# LAB 24

# SOCIAL NETWORKS AND ANALYTICS

## What You Will Learn

- How to create Social Media Accounts and Pages

- How to integrate a variety of Social Media Plugins into your pages.

- How to integrate advertising into your websites

- How to collect stats on your site from popular providers.

## Approximate Time

The exercises in this lab should take approximately 120 minutes to complete.

# Fundamentals of Web Development, 2nd Ed

Randy Connolly and Ricardo Hoar

## Exercise 24.1 — Social Network Home Pages

**1**    This exercise will guide you through the creation of a home page for all 3 social networks covered in this book.

**2**    Defining a Twiiter homepage is probably the least complicated.

https://twitter.com/settings/profile as shown for our twitter account in Figure 24.1 allows you to change many aspects about your twitter homepage. In particular be sure to set a link to your webpage, your bio and set up the photo that is associated with your account on twitter.

Although you can connect with Facebook, we suggest not doing so right now since we will also be setting up a Facebook page.

Note that the URL of you homepage uses your twitter id. Since our ID is funwebdev. Our twitter homepage is twitter.com/funwebdev

*Figure 24.1 The page to change your Twitter homepage's settings*

**3**   Next, log into Facebook to set up a Facebook page. Click on the upper right hand side drop down arrow  (this may change) to bring up the option to "Create Page" as shown in Figure 24.2.
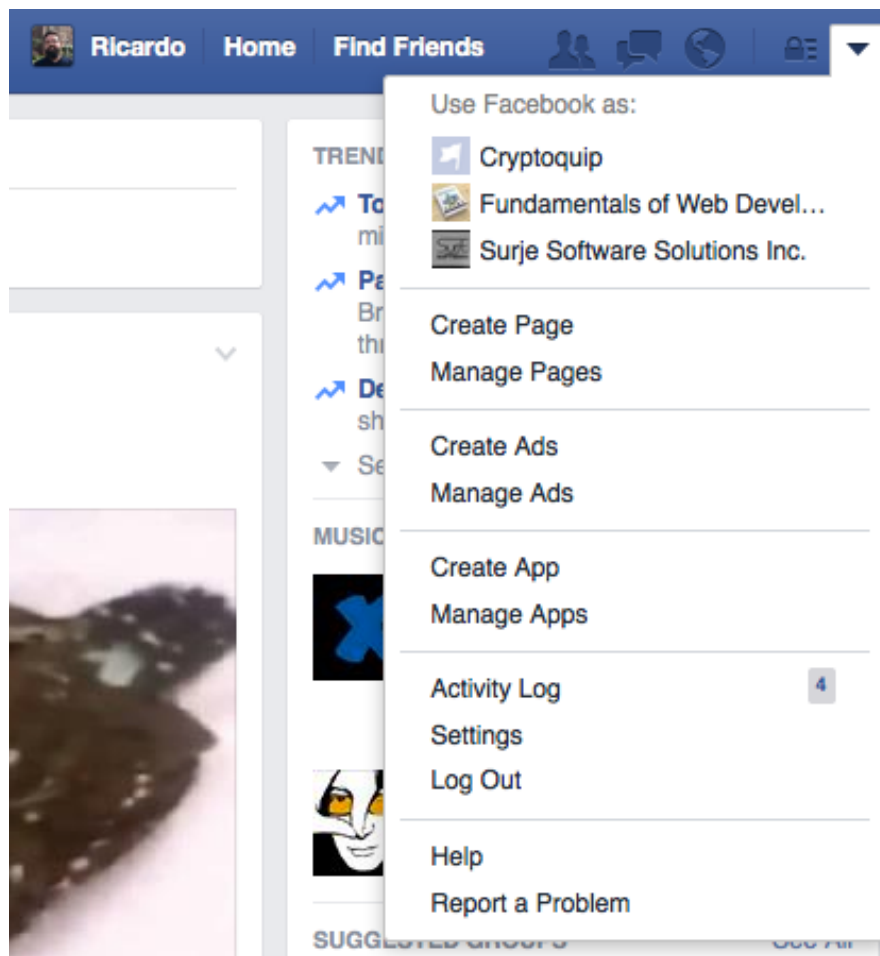


*Figure 24.2 Where to get started creating pages in Facebook*

**4**   This will walk you through a dialog, where you first select what ype of page (business, artist, product, etc), and choose several options about your page.

**5**   Once your page is created you should set up a brief description, and change the Cover and Profile photos to reflect your site just as you would do for your own personal profile.

**6**   Once your page is created you will have the option of logging into facebook as yourself, or switching to your "Page identity" so you can post as the page rather than yourself. As shown in Figure 24.2 you can change identities by clicking the page you created form the list.

7   Creating a google+ Page is relatively straightforward as well. Go to
    https://plus.google.com/u/0/ and log in. In the left you will see a dropdown (may
    change) where you can select a Page as shown in Figure 24.3
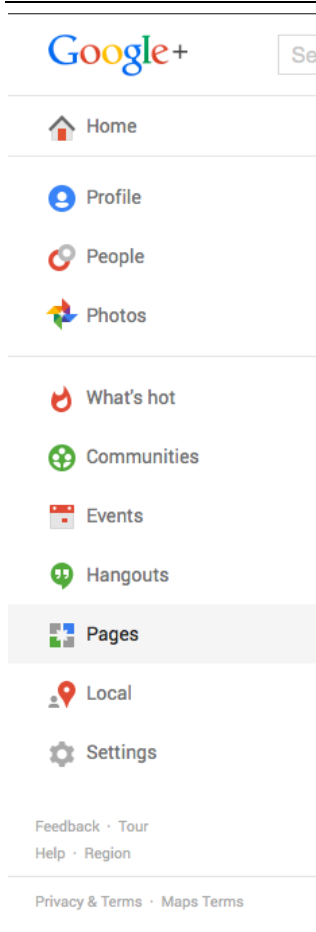


*Figure 24.3 Google+ Page location.*

8   Once you navigate to *Pages* You can click "Create Page" to define a new page for your
    website. There you will be given choices much like Facebook anout what kind of page
    you want to make. Afterwards, you can set up the details of who your are, your web URL,
    and photos for your profile.

9   Finally, the URL of your page (in the URL bar) will begin as a lousy number. To get a
    custom URL click on profile and navigate to Links where you can request a custom URL
    (in our case google.com/+Funwebdev). Be careful through, custom URLs can not be
    changed!

# SOCIAL NETWORK PLUGINS

### Exercise 24.2 — FOLLOW BUTTON

**1**  Choose a place on your website where you want to integrate social media. If you are using WordPress you might consider using a plugin. If you are working on s custom php site, you might create a class or helper function to integrate your social media plugins.

**2**  You need to sign up for a facebook app before you can start integrating the Facebook social plugins.

https://developers.facebook.com/apps/async/create/dialog/

Will walk you through the creation of your app. The details aren't very important, but the creation of the app itself is critical if you want to add plugins.

**3**  Visit https://developers.facebook.com/docs/plugins/follow-button/ to read about the latest documentation on the various follow buttons. Tehre you will see an interactive tool to generate code for your site (it will ask you to select which app id to use in the code.

You will need the URL of your Page created in the last exercise to link to.

You will see a dialog with both the JavaScript and HTML% markup to paste into your site as shown in Figure 24.4

**Your Plugin Code**                                              ✕

| HTML5 | XFBML | IFRAME | URL |

This script uses the app ID of your app    Fundamentals of Web Develop... ▾

Include the JavaScript SDK on your page once, ideally right after the opening <body> tag.

```
<div id="fb-root"></div>
<script>(function(d, s, id) {
  var js, fjs = d.getElementsByTagName(s)[0];
  if (d.getElementById(id)) return;
  js = d.createElement(s); js.id = id;
  js.src = "//connect.facebook.net/en_US/all.js#xfbml=1&
appId=1401923183361422";
  fjs.parentNode.insertBefore(js, fjs);
}(document, 'script', 'facebook-jssdk'));</script>
```

Place the code for your plugin wherever you want the plugin to appear on your page.

```
<div class="fb-follow" data-href="https://www.facebook.com
/fundamentalsofwebdevelopment" data-colorscheme="light" data-layout="standard"
data-show-faces="true"></div>
```

*Figure 24.4 Facebook Follow button code generation dialog*

---

4  In your site add the JavaScript code provided to include Facebook's JavaScript libraries asynchronously. Note: APP_ID should be the ID you received in the last step.

```
$(document).ready(function() {
$.ajaxSetup({ cache: true });
$.getScript('//connect.facebook.net/en_UK/all.js', function(){
FB.init({appId: APP_ID,
channelUrl: $channelURL,
status:true, //status: check fb login
xfbml:true //parse for FB plugins
});
$('#loginbutton,#feedbutton').removeAttr('disabled');
FB.getLoginStatus(updateStatusCallback);
});
});
```

5  Now, in your HTMLpage, where you want to integrate the follow button markup like the following (your will differ slightly, unless you want to make a button to like our page).

```
<fb:follow
href="https://www.facebook.com/fundamentalsOfWebDevelopment"
width="450"
show_faces="true">

</fb:follow>
```

Now, when you refresh your page you will see a familiar follo w widget on the page which will be connected to Facebook.

6  A description of the other Social Network's widgets are provided in Chapter 24. These other plugins/widgets use a similar strategy of logging in to get code to paste on your site. As an exercise integrate Twitter and Google+ into your page as well.

## Exercise 24.3 —Sign up For Ad Network

1  Eventually you will want to see about how you can monetize sites, through ad banners either for yourself or for others.

2  Research and find out what types of ad services you can access (consider Google AdSense, for instance),and register for a ad service.

**3**   Integrate code from the ad service into your sites framework. Consider how a word press site would likely integrate an ad network compared to a custom built site.

# EMAIL MARKETING

### Exercise 24.4 — EMAIL MAILER

**1**   Integrating email capability into your website allows you to leverage one of the most effective communication mediums for marketing, email. Be cautious that email can get you into trouble as many email activities are categorized as spam. Be aware of your local jurisdictions regulations regarding the sending of unsolicited email.

**2**   You may or may not be aware that PHP lets you send email with a simple function call. To see if email is configured correctly create the following script (with your email in the receiver field) and try executing the script to see if the message gets to your inbox.

```php
<?php

mail("<INSERT DESTINATION EMAIL>","Hello","This is the body of the
message");

?>
```

There are many reasons that email might not make it. Email sent from any IP address not associated with a website may look like spam so be sure to check our spam folder.

**3**   Once you confirm that the email is working (again, you may need to test on a production machine to get past spam filters). Create a more powerful mailing script  as shown below:

```php
<?php

function mailform($mailto, $subj, $messageID ,$unsubcode, $accountID){
//define values to use to format the email
$unsubLink="http://funwebdev.com/unsub.php?id=$unsubcode&userID=$accountID"
;
$trackURL="http://example.com/msg=$messageID&userID=$accountID";
$trackImg="http://example.com/img.php?msg=$messageID&userID=$accountID";
$bound = uniqid("FUNWEBDEV_MAIL_EXAMPLE");      //unique boundary string

$rn = "\r\n";
//define a plain (no HTML) footer to illustrate tracking link inclusiopn.
```

```
$plainfooter="$rn$rn$trackURL$rn$rn";
$plainfooter.="--------------------$rn";
$plainfooter.="To unsubscribe from this campaign, please click the
following link.$rn";
$plainfooter.=$unsubLink;

//now define an HTML version of the footer to illustrate tracking through
img
$htmlfooter="<br><br><a href='$trackURL'>example.com</a>";
//hidden image.
$htmlfooter.="<img src='$trackImg'>";
$htmlfooter.="<hr><br>";
$htmlfooter.="<p>To unsubscribe from this campaign, please click the
following link.</p>";
$htmlfooter.="<a href='$unsubLink'>$unsubLink</a>";

// Override SMTP headers
$headers='From: System Administrator <donotreply@example.com>'.$rn;
$headers .= "MIME-Version: 1.0\r\n";     //specify MIME version 1.0
//tell e-mail client this e-mail contains alternate versions
$headers .= "Content-Type: multipart/alternative; boundary =
$bound".$rn.$rn;
$headers .= "This is a MIME encoded message.".$rn.$rn;;

$message = "insert message here";//Message TAKEN FROM DB based on messageID
//declare this is the plain text version
$headers .= "--$bound" . $rn . "Content-Type: text/plain; charset=ISO-8859-
1".$rn . "Content-Transfer-Encoding: base64".$rn.$rn;
//actually output the plaintext version (base64 encoded)
$headers .= chunk_split(base64_encode($message.$plainfooter));

$HTMLMessage = "MESSAGE";//Get HTML message from DB based on messageID
//declare we're about to add the HTML version
$headers .= "--$bound\r\n" . "Content-Type: text/html; charset=ISO-8859-
1\r\n" . "Content-Transfer-Encoding: base64\r\n\r\n";
//actually output the plaintext version (base64 encoded)
$headers .= chunk_split(base64_encode($HTMLMessage. $htmlfooter));

mail($mailto,$subj, "" ,$headers);                    //the all-powerful mail
function
}
```

4   This function can then be used as part of a larger module that pulls content form a marketing campaign database.  In our case we assume there are several campaigns and each campaign has several messages. You may change this to suit your needs.

5   Note that many changes must be made to this script before it is ready for production. You must develop and test the URLs for unsubscribing, set up your own connections to database content for the message, design a proper footer/header and then test against your own campaign system.

6   One note is that you should store both plaintext and HTML version of all your messages, since some email clients cannot interpret HTML. You should also set up the "From" header carefully to be an email under your control since man people will reply to these

messages  (or alternatively use an email that will bounce).

**7**   Remember, spam email  is illegal. You should use this function (and your marketing material) in a responsible, ethical and legal way, for example when sending a "forgotten password" email a "new registrant" email or other requested email.

## Exercise 24.5 — QR Codes

**1**   This will show you how to generate QR codes on the fly. A capability that will help you with marketing campaigns.

**2**   QR codes utilize a complex set of mathematical equation to determine the QR codes. There are different types of QR code, each with greater redundancy than the last (ideal for obscured QR codes). Rather than develop QR code from scratch, download the open source library from http://phpqrcode.sourceforge.net/

**3**   The phpQRCode library requires a temporary location to store the files it generates. The easiest thing is to create a temp folder and make it writable by apache. Open up phpqrcode/index.php to see a simple example script that includes a form to control the generation of QR codes as shown in Figure 24.5.

*Figure 24.5 QR code generator using the phpwrcode Open Source library*

4   Such a utility is useful for generating QR codes for marketing purposes as required. Using a 3rd party generator can be dangerous since some services create links to their own server with a redirect rather than a direct link.

## Exercise 24.5 – Configure an Analytics Package

1   To complete this exercise you must have access to the access log from one of your domains. Ideally, you set up a unique log per domain in Exercise 22.8. AWStats is an open source analytics package that runs on your webserver to create statistics.

Note: If you have access to an interface like Cpanel or otherwise these systems are

sometimes already installed, the underlying ideas are still important, as is configuration.

2   Download the AWStats package to your server (it's a zip file) and extract it inot it's directory. You normally install the package in /usr/local/

3   Run
```
perl awstats_configure.pl
```

To configure ans set up the package. The script will askyou questions, and generate a configuration file.

At the end it will say how it' can't set up a cron tab for you and output the command. The command in our instance is:

```
/usr/local/awstats/wwwroot/cgi-bin/awstats.pl -update -config=funwebdev.com
```

4   Find and edit the configuration file to use the log file you associate with this domain

```
LogFile="var/www/logs/funwebdev_log"
```

5   The configuration will also refer to a DirData. This is where it will store data after each processing.

```
DirData="/var/lib/awstats"
```

Make sure the referenced folder exists.

6   Now run the script once (using the command from step 3)

/usr/local/awstats/wwwroot/cgi-bin/awstats.pl -update -config=funwebdev.com

If there are errors the output will describe the errors. Once you have a working script the output will tell you how many lines were analyzed, similar to below

```
Create/Update database for config "/etc/awstats/awstats.funwebdev.com.conf"
by AWStats version 7.3 (build 20140126)
From data in log file "var/www/logs/funwebdev_log "...
Phase 1 : First bypass old records, searching new record...
Searching new records from beginning of log file...
Phase 2 : Now process new records (Flush history on disk after 20000
hosts)...
    Jumped lines in file: 0
Parsed lines in file: 128303
 Found 0 dropped records,
 Found 0 comments,
 Found 0 blank records,
 Found 3 corrupted records,
 Found 0 old records,
 Found 128300 new qualified records.
```

7   Now create a location (publically, possible protected by a password), where you want to generate the HTML files that summarize your stats.

Run the following command to generate the html pages:

/usr/local/awstats/tools/awstats_buildstaticpages.pl -config=funwebdev.com

8  Consider adding cron jobs to automatically update your stats nightly using the commands used to date (or combine them into a single executable script).

You can now easily see stats on your websites.

9  The power of external system like those from the major search engines also warrant investigation. Investigate one or more of those system to see their benefits.