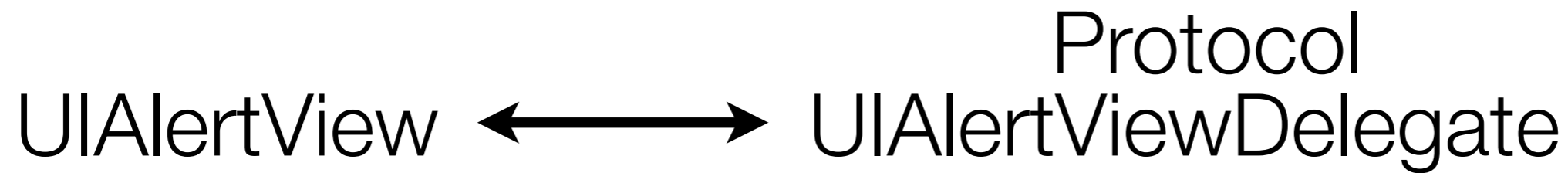
edward benson / eob@csail.mit.edu

Today

- Alerts
- WebRequests

Alerts





```
-initWithTitle:  
    message:  
    delegate:  
cancelButtonTitle:  
otherButtonTitles:  
  
- show
```

Responding to Actions

```
- UIAlertView:clickedButtonAtIndex:
```

Customizing Behavior

```
- willPresentAlertView:
```

```
- didPresentAlertView:
```

```
- UIAlertView:willDismissWithButtonIndex:
```

```
- UIAlertView:didDismissWithButtonIndex:
```

Canceling

```
- UIAlertViewCancel:
```

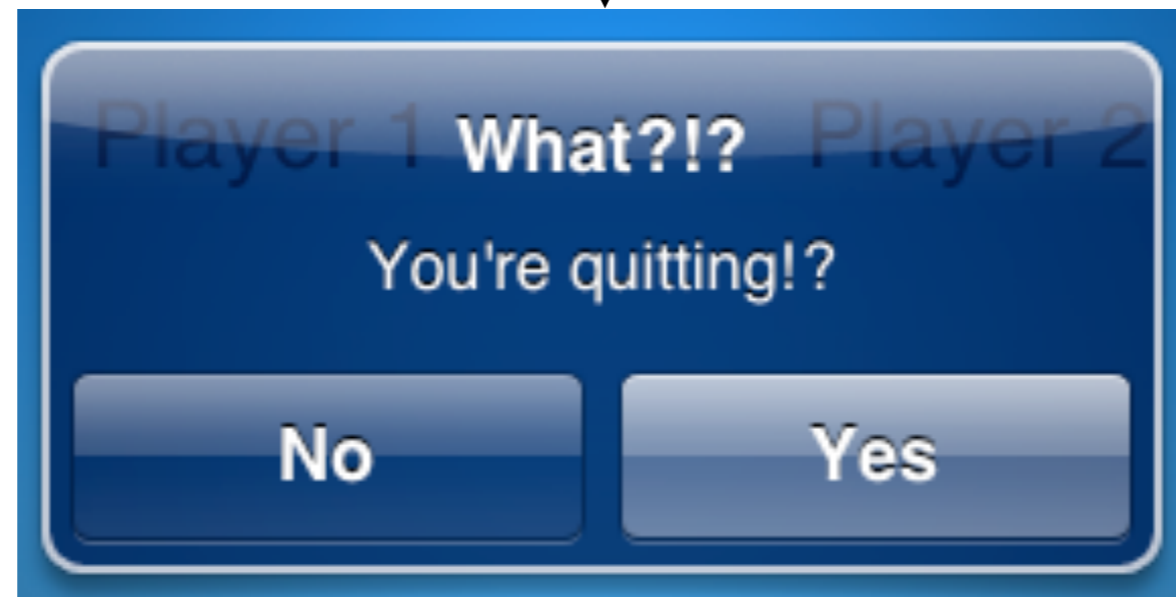
Showing the Alert

```
-initWithTitle: @"Are you kidding!"  
    message: @"Do you really want to quit RPS?"  
    delegate: self  
cancelButtonTitle: @"No"  
otherButtonTitles: @"Yes", nil
```

```
-initWithTitle: @"BOO!"  
    message: nil  
    delegate: self  
cancelButtonTitle: @"OK"  
otherButtonTitles: nil
```

Showing the Alert

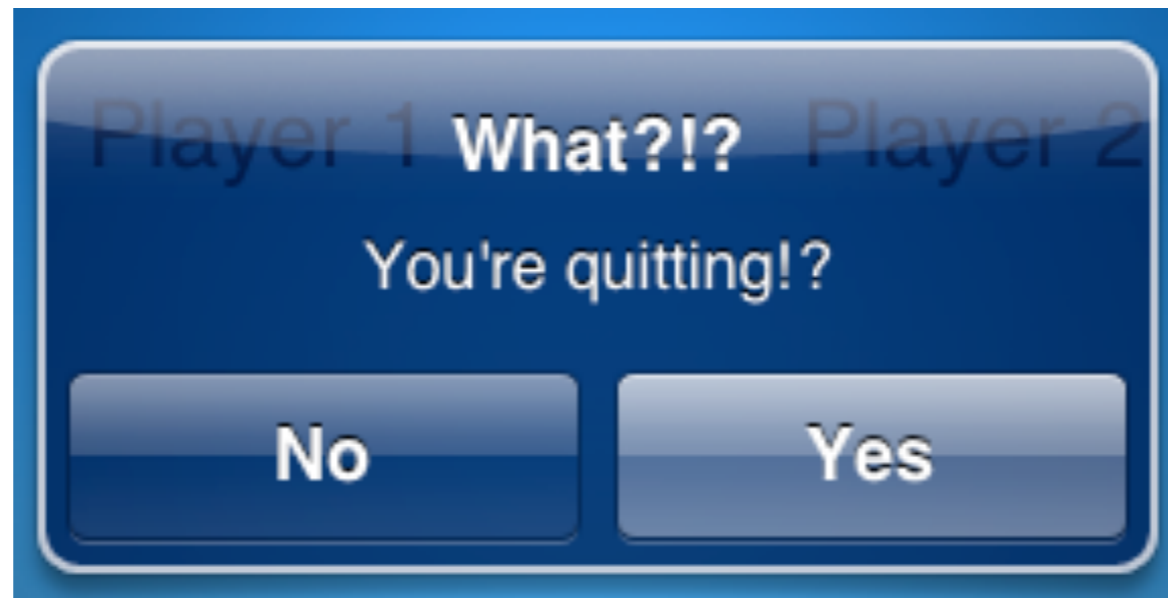
```
-initWithTitle: @"Are you kidding!"  
    message: @"Do you really want to quit RPS?"  
    delegate: self  
cancelButtonTitle: @"No"  
otherButtonTitles: @"Yes", nil
```



```
- (void)alertView:(UIAlertView *)alertView clickedButtonAtIndex:(NSInteger)buttonIndex
```

Conform to the Alert Protocol

```
@interface RPSGameViewController : UIViewController <UIAlertViewDelegate>
```



```
- (void)alertView:(UIAlertView *)alertView clickedButtonAtIndex:(NSInteger)buttonIndex {  
    // 1 index is "Yes" button  
    if(buttonIndex == 1) {  
        [self resetGame];  
    }  
}
```

Right now, we reset the game like this

RPSGameViewController.m

```
- (IBAction)resetGameClicked {
    [self resetGame];
}

-(IBAction)resetGame {
    outcomeLabel.text = nil;
    myMoveLabel.text = nil;
    opponentMoveLabel.text = nil;
    opponentNameLabel.text = @"Player 2";

    rockButton.enabled = YES;
    paperButton.enabled = YES;
    scissorsButton.enabled = YES;
}
```

Let's get a confirmation message first

So Change resetGameClicked to show an alert instead

```
- (IBAction)resetGameClicked {  
    [self resetGame];  
}
```



```
- (IBAction)resetGameClicked {  
    UIAlertView *alert = [[UIAlertView alloc] initWithTitle:@"Really?"  
                                                         message:@"Really reset?"  
                                                         delegate:self  
                                                         cancelButtonTitle:@"No"  
                                                         otherButtonTitles:@"Yes", nil];  
  
    [alert show];  
    [alert release];  
}
```

But now we have to respond to the alert

```
@interface RPSGameViewController :  
UIViewController <UIAlertViewDelegate> {
```

But now we have to respond to the alert

RPSGameViewController.m

```
- (void)alertView:(UIAlertView *)alertView clickedButtonAtIndex:(NSInteger)buttonIndex {  
    if (buttonIndex == 1) {  
        [self resetGame];  
    }  
}
```

buttonIndex is the index of the button clicked,
from left to right



Delegates

App

Tables

Alerts

GameKit

(very important pattern to notice)

Web Connections

Anatomy of a Web API.

Synchronous v. Asynchronous

What would a delegate for an
HTTP connection look like?

Parsing Responses

RPS Tournament Server

<http://courses.csail.mit.edu/iphonedev/rps.php>

Newcomer plays the last player to have registered a
move a move.

[http://courses.csail.mit.edu/iphonedev/rps.php?
player=Ted&move=rock](http://courses.csail.mit.edu/iphonedev/rps.php?player=Ted&move=rock)

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "ht
<dict>
  <key>opponent</key>
  <string>Grace</string>
  <key>opponentMove</key>
  <string>scissors</string>
</dict>
</plist>
```

A dictionary with two key-value pairs

HTTP Libraries

Looking through the code

Thanks