

VirtueMart Template System	2
Users Guide	2
<i>Introduction</i>	<i>2</i>
<i>Themes</i>	<i>2</i>
<i>Template Fields</i>	<i>3</i>
<i>Predefined Global Variables</i>	<i>3</i>
<i>Configuration Fields</i>	<i>3</i>
<i>Coding & PHP</i>	<i>4</i>
Configuration	5
Shop Pages	6
<i>Shop.index</i>	<i>7</i>
<i>shop.browse</i>	<i>9</i>
Supplementary Template Files	10
<i>categoryChildlist.tpl.php (../templates/common)</i>	<i>10</i>
<i>recent.tpl.php (../templates/common)</i>	<i>11</i>
<i>featuredProducts.tpl.php (../templates/common)</i>	<i>12</i>

VirtueMart Template System

Users Guide

Introduction

This, as the title suggests, is a users guide to VirtueMart’s template system. It is not a guide to writing html, php or css or for Joomla templates. Hopefully, it will show which templates each shop page uses and the relationship between template files. This should make it easier for you to locate and edit the files required to make your shop, look the way that you want it.

Some people may be dismayed that the files are not just plain html with a few extra tags added! But, using a few php commands and following a few simple rules, quite complex shop pages can be made. With some items only being displayed when they are required.

Themes

Each theme is stored in it’s own subdirectory within VirtueMart. This means you can work on a new theme and just switch for testing. Which means that you will always have a good theme to to back to.

Themes are stored in your-site/components/com_virtuemart/themes/ (your-site is the root of your-sites directory structure. For the majority of hosting companies, the sites are stored in public_html. So to access the themes on your site called MyShop will be public_html/MyShop/components/com_virtuemart/themes/..... etc).

For all these examples and explanations I shall be using the default template your-site/components/com_virtuemart/themes/default.

The basic structure of each theme is as follows, I won’t go into detail about these files:

	<ul style="list-style-type: none"> All images used by VM All the individual templates Stored values for theme config1 css file for the theme2 Javascript for theme3 php for use in theme4 (temp file for testing) xml file for theme config5
---	--

1. In the admin section certain defaults for the theme can be set. These values are stored in this file.
2. This stores all the css for the frontend of the store.
3. All javascript used by the theme are stored here. This can be extended with new functions for use by the theme.
4. Any php functions that the theme requires.
5. This displays the configuration settings in the admin section for the theme.

Within the templates are these folders:

▶ basket	Shop basket files
▶ browse	Category browse files
▶ checkout	Checkout files, shipping, payment etc
▶ common	Files that are common to many processes
index.html	
▶ order_emails	As the name suggests
▶ pages	Templates that don't apply to the other folders
▶ product_details	Flypages and all their supplemental templates

Template Fields

If you want to output the value of a variable all you need to do is add PHP brackets and echo the variable inside:

```
<!-- Print out the value of a variable -->
<p>Product Name: <?php echo $product_name ?></p>
```

Don't forget to close the PHP brackets and always use valid PHP code. If you don't, it might break your site.

Remember, you don't have to use all the available fields. It is up to you, the store owner or administrator, as to which fields you display. That's the whole point of a template system, not just deciding where to display your data but what data to display.

Predefined Global Variables

The following variables can be used in all of the template files without extra global declaration.

`$VM_LANG`

This is the global language object. It displays strings which are defined in the language file for VirtueMart.

Usage:

```
<?php echo $VM_LANG->_('PHPSHOP_FLYPAGE_LBL') ?>
```

This would print "Product Details".

`$sess`

Used to build (SEF-enabled) URLs that can be used in links.

Usage:

```
<a href="<?php echo $sess->url( $_SERVER['PHP_SELF'] . '?page=shop.product_details&product_id=' .
$product_id ) ?>">Details...</a>
```

Configuration Fields

These are the fields that are defined in the themes configuration. They are normally used to test for their condition e.g. on or off, a list of styles etc.

Usage:

```
if( $this->get_cfg( 'showFeatured', 1 ) ) {  
    //commands to execute  
    ..  
}
```

This checks for the configuration variable showFeatured, if set to 1 the commands are executed.

Coding & PHP

The first line of each template file must be:

```
<?php if( !defined( '_VALID_MOS' ) && !defined( '_JEXEC' ) ) die( 'Direct Access to '.basename(__FILE__).' is not  
allowed.' );
```

This stops the files being accessed outside of Joomla and must be included.

To begin PHP code use `<?php` and to close use `?>`

Each line of php must be ended with a semicolon unless it is a curly brace { or }

These braces are to define blocks of code e.g. a loop or an if statement. They must match otherwise the page will fail and you may end up with a blank screen!

All VirtueMart code is written in an indented fashion. So each block of code and depth of the braces can be easily seen. e.g.

```
<div>  
<?php  
foreach($something as $thing) {  
    //this is within the loop above  
    $command1 = $thing["one"];  
    $command2 = $thing["two"];  
    if($command 1 == $command2 ) {  
        //new block only executed if above is true  
        echo $command1." ".$thing["three"];  
    } //closes the true part of the if  
    else  
    {  
        echo "False";  
    } // closes the else of the if  
} //closes the loop  
>  
</div>
```

as you can see comments can be added by using `//` and php code can be placed anyway. It's just a case of matching tags just as with html tags.

Configuration

Theme Settings

/home/markfco/public_html/noctilucent/photography/components/com_virtuemart/themes/default/theme.config.php :: Writeable

Parameters

Product List Style:

Show the Feed Icon? Yes No

Show the Add-to-cart Button on the product list? Yes No

Show Vendor Link? Yes No

Show Manufacturer Link? Yes No

Show Availability Information? Yes No

Show additional Pathway on the Product Page? Yes No

Show Prev/Next Product Link? Yes No

Open Product Images in a LightBox? Yes No

Customer Checkout in a popup (GreyBox)? Yes No

Use Ajax to add, update or delete products from the cart? Yes No

Show featured products on frontpage? Yes No

Show the latest products on the frontpage? Yes No

Number of recent products to Display?

This is what is generated by theme.xml

Most of the entries are self explanatory, but some may not make sense to some store owners.

Use AJAX: this is a method of loading a page without refreshing the page. A backend process generates the page and just redisplay it.

There are three options in the product list style dropdown. This selects one of three alternative templates when displaying the category pages. These define the overall look of the category pages.

Style	Theme	Description
Product listing with a table	../templates/browse/includes/browse_layouttable.tpl.php	A basic table based layout
Flat product list	../templates/browse/includes/browse_listtable.tpl.php	A 1 product per row, table based category display
Product List, no table, div based	../templates/browse/includes/browse_notables.tpl.php	A basic div based layout

Shop Pages

Each page in a shop uses a backend file to generate all the data for the displayed page, each page will call classes that help generate the data.

Each of these pages will populate the templates and may set data in several templates. The pages I will deal with here are shop.index, shop.browse, shop.product_details and shop.cart. The page is shown in the url of the page you are working on.

e.g. http://photography.noctilucent.me.uk/index.php?page=shop.product_details&flypage=flypage.tpl&product_id=1&category_id=1&option=com_virtuemart&Itemid=53

As you can see page=shop.product_details is the file that generates the data for the page, so in the VirtueMart core we know which page created the page and for any hacks or mods that require customization will be done in this file. This won't be covered here, but it gives you an insight on page creation.

I will also cover the supplementary template files that each template uses. Several of these are shared between pages. So they will only be covered once!

Shop.index

Shop.index is the first page of the shop, which displays a description of the shop and the top level categories.

We have the best tools for do-it-yourselfers. Check us out!


We were established in 1969 in a time when getting good tools was expensive, but the quality was good. Now that only a select few of those authentic tools survive, we have dedicated this store to bringing the experience alive for collectors and master mechanics everywhere.

You can easily find products selecting the category you would like to browse above.


`$vendor_store_desc`

Categories


`$VM_LANG->_('PHPSHOP_CATEGORIES'`



Hand Tools



Power Tools



Garden Tools

Test Category

`$categories`

Recently Viewed Products

- Hammer (Category: Hand Tools)


`$recent_products`

`$ps_product->featuredProducts(true,10,false)`

Featured Products

Circular Saw


\$239.12



Cut rings around wood. This saw can handle the most delicate projects.

Nice Saw


~~\$27.05~~
\$25.05
You Save: \$2.00



This saw is great for getting cutting through downed limbs.

Hand Shovel

\$5.40



Nice hand shovel to dig with in the yard.

Main files

File	Path	Description
shop.index.php	/administrator/components/com/virtuemart/html	Page creation file
shopindex.tpl.php	../templates/common/	Main Template file

Supplementary Template files used

File	Path	Description
categoryChildlist.tpl.php	../templates/common/	Displays the sub categories
recent.tpl.php	../templates/common/	Displays the recently viewed products
featuredProducts.tpl.php	../templates/common/	Displays featured products

Available fields

Field	Template
\$vendor_store_desc	Stored description displayed on the front page
\$ps_product	pointer to the ps_product class so that it can be used within the templates.
\$ps_product->featuredProducts(true, 10,false)	This outputs the featured products using the pointer above. The format is featuredProducts(<i>random,no of products,categories</i>) i.e. random products - true, false, max number of products, category_id to display, false.
\$recent_products	Displays the recent products template
\$categories	Displays the category childlist template

Notes

There is nothing special to note about this template file. But to set the default number of categories per row edit categoryChildlist.tpl.php This will be explained the Supplementary template files.

shop.browse

Supplementary Template Files

These files are not directly used to display pages, but are called by the main templates to keep a consistent look and feel across the whole store. Many of these files will include some basic php to help display the template. I will try to explain any significant lines to make it easier for you to customise.

categoryChildlist.tpl.php (../templates/common)

This file is used to display the sub-categories of the current category. It also checks the current categories per row and if not set defaults it to 4. This can obviously be changed by editing the file.

For those that are not used to php, i'll try to explain what each section does.

Line 6 `$categories_per_row = 4;`

This sets the default number of categories per row, changing the number obviously cahnges this.

```
Lines 10-12  if( empty( $categories ) ) {  
              return; // Do nothing, if there are no child categories!  
            }
```

This just checks to see if there are any sub categories, if not it just stops there and returns with an empty category list.

Line 17 `foreach($categories as $category) {`

This just sets up a loop to display each sub category.

```
Line 26      <a title="<?php echo $category["category_name"] ?>" href="<?php $sess-  
>purl(URL."index.php?option=com_virtuemart&amp;page=shop.browse&amp;category_id=".  
$category["category_id"]) ?>">
```

This line sets up the link from the category list to the actual category displayed.

Available Fields

Each field is referenced using `$category["field_name"]`

Field Name	Description
category_name	Name of the current sub-category
category_id	The category_id of the current sub-category
category_thumb_image	The url of the sub-category's thumb image
number_of_products	The total number of products in this sub-category

recent.tpl.php (../templates/common)

This template file displays the recently viewed products, the number of which is set in the configuration page.

```
Line 2      if( empty( $recent_products)) return;
```

This just checks to see if there are any recently viewed products, if not it just stops there and returns with an empty list.

```
Line 8      foreach( $recent_products as $recent ) {
```

This just sets up a loop to display each recently viewed product.

```
Lines 9-12  foreach( $recent as $attr => $val ) {
            $this->set( $attr, $val );
        }
```

These lines assign the field name to each recent product and then the value to that field name. It may seem a little bit alien, but there is nothing to worry about. Just remember not to delete it or the loop above. Basically, leave well alone, it works and that's all you need to know.

```
line 30-32  <?php
            }
            ?>
```

This basically closes the loop started on line 8. So everything in between is done for each recent product.

Available Fields

Each field is referenced using `$recent["field_name"]`

Field Name	Description
product_name	The name of the recently viewed product
category_name	The category name in which the product was viewed
product_thumb_image	The thumbnail image of the product
product_url	The direct link to the product
category_url	The direct link to the category
product_s_desc	The short description of the product

Notes

The standard file just shows a list of the product name with a link then "Category" with a link to that category. But the available fields allow more to be displayed e.g. a thumbnail which could encompass the product name with a link to the product with the short description beside it without a link to the category.

It could also display the recent products across the page, not just in a list. It really is down to the store owner as to how these are displayed.

featuredProducts.tpl.php (../templates/common)

This template file displays any product which has its status flagged as special (featured product). It also determines how many featured products to show per row. This template also use another template ../templates/browse/includes/addtocart_form.tpl.php

```
line 2      $featured_per_row = 2;
```

Does what is says, sets the number of products to display per row.

```
Lines 9-11  if( empty( $featured_products ) ) {
            return; // Do nothing, if there are no Featured!
        }
```

Returns if there are no featured products.

```
Line 13     foreach( $featured_products as $featured ) {
```

This just sets up a loop to display each featured product.

```
Line 16     <a title="<?php echo $featured["product_name"] ?>" href="<?php $sess-
>purl(URL."index.php?option=com_virtuemart&page=shop.product_details&flypage="
.$featured["flypage"]."&product_id=".$featured["product_id"]) ?>">
```

This outputs a link to the products flypage/detail page.

```
Line 21     <a title="<?php echo $featured["product_name"] ?>" href="<?php $sess-
>purl(URL."index.php?option=com_virtuemart&page=shop.product_details&flypage="
.$featured["flypage"]."&product_id=".$featured["product_id"]) ?>">
```

This starts to output another link to the product page

```
Line 22-23  <?php echo ps_product::image_tag( $featured["product_thumb"], "class=
\"browseProductImage\" border=\"0\" alt=\"".$featured["product_name"]."\"");
?></a><br /><br />
```

This correctly outputs the thumb image in an image tag and resizes if necessary.

Available Fields

Each field is referenced using `$featured["field_name"]`

Field Name	Description
product_name	The name of the featured product
product_sku	sku of the featured product
product_thumb	The thumbnail image of the product
product_id	The actual product_id of the product
product_price	The calculated price, including discounts etc
product_s_desc	The short description of the product

Field Name	Description
product_url	Direct link to the products flypage
product_full_image	Full image of the product
flypage	Flypage of the featured product
form_addtocart	Created by the template file ../templates/browse/includes/addtocart_form.tpl.php
has_addtocart	Set if the product has an addtocart button
i	count for the total number of featured products
product_in_stock	number of products held in stock
ps_product_attribute	pointer to the class ps_product_attribute.php