

# **AShop V Design Guide**

AShop Software © Copyright 2011

# Table of Contents

	0
<b>Part I Overview Of AShop</b>	<b>3</b>
<b>Part II Designguide</b>	<b>3</b>
1 Page Templates.....	4
Design Your Page Templates .....	6
Placeholder Codes .....	7
2 Modular Storefront.....	8
Main Storefront Modules .....	9
Newsletter Module .....	11
Category List Module .....	13
Language Selection Module .....	14
The TopForm Module .....	15
3 TopLists.....	17
4 Style Sheets - CSS.....	19
5 Themes .....	20
6 Receipt Templates.....	22
7 Message Templates.....	25
8 Multiple Languages.....	27
9 Buy Buttons or Links.....	28
<b>Index</b>	<b>29</b>

# 1 Overview Of AShop

**AShop is a shopping cart program**, which can be used in a variety of different ways to manage and sell products and services over the Internet.

AShop can be used to sell many different types of products and services:

- Tangible Products
- Downloadable Products
- Services, including billing for custom work
- Access to Password Protected Areas of a Website
- Key/PIN codes
- Auctions and penny auctions
- Wholesale

AShop is a commercial software product, which is licensed per domain. It is delivered as IonCube encoded PHP scripts.

We offer a 30 day trial to let anyone try the software before they buy it.

Get [Trial Key](#) and [Download Files](#).

## Multi Vendor Shopping Mall

The Shopping Mall features of AShop lets other businesses and individuals sell their products through your online shopping cart. It includes a system for members to join, manage their products and earn commissions through a central order processing, payment processing, and digital product delivery system. Together with the affiliate tracking features of AShop you can use this to set up a service similar to Clickbank on your own website.

When using the shopping mall you should decide if you are going to show the storefront as your frontpage or the mall.php script, which generates a list of all available shops in the mall. Designing for a shopping mall site can also be quite different from designing for a regular shopping cart site. The vendor signup page, the shop list and possibly also the affiliate pages might need a more prominent position in the design.

# 2 Designguide

This guide focuses on the aspects of designing an AShop shopping cart site. It is meant for shop owners who want to customize the look of their own online shop but also for web designers who wish to design AShop based sites for their clients or even sell commercial layout themes for the software.

You will only find information related to layout and design in this document. To learn more about how to use AShop refer to the online help, which is linked from the AShop admin panel menu.

## 2.1 Page Templates

The AShop page templates are simply HTML pages with the AShop start and stop tags inserted where the AShop content is to appear. They can include HTML, images, links, CSS, and JavaScript just like any other web page but they can not include PHP or any other server side code. If you are using a WYSIWYG HTML editor such as Dreamweaver, you can add the AShop script folders to your website and create the AShop templates just as you would any other page within your website.

There are six templates used for the retail shopping cart and affiliate pages. If the wholesale shopping cart is activated, it has an additional 4 templates. The page templates can have the same or different content, depending on your needs.

Example templates are included in the distribution package for new installations. They give the shopping cart a simple but effective black and white look. This set of templates and other color variations of it can be downloaded from [AShopSoftware.com](http://AShopSoftware.com) which gives you a good start for creating your own templates.

### Retail Shopping Cart & Affiliate Page Templates

The following templates are located in the templates directory of AShop.

**aboutus.html** – used by the About Us page

**affiliate.html** – used by the affiliate sign up, login, and support pages

**cart.html** – used by the view cart page and shipping address page

**checkout.html** – used by the checkout page

**customer.html** – used by the customer profile and order history pages

**delivery.html** – used by the digital goods delivery pages

**signup.html** – used by the customer signup page

**terms.html** – used by the Terms & Conditions page

**thankyou.html** – used by the payment confirmation page

**upsell.html** – used by the upsell page

If you are not using the default advanced storefront (index.php) but prefer to use the simplified version (catalogue.php) you will also need to modify this template:

**catalogue.html** – used by the simplified product catalog and search pages

**IMPORTANT!** Some payment gateways will host your payment confirmation page through their SSL connection. When they do this, they take a snap shot of your page and nest it within their page at their URL. If this is the case, the hyperlinks within the thankyou.html template must use absolute URL's for links to work and for images to appear. In addition, image links should be https links so that browser secure/non-secure content warning messages won't appear for customers when the payment processor redirects to the thankyou (payment confirmation) page.

## Wholesale Shopping Cart & Sign up Page Templates

These templates are used by the wholesale pages.

**wscart.html** - used by the wholesale basket and checkout pages

**wscatalogue.html** - used by the wholesale product catalog and search pages

**wssignup.html** - used by pages where wholesale customers sign up for an account

**wsthankeyou.html** - used to display a thank you message page after a customer completes a wholesale order

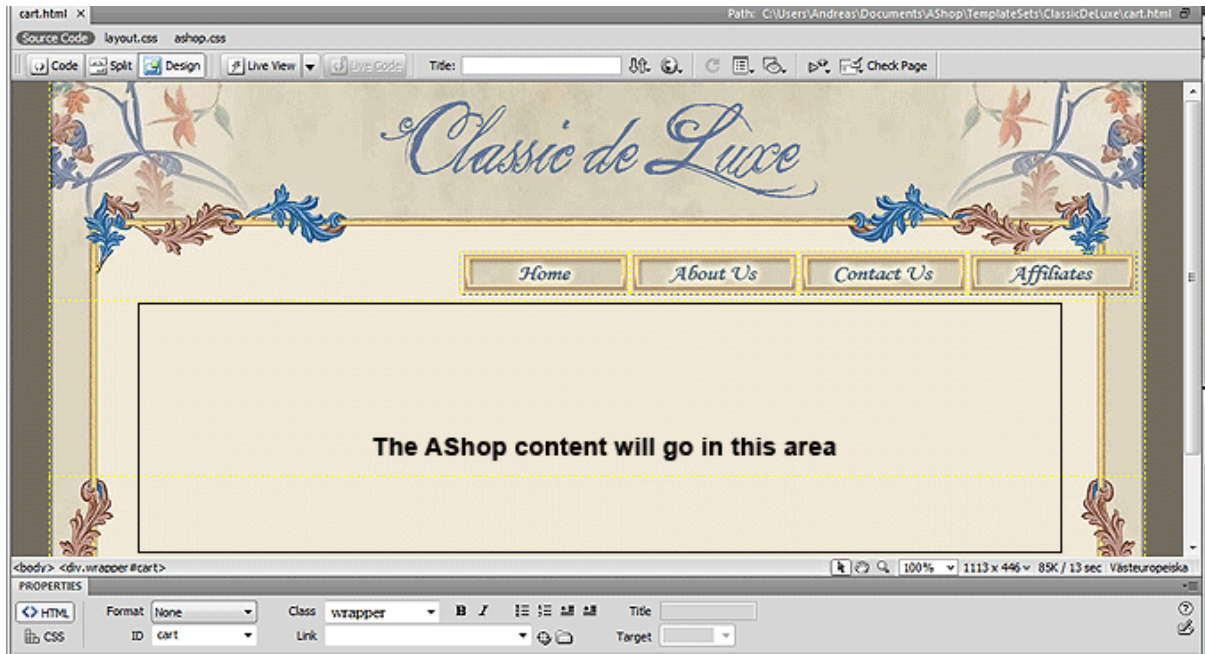
## Shopping Mall Page Templates

**mall.html** - used by mall.php, the member shop directory

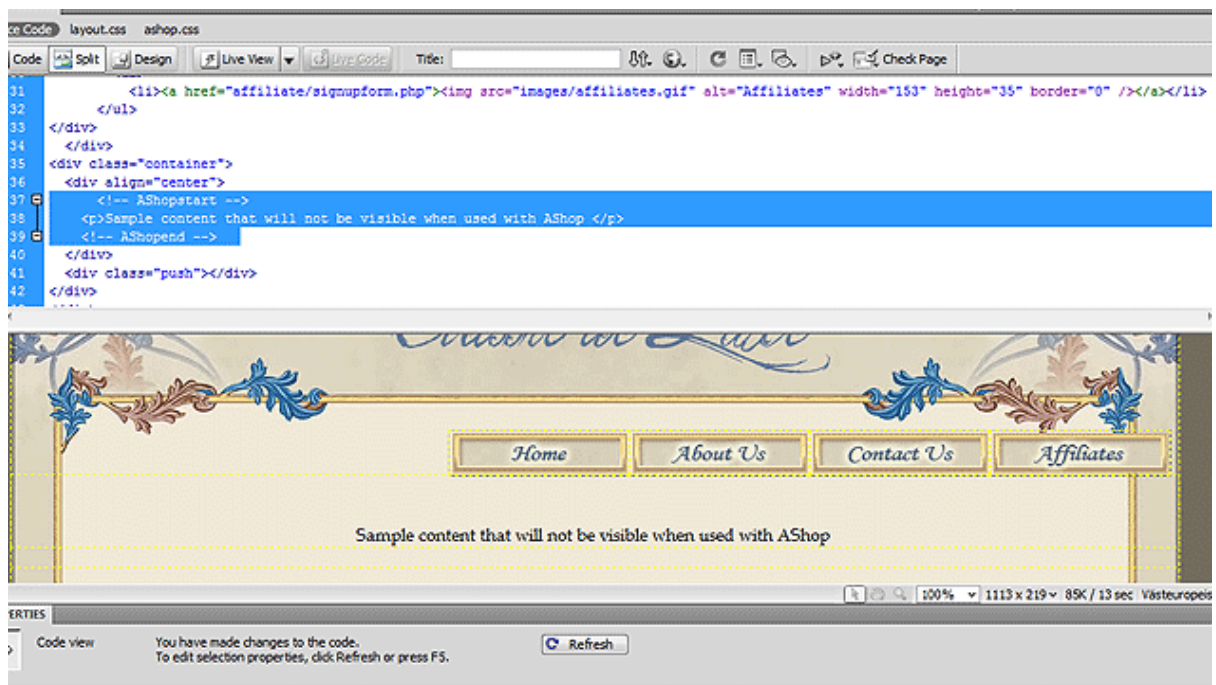
**membersignup.html** - used by pages where members sign up for an account

### 2.1.1 Design Your Page Templates

Start by designing a page with an empty area that can be used for content such as signup forms, login forms or shopping cart contents list. You can use any web page editor or design tool you like, for example Dreamweaver or Microsoft Expression Web. Here is an example of a page template designed in Dreamweaver:



Add the two comment tags: `<!-- AShopstart -->` and `<!-- AShopend -->` to the empty area of the page. All the content generated by AShop will be displayed between those comments. To make it easier to design your page and see how it will look when it is being used, you can put sample content between those comments. This will be discarded by AShop and replaced by its signup forms, shopping cart contents and other content.



**Note:** The templates must use paths relative to the root folder of your AShop, not the templates folder. If you design your templates while keeping them in the templates folder of a local copy of AShop you may need to remove "../" on image src parameters, link and CSS href parameters and any other code that refers to a local file, before publishing and using the templates. It might be easier to just move the templates up to the top folder of your site and then put them back in the templates folder when you are done designing them.

AShop is designed and tested with the doctype tag set to the following:

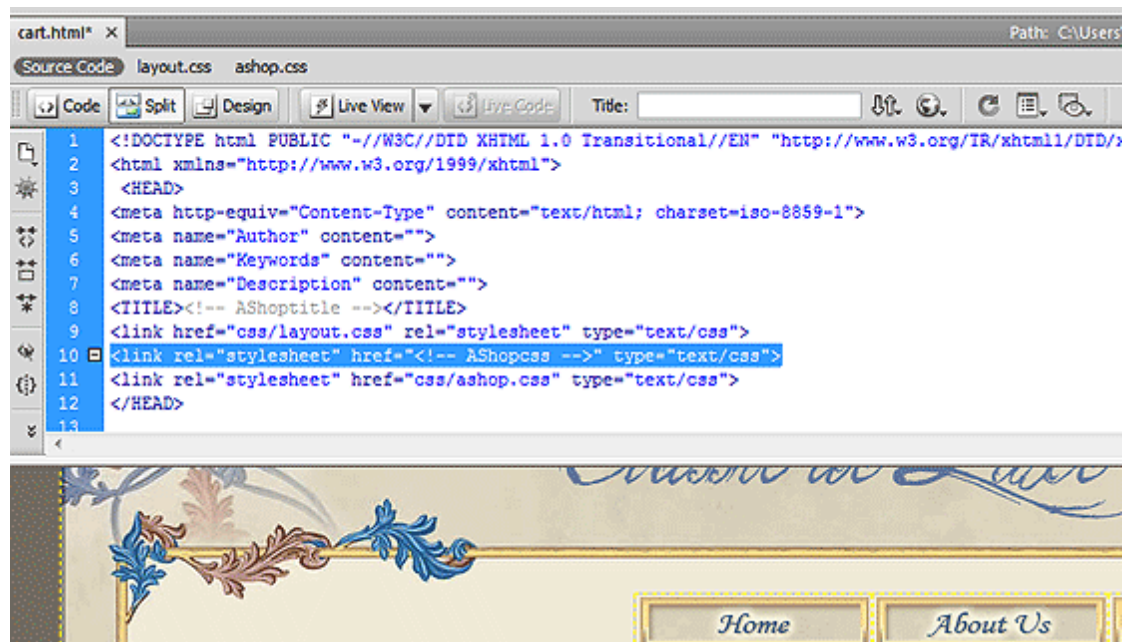
```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

It is recommended that you start your templates with that tag since using other doctypes can give unexpected results. The doctype tag should be placed at the top of the html source code of your page.

## 2.1.2 Placeholder Codes

In addition to the start and end tags you will also need to add some code to make the templates use the AShop CSS file. Without this the layout and design elements of your AShop will not be activated and the shopping cart will be presented as unformatted text. Add the following line to the head tag of your templates to use the dynamic CSS file:

```
<link rel="stylesheet" href="<!-- AShopcss -->" type="text/css" />
```



You can override any of the styles generated by the AShop CSS script to tweak and customize all the design elements in great detail. A copy of all the CSS classes is included in the sample `ashop.css` file, found in the `css` folder. Just make sure you put any style tags or custom CSS files below the default AShop CSS tag in your templates and storefront files to make them override the default classes.

There are some other comment codes available that you can use in your page templates:

- `<!-- AShopname -->` - displays the active Shop Name
- `<!-- AShoptitle -->` - displays the page title, see the example above where this is used in the `<TITLE>` tag of the page.
- `<!-- AShopURL -->` - displays the URL to the shopping cart
- `<!-- AShopmetakeywords -->` - displays the meta keywords for the current page
- `<!-- AShopmetadescription -->` - displays the meta description for the current page
- `<!-- AShoplogo -->` - displays the shop's Logo Image
- `<!-- AShopcart -->` - displays a shopping cart box
- `<!-- AShopcategories -->` - lists the product categories **Note!** Use this one with caution. It is not compatible with pages in subfolders, the delivery pages or the customer profile pages.
- `<!-- AShopbgcolor -->` - the page background color
- `<!-- AShopmemberheader -->` - a custom header defined by a shopping mall member

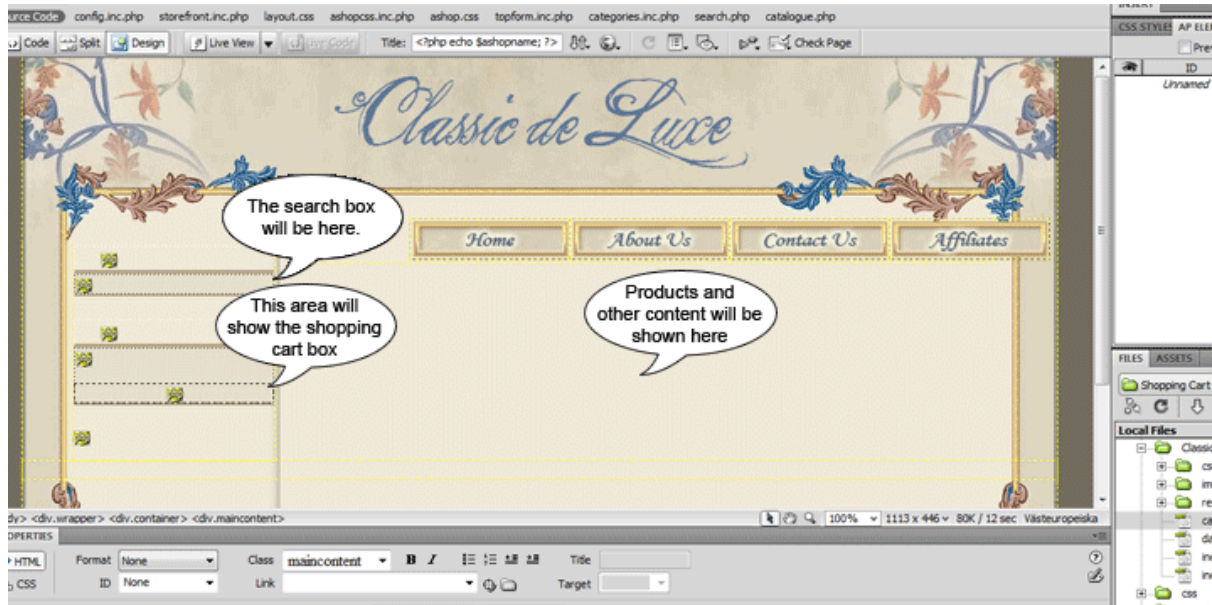
## 2.2 Modular Storefront

The `index.php` script is an example of a storefront for AShop. It doesn't need to be named `index.php` and you can actually create multiple storefronts if you want to. By using the various content modules found in the `includes` directory you can put together the frontpage of your shopping cart any way you like. This way the storefront is an exception from how the pages usually work in AShop, in that it is its own page template. It is also one of the few scripts included with AShop that isn't ioncube encoded, which means that you can edit it, for example with a web page editor like Dreamweaver or a text editor.

To design your storefront you use a similar approach to when you design page templates, but rather than preparing just one main area for all AShop generated content, you need to prepare areas for all the



different types of content modules you wish to include on your storefront: product categories, product list, search box, shopping cart box, newsletter subscription box... Here is an example designed in Dreamweaver:



As you can see we have prepared areas for various types of content. The small yellow icons are shown by Dreamweaver when there is a PHP code section on a page. They represent the PHP code we used to actually get the AShop content to display in those areas.

Save your storefront file with the filename `index.php`. You can actually use any filename, but that is the default name of the storefront in AShop. The most important thing is that you save it with a file extension (the part after the dot) of `"php"` rather than `"html"` or `"htm"`.

## 2.2.1 Main Storefront Modules

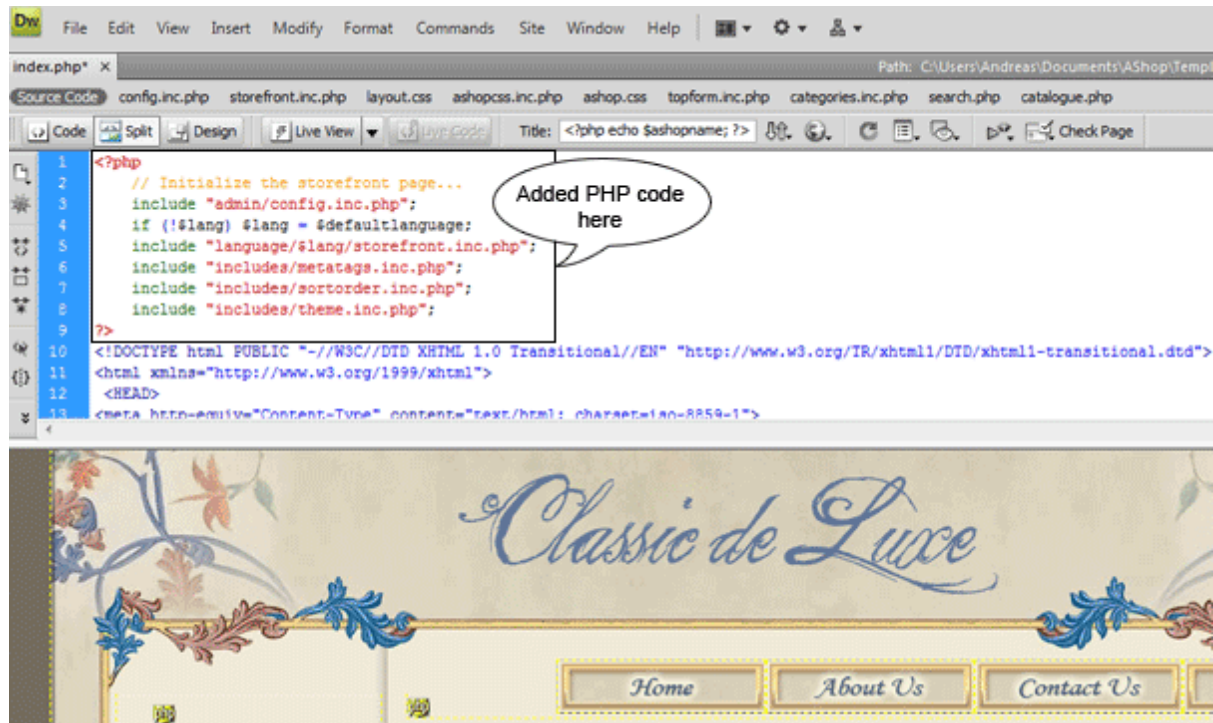
To make the storefront page work with AShop you need to add some PHP code to it. If you don't know your way around PHP programming, you can just copy and paste the code segments from this guide into your storefront page. You may need to switch to code view in your web page editor to be able to add the PHP code.

The storefront script always needs to begin with the following PHP code:

```
<?php
    // Initialize the storefront page...
    include "admin/config.inc.php";
    if (!$lang) $lang = $defaultlanguage;
    include "language/$lang/storefront.inc.php";
    include "includes/metatags.inc.php";
    include "includes/sortorder.inc.php";
    include "includes/theme.inc.php";
?>
```

This is how it would look when you add the PHP code above to your storefront in Dreamweaver's split

Design/Code view:



This makes the script ready to use the content modules and also makes it multilingual, as long as you put all text used in the page in the storefront.inc.php language module. To use the text from the language module you need to replace it with this type of code:

```
<?php echo HOME; ?>
```

Where the "HOME" part should be the name of the language constant you want to use, in this case it will show the word "Home" in the language selected by the visitor. Open the storefront.inc.php language module (you will find the English version in the folder language/en in your AShop) in a text editor to see which constants are available and to add your own constants if needed. If you will only be using one language in your shopping cart, you can just keep the text of your storefront page unchanged and skip this.

### Displaying the product list or search results

The catalogue.php script shows the list of products for a selected category while the search.php script shows the results of a search. They can either be used standalone or as part of a modular storefront script. To use them in a storefront script add the following code:

```
<?php
    // Show product list or search results...
    $topform = "off";
    $categories = "off";
    if($searchstring) {
        $search = "index.php";
        include "search.php";
    } else if ($product) {
        $catalog = "index.php";
```

```

        include "product.php";
    } else {
        $catalog = "index.php";
        include "catalogue.php";
    }
?>
```

By adding the line: `$topform = "off"`, the output will not include a box at the top of the list, with a search field, the subtotal of the shopping cart contents and the View Cart/Checkout buttons. Remove that line if you want that box to be shown instead of putting those parts in separate places on your page.

By adding the line: `$categories = "off"`, the output will not include a categories list to the left of the products/search results. Use this if you want to place the categories using the categories list module instead or remove it if you want the categories to be shown together with the product/search results.

The \$search parameter must be set to the filename of the script that shows the search results and the \$catalog parameter must be set to the filename of the script that shows the product list. Both should be set to "index.php" when you use the default storefront script.

## Using Custom Modules

The storefront script can of course show any other module or dynamic content you want to use. As an example of this the default `index.php` script includes a SimplePie RSS reader, which can show news feeds from an external site or even article links from your own blog. This is the code that runs this script:

```
<?php
// Demonstrates how to include a news feed reader on your storefront...
include "includes/simplepie.inc.php";
$feed = new SimplePie();
$feed->set_cache_location("./updates");
$feed->set_feed_url($ashopnewsfeed);
$feed->init();
if ($feed->data) {
    $items = $feed->get_items(0,5);
    foreach($items as $item) echo " &nbsp;&nbsp;<img src=\"images/bullet.
gif\"> &nbsp;&nbsp;<a href=\"\". $item->get_permalink().\"\" target=\"_blank\">\". $item-
>get_title().\"</a><br>";
}
?>
```

You can change the news feed to any RSS news feed of your choice through the setting labeled "News Feed" on the page Configuration->Shop Parameters in your AShop admin panel. The default is to show any links you have added to your AShop catalog.

### 2.2.2 Newsletter Module

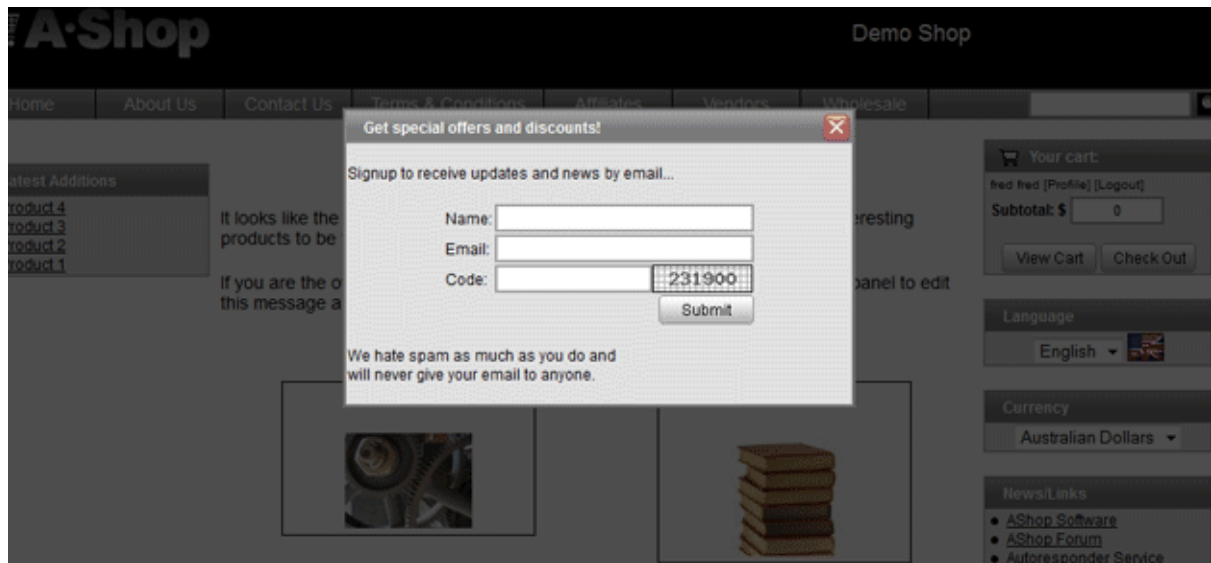
If you are using the Autoresponder-Service.com account that is included for free with your full AShop V license, you can enable the newsletter module to let visitors sign up with your mailing list, which can of course be an autoresponder. You need to select the list you wish to use for this on the Configuration->Shop Parameters page to make it work. There should be an option labeled "Newsletter Autoresponder" on that page with a drop down list showing all your available autoresponders/lists.

By using the DHTML popup mode of the newsletter module you can make the storefront page ask every new visitor to sign up with your list before they start shopping. The whole page will be greyed out while a signup box shows on top of it. To enable this, put the following code directly after the BODY tag of your

storefront script:

```
<?php
    // Show newsletter subscription form...
    $layout = 2;
    $subscribe = "index.php";
    include "includes/newsletter.inc.php";
?>
```

The \$subscribe parameter must be set to the filename of the page you are using this code on, which would be index.php when it is used on the storefront. This is how it will look to a new visitor on your site:



To just show a form somewhere on the page, use the following code in the location where you want the form to show up:

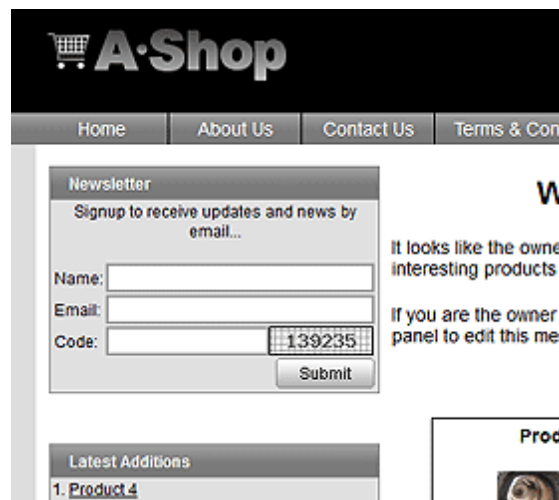
```
<?php
    // Show newsletter subscription form...
    $layout = 1;
    $subscribe = "index.php";
    include "includes/newsletter.inc.php";
?>
```

You can also use the following optional parameter to disable the captcha code:

```
$captcha = "off";
```

The captcha parameter must be set before the include statement.

Here is an example of the newsletter signup form:

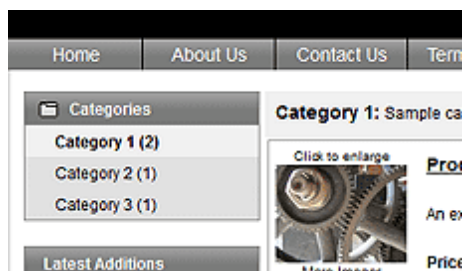


### 2.2.3 Category List Module

To list your product categories you can use the categories.inc.php module. For example:

```
<?php
    // Show categories...
    $catalog = "index.php";
    include "includes/categories.inc.php";
?>
```

The name of each category is a link that should lead to a page that shows the products that belong to the category, the \$catalog parameter determines which page the links lead to and when used in the default storefront script it should be set to index.php.



To create a custom list of categories you can use the layout parameter to get the full category tree as an unordered list:

```
<?php
    // Show categories...
    $layout = "2";
    $catalog = "index.php";
    include "includes/categories.inc.php";
?>
```

The default value of the layout parameter is "1", which gives you the standard categories list in a table. Using mode "2" will, for example, let you create vertical or horizontal drop down lists, using CSS.

You can limit the listing to just one category and its subcategories by using the cattree parameter. It must be set to the category ID of the top category, for example:

```
<?php
    // Show categories...
    $cattree = "1";
    $catalog = "index.php";
    include "includes/categories.inc.php";
?>
```

Which will show category 1 with all its subcategories.

You can also exclude certain categories from the listing by using the exclude parameter, set to a pipe separated list of category IDs to exclude, for example:

```
<?php
    // Show categories...
    $exclude = "1|2";
    $catalog = "index.php";
    include "includes/categories.inc.php";
?>
```

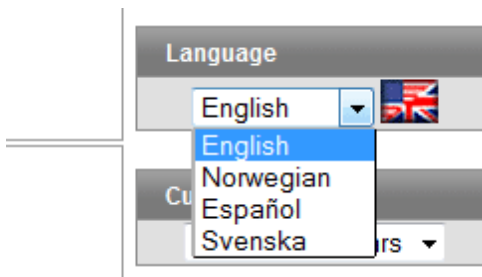
Which will skip categories 1 and 2 from the list.

## 2.2.4 Language Selection Module

This module will display a drop down list of all available languages.

```
<?php
    // Show language selection...
    $redirect="index.php";
    include "includes/language.inc.php";
?>
```

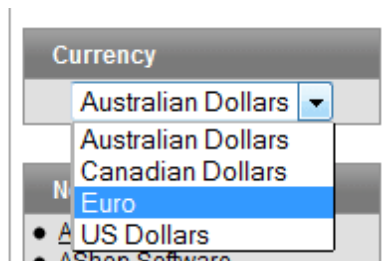
It only takes one parameter: redirect, which should be set to the page that will be shown after the customer selects a language. If you skip this parameter or leave it blank, the redirect URL configured for each language in its lang.cfg.php file will be used instead, which can be used to redirect the customer to a different page for each language.



You can also add a similar box that will allow your customers to select which currency to view your product prices in. The currency rates are automatically updated and changing to a different currency than the one you used when you set your prices should give a correct price in the selected currency. This is the code you need to add to your storefront to display the currency selection box:

```
<?php
    // Show currency selection...
    $redirect="index.php";
    $currencies="usd,cad,aud,eur";
    include "includes/currency.inc.php";
?>
```

The variable \$currencies should be set to a comma separated list of currencies you wish to display, in this case US dollar, Canadian dollar, Australian dollar and Euro. This is how it will look:



## 2.2.5 The TopForm Module

If you use the catalogue.php script as your storefront instead of the more advanced modular index.php script, it will include a form at the top of the page with the search box, the subtotal of the cart contents and the View Cart and Checkout buttons. These can be placed individually instead, using the topform.inc.php module, if you wish to use a modular storefront. Here are some examples...

This code will just show the search box:

```
<?php
    // Show search box...
    $layout = 3;
    $search = "index.php";
    include "includes/topform.inc.php";
?>
```

This one will show the subtotal field, the customer profile links and a placeholder for "added to cart" messages:

```
<?php
    // Show subtotal...
    $layout = 4;
    include "includes/topform.inc.php";
?>
```

This will show only the subtotal field, without the customer profile links or the placeholder:

```
<?php
    // Show shopping cart buttons...
    $customerlogin = "off";
```

```
$confirmmessage = "off";  
$layout = 4;  
include "includes/topform.inc.php";  
?>
```

This code will show the shopping cart buttons:

```
<?php  
    // Show shopping cart buttons...  
    $layout = 5;  
    include "includes/topform.inc.php";  
?>
```

This will show the default box, but wherever you want to place it:

```
<?php  
    // Show shopping cart buttons...  
    $layout = 2;  
    include "includes/topform.inc.php";  
?>
```

This will show only the customer profile links:

```
<?php  
    // Show shopping cart buttons...  
    $layout = 6;  
    include "includes/topform.inc.php";  
?>
```



## 2.3 Top Lists

The top products and latest additions may be displayed in shopping cart catalog and search pages by inserting PHP code in the page where it is to appear.



The scripts that produce these lists may be included in...

- [page templates](#)
- any custom web page of your choice
- the storefront index.php script or any other custom php script

There are two ways to include the lists, either through Javascript or a PHP include. This is how to use the Javascript version...

```
<script language="JavaScript" src="top.js.php"></script>
```

And this is how to use the PHP include version...

```
include "includes/toplist.inc.php";
```

Note that the second version only works in a PHP script.

Parameters may be added to both versions of the script to customize the output. To add a parameter to the Javascript version, first add a question mark "?" and then add the parameter. If more than one parameter is specified, add an ampersand "&" after the first parameter and to separate additional parameters. Here is an example.

```
<script language="JavaScript" src="top.js.php?items=5&redirect=product.html"></script>
```

To add a parameter to the PHP include version, set it as a variable before the include statement. For example:

```
$redirect="index.php";  
include "includes/toplist.inc.php";
```

### Number of Products Shown

If not otherwise specified, the top.js.php script will display the last 10 new products and latest additions. To display a lower number of products, add the parameter "items=x" where x is the number of products to display.

This include will display the 5 most popular and latest additions:

```
<script language="JavaScript" src="top.js.php?items=5"></script>
```

Or for the PHP include version use:

```
$items = 5;  
include "includes/toplist.inc.php";
```

### Redirect To A Different Product Page

In shops where only custom catalog pages are used and a direct link to the default storefront script is not desired, the top.js.php product links can be redirected to the custom catalog pages by adding a "redirect=x" parameter, where x is the local file name or URL.

This include will display the 10 most popular and latest additions in a custom product page named "product.html".

```
<script language="JavaScript" src="top.js.php?redirect=product.html"></script>
```

### The Layout Parameter

By setting the layout parameter you can control which lists are shown. These are the possible values of the parameter:

- layout = 0 : show top sellers and latest additions below top sellers (the default mode)
- layout = 1 : show only top sellers
- layout = 2 : show only latest additions
- layout = 3 : show latest additions as boxes, this mode is currently only supported by the Javascript version

The PHP include version also supports a \$mode variable which can be set to "table" (default) for a top list formatted as a HTML table, or "list" for a top list formatted as an unordered list. The "list" mode can be used together with CSS code to completely customize the look of the top list.

This example will show only the latest additions, using the PHP include, and will limit the list to 4 items:

```
$layout = 2;  
$items = 4;  
include "includes/toplist.inc.php";
```

### Colors, and Fonts

The table borders, background colors, and fonts can be controlled through the Layout menu and with [style sheets \(CSS\)](#).

## 2.4 Style Sheets - CSS

The settings in the Layout menu are applied to the styles in the AShop pages through the includes/ashopcss.inc.php script, which is read by the page templates in the templates directory and by the storefront script. You can override individual classes in this script with your own to fine tune the layout. The css/ashop.css file is an example which lists all the different classes you can change. They have self explanatory names to make it easier to identify which ones to change.

Here are a few examples:

.ashopcategoriesheader - controls the header above the list of categories, where the text "Categories" will usually appear.

.ashopsearchfield - controls the search input field, where the customer can enter keywords to search for.

.ashopitembox - the product listing boxes.

To give total flexibility we have decided to use new classes for each type of content, which means that there are a lot of classes available for tweaking, too many to list them all here. Each AShop class has a name that begins with "ashop" to set them apart from other CSS classes that may be used in the templates.

## 2.5 Themes

A theme is basically a set of templates and button images but by adding a configuration file and putting the files in a subdirectory of the themes directory, the theme can be selected from the AShop admin panel on the page Configuration->Layout which makes it possible to change the look of a whole AShop, or shopping mall member shop, by just switching to a different theme.

Each shopping mall shop can have its own theme, which makes them suitable for an AShop that uses the shopping mall feature. Another way to use themes could be to prepare holiday or seasonal versions of the shopping cart design, which can then easily be used when the time comes.

A theme can include a set of templates or a set of button images or both. It must always include a file named theme.cfg.php which defines how the theme will work. Page templates and storefront script, if included, should be in the main directory of the theme while button images and other image files used by the theme should be in a subdirectory named "images".

The theme.cfg.php file should contain the following variables:

- \$themenam - the name of the theme as shown in the drop down list where the shop owner selects which theme to use.
- \$themelanguages - an array containing the two letter language abbreviations supported by the theme, for example: array("en","sv"); which would define the theme as available in English and Swedish. Only put the languages that the theme has button images and templates for in this array.
- \$usethebuttons - when set to "true" the theme will use its own button images.
- \$usethetemplates - when set to "true" the theme will use its own templates.
- \$usecondensedlayout - when set to "true" the theme will force the use of the condensed layout mode while setting it to "false" will force the standard layout mode.

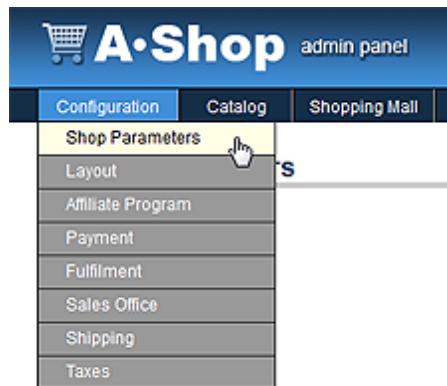
There are also a set of layout variables available, which correspond to the settings on the Configuration->Layout page of the AShop admin panel:

- \$bgcolor - set this to a six hexadecimal color code to override the current background color setting, for example: "#FFFFFF" to use a white background.
- \$textcolor - overrides the current text color.
- \$linkcolor - the text color.
- \$formsbgcolor - the background color of forms, such as the customer registration form.
- \$formstextcolor - the text color of forms.
- \$categorytextcolor - the text color of unselected categories in the categories list.
- \$selectedcategorytext - the text color of the currently selected category.
- \$catalogheader - the background color of the header at the top of the categories list, the products list, search results etc.
- \$catalogheadertext - the text color of the headers.
- \$itembordercolor - the border color of product listings.
- \$itemborderwidth - the border width of product listings.
- \$itembgcolor - the background color of product listings.
- \$categorycolor - the background color of unselected categories in the categories list.
- \$selectedcategory - the background color of the currently selected category in the categories list.
- \$font - a comma separated list of fonts to use for all text, for example: "Arial, Helvetica, sans-serif".
- \$fontsize1 - the smallest text size.
- \$fontsize2 - medium text size.
- \$fontsize3 - maximum text size.



## 2.6 Receipt Templates

Receipts in AShop can be sent in either HTML or plain text format. Select the receipt format in Configuration > Shop Parameters.



**Catalog Options**

Default language: English ▾

② Currency: US Dollars ▾

Thousand Separator: none ▾

Decimal Separator: dot ▾

② Catalog mode: Shopping cart ▾

② SEO Friendly URLs: on ▾

② Enable Wishlists: ☐

② Add To Cart Confirmation Window: ☐ Yes ☒ No

② Link From Thank You Page:

② Items per page:

② AShop Affiliate ID:

Receipt mail format: HTML ▾

Collect:  information ▾ for gifts.

Up sell: 0 ▾ items before check out.

Customer Email Confirmation: ☐

Activate Autoresponder: ☒

English language receipt templates are included in the AShop installation package. Receipt templates for other languages are [downloadable from our web site](#). The receipt templates can be edited to suit your needs. Customer and order information is included in the receipts by using %tokens%. See below for more about [editing receipt templates](#).

Order Date: 2007-01-26 Reference: 9567685		AShop Deluxe RECEIPT	
<b>Sold To:</b> Test Customer 848 N Rainbow Blvd #539 Las Vegas, NV 89107 US test@ashopsoftware.com, 702-966-9392			
<b>Qty</b>	<b>Description</b>	<b>Price</b>	<b>Amount</b>
1	Music Product	0.02	0.02
		<b>Subtotal:</b>	0.02
		<b>Tax:</b>	0.00
		<b>Shipping:</b>	0.00
<b>Paid By:</b> Authorize Net SIM (\$0.00 fee)		<b>Total:</b>	<b>\$0.02</b>
<b>Special Instructions</b>			
<b>Ship To:</b> Rob Harris 848 N Rainbow Blvd #539 Las Vegas, NV 89107 United States  Selected shipping option: Regular Ground Shipping - Included			
<b>Product File Download:</b>			
<a href="http://www.ashopsoftware.com/ashopdeluxe/deliver.html">http://www.ashopsoftware.com/ashopdeluxe/deliver.html</a> <b>Login eMail:</b> test@ashopsoftware.com <b>Password:</b> erh7s8s  If you experience any difficulty with this download or lose your connection before the download is			

## Editing Receipt Template Layout and Content

Receipt templates are in the *templates* directory. There are two receipt templates for each language; html and text. Receipt templates are named with -en, -sp, -it, -jp, and so on to indicate the language. Example: English language receipts are named receipt-en.html and receipt-en.txt. Modify the text in receipt templates as needed, but do not change or remove the divider comment tags and token names.

Here is a list of tokens and how they are used in receipt templates.

%ashopname%	The name of your store
%dateshort%	Date of the order
%invoice%	Invoice/Order number
%customer_firstname%	Customer's first name
%customer_lastname%	Customer's last name
%customer_address%	Customer's street address
%customer_city%	Customer's city
%customer_state%	Customer's state
%customer_zip%	Customer's zip code
%customer_country%	Customer's country
%customer_email%	Customer's email address
%customer_phone%	Customer's phone number
%customer_info%	Extra form fields defined by the shop owner
%receipt_description%	
%payoption%	Method of payment used
%payoptionfee%	Fee charged for using specified payment method
%subtotal%	Total before taxes and shipping
%salestax%	Tax total
%shipping%	Shipping costs
%amount%	Total cost billed
<!-- Shippingdiscount -->	Display shipping discount (if any)
%shippingdiscount%	Discount on shipping
<!-- /Shippingdiscount -->	

<!-- Specialinstructions -->	Determines whether any special instructions are to be sent with the receipt. Necessary for all the following sections of the receipt to be displayed.
<!-- Shippingaddress --> %customer_shippingaddress%	Display shipping information (if product is shippable) Customer's ship to address
<!-- Shippingoption --> %shipoptions% <!-- /Shippingoption -->	Display method of shipment (if any) Method of planned shipment
<!-- /Shippingaddress -->	
<!-- Downloads --> %ashopurl% %customer_email% %password% %downloadtimes% %downloaddays% <!-- /Downloads -->	Display information for download instructions (if applicable) Your store's location as a link for product delivery Customer's email address Password for customer to download product file Number of times download can occur Number of days download is available for
<!-- Unlockkeys --> %unlockkeys% <!-- /Unlockkeys -->	Display unlock keys (if any) Any software registration keys or PIN codes
<!-- Subscriptions --> (if applicable) %subscriptionlinks% %password% <!--/ Subscriptions -->	Display instructions on how to access the subscribed directory  Link to the password protected directory subscribed Password to access protected directory
<!-- Manualpayment --> Handling payment option. <!-- /Manualpayment -->	Display notice that the payment was made using the Manual
<!-- /Specialinstructions -->	End Special Instructions portion of the receipt.
%ashopname% %ashopaddress% %ashopemail% %ashopphone%	Your Shop Name Your Shop Address Your Shop Email Your Shop phone number
<!-- Membership -->  %membershop% <!-- /Membership -->	Displays if order came from a Shopping Mall member's catalog  Shopping Mall member's shop name



## 2.7 Message Templates

AShop sends out email messages under some circumstances and most of them can be customized by editing the corresponding message template. All message templates can be found in the templates directory, where you also find the receipt templates and page templates. These are the ones that are currently available:

- *affiliatesignupmessage.html* - this template defines the message that is sent to affiliates when they sign up.
- *membersignupmessage.html* - this template defines the message that is sent to shopping mall members when they sign up. **Note:** if you are going to have shopping mall member accounts activated automatically, without review, you will need to make sure that this message template contains the code `%password%` to give your new members their password. A second template should be included to illustrate this: *membersignupmessage2.html*. Just rename it, removing the "2", to use it instead of the default one.
- *orderactivationmessage.html* - this is sent to customers when their order is activated. It is used for orders where the payment is delayed, for example check payments or bank deposits, which will result in an activation link in the order notice that the shop owner receives, which can be clicked to activate the order upon full payment.
- *signupmessage.html* - this is sent to customers when they sign up for a customer profile.
- *wssignupmessage.html* - this is sent to customers when they sign up for a wholesale account.
- *outbidmessage.html* - this is sent to customers when they are outbid on an auction.
- *winningbidmessage.html* - this is sent to customers when they win an auction.

Each of these templates can be made language specific by adding a dash and the two letter language code after the word "message", for example: *signupmessage-sp.html* which would be sent to Spanish customers when they sign up for a customer profile, or *affiliatesignupmessage-jp.html* which would be sent to Japanese affiliates.

The order activation message template can use the same codes that are available for the [receipt templates](#). The other message templates can use the following codes to personalize the text:

- `%ashopname%` - replaced with the name of the shop.
- `%firstname%` - replaced with the first name of the recipient.
- `%lastname%` - replaced with the last name of the recipient.
- `%username%` - the username.
- `%password%` - the password.
- `%email%` - the email address of the recipient.

The wholesale, affiliate and shopping mall signup messages can use these additional codes:

- `%address%` - the postal address of the recipient.
- `%state%` - the recipient's state or province.
- `%zip%` - the recipient's zip/postal code.
- `%city%` - the recipient's city.
- `%country%` - the recipient's country.
- `%phone%` - the recipient's phone number.
- `%url%` - the recipient's website URL, if applicable.
- `%date%` - the date the recipient signed up.

The following codes are only available in the affiliate signup message:

- `%referralcodes%` - the manual referral code for affiliates, if applicable..

%loginlink% - a link to the affiliate area, if applicable..

%affiliateurl% - the default affiliate link URL for the new affiliate, if applicable..

%business% - the business name of the recipient, if applicable.

The following codes are only available in the shopping mall signup message:

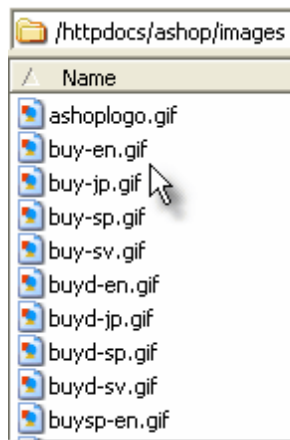
%shopname% - the name of the shopping mall shop.

%description% - a description of the shopping mall shop.

## 2.8 Multiple Languages

There are three areas that have language specific files. Each language is identified by a 2 character abbreviation, which is used for language subdirectory name, for image file names, and in receipt.html file names. For instance; en for English, sv for Swedish, etc.

1. **Each language should have its own set of button images**, which are all kept in the images directory. The language for button images is identified by the 2 character suffix. For instance, buy-en.gif is the name of a buy button image that is used for English language.



2. **Each language should have its own set of templates**. The templates are located in the templates directory. Each of them can be made language specific by adding the two letter language abbreviation to the filename, like this: cart-sp.html which would be the Spanish language template for the shopping cart pages. The message and receipt templates can be made language specific this way too.

## 2.9 Buy Buttons or Links

Each product has a small chain link icon in the top right corner of the product listing on the page Edit Catalog in the AShop admin panel. Clicking that icon will bring up a popup window with two code boxes. The first one can be used to create a simple link to buy the products while the second one results in a form with a submit button that does the same thing.

This feature can be used to create Buy Now buttons and use them in any web page, both on the same site as the AShop installation and on external sites. This way you can create squeeze pages, sales letters, minisites, splash pages or any other web page where you want to sell your products and link them to AShop to let your customers buy the products without first locating them in the AShop product catalog.

# Index

## - B -

buy buttons 28

## - C -

customization 4, 22, 27, 28

## - I -

Includes 17

installation 4, 22, 27, 28

## - L -

language 27

## - N -

navigation 28

## - O -

OrderLink 28

overview of AShop Deluxe 3

## - P -

page labels 27

page templates 4

Products 17

## - R -

receipt templates 22

## - S -

setup shopping cart 4, 22, 27, 28

## - T -

Templates 17

top.js.php 17