**eBay API Dev Camp**

Innovate 2011 Conference

October 14, 2011

San Francisco

About Jimmy Wong

Jimmy Wong is a Developer Technical Support Engineer at X.commerce. Prior to this role, he was been a QA engineer at eBay Marketplaces. He went to college at the School of Engineering at San Jose State University from 2001-2005, where he studied Computer Engineering.

**Contact Information**

email: jimmy@ebay.com

twitter: @jwong77

Setup and Configuration

1. Make sure that XAMP is installed, and Apache is running
2. Go to c:\xamp\php\php.ini and open it in notepad
3. Search for “extension=php\_curl.dll”
4. Delete the “;” in front of this statement, if it is there
5. Search for “error\_reporting”
6. Replace “error\_reporting = E\_ALL | E\_STRICT” with “error\_reporting = E\_ALL & ~E\_NOTICE”
7. Search for “default\_socket\_timeout”
8. Replace “default\_socket\_timeout = 60” with “default\_socket\_timeout = 3000”
9. Restart XAMP

**And code is in boxes like this.**

Grey text indicates code that is already in your lab files, and do not need to be entered.

Answers are in these blue boxes.

**Required Resources**

This hands-on lab requires that you have the following installed:

* Chrome or FireFox Browser
* An IDE such as Eclipse or NetBeans
* If using Eclipse please install the PHP plugin
* Files for this lab (eBay-API-XDevCamp.zip file)

The file **eBay-API-XDevCamp.zip** needs to be in your htdocs folder. Extract the folder within the file directly within the htdocs folder.

**Directory Structure**

C:/xampp/htdocs/eBay-API-XDevCamp/

**Lab 1 – Find the product id using FindProducts**

**Open the file for this lab.**

1. Go to your IDE.
2. Open the **File** menu and select **Open** …
3. Open the file **FindProducts.php** in the **eBay-API-XDevCamp** folder located in the **htdocs** directory.

.

Create the FindProducts request.

*Enter Code 1.1*

**include\_once** "latlong.php";

**$url =** [**http://open.api.ebay.com/shopping?**](http://open.api.ebay.com/shopping?)**;**

$debug = $\_GET['debug'];

$debug2 = $\_GET['debug2'];

$debug3 = $\_GET['debug3'];

$keyword = $\_GET['q'];

**$strPost = '<?xml version="1.0" encoding="utf-8"?>**

**<FindProductsRequest xmlns="urn:ebay:apis:eBLBaseComponents">**

**<QueryKeywords>'.$keyword.'</QueryKeywords>**

**<MaxEntries>3</MaxEntries>**

**</FindProductsRequest>';**

**if**($debug == "yes")

{

header("Content-type: text/xml");

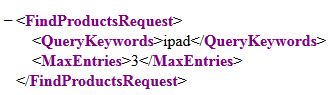
print\_r($strPost);

**exit**();

}

**Test Your Code**

1. Launch a Browser.
2. Open the **File** menu and select **Open** File
3. Enter in <http://localhost/eBay-API-XDevCamp/FindProducts.php?q=ipad&debug=yes>
4. The response should show us the FindProducts call we are making



Now lets construct the headers and the curl statements for making the POST request.

*Enter Code 1.2*

**$headersArray = array(**

**'X-EBAY-API-APP-ID: PUT YOUR APP ID HERE',**

**'X-EBAY-API-SITEID: 0',**

**'X-EBAY-API-CALL-NAME: FindProducts',**

**'X-EBAY-API-VERSION: 733',**

**'X-EBAY-API-REQUEST-ENCODING: XML',**

**'Content-Type: ' . 'text/xml',**

**);**

$ch = curl\_init();

curl\_setopt($ch, CURLOPT\_SSL\_VERIFYPEER, **false**);

curl\_setopt($ch, CURLOPT\_SSL\_VERIFYHOST, **false**);

curl\_setopt($ch, CURLOPT\_POST, 1);

curl\_setopt($ch, CURLOPT\_HTTPHEADER, $headersArray);

curl\_setopt($ch, CURLOPT\_URL, $url);

curl\_setopt($ch, CURLOPT\_POSTFIELDS, $strPost);

curl\_setopt($ch, CURLOPT\_FOLLOWLOCATION, 1);

curl\_setopt($ch, CURLOPT\_SSL\_VERIFYPEER, **false**);

curl\_setopt($ch, CURLOPT\_RETURNTRANSFER, **true**);

$FindProductsResponseXml = curl\_exec($ch);

**if**($debug2 == "yes")

{

header("Content-type: text/xml");

print\_r($FindProductsResponseXml);

**exit**();

}

**Test Your Code**

1. Launch a Browser.
2. Open the **File** menu and select **Open** File
3. Enter in [http://localhost/eBay-API-XDevCamp /FindProducts.php?q=ipad&debug2=yes](http://localhost/eBay-API-XDevCamp%20/FindProducts.php?q=ipad&debug2=yes)
4. The response should show us the response from the FindProducts call

This is a snippet of what you should expect to see



Load the FindProducts response into and object and parse the product id and product id type. We will use the variable names $pid and $pidType for these values.

We’ll use “href” to make each product that returns a clickable link. Where the link leads to will be discussed in lab 2.

Enter Code 1.3

**$xml = simplexml\_load\_string($FindProductsResponseXml);**

**print** "<pre>";

$ProductArray = $xml[0]->Product;

**foreach**($ProductArray **as** $Product)

{

**$ProductIDArray = $Product->ProductID;**

**$ProductIDTypeArray = $Product->ProductID->attributes()->type;**

**$Title = $Product->Title;**

**foreach**($ProductIDArray **as** $ProductID)

{

**$pid = $ProductID[0];**

**$pidType = $pid->attributes()->type;**

**if** ($pidType == 'Reference')

$pidType = $pidType.'ID';

**$url = "http://localhost/eBay-API-XDevCamp/getMap.php?pId=".$pid."&pidType=".$pidType;**

**print** '<a href="'.$url.'">';

**print** $Title[0]."</a>";

**print** "<br>";

}

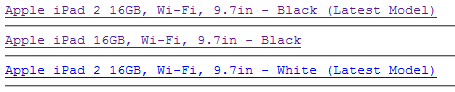
**print** "<hr>";

}

**print** "</pre>";

**Test Your Code**

1. Launch a Browser.
2. Open the **File** menu and select **Open** File
3. Enter in <http://localhost/eBay-API-XDevCamp/FindProducts.php?q=ipad>
4. The response should show us a list of available products that match the keywords we passed in (“ipad”).



**Take Aways From Lab1**

1. Make sure the endpoint you are making your request against is correct. Please refer to <http://developer.ebay.com/DevZone/shopping/docs/Concepts/ShoppingAPI_FormatOverview.html>
2. FindProducts takes the QueryKeywords field as an input.
3. The output contains product id and product id type.

**Lab 2 – Make a findItemsByProduct call**

**Open the file for this lab.**

1. Go to your IDE.
2. Open the **File** menu and select **Open** …
3. Open the file **findItemsByProduct.php** in the **eBay-API-XDevCamp** folder located in the **htdocs** directory.

Defining dependencies and defining the post request

The request is a findItemsByProduct request, where we want to find items within 200 mile radius of the postal code 95054.

<?xml version="1.0" encoding="utf-8"?>

<findItemsByProductRequest xmlns="http://www.ebay.com/marketplace/search/v1/services">

<productId type="'.$pidType.'">'.$pId.'</productId>

<buyerPostalCode>95054</buyerPostalCode>

<itemFilter>

<name>MaxDistance</name>

<value>200</value>

</itemFilter>

</findItemsByProductRequest>';

*Enter Code 2.1*

$debug = $\_GET['debug'];

$debug2 = $\_GET['debug2'];

$debug3 = $\_GET['debug3'];

$keyword = $\_GET['q'];

$pId = $\_GET['pId'];

$pidType = $\_GET['pidType'];

$zip = $\_GET['zip'];

**include\_once** "latlong.php";

**if** ($debug == "yes" || $debug2 == "yes" || $debug3 == "yes")

{

$pId = 103127916; //Dummy values to test our code

$pidType = ReferenceID;

}

**$url = "http://svcs.ebay.com/services/search/FindingService/v1?";**

**$strPost = '<?xml version="1.0" encoding="utf-8"?>**

**<findItemsByProductRequest xmlns="http://www.ebay.com/marketplace/search/v1/services">**

**<productId type="'.$pidType.'">'.$pId.'</productId>**

**<buyerPostalCode>95054</buyerPostalCode>**

**<itemFilter>**

**<name>MaxDistance</name>**

**<value>200</value>**

**</itemFilter>**

**</findItemsByProductRequest>';**

**if** ($debug == **true**)

{

header("Content-type: text/xml");

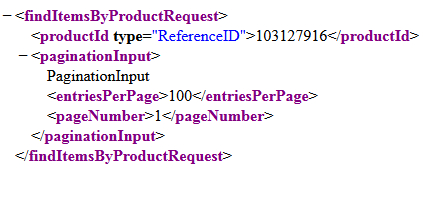
print\_r($strPost);

**exit**();

}

**Test Your Code**

1. Launch a Browser.
2. Open the **File** menu and select **Open** File
3. Enter in <http://localhost/eBay-API-XDevCamp/findItemsByProduct.php?debug=yes>
4. We should see the following



Now lets construct the headers and the curl statements for making the POST request.

*Enter Code 2.2*

**$headersArray = array(**

**'X-EBAY-SOA-SECURITY-APPNAME: PUT YOUR APP ID HERE',**

**'X-EBAY-SOA-GLOBAL-ID: EBAY-US',**

**'X-EBAY-SOA-OPERATION-NAME: findItemsByProduct',**

**'X-EBAY-API-REQUEST-ENCODING: XML',**

**'Content-Type: ' . 'text/xml',**

**);**

$ch = curl\_init();

curl\_setopt($ch, CURLOPT\_SSL\_VERIFYPEER, **false**);

curl\_setopt($ch, CURLOPT\_SSL\_VERIFYHOST, **false**);

curl\_setopt($ch, CURLOPT\_POST, 1);

curl\_setopt($ch, CURLOPT\_HTTPHEADER, $headersArray);

curl\_setopt($ch, CURLOPT\_URL, $url);

curl\_setopt($ch, CURLOPT\_POSTFIELDS, $strPost);

curl\_setopt($ch, CURLOPT\_FOLLOWLOCATION, 1);

curl\_setopt($ch, CURLOPT\_SSL\_VERIFYPEER, **false**);

curl\_setopt($ch, CURLOPT\_RETURNTRANSFER, **true**);

$responseXml = curl\_exec($ch);

**if**($debug2 == "yes")

{

header("Content-type: text/xml");

print\_r($responseXml);

**exit**();

}

**Test Your Code**

1. Launch a Browser.
2. Open the **File** menu and select **Open** File
3. Enter in <http://localhost/eBay-API-XDevCamp/findItemsByProduct.php?debug2=yes>
4. The following is a snippet of what you should see



Now parse the response and get the gallery image url, title, and postal code of the item.

As was shown in Lab 1 Code 1.3, load the xml response into a simplexml object.

Then parse the parts of the findItemsByProduct response that we need.

**Note: Markers** is the xml container we need to pass into Google Maps. Its not important to completely understand this, just realize that this is needed to build the Google Maps display.

*Enter Code 2.6*

**$xml = simplexml\_load\_string($responseXml);**

**$strXML = '<?xml version="1.0"?><markers>';**

**$searchresults = $xml->searchResult;**

**$items = $searchresults[0]->item;**

**foreach**($items **as** $item)

{

**$itemID = $item->itemId;**

**$title = $item->title;**

**$price = $item->sellingStatus->currentPrice;**

**$postalCode = $item->postalCode;**

**$location = $item->location;**

**$country = $item->country;**

**$point = getPoint($postalCode);**

**$lon = $point[0];**

**$lat = $point[1];**

**$imageUrl = $item->galleryURL;**

$imageHtml = '<img width="50" height="50" src="'.$imageUrl.'">';

**if**($debug3 != "yes")

{

**i**nclud**e\_once** "GetShippingCosts.php";

$shippingCost = getShippingCost($itemID, $zip);

}

$strXML .= '<marker name="'.htmlspecialchars($imageHtml.' : '.$title .' Cost to Ship: '.$shippingCost).'" address="'.$postalCode.'" lat="'.$lat.'" lng="'.$lon.'" distance="'.$distance.'"/>';

}

header('Content-Type:text/xml');

$strXML .= '</markers>';

**print** $strXML;

**Test Your Code**

1. Launch a Browser.
2. Open the **File** menu and select **Open** File
3. Enter in <http://localhost/eBay-API-XDevCamp/findItemsByProduct.php?debug3=yes>
4. The following is a snippet of what you should see



**Takeaways from Lab 2**

1. findItemsByProduct uses the Finding API endpoint and the Finding API standard headers. For more information about the Finding API please refer to : <http://developer.ebay.com/devzone/finding/callref/findItemsAdvanced.html>
2. API headers for the Finding API’s (findItemsByProduct) differ from those in the Shopping API

**Lab 3 – Get Shipping Costs by calling GetShippingCosts**

**Open the file for this lab.**

1. Go to your HTML text editor.
2. Open the **File** menu and select **Open** …
3. Open the file **GetShippingCosts.php** in the **eBay-API-XDevCamp** folder located in the **htdocs** directory.

**Making your getShippingCost Function**

Use the function keyword to create a function called getShippingCost. Create the GetShippingCosts request. In the first if statement, we put in some dummy values to test our request.

*Enter Code 3.1*

//Code 3.1

$debug = $\_GET['debug'];

$debug2 = $\_GET['debug2'];

$debug3 = $\_GET['debug3'];

**if**($debug == "yes" || $debug2 == "yes" || $debug3 == "yes")

{

//Here we're simply testing getShippingCost by a dummy itemid and a zip code

getShippingCost(150660130087, 95054);

**exit**();

}

**function** getShippingCost($itemID,$zip){

**global** $debug, $debug2, $debug3;

$url = "http://open.api.ebay.com/shopping?";

**$strPost = '<?xml version="1.0" encoding="utf-8"?>**

**<GetShippingCostsRequest xmlns="urn:ebay:apis:eBLBaseComponents">**

**<ItemID>'.$itemID.'</ItemID>**

**<DestinationCountryCode>US</DestinationCountryCode>**

**<DestinationPostalCode>'.$zip.'</DestinationPostalCode>**

**<QuantitySold>1</QuantitySold>**

**</GetShippingCostsRequest>';**

**if**($debug == "yes")

{

header("Content-type: text/xml");

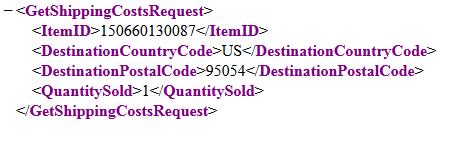
print\_r($strPost);

**exit**();

}

**Test Your Code**

1. Launch a Browser.
2. Open the **File** menu and select **Open** File
3. Enter in <http://localhost/eBay-API-XDevCamp/GetShippingCosts.php?debug=yes>
4. The following is a snippet of what you should see



Enter the Headers that correspond to making the GetShippingCosts request.

*Enter Code 3.2*

//Code 3.2

**$headersArray = array(**

**'X-EBAY-API-APP-ID: EBAYMERCHAF15555F8CJS66713K531',**

**'X-EBAY-API-SITEID: 0',**

**'X-EBAY-API-CALL-NAME: GetShippingCosts',**

**'X-EBAY-API-VERSION: 733',**

**'X-EBAY-API-REQUEST-ENCODING: XML',**

**'Content-Type: ' . 'text/xml',**

**);**

$ch = curl\_init();

curl\_setopt($ch, CURLOPT\_SSL\_VERIFYPEER, **false**);

curl\_setopt($ch, CURLOPT\_SSL\_VERIFYHOST, **false**);

curl\_setopt($ch, CURLOPT\_POST, 1);

curl\_setopt($ch, CURLOPT\_HTTPHEADER, $headersArray);

curl\_setopt($ch, CURLOPT\_URL, $url);

curl\_setopt($ch, CURLOPT\_POSTFIELDS, $strPost);

curl\_setopt($ch, CURLOPT\_FOLLOWLOCATION, 1);

curl\_setopt($ch, CURLOPT\_SSL\_VERIFYPEER, **false**);

curl\_setopt($ch, CURLOPT\_RETURNTRANSFER, **true**);

$GetShippingCostsResponseXml = curl\_exec($ch);

**if**($debug2 == "yes")

{

header("Content-type: text/xml");

print\_r($GetShippingCostsResponseXml);

**exit**();

}

**Test Your Code**

1. Launch a Browser.
2. Open the **File** menu and select **Open** File
3. Enter in <http://localhost/eBay-API-XDevCamp/GetShippingCosts.php?debug2=yes>
4. The following is a snippet of what you should see



Load the GetShippingCostsResponseXML into a simpleXML object. Next parse the ShippingServiceCost from the response.

*Enter Code 3.3*

//Code 3.3

**$xml = simplexml\_load\_string($GetShippingCostsResponseXml);**

**$ShippingCostSummary = $xml->ShippingCostSummary;**

**$ShippingServiceCost = $ShippingCostSummary->ShippingServiceCost;**

**if**($debug3 == "yes")

{

**echo** "This is the cost to ship $ $ShippingServiceCost";

**exit**();

}

**return** $ShippingServiceCost;

}

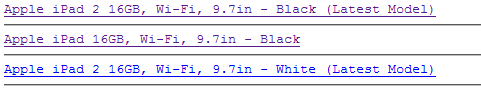
?>

**Test Your Code**

1. Launch a Browser.
2. Open the **File** menu and select **Open** File
3. Enter in <http://localhost/eBay-API-XDevCamp/GetShippingCosts.php?debug3=yes>
4. The following is a snippet of what you should see

**Test your code**

1. In your browser type in the following URL: <http://localhost/eBay-API-XDevCamp/FindProducts.php?q=ipad>
2. You should see the following:



1. Click on “Apple iPad 2 16GB, Wi-Fi, 9.7in - Black (Latest Model)”
2. Type in an address.
3. You should see a basic Google Maps with the list of items found using our series of API calls.

