

Creating Professional Apps with LinkedIn

Kirsten Jones
Developer Advocate
LinkedIn

LinkedIn Platform



- Leading platform for building the professional web
- Professional profile of record
- Committed to building a healthy developer ecosystem

LinkedIn

Developer Relations

- Dedicated to developer success
- Documentation, tutorials, answers and talks
- Enthusiastic team built from industry pros



Getting LinkedIn Data with REST APIs

- Profiles
- Search
- Connections
- Network Updates
- Messages

LinkedIn Authentication

- OAuth
- Industry standard - libraries for most languages
- Application and user credentials
- Combination allows LinkedIn to know who is making the request

Getting Started with iPhone Development

- Open source client written by Lee Whitney, community member
- Uses OAuthConsumer library
- Fully functional application (OAuth flow, profile, network read and write)
- Works with just key and secret
- Excellent starting place

Getting Member Data is Simple

```
- (void)profileApiCall
{
    NSURL *url = [NSURL URLWithString:@"http://api.linkedin.com/v1/people/~"];
    OAMutableURLRequest *request =
        [[OAMutableURLRequest alloc] initWithURL:url
                                             consumer:oAuthLoginView.consumer
                                             token:oAuthLoginView.accessToken
                                             callback:nil
                                             signatureProvider:nil];

    [request setValue:@"json" forHTTPHeaderField:@"x-li-format"];

    ODataFetcher *fetcher = [[ODataFetcher alloc] init];
    [fetcher fetchDataWithRequest:request
                        delegate:self
                        didFinishSelector:@selector(profileApiCallResult:didFinish:)
                        didFailSelector:@selector(profileApiCallResult:didFail:)];
    [request release];
}
```


Getting Member Data is Simple

```
- (void)profileApiCall
{
    NSURL *url = [NSURL URLWithString:@"http://api.linkedin.com/v1/people/~"];
    OAMutableURLRequest *request =
        [[OAMutableURLRequest alloc] initWithURL:url
                                             consumer:oAuthLoginView.consumer
                                             token:oAuthLoginView.accessToken
                                             callback:nil
                                             signatureProvider:nil];

    [request setValue:@"json" forHTTPHeaderField:@"x-li-format"];

    ODataFetcher *fetcher = [[ODataFetcher alloc] init];
    [fetcher fetchDataWithRequest:request
                        delegate:self
      didFinishSelector:@selector(profileApiCallResult:didFinish:)
      didFailSelector:@selector(profileApiCallResult:didFail:)];
    [request release];
}
```



Getting Member Data is Simple

```
- (void)profileApiCall
{
    NSURL *url = [NSURL URLWithString:@"http://api.linkedin.com/v1/people/~"];
    OAMutableURLRequest *request =
        [[OAMutableURLRequest alloc] initWithURL:url
                                             consumer:oAuthLoginView.consumer
                                             token:oAuthLoginView.accessToken
                                             callback:nil
                                             signatureProvider:nil];

    [request setValue:@"json" forHTTPHeaderField:@"x-li-format"];

    ODataFetcher *fetcher = [[ODataFetcher alloc] init];
    [fetcher fetchDataWithRequest:request
                        delegate:self
      didFinishSelector:@selector(profileApiCallResult:didFinish:)
      didFailSelector:@selector(profileApiCallResult:didFail:)];
    [request release];
}
```


Getting Member Data is Simple

```
- (void)profileApiCall
{
    NSURL *url = [NSURL URLWithString:@"http://api.linkedin.com/v1/people/~"];
    OAMutableURLRequest *request =
        [[OAMutableURLRequest alloc] initWithURL:url
                                             consumer:oAuthLoginView.consumer
                                             token:oAuthLoginView.accessToken
                                             callback:nil
                                             signatureProvider:nil];

    [request setValue:@"json" forHTTPHeaderField:@"x-li-format"];

    ODataFetcher *fetcher = [[ODataFetcher alloc] init];
    [fetcher fetchDataWithRequest:request
                        delegate:self
      didFinishSelector:@selector(profileApiCallResult:didFinish:)
      didFailSelector:@selector(profileApiCallResult:didFail:)];
    [request release];
}
```


JSON Results

```
{  
  "headline": "Developer Advocate at LinkedIn",  
  "lastName": "Jones",  
  "firstName": "Kirsten"  
}
```


Parsing Responses with JSON

```
- (void)profileApiCallResult:(OAServiceTicket *)ticket didFinish:(NSData *)data
{
    NSString *responseBody = [[NSString alloc] initWithData:data
                                                                encoding:NSUTF8StringEncoding];

    NSDictionary *profile = [responseBody objectFromJSONString];
    [responseBody release];

    if ( profile )
    {
        name.text = [[NSString alloc] initWithFormat:@"%s %s",
                                                         [profile objectForKey:@"firstName"],
                                                         [profile objectForKey:@"lastName"]];
        headline.text = [profile objectForKey:@"headline"];
    }

    // The next thing we want to do is call the network updates
    [self networkApiCall];
}
```


Demo

Learn More

- Developer portal:
<http://developer.linkedin.com>
- Client at
<https://github.com/synedra/LinkedIn-OAuth-Sample-Client>
- Questions?