easyPartnerShop

0.0.6

Generated by Doxygen 1.7.5

Mon Jun 11 2012 01:49:53

Contents

Chapter 1

easyPartnerShop Documentation

Welcome to *easyPartnerShop* Documentation.

There are so many shop systems in the internet. Look at Amazon, Spreadshirts or - Fotoportopro ... to count up all of them would just eat up my time. The point is that all those providers give you a free partner shop where they offer you the goods to sale, but then you have a shop with sublevel domain or even a long URL in a subfolder. So your customers will not find you and you will not make money. If one customer comes over your shop, they mostly go then to the original shop and you will not make money. This system is not a waranty that a customer orders in your shop, *easyPartnerShop* is developed so you can install those partner shops on your own domain, at a webhoster of your choice, with a easy Setup. Yes there are some partner shops that offer you a own domain, but mostly they are expensiv.

1.1 What is easyPartnerShop

easyPartnerShop is a PHP script with an XML setup and a XHTML theme system that help you to install all your partner shops in one install under your own domain. Once you Installed *easyPartnerShop* you can add any time a new shop within a few minutes.

easyPartnerShop is a script that you can install on your webserver and include your partner shop like Amazon.

With *easyPartnerShop* you can include as many partner shops as you like in one installation on your webspace. It does not matter if Amazon, Spreadshirts or any other shop. Just configure your shop and you are good to go.

Before installing the shop you should check if your webserver meets the Requirements.

This manual for *easyPartnerShop* will show you how to Install and Setup your own shop. It is real easy to set it up, but your webserver should meet the Requirements.

1.2 License

The shop license.

- GNU General Public License 3.0 http://yoursl.in/GPL

```
*
* This program is distributed in the hope that it will be useful,
* but WITHOUT ANY WARRANTY; without even the implied warranty of
* MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
* GNU General Public License for more details.
*
```

1.3 Requirements

What are the requirements to run *easyPartnerShop*.

You need a webserver, webspace that will have PHP5 and the possibility to route all domain or sublevel domains on one folder.

Attention

> This shop will not run on subfolders, you have to configure for each shop a own domain

If you can not configure your domains that way, you only can run one shop in the system

- PHP5
- Webserver / Webspace
- Be able to configure your webspace that all domains go on one folder
- A partner shop like Amazon or Spreadshirts

1.3.1 Webserver

The Webserver.

You have to setup your server that all domains or sublevel domains point to the install folder of the shop.

Attention

> No Webforward or Alias, use a A-Record or CNAME and set it up as VirtualHosts in Apache

1.4 Features

Shop features.

Setup with one install as many shops as you like, each one with its own design or all with the same. Every shop will have a own domain or sublevel domain.

- No Database needed
- One install, as many shop as you like
- Every shop with a own design
- XHTML Theme, easy to make your own
- easy setup, one php file and the XML files
- use a Webhoster of your choice
- No costs for *easyPartnerShop* because of its License

Attention

If you like *easyPartnerShop* please consider to donate

1.5 Install

How to install *easyPartnerShop*.

It easy to install the shop, you can do it in a few steps. Before you install *easyPartnerShop* please look at the Setup and go through those steps. When you are finished with the Setup, you come back here.

Attention

Did you allready follow the documentation, and your Setup is finished? If yes, keep going in this part.

Now you just upload the contents in this folder to the root of your webserver. To do this you need a FTP program.

Now you are finished and your newly installed *easyPartnerShop* is running.

Attention

If you find a bug, or something that you think is not correct, see the Bugs section

1.5.1 Setup

How to Setup *easyPartnerShop*.

After you finished with the Install of *easyPartnerShop* you can Setup more shops in a minute, it is real easy.

Attention

> You need a Texteditor to Setup *easyPartnerShop*, no MS Word, just a editor like kwrite, gedit, or notepad.

On how to set it up, please look at the Shop Settings, after you are finished with those steps, you go to the Install and finish the installation of your shop.

- SHOP_THEME
- SHOP_THEME_SETTINGS
- SHOP_LANG
- SHOW_EPS_INFO

1.6 Bugs

You found a bug!!!

Please look here if the bug is allready known, if not open a ticket at *GitHub issuess*

Author

> Ruben Storm

Version

> 0.0.6

Since

> Version 0.0.6

See also

> `http://yoursl.in/epsIssuess` GitHub issuess

Chapter 2

Shop Settings

Shop setup

This is the setup file, here you have to set up your shop. Please make sure you have everything setup right, so the shop can work.

There are four points you have to configure in eps_settings.php, that's all. Do configure those steps and move on to the XML configuration

- SHOP_THEME
- SHOP_THEME_SETTINGS
- SHOP_LANG
- SHOW_EPS_INFO

The other configuration is the theme XML and the shop XML. For this setup you go to Database Settings

When you are finished with that, go to the Install and Install you shop online.

Attention

Make sure you made all setups right

Author

Ruben Storm

Version

0.0.6

Since

Version 0.0.6

Chapter 3

CHANGELOG

easyPartnerShop CHANGELOG

This is the CHANGELOG for *easyPartnerShop*

3.1 Maintainers

- Ruben Storm

3.2 Translators

None

3.3 CHANGELOG

3.3.1 0.0.6

- Better Documentation
- Code cleanup
- header.xml deleted, no need anymore
- body.xml deleted, no need anymore

3.3.2 0.0.5

- First Release

Date

2012-06-01

3.3.3 0.0.1

- First Testing

Author

Ruben Storm

Copyright

Copyright (c) 2012, Ruben Storm

Version

0.0.6

Since

Version 0.0.6

See also

`http://yoursl.in/BS5V4V` Current CHANGELOG

Chapter 4

Database Settings

Setup your shop Database

After you are finished with the Shop Settings you keep going with this part.

In this part you have to configure the Main Shop Database. In this file you put all your shops in you wan to host.

After your finished with this file your shop is ready to run. But you might want more, like a own theme, so how to make your own theme is explained in a different place.

4.1 Main Shop Database

Database setup

You have to setup a own container for each shop you have in your system. Here you have an example how a container look like. The file for the databse you find in the folder *eps_content/eps_DB/setup/* and it is named *settings.xml*

The main container is <shops> so you have to put in your shops in between this.
Main container:

```
<shops>
     // container for all shops
</shops>
```

This is XML, and it is not hard to set up, in the distribution of your shop is a file included, so just look at it and it is self explaining. There are a few example shop allready set up for you, so you have an example.

In this tutorial are some example hot to set up your shop.

- Example
- Name
- Host

- frameLink
- Owner
- Slogan
- Header
- Template

Here is a Database example for you.

4.1.1 Example

In the main container you put the following code for each shop into it:

```
<shop>
     <name>Ruben’s Bookstore</name>
     <host>books.ruben-storm.eu</host>
     <frameLink>http://astore.amazon.de/power2search-21</frameLink>
     <owner>Ruben Storm</owner>
     <slogan>Ruben’s Bookstore - Bücher in ca. 1000 Kategorien</slogan>
     <header>
         <title>Ruben’s Bookstore - Bücher bei books.ruben-storm.eu</title>
         <contlang>de</contlang>
         <meta>
             <descr>Ruben’s Bookstore, Bücher in ca. 1000 Kategorien sortiert.
     </descr>
             <keywords>books, Biografien, Erinnerungen, Business, Karriere</
     keywords>
         </meta>
     </header>
     <template>default</template>
</shop>
```

This is the container for each shop, it is an XML file, so please keep up with the standards.

Now i will explain you on what are the fields for.

4.1.2 Name

The field name:

```
<name>
     // write here the name of your shop
</name>
```

4.1.3 Host

The hostname:

```
<host>
     // your host, where the shop will run under
     // make sure it is writen the way on the example
</host>
```

Attention

This is impotant, dont use `http://` in front of it, your shop will not work

4.1.4 frameLink

The partner shop:

```
<frameLink>
     // The link to your partner shop, like Spreadshirts or Amazon
     // The example is a Amazon partner shop
</frameLink>
```

4.1.5 Owner

The Owner:

```
<owner>
     // The name of the shop owner
</owner>
```

4.1.6 Slogan

Your Slogan:

```
<slogan>
     // The slogan for your shop
</slogan>
```

4.1.7 Header

The header:

```
<header>
     <title>
         // Write your shop title here
     </title>
     <contlang>
         // Write your shop language here
     </contlang>
     <meta>
         <descr>
             // Write your meta description here
         </descr>
         <keywords>
             // Write your meta keywords here
         </keywords>
     </meta>
</header>
```

The header container is the data for each shop that is used in the HML header

4.1.8 Template

The template:

```
<template>
     // Write the name of your theme here,
     // used if you have setup each shop with a own theme
</template>
```

This container is used if you have setup the SHOP_THEME_SETTINGS to *1*

```
     define('SHOP_THEME_SETTINGS', 1);
```

4.2 Database example

How a Database looks like

Here is an example file, the way it should look like if you have two shops installed, shop number 1 with the theme *blue* and shop numer two with the theme *red*

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
This is an example shopfile with two shops installed.
Don't use this file, change it to fit your own shop.
-->
<shops>
     <shop>
         <name>Ruben's Bookstore</name>
         <host>books.ruben-storm.eu</host>
         <frameLink>http://astore.amazon.de/power2search-21</frameLink>
         <owner>Ruben Storm</owner>
         <slogan>Ruben's Bookstore - Bücher in ca. 1000 Kategorien</slogan>
         <header>
             <title>Ruben's Bookstore - Bücher bei books.ruben-storm.eu</title
     >
             <contlang>de</contlang>
             <meta>
                 <descr>Ruben's Bookstore, Bücher in ca. 1000 Kategorien
      sortiert.</descr>
                 <keywords>books, Biografien, Erinnerungen, Business,
      Karriere</keywords>
             </meta>
         </header>
         <template>blue</template>
     </shop>
     <shop>
         <name>Ruben's Poster und Leinwand Shop</name>
         <host>poster.ruben-storm.eu</host>
         <frameLink>http://rubenstorm.fotoportopro.de/</frameLink>
         <owner>Ruben Storm</owner>
         <slogan>Fotos und Leinwände für Ihr neues Zuhause</slogan>
         <header>
             <title>Ruben's Poster und Leinwand Shop -
      poster.ruben-storm.eu</title>
```

```
              <contlang>de</contlang>
              <meta>
                  <descr>Bestellen Sie Poster oder Leinwände für Ihre Wände,
       die einfach so kahl sind.</descr>
                  <keywords>Poster, Leinwand, Wildlife, Enten, Gans,
       Schwan</keywords>
              </meta>
          </header>
          <template>red</template>
      </shop>
 </shops>
```

The themes red and blue must exist in your installation, on a default install is only one theme in it, it is the default.

Attention

If now both shops have the same theme, go and read the Template

Author

Ruben Storm

Version

0.0.6

Since

Version 0.0.6

Chapter 5

Make Your Theme

How to make your own theme

This is a short documentation on how you make your own theme. The easy way is to go in the folder *eps_conten/themes/default/* and look at the files.

The Theme Library is not finished yet, there is a bit more work todo, but so far it works.

Todo Finish the Theme Library

A Theme for *easyPartnerShop* has two important files and then any file you want and need (ment .css and .js).

To see the files you can look on GitHub

First make a folder with the name of your theme under *eps_conten/themes/* lets say you want to name it *blue* so it should look like *eps_conten/themes/blue* and in that folder you make a file called *theme.xml*, this file is your theme settings file and then make a file named *template_1.xhtml*, this is your HTML template file for your theme.

For help you can look in the GitHub Repo.

Attention

The filenames are very important, dont change them

See also

http://yoursl.in/epsGitHub GitHub Repo
http://yoursl.in/epsThemeXML Theme Settings File
http://yoursl.in/epsTemplateXHTML Template File

Author

Ruben Storm

Copyright

Copyright (c) 2012, Ruben Storm

Version

0.0.6

Since

Version 0.0.6

Chapter 6

Todo List

Class libErrors

Check it and fix some controler bugs

Global libErrors ()

manual and over work it, might be depricated

Class libTheme

make an error controler, so initialize if installed theme is old or new

Page Make Your Theme

Finish the Theme Library

Global SHOP_LANG

make or find a list with posix locale for example, not used in Version 0.0.5, implement in next version

Chapter 7

Bug List

Global SHOP_LANG

Not implemented in this version, the coding is not working

Chapter 8

Data Structure Index

8.1 Data Structures

Here are the data structures with brief descriptions:

Chapter 9

File Index

9.1 File List

Here is a list of all files with brief descriptions:

Chapter 10

Data Structure Documentation

10.1 libDefine Class Reference

Library for definitions.

Static Public Member Functions

- static mk_define ()

  Make definitions.
- static _definer ($def, $data)

  Single definer.

10.1.1 Detailed Description

Library for definitions.

The libDefine is making sure everything gets defined what the system needs to be running

Author

Ruben Storm

Version

0.0.5

Since

Version 0.0.5

10.1.2 Member Function Documentation

10.1.2.1 _definer ($ *def,* $ *data*) [static]

Single definer.

Define a single part, so if you have to restup a special information, it will get directly in here

Version

0.0.5

Since

Version 0.0.5

Author

Ruben Storm

Parameters

string	*$def*	this is the definition
string	*$data*	this is the data that has to be saved

Returns

10.1.2.2 mk_define ( ) [static]

Make definitions.

This is the main door for the libDefine, here it defines the host so the shop can find the special settings.

Version

0.0.5

Since

Version 0.0.5

Author

Ruben Storm

Returns

The documentation for this class was generated from the following file:

- lib/libDefine.php

10.2 libErrors Class Reference

Error handler.

Static Public Member Functions

- static mk_Errors ($message, $id)

  Make the error.
- static _getError ()

  check errors

10.2.1 Detailed Description

Error handler.

The error handler is checking for errors and if there is an error it throws an exeption

Todo Check it and fix some controler bugs

Author

Ruben Storm

Version

0.0.5

Since

Version 0.0.5

10.2.2 Member Function Documentation

10.2.2.1 function _getError ( ) [static]

check errors

Check if an error happen, then go to next and throw exception

Version

0.0.5

Since

Version 0.0.5

Author

Ruben Storm

Todo manual and over work it, might be depricated

Returns

10.2.2.2 mk_Errors ($ *message*, $ *id*) [static]

Make the error.

The error maker, main object to make an error, call this from other classes if an error happen

Version

0.0.5

Since

Version 0.0.5

Author

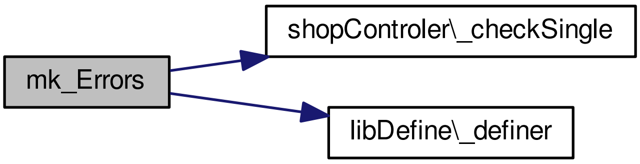
Ruben Storm

Parameters

string	*$message*	the error message that throw
string	*$id*	the id, 0 or 1, if 1 it will throw an error

Returns

Here is the call graph for this function:



The documentation for this class was generated from the following file:

- lib/libErrors.php

10.3 libShopData Class Reference

The Data Library.

Static Public Member Functions

- static getShop ()

  Shop data starter.
- static _getShopTheme ()

  Theme data.

10.3.1 Detailed Description

The Data Library.

The shop is makes the connection to the XML database and loading the required data. It is loading the shop settings, theme settings and the data for the shop.

Author

Ruben Storm

Version

0.0.5

Since

Version 0.0.5

10.3.2 Member Function Documentation

10.3.2.1 _getShopTheme ( ) [static]

Theme data.

Get the shop theme, if the shop has a mutli setup.

Version

0.0.5

Since

Version 0.0.2

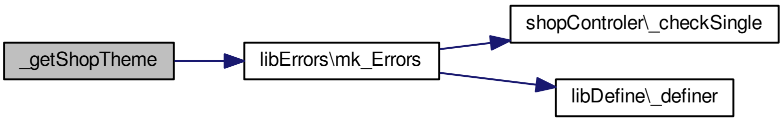
Author

Ruben Storm

Returns

string $shop->theme the name of the theme

Here is the call graph for this function:



10.3.2.2 getShop ( ) [static]

Shop data starter.

Starting to initialize the shop data system

Version

0.0.5

Since

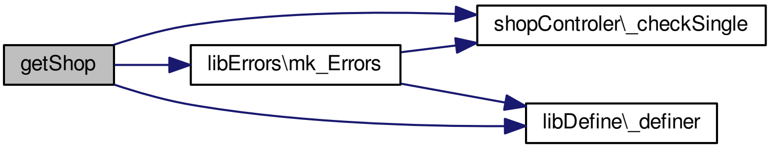
Version 0.0.5

Author

Ruben Storm

Returns

Here is the call graph for this function:



The documentation for this class was generated from the following file:

- lib/libShopData.php

10.4 libTheme Class Reference

Theme library.

Static Public Member Functions

- static mk_theme ()

  Theme maker.

10.4.1 Detailed Description

Theme library.

The theme driver, for version 0.0.5 totally new made up, so old themes will not run anymore in this version. The new theme driver runs with a XHTML theme system and has a new spaceholder

Author

Ruben Storm

Version

0.0.5

Since

0.0.5

Todo make an error controler, so initialize if installed theme is old or new

10.4.2 Member Function Documentation

10.4.2.1 mk_theme ( ) [static]

Theme maker.

Make the theme, this has to be twice taken in the driver

Author

Ruben Storm

Version

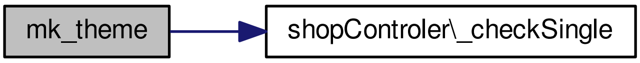
0.0.5

Since

0.0.5

Returns

Here is the call graph for this function:



The documentation for this class was generated from the following file:

- lib/libTheme.php

10.5 shopControler Class Reference

Shop controler.

Static Public Member Functions

- static checkShop ()

  Check the shop.
- static _checkSingle ($toCheck)

  Single checker.

10.5.1 Detailed Description

Shop controler.

This is the main class for the Shop controler

Author

Ruben Storm

Version

0.0.5

Since

Version 0.0.5

10.5.2 Member Function Documentation

10.5.2.1 _checkSingle ($ *toCheck*) `[static]`

Single checker.

The public method to check a single possition if it exists and is ok it will return true<

This method is used to check a single settings value.

Author

Ruben Storm

Version

0.0.5

Since

0.0.5

Parameters

string	*$toCheck*	the name string to check on

Returns

boolean true all is ok, true or nothing is ok, false

10.5.2.2 checkShop ( ) `[static]`

Check the shop.

The public method to enter the main controler

Author

Ruben Storm

Version

0.0.5

Since

0.0.5

Returns

The documentation for this class was generated from the following file:

- cmd/shopControler.php

10.6 shopManager Class Reference

Manager class.

Static Public Member Functions

- static eps_register ()

  Register.

10.6.1 Detailed Description

Manager class.

This is the shopManager class for the shop, it builds the shop and then gives the output

This class get automatic detected because of the sysLoader

Author

Ruben Storm

Version

0.0.5

Since

Version 0.0.5

10.6.2 Member Function Documentation

10.6.2.1 static public function eps_register ( ) `[static]`

Register.

Register the system and starting the shop

Author

Ruben Storm

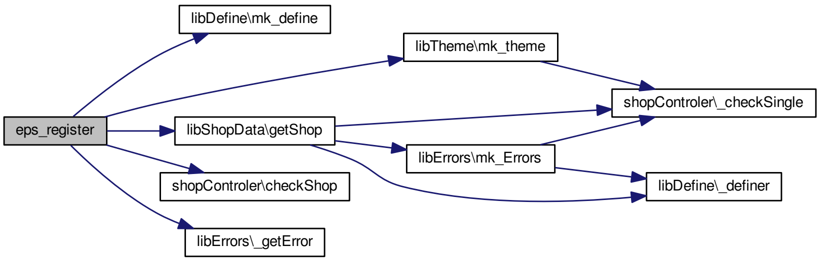
Version

0.0.5

Since

0.0.5

Here is the call graph for this function:



The documentation for this class was generated from the following file:

- cmd/shopManager.php

10.7 sysLoader Class Reference

The system loader.

Static Public Member Functions

- static init ($name)

    Entry for autoloader.

10.7.1 Detailed Description

The system loader.

This is the sysLoader.

This class is checking the filesystem if a class exists and then loading what it need to run.

Author

Ruben Storm

Version

0.0.5

Since

Version 0.0.5

See also

__autoload

10.7.2 Member Function Documentation

10.7.2.1 init ($ *name*) [static]

Entry for autoloader.

Required by the autoloader.

This part is needed by the __autoload method, so it can run the system

Author

Ruben Storm

Version

0.0.5

Since

Version 0.0.5

Parameters

string	*$name*	the name of the class to load

Returns

string $file the file with path, where to find the requested class

The documentation for this class was generated from the following file:

- sysLoader.php

Chapter 11

File Documentation

11.1 cmd/shopControler.php File Reference

The shopControler.

Data Structures

- class shopControler

  Shop controler.

11.1.1 Detailed Description

The shopControler. This is the Controler for the Shop

Author

Ruben Storm

Copyright

Copyright (c) 2012, Ruben Storm - GNU General Public License 3.0 (GPL) `http://www.gnu.org/licenses/gpl.html`

Version

0.0.5

Since

Version 0.0.5

```
 *
 * This program is distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
 * GNU General Public License for more details.
 *
```

11.2 cmd/shopManager.php File Reference

The shopManager.

Data Structures

- class shopManager

  Manager class.

11.2.1 Detailed Description

The shopManager. This is the shop manager, the shop manager is making the main work for the system. Everything gets managed from here. So it is like the main class in the system.

Author

Ruben Storm

Copyright

Version

0.0.5

Since

Version 0.0.5

```
 *
 * This program is distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
 * GNU General Public License for more details.
 *
```

11.3 eps_settings.php File Reference

Settings file.

Enumerations

eps_settings

- enum SHOP_THEME

  The theme name.
- enum SHOP_THEME_SETTINGS

  Theme settings.
- enum SHOP_LANG

  Shop language.
- enum SHOW_EPS_INFO

  Set the shop information.

11.3.1 Detailed Description

Settings file. This is the file where you set up your shop

This is the setup file, here you have to set up your shop. Please make sure you have everything setup right, so the shop can work.

11.3.2 Enumeration Type Documentation

11.3.2.1 enum SHOP_LANG

Shop language.

Setting up your language

In this section you set up the language of your shop. The default is german, for german language it is de_DE, see the default settings

Default settings:

```
define("SHOP_LANG", "de_DE");
```

Author

Ruben Storm

Bug Not implemented in this version, the coding is not working

Todo make or find a list with posix locale for example, not used in Version 0.0.5, implement in next version

Version

0.0.5

11.3.2.2 enum SHOP_THEME

The theme name.

Setting up the Theme for the Shop.

Define the name of the Shop Theme, it is the same name as the folder of the theme.

You can here set up the shop, so every shop instance is using the same theme or each Shop has a different theme. If you want that each shop has his own theme, then ignore this settings and go to the next one

Make sure that the theme, that you configure here, is installed. The theme name *default* is saying that your theme is in the folder default.

This is only used if SHOP_THEME_SETTINGS is set to 0

Default settings:

```
define('SHOP_THEME', 'default');
```

use the default theme, it is in the folder default of your main theme folder

Warning

The theme name must be the name of the folder where the theme resides

Author

Ruben Storm

Version

0.0.5

11.3.2.3 enum SHOP_THEME_SETTINGS

Theme settings.

Default is that all shops have the same theme. With this setup you can change it.

Configure your Shop for single or multi themes.

The default ist that all shops use the same theme, i call it single setup. But you can install more than one shop, they just use the same theme. With those settings you configure the Shop to use single or multi theme, possible settings are *0* or *1*

If you want to use the theme that is defined in the XML settings file, then set the this to *1*

Default settings:

```
define('SHOP_THEME_SETTINGS', 0);
```

All shops you install use the same theme

Every shop with a own theme:

```
define('SHOP_THEME_SETTINGS', 1);
```

Now each shop has a won theme, to set this up go to your XML setup file

With this settings you have to make sure you have the theme written in the setup file and that each shop has its own theme installed. Now you can ignore the settings for SHOP_THEME

Warning

If you turn this to 1, make sure you have the theme setup in the XML file

Author

Ruben Storm

Version

0.0.5

11.3.2.4 enum SHOW_EPS_INFO

Set the shop information.

You like this shop, you want to support it?

Please show where i come from and what is this for a shop

With other words, it has a copyright on it, you can configure here if you like to show it or not.

The default:

```
define('SHOW_EPS_INFO', 1);
```

Now it shows the copyright

Don't show:

```
define('SHOW_EPS_INFO', 0);
```

Now it will hide the copyright

Attention

Would be nice if you support me and let it on 1, but you can set it on 0, if you don't want to show it

Author

Ruben Storm

Version

0.0.5

11.4 index.php File Reference

The index file.

11.4.1 Detailed Description

The index file. This is the index.php file, here is your shop start. A Webserver is looking for a index.html, index.php or default.php ... make sure your Webserver is configured to take the index.php in first place.

Go to Home

Read how to Setup your shop

Author

Ruben Storm

Copyright

Version

0.0.5

Since

Version 0.0.5

```
*
* This program is distributed in the hope that it will be useful,
* but WITHOUT ANY WARRANTY; without even the implied warranty of
* MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
* GNU General Public License for more details.
*
```

See also

http://yoursl.in/easyPartnerShop Project Homepage
http://yoursl.in/GPL License

11.5 lib/libDefine.php File Reference

The libDefine.

Data Structures

- class libDefine

  Library for definitions.

11.5.1 Detailed Description

The libDefine. This is a library where the definitions made and controled.

Author

Ruben Storm

Copyright

Version

0.0.5

Since

Version 0.0.5

11.6 lib/libErrors.php File Reference

The libErrors.

Data Structures

- class libErrors

  Error handler.

11.6.1 Detailed Description

The libErrors. This is a library where the error made, controled and checked.

Author

Ruben Storm

Copyright

Version

0.0.5

Since

Version 0.0.5

```
 *
 * This program is distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
 * GNU General Public License for more details.
 *
```

11.7 lib/libShopData.php File Reference

The libShopData.

Data Structures

- class libShopData

  The Data Library.

11.7.1 Detailed Description

The libShopData. This is a library for data connection and definition.

This file is responible to get and make the data that your shop needs. It takes the required shop data for the requested shop and returns it. It is like a XML Database driver.

Author

Ruben Storm

Copyright

Copyright (c) 2012, Ruben Storm - GNU General Public License 3.0 (GPL) http-://www.gnu.org/licenses/gpl.html

Version

0.0.5

Since

Version 0.0.5

```
*
* This program is distributed in the hope that it will be useful,
* but WITHOUT ANY WARRANTY; without even the implied warranty of
* MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
* GNU General Public License for more details.
*
```

11.8 lib/libTheme.php File Reference

The libTheme.

Data Structures

- class libTheme

  Theme library.

11.8.1 Detailed Description

The libTheme. This is the theme library, here it will design the shop

Author

Ruben Storm

Copyright

Copyright (c) 2012, Ruben Storm - GNU General Public License 3.0 (GPL) http-://www.gnu.org/licenses/gpl.html

Version

0.0.5

Since

Version 0.0.5

```
*
* This program is distributed in the hope that it will be useful,
* but WITHOUT ANY WARRANTY; without even the implied warranty of
* MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
* GNU General Public License for more details.
*
```

11.9 sysLoader.php File Reference

The sysLoader.

Data Structures

- class sysLoader

  The system loader.

sysLoader

- enum INCLUDEBASE

  Define the INCLUDEBASE.
- __autoload ($name)

  Autoloader.

11.9.1 Detailed Description

The sysLoader. This is the file where it is looking for classes and includes them automatic so it is easy to develop new classes for the system.

Author

Ruben Storm

Copyright

Copyright (c) 2012, Ruben Storm - GNU General Public License 3.0 (GPL) http://www.gnu.org/licenses/gpl.html

Version

0.0.5

Since

Version 0.0.5

```
 *
 * This program is distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
 * GNU General Public License for more details.
 *
```

See also

http://projects.ruben-storm.eu/easypartnershop/ Project Page
http://www.gnu.org/licenses/gpl.html License

11.9.2 Enumeration Type Documentation

11.9.2.1 enum INCLUDEBASE

Define the INCLUDEBASE.

Set the include Base to the Document Root.

Warning

Don't change this settings, this has to be set the way it is

Author

Ruben Storm

Version

0.0.5

Since

Version 0.0.5

11.9.3 Function Documentation

11.9.3.1 __autoload ($ *name*)

Autoloader.

This is the autoloader, this is getting loaded to load everything needed during shop time

If class not exists it produces the class on the fly and throws an exception

Author

Ruben Storm

Version

0.0.5

Since

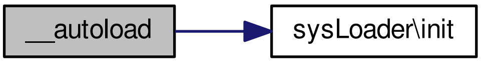
Version 0.0.5

Parameters

string	*$name*	the name of the class to load

Returns

Here is the call graph for this function:


__autoload
sysLoader\init