



How do I setup Google Analytics GA4 and eCommerce conversion tracking in ShopFactory?

Merliza N. - 2024-06-03 - Global Settings

Latest update of tracking codes on: October 12, 20, 2023

Google Analytics GA4 is a very powerful tool to track your visitors on your website and to see what actions they have taken on your website. In this article, we will explain how to set this up and to start measuring:

- Purchase event with product items
- View item details (Navigating to a detailed product page)
- Add/remove a product from a shopping cart
- Add a product to the wishlist (Favorites)
- Initiate the checkout process.

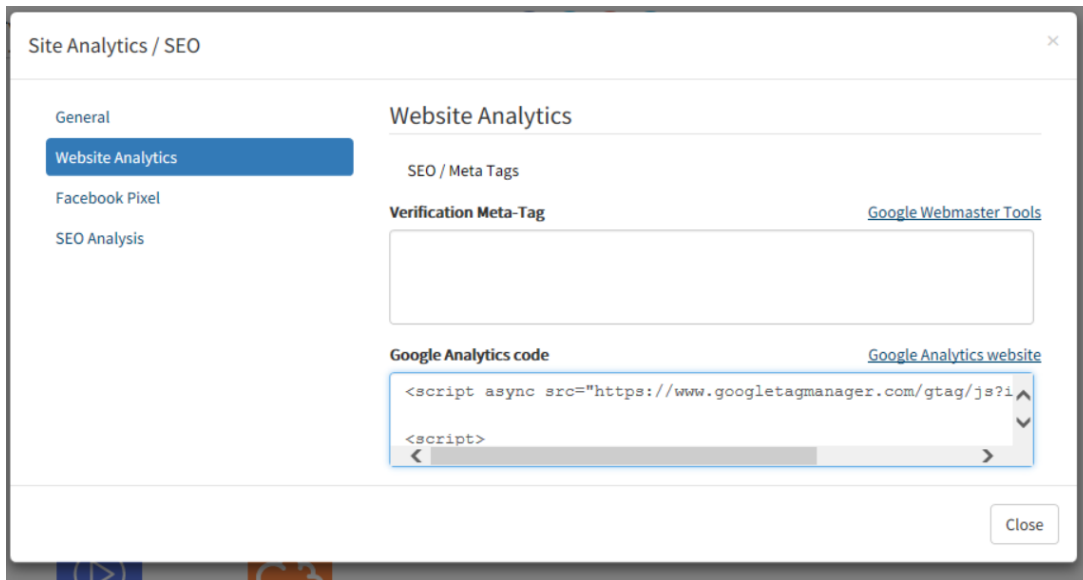
To add this tool to your website, follow the instructions below.

You will need to add some code to your website on 2 locations:

Location 1

For the tracking to work, you'll need to install the tracking tag . This global site tag must be added in the **<head>** section of your website.

1. Open your shop in ShopFactory software and go to your ShopFactory Central page
2. Click "Site Analytics / SEO" button
3. Click "Website Analytics"
4. Locate the "Google Analytics Code" box and then paste your code as shown in screen shot:



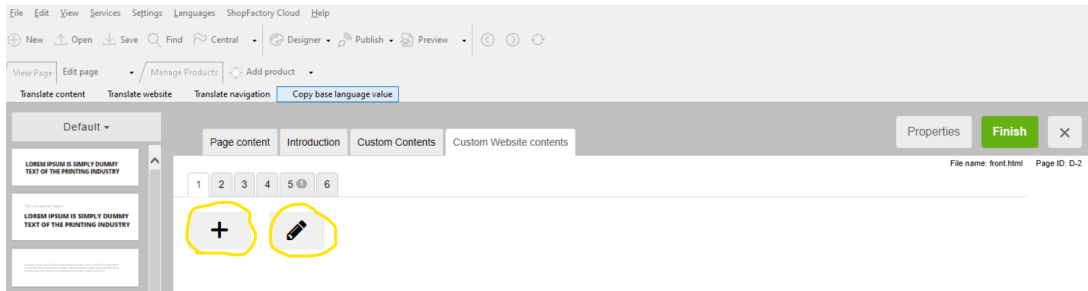
Your tag manager code should look like this, all you need to do is make sure that, in 2 locations in the script, G-XXXXXXXXXX is replaced with your actual Google Analytics Measurement ID (the easiest way to find your GA4 Measurement ID is by typing measurement ID in the Google Analytics Search bar, you can copy your ID from the search result):

```
<!-- Google tag (gtag.js) -->
<script async src="https://www.googletagmanager.com/gtag/js?id=G-XXXXXXXXXX"></script>
<script>
  window.dataLayer = window.dataLayer || [];
  function gtag(){dataLayer.push(arguments);}
  gtag('js', new Date());

  gtag('config', 'G-XXXXXXXXXX');
</script>
```

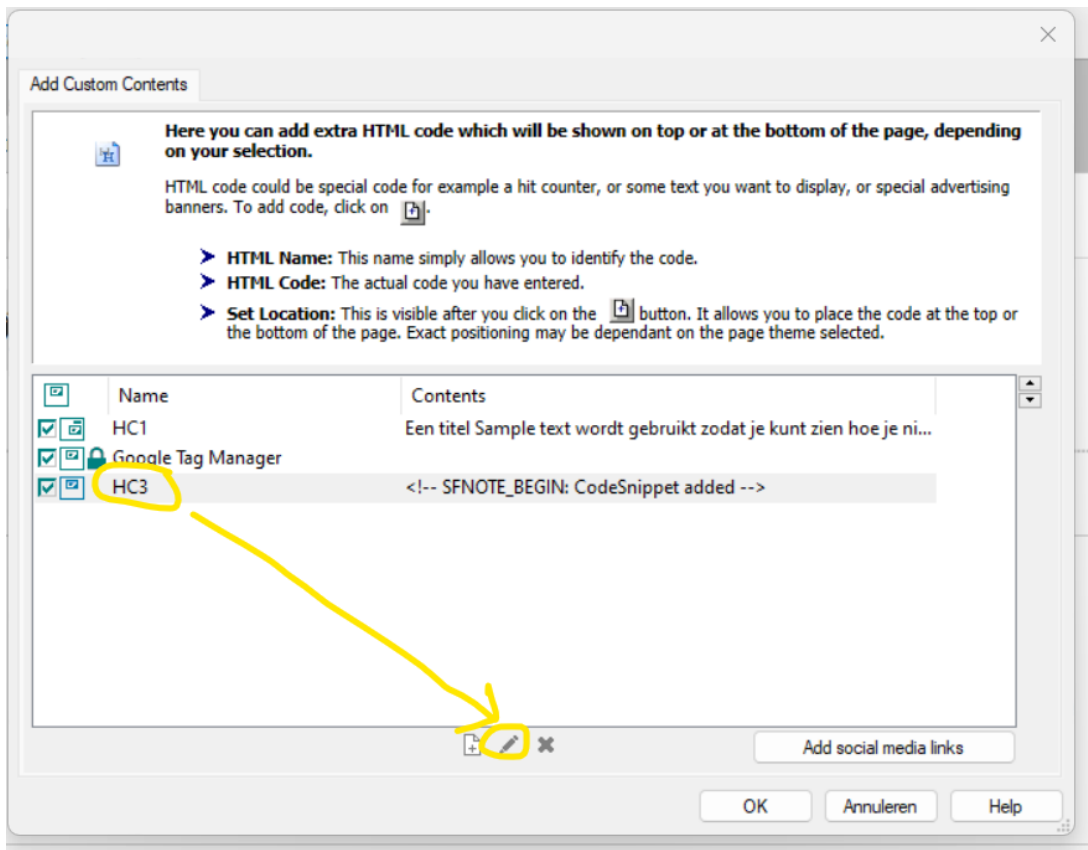
Location 2

1. Open your shop in ShopFactory software and go to your ShopFactory Central page
2. Click "My Store" button
3. Click "Custom Website Contents" (if a ShopFactory alert pops up, click Close)
4. Click on the + sign to add a new content block (see image below)



5. Click on the Pencil to edit it (see image above) (if a ShopFactory alert pops up, click Close)

6. The “Add Custom Contents” wizard will appear. Now locate the content block that was just created in step 4, it is the bottom one in the list (if you already have other pieces of content, otherwise it will be the only one listed there) and the name starts with HC. Please select it and click the pencil to edit it, see the image below



7. Rename the block to “Google GA4” in the identification name area, and past the script below in the main contents window AND make sure to tick the box in front of “Field contains Javascript code”. See image below for the correct settings

This is the code you need to copy (it starts with <script> and ends with } </script>, all you need to is make sure that **G-XXXXXXXXXX** is replaced at the top of the script with your

actual measurement ID:

```
<script>
```

```
var GA_MEASUREMENT_ID = 'G-XXXXXXXX';
```

```
window.dataLayer = window.dataLayer || [];
```

```
function gtag() { dataLayer.push(arguments); }
```

```
gtag('js', new Date());
```

```
gtag('config', GA_MEASUREMENT_ID);
```

```
var _sfAnalyser = {
```

```
  processing: false,
```

```
  init: function () {
```

```
    this.processing = true;
```

```
    var qs = location.search.substr(1);
```

```
    if (qs) {
```

```
      var pairs = qs.split('&'), len = pairs.length;
```

```
      for (var i = 0; i < len; i++) {
```

```
        temp = pairs[i].split('=');
```

```
        tf.parm[unescape(temp[0])] = unescape(temp[1]);
```

```
      }
```

```
    }
```

```
    if (tf.parm['orderno']) this.run();
```

```
  },
```

```
  run: function () {
```

```
    if (!tf.core || (!tf.nametag.get('xml_basket') && tf.pgid == 'D-8') || (tf.core.Basket.items._cnt == 0 && tf.pgid == 'D-6')) {
```

```
      setTimeout(_sfAnalyser.run, 100);
```

```
      return;
```

```
    }
```

```
function unfmt(v) {
```

```
  var vv = v.toString().replace(/[\^\\d.]/g, "");
```

```
  if (vv.match(/^[^\\d.]+(\\d+)$/)) {
```

```
    vv = RegExp.$1.replace(/./g, ".") + '.' + RegExp.$2;
```

```
  }
```

```
  else {
```

```
    vv = vv.replace(/./g, ".");
```

```
  }
```

```

return vv;
}
function attributify(xml, tag) {
var s = xml.match(RegExp('<' + tag + '.*?>', 'g'));
if (s && s.length > 0) {
s = s[0];
eval('var o={ ' + s.replace(RegExp('<' + tag + ' '), '').replace(/\/?>$/,
'').replace(/(\\S+?)="(.*?)"/g, '"$1":"$2",').replace(/,$/, '') + ' }');
return o;
}
else return null;
}

var total = unfmt(tf.parm['orderAmount'] || '0'), tax = unfmt(tf.parm['tax'] || '0'), shipping =
unfmt(tf.parm['shipping'] || '0'), ccy = "", coupon = "";
var xmlBasket = tf.nametag.get('xml_basket').replace(/%3e/g, '>').replace(/%3c/g,
'<').replace(/%22/g, '"').replace(/%26/g, '&').replace(/%25/g, '%');

var bskAttr = attributify(xmlBasket, 'basket');
if (bskAttr) {
total = unfmt(bskAttr['total']);
tax = unfmt(bskAttr['tax_total']);
shipping = unfmt(bskAttr['ship_total']);
if (tf.core.nfmt) {
total = tf.core.nfmt.round(total);
tax = tf.core.nfmt.round(tax);
shipping = tf.core.nfmt.round(shipping);
}
}

var shopAttr = attributify(xmlBasket, 'shop');
if (shopAttr) {
ccy = shopAttr['currency'];
}

var orderAttr = attributify(xmlBasket, 'order');
if (orderAttr) {
coupon = orderAttr['voucherApplied'];
}

var itmsForAnalytics = xmlBasket.match(/<item.*?>/g), items = [];
if (itmsForAnalytics) {
for (var n = 0, cnt = itmsForAnalytics.length; n < cnt; n++) {

```

```

var itm = itmsForAnalytics[n];
itm = attributify(itm, 'item');
items.push({
'item_id': itm['itemNo'] ? itm['itemNo'] : itm['prodid'],
'item_name': itm['title'],
'affiliation': tf.core['shopName'],
'quantity': " + itm['quantity'],
'price': " + unfmt(itm['price'])
})
}
}

if (!xmlBasket && items.length == 0 && tf.pgid == 'D-6') {
var itms = tf.core.Basket.items;
for (var n = 0; n < itms._cnt; n++) {
var itm = itms.get(n);
items.push({
'item_id': itm['itemNo'] ? itm['itemNo'] : itm['prodid'],
'item_name': itm['title'],
'affiliation': tf.core['shopName'],
'quantity': " + itm['quantity'],
'price': " + unfmt(itm['price'])
});
}
total = unfmt(tf.core.Basket.chk_total);
tax = unfmt(tf.core.nfmt.round(tf.core.Basket.chk_total_tax));
}

if (!xmlBasket && items.length == 0) {
var isDetailedView = $('#extra').is('.is-detailed-view');

if (isDetailedView) {
var $itmsForAnalytics = $('script[type="application/ld+json"]');
$itmsForAnalytics.each(function () {
var itm = window.JSON.parse($(this).text().replace(/\\*.*?\\*/g, ""))[0];
if (itm && itm['@type'] == 'Product') {
var o = {
'item_id': itm.skun || itm['@id'].replace(/^Product-/),
'item_name': itm.name,
'affiliation': tf.core['shopName'],
'currency': tf.core.shopCurrency.iso
}
}
}
}
}

```

```
}  
var prc = ((itm.offers && itm.offers.price) ? itm.offers.price : 0);  
if (prc) o['price'] = unfmt(prc);  
items.push(o);  
ccy = tf.core.shopCurrency.iso;  
total = unfmt(prc);  
}  
});
```

```
if (items.length > 0) {  
  gtag('event', 'view_item', {  
    'value': total,  
    'currency': ccy,  
    'items': items  
  });  
}  
}
```

```
$(tf.core.Basket).off('Add.GA4').on('Add.GA4', function (e, itmNo, itm) {  
  if (itm) {  
    items = [];  
    items.push({  
      'item_id': itmNo || itm.itemNo,  
      'item_name': itm.title,  
      'affiliation': tf.core['shopName'],  
      'price': unfmt(itm.price),  
      'currency': tf.core.currentCurrency.iso  
    });  
    ccy = tf.core.currentCurrency.iso;  
    total = unfmt(itm.price);  
  }  
  gtag('event', 'add_to_cart', {  
    'value': total,  
    'currency': ccy,  
    'items': items  
  });  
});
```

```
$(tf.core.Favorite).off('Add.GA4').on('Add.GA4', function (e, itmNo, itm) {  
  if (itm) {  
    items = [];
```

```

items.push({
  'item_id': itmNo || itm.itemNo,
  'item_name': itm.title,
  'affiliation': tf.core['shopName'],
  'price': unfmt(itm.price),
  'currency': tf.core.currentCurrency.iso
});
ccy = tf.core.currentCurrency.iso;
total = unfmt(itm.price);
}
gtag('event', 'add_to_wishlist', {
  'value': total,
  'currency': ccy,
  'items': items
});
});
}

```

```

if (tf.pgid == 'D-6') {
  gtag('event', 'view_cart', {
    'value': total,
    'currency': ccy,
    'items': items
  });
}

```

```

$(document).off('click.remove_from_cart.GA4').on('click.remove_from_cart.GA4', function ()
{
  var idx = $(arguments[0].target.parentElement).attr('onclick');
  if (idx) idx = idx.replace(/rmv\('(.*?)'\)/, '$1');
  if (items[idx]) {
    gtag('event', 'view_cart', {
      'value': total,
      'currency': ccy,
      'items': items.slice(idx)
    });
  }
});

```

```

var idBeginCheckout = setInterval(function () {
  if (tf.utils.dlgModal.win) {
    clearInterval(idBeginCheckout);
  }
}

```



```

gtag('event', 'begin_checkout', {
'value': tf.core.nfmt.round(unfmt(tf.core.Basket.chk_total)),
'currency': ccy,
'tax': tf.core.nfmt.round(unfmt(tf.core.Basket.chk_total_tax)),
'shipping': tf.core.nfmt.round(unfmt(tf.core.Basket.shpP)),
'coupon': tf.exNo || '',
'items': items
});
}
}, 500);

}
else if (tf.pgid == 'D-8') {
gtag('event', 'conversion', {
'send_to': GA_MEASUREMENT_ID,
'value': unfmt(tf.parm['orderAmount'] || '0'),
'currency': ccy,
'transaction_id': tf.parm['orderno']
});
gtag('event', 'purchase', {
'transaction_id': tf.parm['orderno'],
'affiliation': tf.core['shopName'],
'value': total,
'currency': ccy,
'tax': tax,
'shipping': shipping,
'coupon': coupon,
'items': items,
'user_data': {
'email': tf.parm['email'],
'email_hashed': window.sha256(tf.parm['email'])
}
});
}

_sfAnalyser.processing = false;
}
};
if (tf.pgid == 'D-8') {
_sfAnalyser.init();
}
}

```

```

else {
$(function () {
_sfAnalyser.init();
_sfAnalyser.run();
});
}

var sha256 = function sha256(ascii) {
function rightRotate(value, amount) {
return (value>>>amount) | (value<<(32 - amount));
};

var mathPow = Math.pow;
var maxWord = mathPow(2, 32);
var lengthProperty = 'length'
var i, j; // Used as a counter across the whole file
var result = ""

var words = [];
var asciiBitLength = ascii[lengthProperty]*8;

/* caching results is optional - remove/add slash from front of this line to toggle
// Initial hash value: first 32 bits of the fractional parts of the square roots of the first 8
primes
// (we actually calculate the first 64, but extra values are just ignored)
var hash = sha256.h = sha256.h || [];
// Round constants: first 32 bits of the fractional parts of the cube roots of the first 64
primes
var k = sha256.k = sha256.k || [];
var primeCounter = k[lengthProperty];
*/
var hash = [], k = [];
var primeCounter = 0;
/*/

var isComposite = {};
for (var candidate = 2; primeCounter < 64; candidate++) {
if (!isComposite[candidate]) {
for (i = 0; i < 313; i += candidate) {
isComposite[i] = candidate;
}
}
}

```

```

hash[primeCounter] = (mathPow(candidate, .5)*maxWord)|0;
k[primeCounter++] = (mathPow(candidate, 1/3)*maxWord)|0;
}
}

ascii += '\x80' // Append C' bit (plus zero padding)
while (ascii[lengthProperty]%64 - 56) ascii += '\x00' // More zero padding
for (i = 0; i < ascii[lengthProperty]; i++) {
j = ascii.charCodeAtAt(i);
if (j>>8) return; // ASCII check: only accept characters in range 0-255
words[i>>2] |= j << ((3 - i)%4)*8;
}
words[words[lengthProperty]] = ((asciiBitLength/maxWord)|0);
words[words[lengthProperty]] = (asciiBitLength)

// process each chunk
for (j = 0; j < words[lengthProperty];) {
var w = words.slice(j, j += 16); // The message is expanded into 64 words as part of the
iteration
var oldHash = hash;
// This is now the undefinedworking hash", often labelled as variables a...g
// (we have to truncate as well, otherwise extra entries at the end accumulate
hash = hash.slice(0, 8);

for (i = 0; i < 64; i++) {
var i2 = i + j;
// Expand the message into 64 words
// Used below if
var w15 = w[i - 15], w2 = w[i - 2];

// Iterate
var a = hash[0], e = hash[4];
var temp1 = hash[7]
+ (rightRotate(e, 6) ^ rightRotate(e, 11) ^ rightRotate(e, 25)) // S1
+ ((e&hash[5])^((~e)&hash[6])) // ch
+ k[i]
// Expand the message schedule if needed
+ (w[i] = (i < 16) ? w[i] : (
w[i - 16]
+ (rightRotate(w15, 7) ^ rightRotate(w15, 18) ^ (w15>>>3)) // s0
+ w[i - 7]

```

```

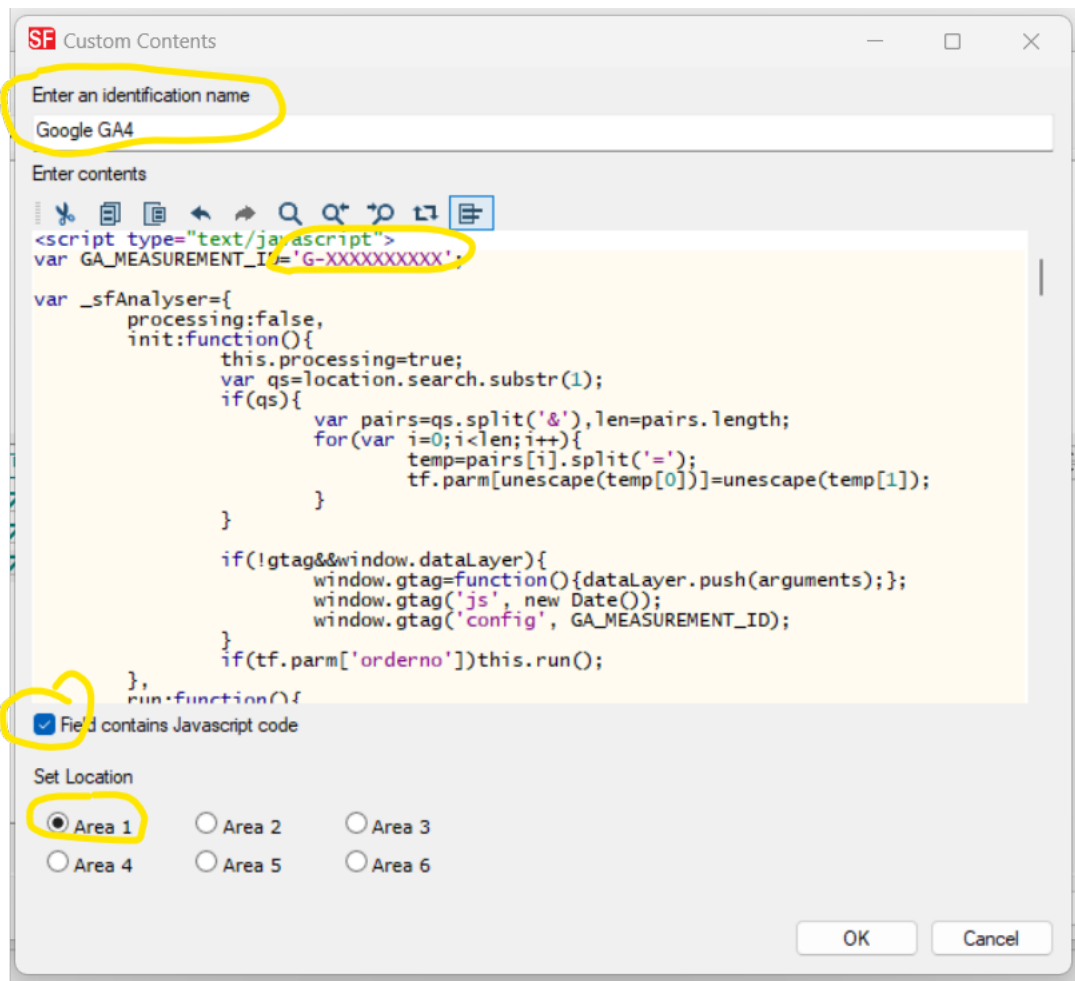
+ (rightRotate(w2, 17) ^ rightRotate(w2, 19) ^ (w2>>>10)) // s1
)|0
);
// This is only used once, so *could* be moved below, but it only saves 4 bytes and makes
things unreadable
var temp2 = (rightRotate(a, 2) ^ rightRotate(a, 13) ^ rightRotate(a, 22)) // S0
+ ((a&hash[1])^(a&hash[2])^(hash[1]&hash[2])); // maj

hash = [(temp1 + temp2)|0].concat(hash); // We don't bother trimming off the extra ones,
they're harmless as long as we're truncating when we do the slice()
hash[4] = (hash[4] + temp1)|0;
}

for (i = 0; i < 8; i++) {
hash[i] = (hash[i] + oldHash[i])|0;
}
}

for (i = 0; i < 8; i++) {
for (j = 3; j + 1; j--) {
var b = (hash[i]>>(j*8))&255;
result += ((b < 16) ? 0 : " ") + b.toString(16);
}
}
return result;
};
</script>

```



8. Click OK, Click OK again, and click Finish. Now save your shop and Publish it to the Internet.

NOTE: Accepting PayPal payments?

To ensure your orders paid in PayPal are correctly tracked by Google - setup up the Auto Return URL so buyers are immediately redirected back to your shop.

<https://shopfactory.deskpro.com/en/kb/articles/en-paypal-payments-setting-up-auto-return-url-so-buyers-are-immediately-redirected-back-to-shop>

With Auto Return for Website Payments, your buyers are redirected back to your site immediately after clicking the Pay button on the Payment Confirmation page.

Refer to this PayPal article for more information,

https://www.paypal.com/au/cgi-bin/webscr?cmd=p/mer/express_return_summary-outside