

Site Search: Quick & Easy

by
Peter Lavin

January 6, 2005

Overview

A site-specific search capability is a nice feature to be able to add to a website. If a site is large and content-rich such an addition can be an indispensable aid.

However, the complexity of creating your own search engine can be intimidating even for an experienced web programmer. This article will show how to create a site-specific search engine. No programming skills will be required to implement the code presented here but the reader should be familiar with HTML and would benefit by having some understanding of scripting languages.

How We'll Do It

To give a quick overview, we will be using an HTML “form” with a “text input” box and a “submit” button. This form in conjunction with the Google API and an open-source PHP script is all that will be needed. We'll show you how to put these elements together so that a simple form may be placed on a sidebar, or wherever appropriate, to assist in searching your site.

There are four steps to creating this site-specific search engine. 1) Get a license key from Google, 2) download the PHP script, “nusoap” and finally, 3) install a short script to initiate the search and 4) format the HTML output.

Google API

API stands for “Application Programming Interface” – it is simply a means to tap into the Google search engine without having to actually point your browser at their site. It allows you to perform Google searches programmatically.

You may find out about the Google API at <http://www.google.com/apis/>. You need not download the example code but you will need to create a Google account and get a license key. This license key will allow you to initiate 1000 searches per day using the Google API. Check your website statistics and if you are getting fewer than 500 visits per day then this number of searches will most likely be more than adequate.

nusoap

nusoap is a PHP script that will facilitate use of the Google API. It is available from the URL, <http://sourceforge.net/projects/nusoap/>. Go to this site and find the download link about halfway down the page. At the time of writing the latest version was 0.6.7. The file is zipped but don't let that discourage you from using it on a non-Windows server. It works just as well using Apache and Linux as it does using Internet Information Server and Windows. You will, of course have to have PHP installed on your server. If it is not already installed then it is probably time to find a new web host.

There are a number of files compressed into the zipped file but “nusoap.php” is the only one we need to be concerned about. If you are familiar with PHP you may want to have a look at the other files. I will just mention though that the script we develop is a modification of the “client3.php” file.

The Code

Let's first create the form that will invoke the search page:

```

Search this site: <br />
<form method="get" name="search" action="search.php">
<input type="text" name="criterion" style="width:100px" /><br />
<input style="margin-top:5px" type="submit" value="Submit" />
</form>

```

Insert this code into the page that you intend to search from – for the sake of easy reference let’s call that page “searchfrom.html”.

The code that actually performs the search is below (based on the “client3.php” file):

```

<html>
<head>
<title>Site Search Page</title>
</head>
<body>
<?php
require_once("nusoap.php");
$criterion=@$_GET["criterion"];
if(strpos($criterion, "\"")){
    $criterion = stripslashes($criterion);
    echo "<b>$criterion</b>";
}else
    echo "\"<b>$criterion</b>\"</p>";
$query=$criterion;
//your site here
$query .= " site:www.yoursite.com";
//your Google key goes here
$key = "yourgooglekey";
//change the value below if you like
$maxresults = 10;
$start = 0;
$parameters = array(
    'Googlekey'=>$key,
    'queryStr'=>$query,
    'startFrom'=>$start,
    'maxResults'=>$maxresults,
    'filter'=>true,
    'restrict'=>'',
    'adultContent'=>true,
    'language'=>'',
    'iencoding'=>'',
    'oencoding'=>''
);
$client = new soapclient("http://api.google.com/search/beta2");
$result = $client->call("doGoogleSearch", $parameters,
"urn:GoogleSearch");
$searchtime = $result["searchTime"];
$total = $result["estimatedTotalResultsCount"];
if($total > 0){
    $rs = $result["resultElements"];

```

```

$output="";
for ($i = 0; $i < $total; $i++){
    if (!isset($rs[$i])) break;
    $element = $rs[$i];
    //$title=$element["title"];
    $url = $element["URL"];
    $snippet = $element["snippet"];
    $output.= "<p><a href=\"\$url\">".basename($url)."</a>
$snippet</p>\n";
}
echo $output;
echo "<br /><br />Search time: $searchtime seconds.";
}else
    echo "<br /><br />Nothing found.";
?>
</body>
</html>

```

Save this code as “search.php” making sure that you use the file extension “php”.

The changes you need to make to this script are shown in bold. Substitute your domain name for “www.yoursite.com” and your Google key for “yourgooglekey” making sure to enclose both items in quotation marks and that you preserve the leading space before “site”. Change the value of “\$maxresults” if you wish to have fewer or more results.

To sum up, we now have three files that need to be located in the same directory, “searchfrom.html”, “nusoap.php” and “search.php”. Entering a criterion into the textbox on the “searchfrom.html” page will open the “search.php” page and display the results of your search limited to the specified site. These results will show a snippet of text and a hyperlink to the page on which your criterion appears. Visitors to your site can now easily find items of interest and quickly navigate to them.

Changes & Improvements

Depending upon your knowledge of PHP and HTML this code can be customised and changed in a number of ways. We have already shown how the number of results returned can be adjusted but you might also want to add a pop-up window of search hints for your site visitors. Let them know that they can search for a single word, or a group of words or a specific phrase if they enclose the expression in quotation marks.

Another useful element returned from Google is the page title. I’ve put it into the “search.php” page to show how to access it but have commented it out. One possible use for this piece of information would be to limit search results to a specifically named page. I’ll let you determine how this might be done.

As you can see our script uses one of the Google advanced search techniques (see <http://www.google.com/help/refinerearch.html>), a criterion, the word “site” followed by a semicolon and then a domain name. In the same way as we have done here, you could easily change this code to search instead for specific file types by using something like “ filetype:pdf” instead of “ site:www.yoursite.com”.

You may also want to indicate to visitors that your site search is Google-based. You may certainly do this but check out the terms and conditions of use at http://www.google.com/apis/api_terms.html.

Some Limitations

It might be said that any web surfer could do exactly what is described here by opening a separate instance of their browser and doing a site-specific search themselves. Quite true, but how many people would know how to do this and even if they knew, how many would actually do it? Most web surfers will appreciate the convenience of being provided with a site-specific search capabilities embedded right into your site.

As already mentioned, this is not the solution for a high-traffic site where many searches will be initiated. Nor is it a solution for a newly posted site. Until a site is indexed by Google no search results will be returned. Likewise, recent changes to a site will not be found until the Googlebot visits and registers them.

Additionally, some of the less common character entities are not rendered correctly. I noticed no problem with common entities such as “<,” and “&,” but less common ones such as “…,” were simply rendered as question marks.

That said though, using the Google API in conjunction with the “nusoap” script is a quick and easy way to implement a site-specific search. This is particularly useful for sites that are content-rich and it is an excellent additional service to propose to your clients once their newly created site shows up on Google.

Resources

The Google API - <http://www.google.com/apis/>

Advanced Google searches - <http://www.google.com/help/refinerearch.html>

Terms of Use - http://www.google.com/apis/api_terms.html

PHP script “nusoap” - <http://sourceforge.net/projects/nusoap/>

About the Author

Peter Lavin runs a Web Design/Development firm in Toronto, Canada. For more information visit <http://www.softcoded.com>. He may be reached at peterlavin@sympatico.ca.